aws

Guia do desenvolvedor

OpenSearch Serviço Amazon



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

OpenSearch Serviço Amazon: Guia do desenvolvedor

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

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestigie a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

O que é o Amazon OpenSearch Service? ... 1
Características do Amazon OpenSearch Service ... 2
Quando usar ... 3
Versões compatíveis do Elasticsearch e OpenSearch ... 4
Suporte padrão e estendido ... 4
Versões com suporte padrão e estendido ... 6
Cálculo de custos do suporte estendido ... 7
Preços ... 8
Serviços relacionados ... 9
Configurar ... 11
Conceder permissões ... 11
Conceder acesso programático ... 11
Configure o AWS CLI ... 13
Abra o console do 14
Introdução ... 15
Criar um domínio ... 16
Carregar dados para indexação ... 17
Opção 1: Carregar um único documento ... 18
Opção 2: carregar vários documentos ... 18
Pesquisar documentos ... 19
Para pesquisar documentos via linha de comando ... 19
Pesquise documentos usando OpenSearch painéis ... 20
Excluir um domínio ... 21
OpenSearch Ingestão da Amazon ... 22
Principais conceitos ... 22
Benefícios ... 24
Limitações ... 24
Versões do Data Prepper compatíveis ... 25
Pipelines de escalabilidade ... 26
Preços ... 28
Suportado Regiões da AWS ... 28
Configurar funções e usuários ... 28
Perfis do pipeline ... 30
Perfil de ingestão ... 33

Concedendo acesso aos pipelines aos domínios ... 35
Conceder aos pipelines acesso às coleções ... 39
Começando com o OpenSearch Ingestion ... 45
Tutorial: ingerir dados em um domínio ... 45
Tutorial: Ingestão de dados em uma coleção ... 54
Atributos do pipeline ... 60
Armazenamento em buffer persistente ... 61
Dividindo ... 63
Encadeamento ... 64
Filas de mensagens não entregues ... 65
Gerenciamento de índices ... 67
End-to-end reconhecimento ... 71
Pressão oposta da origem ... 71
Como criar pipelines ... 72
Pré-requisitos e perfil do IAM necessário ... 73
Permissões obrigatórias do IAM ... 73
Como especificar a versão do pipeline ... 75
Como especificar o caminho de ingestão ... 76
Como criar pipelines ... 77
Acompanhar o status da criação do pipeline ... 82
Trabalhar com esquemas ... 83
Visualizar pipelines ... 85
Atualizar pipelines ... 87
Considerações ... 88
Permissões obrigatórias ... 88
Atualizar pipelines ... 89
Implantações azul/verde para atualizações de pipeline ... 90
Gerenciar os custos do pipeline ... 91
Como interromper um pipeline ... 92
Como iniciar um pipeline ... 93
Exclusão de pipelines ... 94
Plug-ins e opções compatíveis ... 95
Plug-ins compatíveis ... 95
Processadores sem estado x processadores com estado ... 98
Requisitos e restrições de configuração ... 98
Integração de pipelines ... 104

Criar o endpoint de ingestão ... 104
Criação de uma função de ingestão ... 105
Serviços da Atlassian ... 107
Amazon Aurora ... 124
Amazon DynamoDB ... 154
Amazon DocumentDB ... 170
Confluent Cloud Kafka ... 189
Amazon MSK ... 199
Amazon RDS ... 207
Amazon S3 ... 236
Amazon Security Lake ... 246
Fluent Bit ... 251
Fluentd ... 253
Inferência em lote offline de ML ... 255
OpenTelemetry Colecionador ... 271
Amazon Managed Service for Prometheus ... 278
Kafka autogerenciado ... 286
Clusters autogerenciados OpenSearch ... 294
Amazon Kinesis Data Streams ... 302
Próximas etapas ... 314
AWS Lambda ... 314
Migração de dados entre domínios e coleções ... 318
Limitações ... 319
OpenSearch Serviço como fonte ... 319
Especificação de vários coletores OpenSearch de domínio de serviço ... 321
Migração de dados para uma coleção de OpenSearch VPC sem servidor ... 322
Gerenciando pipelines com o AWS SDKs ... 323
Python ... 323
Segurança na OpenSearch ingestão ... 327
Como configurar o acesso à VPC para pipelines ... 328
Configurar pipelines para ingestão entre contas ... 333
Gerenciamento de Identidade e Acesso ... 340
Monitoramento com CloudTrail ... 348
Endpoints da VPC (AWS PrivateLink) ... 352
Uso de tags com pipelines ... 355
Permissões obrigatórias ... 356

Uso de tags (console) ... 356
Uso de tags (AWS CLI) ... 357
Registro em log e monitoramento ... 357
Monitoramento dos logs de pipeline ... 357
Métricas do pipeline de monitoramento ... 359
Práticas recomendadas ... 392
Práticas recomendadas gerais ... 393
CloudWatch Alarmes recomendados ... 393
Amazon sem OpenSearch servidor ... 400
Benefícios ... 400
O que é Amazon OpenSearch Serverless? ... 401
Casos de uso do OpenSearch Serverless ... 402
Como funciona ... 402
Escolha de um tipo de coleção ... 404
Preços ... 405
Suportado Regiões da AWS ... 406
Limitações ... 406
Comparando OpenSearch serviços e sem OpenSearch servidor ... 407
Tutorial: Introdução ao OpenSearch Serverless ... 411
Etapa 1: configurar permissões ... 412
Etapa 2: criar uma coleção ... 413
Etapa 3: Transferir e pesquisar dados ... 414
Tratamento de erros ... 415
Etapa 4: Excluir a coleção ... 416
Próximas etapas ... 416
Coleções ... 417
Gerenciar coleções ... 417
Trabalho com coleções de pesquisa vetorial ... 441
Usar políticas de ciclo de vida de dados ... 449
Gerenciando coleções com o AWS SDKs ... 458
Criação de coleções com CloudFormation ... 470
Fazer backup de coleções usando snapshots ... 471
Suporte ao codec Zstandard no Amazon Serverless OpenSearch ... 478
Economize armazenamento usando a fonte derivada ... 480
Grupos de coleções ... 482
Principais conceitos ... 483

Limites de capacidade do grupo de coleta 484
Criptografia e chaves KMS em grupos de coleção 487
Crie grupos de coleções 488
Gerenciar grupos de coleções 491
Gerenciamento de limites de capacidade 494
Definição de configurações de capacidade 497
Limites máximos de capacidade 497
Monitoramento do uso da capacidade 498
Ingestão de dados em coleções 498
Permissões mínimas necessárias 499
OpenSearch Ingestão 500
Fluent Bit 500
Amazon Data Firehose 501
Go 501
Java 504
JavaScript 505
Logstash 508
Python 510
Ruby 512
Outros clientes 513
Configurar o Machine Learning 514
Machine Learning 514
Conectores 514
Modelos da 515
Configurar permissões para Machine Learning 516
Não suportado APIs e recursos 517
Configurar a pesquisa neural e pesquisa híbrida 518
Pesquisa neural 518
Pesquisa híbrida 518
Consultas neurais e híbridas 519
Configurar permissões do 520
Configurar fluxos de trabalho 523
Fluxos de trabalho 523
Configurar permissões do 523
Segurança sem OpenSearch servidor 525
Políticas de criptografia 527

Políticas de rede ... 527
Políticas de acesso a dados ... 528
Autenticação SAML e IAM ... 529
Segurança da infraestrutura ... 530
Conceitos básicos da segurança ... 530
Gerenciamento de Identidade e Acesso ... 545
Criptografia ... 563
Acesso à rede ... 574
Conformidade com os FIPS ... 586
Controle de acesso a dados ... 591
Endpoint VPC do plano de dados ... 602
Endpoint VPC do plano de controle ... 614
Autenticação SAML ... 617
Validação de conformidade ... 627
Aplicação de tags nas coleções ... 627
Permissões obrigatórias ... 628
Aplicar tags em coleções (console) ... 628
Aplicar tags em coleções (AWS CLI) ... 629
Operações e plug-ins com suporte ... 629
Operações e permissões de OpenSearch API suportadas ... 629
OpenSearch Plugins compatíveis ... 639
Monitoramento OpenSearch sem servidor ... 640
Monitoramento com CloudWatch ... 641
Monitoramento com CloudTrail ... 648
Monitoramento com EventBridge ... 653
Criação e gerenciamento de domínios ... 657
Criação OpenSearch de domínios de serviço ... 657
Criação OpenSearch de domínios de serviço (console) ... 657
Criação OpenSearch de domínios de serviço ()AWS CLI ... 664
Criação OpenSearch de domínios de serviço ()AWS SDKs ... 666
Criação OpenSearch de domínios de serviço ()AWS CloudFormation ... 666
Configuração de políticas de acesso ... 667
Migração para o OpenSearch serviço com o Migration Assistant ... 667
Visão geral do ... 667
Cenários de migração ... 668
Introdução ao Migration Assistant ... 669

Arquitetura da solução ... 669
Configurações avançadas do cluster ... 670
Enriquecimento semântico automático ... 671
Introdução ... 671
Detalhes do modelo e benchmark de desempenho ... 672
Idiomas suportados ... 673
Configure um índice automático de enriquecimento semântico para domínios ... 673
Ingestão e pesquisa de dados ... 675
Configurar permissões para enriquecimento semântico automático ... 675
Regravações de consultas ... 679
Limitações do enriquecimento semântico automático ... 679
Preços ... 680
Alterações de configuração ... 680
Mudanças que geralmente causam blue/green implantações ... 681
Mudanças que geralmente não causam blue/green implantações ... 682
Opções de implantação azul/verde ... 683
Determinar se uma alteração causará uma blue/green implantação ... 685
Rastrear uma alteração de configuração ... 689
Etapas de uma alteração de configuração ... 691
Impacto no desempenho das blue/green implantações ... 695
Cobranças para alterações de configuração ... 695
Solução de problemas de erros de validação ... 696
Atualizações de software de serviço ... 702
Atualizações opcionais x obrigatórias ... 703
Atualizações de patch ... 704
Considerações ... 704
Como iniciar uma atualização ... 704
Janelas fora do horário de pico ... 708
Atualizações de monitoramento ... 709
Quando os domínios não são elegíveis para uma atualização ... 709
Janelas fora do horário de pico ... 710
Atualizações de software de serviço fora do horário de pico ... 711
Otimizações do Auto-Tune fora do horário de pico ... 712
Ativar a janela fora do horário de pico ... 713
Configurar uma janela personalizada fora do horário de pico ... 713
Exibir ações agendadas ... 714

Ações de reagendamento ... 716
Migração das janelas de manutenção do Auto-Tune ... 718
Notificações ... 719
Conceitos básicos das notificações ... 719
Gravidades das notificações ... 720
Exemplo de EventBridge evento ... 721
Configuração de um domínio Multi-AZ ... 722
Multi-AZ com modo de espera ... 722
Multi-AZ sem modo de espera ... 724
Interrupções na zona de disponibilidade ... 728
Suporte à VPC ... 730
VPC versus domínios públicos ... 730
Limitações ... 731
Arquitetura ... 731
Criação de snapshots de índices ... 739
Pré-requisitos ... 740
Registro de um repositório de snapshots manuais ... 744
Obtenção manual de snapshots ... 748
Restauração de snapshots ... 750
Excluir snapshots manuais ... 753
Automação de snapshots com o Snapshot Management ... 753
Automação de snapshots com o Gerenciamento de estados de índices ... 756
Uso do Curator para snapshots ... 756
Atualização de domínios ... 757
Caminhos de atualização com suporte ... 757
Atualizar um domínio (console) ... 762
Atualizar um domínio (CLI) ... 762
Atualizar um domínio (SDK) ... 763
Solução de problemas de falha de validação ... 764
Solução de problemas em uma atualização ... 765
Como usar um snapshot para migrar dados ... 767
Criar um endpoint personalizado ... 775
Endpoints personalizados para novos domínios ... 775
Endpoints personalizados para domínios existentes ... 776
Mapeamento CNAME ... 776
Auto-Tune ... 777

Tipos de alterações ... 777
Habilitação ou desabilitação do Auto-Tune ... 779
Agendamento de melhorias no Auto-Tune ... 780
Monitoramento de alterações no Auto-Tune ... 781
Marcação de domínios ... 781
Exemplos de marcação com tags ... 782
Marcação de domínios (console) ... 782
Marcação de domínios (AWS CLI) ... 783
Marcação de domínios (AWS SDKs) ... 784
Executar ações administrativas ... 786
Reiniciando o OpenSearch processo em um nó de dados ... 786
Reinicializar um nó de dados ... 787
Reiniciar o processo do Dashboards ... 787
Limitações ... 788
Trabalhar com consultas diretas ... 790
Preços ... 790
Limitações ... 791
Limitações gerais ... 791
Limitações do Amazon S3 ... 791
Limitações do Amazon CloudWatch Logs ... 792
Limitações do Amazon Security Lake ... 792
Recomendações ... 793
Recomendações gerais ... 793
Recomendações para o Amazon S3 ... 794
CloudWatch Recomendações de registros ... 794
Recomendações sobre o Security Lake ... 795
Cotas ... 796
Cotas para o Amazon S3 ... 796
Cotas para registros CloudWatch ... 797
Cotas para o Security Lake ... 798
Suportado Regiões da AWS ... 799
Disponível Regiões da AWS para Amazon S3 ... 799
Disponível Regiões da AWS para CloudWatch registros ... 800
Disponível Regiões da AWS para Security Lake ... 801
Consultas diretas no S3 ... 801
Criar uma fonte de dados do S3 ... 802

Configurar uma fonte de dados do S3 ... 810
Consultas diretas em registros CloudWatch ... 813
Criação de uma fonte CloudWatch de dados de registros ... 813
Configurando uma fonte de dados de CloudWatch registros ... 820
Consultas diretas no Security Lake ... 822
Criar uma fonte de dados do Security Lake ... 822
Configurar uma fonte de dados do Security Lake ... 828
Gerenciar uma fonte de dados ... 831
Monitoramento com fontes de dados de CloudWatch métricas ... 831
Habilitação e desabilitação de fontes de dados ... 835
Monitoramento com AWS orçamento ... 835
Excluir uma fonte de dados ... 836
Otimizar a performance da consulta ... 837
Índices de salto ... 837
Visões materializadas ... 838
Índices de abrangência ... 838
Comandos SQL e PPL compatíveis ... 838
Comandos SQL compatíveis ... 840
Comandos PPL compatíveis ... 1048
Domínios de monitoramento ... 1232
Monitoramento de métricas de cluster ... 1233
Visualizando métricas em CloudWatch ... 1234
Interpretando prontuários de saúde em serviço OpenSearch ... 1235
Métricas de cluster ... 1235
Métricas de nó principal dedicado ... 1243
Métricas do nó Coordenador dedicado ... 1245
Métricas de volume do EBS ... 1245
Métricas de instância ... 1248
Métricas quentes ... 1261
Métricas de armazenamento de baixa atividade ... 1268
OpenSearch Métricas de instâncias otimizadas (OR1) ... 1269
Métricas de alerta ... 1270
Métricas de detecção de anomalias ... 1271
Métricas de pesquisa assíncrona ... 1273
Métricas do Auto-Tune ... 1275
Métricas do multi-AZ com modo de espera ... 1276

Métricas pontuais ... 1278
Métricas de SQL ... 1279
Métricas de k-NN ... 1280
Métricas de pesquisa entre clusters ... 1284
Métricas de replicação entre clusters ... 1285
Métricas de Learning to Rank ... 1286
Métricas da Piped Processing Language ... 1287
Monitoramento de logs ... 1288
Habilitação da publicação de logs (console) ... 1290
Habilitação da publicação de logs (AWS CLI) ... 1292
Habilitação da publicação de logs (AWS SDKs) ... 1294
Habilitação da publicação de logs (CloudFormation) ... 1294
Como definir limites de log lento para solicitações de pesquisa ... 1296
Como definir limites de logs lentos de fragmentos ... 1297
Teste logs lentos ... 1297
Visualizar logs ... 1298
Monitoramento de logs de auditoria ... 1299
Limitações ... 1299
Habilitação dos logs de auditoria ... 1300
Ative o registro de auditoria usando o AWS CLI ... 1302
Habilitar o registro de auditoria em log usando a API de configuração ... 1302
Camadas e categorias do log de auditoria ... 1302
Configurações do log de auditoria ... 1305
Exemplo de log de auditoria ... 1308
Configuração de logs de auditoria usando a API REST ... 1311
Monitoramento de eventos ... 1313
Eventos de atualização de software de serviço ... 1314
Auto-Tune de eventos ... 1320
Eventos de integridade do cluster ... 1325
Eventos de endpoint da VPC ... 1339
Eventos de desativação do nó ... 1341
Eventos de retirada do nó degradado ... 1343
Eventos de erro de domínio ... 1345
Tutorial: Ouvindo eventos OpenSearch de serviço ... 1347
Tutorial: Envio de alertas do SNS para atualizações disponíveis ... 1350
Monitoramento com CloudTrail ... 1351

Informações OpenSearch do Amazon Service em CloudTrail 648
Entendendo as entradas do arquivo de log do Amazon OpenSearch Service 652
Informações sobre o cluster 1356
Benefícios 1356
Acesse o Cluster Insights por meio do console 1357
Acesse o Cluster Insights e métricas detalhadas por meio da OpenSearch interface 1357
Crie e configure um aplicativo para visualizar o Cluster Insights 1358
Compreendendo o Cluster Insights 1360
Painel de visão geral 1361
Status atual do cluster 1361
Tendência de insights 1361
Insights abertos atuais 1362
OpenSearch Clusters de serviços 1362
Principais insights por severidade 1362
Detalhes dos insights 1362
Detalhes do cluster 1362
Visão geral do 1362
Práticas recomendadas para configuração 1363
Insights 1363
Seção de métricas 1364
Visualizações Node, Index e Shard 1364
Visualização da consulta 1365
Acesse Insights por meio de EventBridge eventos da Amazon 1367
Catálogo de insights 1367
Segurança 1371
Proteção de dados 1372
Criptografia em repouso 1373
Node-to-node criptografia 1377
Gerenciamento de Identidade e Acesso 1378
Tipos de políticas 1378
Fazendo e assinando solicitações OpenSearch de serviço 1387
Quando há colisão de políticas 1389
Referência de elementos da política 1390
Opções avançadas e considerações sobre a API 1396
Configuração de políticas de acesso 1400
Exemplos adicionais de políticas 1400

Referência de permissões da API ... 1400
AWS políticas gerenciadas ... 1401
Prevenção contra o ataque do “substituto confuso” em todos os serviços ... 1411
Controle de acesso refinado ... 1413
Visão geral: controle de acesso refinado e segurança de serviços OpenSearch ... 1414
Principais conceitos ... 1417
Sobre o usuário-mestre ... 1418
Habilitar o controle de acesso detalhado ... 1419
Acessando OpenSearch painéis como usuário principal ... 1423
Gerenciar permissões ... 1425
Configurações recomendadas ... 1431
Limitações ... 1434
Modificação do usuário primário ... 1436
Usuários primários adicionais ... 1436
Snapshots manuais ... 1438
Integrações ... 1438
Diferenças de API REST ... 1439
Tutorial: Controle de acesso minucioso com autenticação Cognito ... 1441
Tutorial: Banco de dados interno de usuários com autenticação básica ... 1446
Validação de conformidade ... 1449
Resiliência ... 1450
JSON Web Tokens ... 1450
Considerações ... 1450
Modificar a política de acesso ao domínio ... 1451
Configurar autenticação e autorização do JWT ... 1451
Usar um JWT para enviar uma solicitação de teste ... 1452
Segurança da infraestrutura ... 1454
Trabalhando com OpenSearch VPC endpoints gerenciados por serviços ... 1455
Autenticação SAML para painéis OpenSearch ... 1459
Visão geral da configuração do SAML ... 1459
Considerações ... 1460
Autenticação SAML para domínios de VPC ... 1460
Modificar a política de acesso ao domínio ... 1461
Configurar a autenticação iniciada por SP ou IdP ... 1462
Configurar a autenticação iniciada por SP ou IdP ... 1469
Configurar a autenticação SAML (AWS CLI) ... 1469

Configurar a autenticação SAML (API de configuração) 1470
Solução de problemas de SAML 1470
Desabilitação da autenticação SAML 1474
Suporte do IAM Identity Center para OpenSearch 1475
Considerações 1475
Modificar a política de acesso ao domínio 1476
Configurar autenticação e autorização do IAM Identity Center (console) 1476
Habilitar o controle de acesso refinado 1477
Configurar autenticação e autorização do IAM Identity Center (CLI) 1477
Desabilitar a autenticação do IAM Identity Center no domínio 1477
Autenticação do Amazon Cognito para painéis OpenSearch 1478
Pré-requisitos 1479
Configuração de um domínio para uso da autenticação do Amazon Cognito 1482
Como permitir a função autenticada 1486
Configuração de provedores de identidade 1487
(Opcional) Configuração de acesso granular 1488
(Opcional) Personalização da página de login 1489
(Opcional) Configuração da segurança avançada 1489
Teste 1490
Cotas 1490
Problemas de configuração comuns 1490
Desabilitando a autenticação do Amazon Cognito para painéis OpenSearch 1495
Excluindo domínios que usam a autenticação do Amazon Cognito para painéis OpenSearch 1495
Uso de perfis vinculados ao serviço 1495
Perfil de criação de domínio e fonte de dados de VPC 1496
Função de criação de coleção 1499
Perfil de criação de pipeline 1502
Código de exemplo 1506
Compatibilidade com clientes Elasticsearch 1506
Compactação de solicitações HTTP 1507
Habilitação da compactação gzip 1507
Cabeçalhos obrigatórios 1508
Código de exemplo (Python 3) 1508
Usando o AWS SDKs 1510
Java 1510

Python 1521
Nó 1524
Indexação de dados 1527
Restrições de nomenclatura para índices 1527
Redução do tamanho da resposta 1528
Codecs de índice 1530
Carregando dados de streaming no OpenSearch Serviço 1530
Carregando dados de streaming do OpenSearch Ingestion 1531
Carregamento de dados de transmissão do Amazon S3 1531
Carregamento dados de transmissão do Amazon Kinesis Data Streams 1537
Carregamento de dados de transmissão do Amazon DynamoDB 1541
Carregamento de dados de transmissão do Amazon Data Firehose 1546
Carregando dados de streaming da Amazon CloudWatch 1547
Carregando dados de streaming de AWS IoT 1547
Carregamento de dados com o Logstash 1547
Configuração 1547
Pesquisa de dados 1551
Pesquisas de URI 1552
Pesquisas de corpo da solicitação 1553
Impulsão de campos 1555
Destaques de resultados da pesquisa 1555
API de contagem 1557
Paginação de resultados da pesquisa 1558
Ponto de tempo 1558
Os parâmetros `from` e `size`. 1558
Dashboards Query Language 1559
Pacotes 1561
Permissões obrigatórias 1562
Carregar pacotes para o Amazon S3 1562
Importação e associação de pacotes 1563
Usando pacotes com OpenSearch 1564
Atualização de pacotes 1569
Atualizar manualmente os índices com um novo dicionário 1572
Dissociação e remoção de pacotes 1574
Plugins personalizados 1575
Plug-ins de terceiros 1593

Suporte a SQL 1597
Chamada de exemplo 1599
Notas e diferenças 1599
SQL Workbench 1600
SQL CLI 1453
Driver JDBC 1600
Driver ODBC 1600
Pesquisa entre clusters 1600
Limitações 1601
Pré-requisitos da pesquisa entre clusters 1602
Preços da pesquisa entre clusters 1602
Configuração de uma conexão 1602
Remoção de uma conexão 1604
Configuração da segurança e demonstração de exemplo 1604
OpenSearch Painéis 1610
Learning to Rank 1611
Conceitos básicos do Learning to Rank 1611
API do Learning to Rank 1633
Pesquisa assíncrona 1640
Exemplo de chamada de pesquisa 1640
Permissões da pesquisa assíncrona 1642
Configurações da pesquisa assíncrona 1643
Pesquisa entre clusters 1643
UltraWarm 1645
Ponto de tempo 1645
Considerações 1645
Criar um PIT 1646
Permissões pontuais 1648
Configurações do PIT 1649
Pesquisa entre clusters 1649
UltraWarm 1649
Pesquisa de agentes 1649
Pesquisa semântica 1650
Pesquisa simultânea de segmento 1650
Geração de consulta em linguagem natural 1651
Pré-requisitos 1652

Introdução 1652
Configurar permissões do 1652
Automação de configuração 1653
Pesquisa vetorial 1654
Importação dos vetores do Amazon S3 para o Serverless OpenSearch 1655
Arquiteturas de integração 1656
Permissões obrigatórias do IAM 1658
Configurando a integração do Amazon S3 Vectors com OpenSearch 1661
Recursos avançados de pesquisa com um mecanismo vetorial Amazon S3 1663
Habilitar vetores do S3 1664
Criar índices com o mecanismo vetorial do S3 1664
Limitações de funcionalidade 1667
Indexar documentos 1668
Pesquisar documentos 1669
Parâmetros de mapeamento compatíveis 1670
Medição e faturamento 1671
Desabilitar o mecanismo s3vector 1671
Pesquisa de k-NN 1672
Conceitos básicos do k-NN 1672
Diferenças, ajustes e limitações do k-NN 1675
Ingestão de vetores 1675
Como funciona 1676
OpenSearch Preços de ingestão de vetores 1676
Pré-requisitos 1676
Criar banco de dados vetoriais 1676
Características relacionadas 1677
Exportar índice vetorial do Amazon S3 para o mecanismo vetorial OpenSearch de serviço 1678
Importar o namespace vetorial do Amazon S3 para o mecanismo vetorial Service OpenSearch 1679
Visualize trabalhos de ingestão de vetores e histórico de importação 1681
Otimização automática 1682
Benefícios 1682
Como funciona 1683
Pré-requisitos 1684
Casos de uso para otimização automática 1685
Limitações 1686

Faturamento e custos 1687
Formatos de dados compatíveis 1687
Usando a otimização automática no console 1690
Características relacionadas 1696
Aceleração por GPU para indexação vetorial 1697
Pré-requisitos 1697
Como funciona 1697
Preços de aceleração de GPU 1698
Operações de aceleração e gravação de GPU 1698
Configurações de índice suportadas 1699
Suportado Regiões da AWS 1699
Ativar aceleração por GPU 1699
Crie índices vetoriais acelerados por GPU 1702
Indexação e fusão forçada 1704
Práticas recomendadas 1705
OpenSearch Painéis 1706
Controlar acesso ao Dashboards 1707
Usando um proxy para acessar o OpenSearch serviço a partir de painéis 1707
Configurar o OpenSearch Dashboards para usar um servidor de mapas WMS 1711
Conectando um servidor local de painéis ao serviço OpenSearch 1712
Gerenciar índices no Dashboards 1714
Recursos adicionais 1714
OpenSearch UI 1715
Histórico de versões 1716
Introdução 1718
Permissões necessárias para criar aplicativos do Amazon OpenSearch Service 1719
Como criar uma aplicação do 1722
Gerenciar administradores de aplicação 1731
IA agente 1734
Capacidades gerais 1735
Limites de preço e uso 1736
Segurança 1736
Ativando e desativando recursos de IA agentes 1737
Disponibilidade de regiões 1737
Chat Agêntico 1738
Agente de investigação 1742

Memória Agêntica 1744
Criptografia e chave gerenciada pelo cliente 1745
Pré-requisitos para usar chaves gerenciadas pelo cliente 1746
Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o console 1748
Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o AWS CLI 1748
Monitorando o uso da chave gerenciada pelo cliente 1749
Atualizando as configurações de criptografia 1749
Habilitar federação SAML com o IAM 1750
Etapa 1: configurar a aplicação de provedor de identidades (Okta) 1751
Etapa 2: AWS Configurar o Okta 1755
Etapa 3: criar a política OpenSearch de acesso ao Amazon Service no IAM 1757
Etapa 4: verificar a experiência de login único iniciada por provedor de identidades com o SAML 1760
Etapa 5: configurar o controle de acesso refinado baseado em atributos SAML 1764
Gerenciar associações de fontes de dados e permissões de acesso à VPC 1768
Associando uma fonte de dados a um aplicativo de OpenSearch interface do usuário 1768
Gerenciar o acesso a domínios em uma VPC 1769
Configurando o acesso a coleções OpenSearch sem servidor em uma VPC 1770
Usando espaços de trabalho OpenSearch do Amazon Service 1774
Criação de espaços de trabalho de aplicativos de OpenSearch UI 1774
Privacidade do espaço de trabalho e colaboradores 1775
Tipo de espeço ded trabalho 1776
Acesso a dados entre regiões e contas 1777
Acesso a dados entre contas a OpenSearch domínios 1778
Pesquisa entre clusters 1801
Gerenciando o acesso à OpenSearch interface do usuário a partir de um VPC endpoint 1811
Criação de uma conexão privada entre uma VPC e uma interface do usuário OpenSearch 1812
Atualização da política de VPC endpoint para permitir acesso ao aplicativo de interface do usuário OpenSearch 1813
Revogando o acesso à OpenSearch interface do usuário em uma política de VPC endpoint 1814
Migrar do painel OpenSearch 1815
Visão geral da migração 1815
Pré-requisitos 1816

Etapa 1: exportar objetos salvos dos OpenSearch painéis 1817
Etapa 2: importar objetos salvos para a OpenSearch interface do usuário 1819
Endpoints e cotas 1821
OpenSearch Endpoints de interface do usuário 1822
OpenSearch Cotas de serviços de interface do usuário 1825
Gerenciamento de Índices 1826
OpenSearch instâncias otimizadas 1826
Limitações 1827
Ajustar para obter melhor throughput de ingestão 1828
Como as instâncias OpenSearch otimizadas diferem de outras instâncias 1828
Como as instâncias OpenSearch otimizadas diferem das UltraWarm instâncias 1829
Provisionamento de um domínio com OpenSearch instâncias otimizadas 1829
Armazenamento em vários níveis 1831
Arquitetura de armazenamento de vários níveis 1832
UltraWarm armazenamento 1838
Pré-requisitos 1839
UltraWarm requisitos de armazenamento e considerações de desempenho 1841
UltraWarm preços 1842
Habilitando UltraWarm 1842
Migração de índices para armazenamento UltraWarm 1845
Automatização de migrações 1848
Ajuste de migrações 1848
Cancelamento de migrações 1849
Listagem de índices quentes e mornos 1849
Retorno de índices warm para o armazenamento quente 1849
Restauração de índices quentes de snapshots 1850
Snapshots manuais de índices mornos 1851
Migração de índices mornos para o armazenamento frio 1852
Práticas recomendadas para índices KNN 1852
Desativando UltraWarm 1853
Armazenamento de baixa atividade 1853
Pré-requisitos 1854
Requisitos de armazenamento e considerações de performance do armazenamento de baixa atividade 1856
Preços do armazenamento de baixa atividade 1856
Habilitação do armazenamento de baixa atividade 1857

Gerenciando índices frios em painéis OpenSearch ... 1859
Migração de índices para o armazenamento frio ... 1859
Automatização de migrações para o armazenamento frio ... 1861
Cancelando migrações para armazenamento frio ... 1861
Listagem de índices de baixa atividade ... 1862
Migração de índices frios para o armazenamento warm ... 1865
Restauração de índices frios de snapshots ... 1867
Cancelamento de migrações do armazenamento de baixa atividade para o armazenamento de alta atividade ... 1867
Atualizando metadados de índice de baixa atividade ... 1868
Exclusão de índices de baixa atividade ... 1868
Desabilitação do armazenamento de baixa atividade ... 1869
Gerenciamento de estados de índice ... 1869
Criar uma política do IAM ... 1870
Políticas de exemplo ... 1871
Modelos do ISM ... 1875
Diferenças ... 1875
Tutorial: Automatização de processos do ISM ... 1877
Totalizações de índices ... 1882
Criação de um trabalho de totalização de índices ... 1882
Transformações de índices ... 1884
Criação de um trabalho de transformação de índice ... 1884
Replicação entre clusters ... 1886
Limitações ... 1887
Pré-requisitos ... 1887
Requisitos de permissão ... 1888
Configurar uma conexão entre clusters ... 1889
Como iniciar a replicação ... 1890
Confirmar replicação ... 1891
Interromper e retomar a replicação ... 1892
Encerrar a replicação ... 1893
Seguir automaticamente ... 1893
Atualizar domínios conectados ... 1895
Reindexação remota ... 1895
Pré-requisitos ... 1896
Reindexar dados entre os domínios da Internet OpenSearch do Serviço ... 1897

Reindexe os dados quando o domínio remoto estiver em uma VPC ... 1898
Reindexe dados entre domínios que não são OpenSearch de serviço ... 1903
Reindexar conjuntos de dados grandes ... 1903
Configurações da reindexação remota ... 1905
Streams de dados ... 1906
Conceitos básicos de fluxos de dados ... 1906
Monitoramento de dados ... 1910
Geração de alertas ... 1910
Permissões de alertas ... 1911
Conceitos básicos dos alertas ... 1911
Notificações ... 1912
Diferenças ... 1913
Detecção de anomalias ... 1914
Pré-requisitos ... 1915
Conceitos básicos da detecção de anomalias ... 1915
Tutorial: Detectar uso elevado da CPU com detecção de anomalias ... 1918
Assistente de IA ... 1922
Suportado Regiões da AWS ... 1922
Configurando o AI Assistant for OpenSearch Service ... 1923
Gerar visualizações usando linguagem natural ... 1923
Visualizar resumos e insights de alertas ... 1924
Antes de começar ... 1927
Visualizar resumos e insights de alertas ... 1928
Veja os resumos dos resultados da consulta gerados pelo AI Assistant na página Discover ... 1929
Visualizar os detectores de anomalias recomendados ... 1930
Acesse o chat do AI Assistant para perguntas sobre OpenSearch serviços ... 1931
Machine learning ... 1934
Conectores para Serviços da AWS ... 1934
Pré-requisitos ... 1935
Crie um conector OpenSearch de serviço ... 1938
Conectores para plataformas externas ... 1940
Pré-requisitos ... 1941
Crie um conector OpenSearch de serviço ... 1944
CloudFormation integrações de modelos ... 1946
CloudFormation Modelos disponíveis ... 1947
Pré-requisitos ... 1948

Modelos do Amazon Bedrock 1950
Configurando a pesquisa de agentes com Bedrock Claude 1951
Modelos de integração do servidor MCP 1952
SageMaker Modelos da Amazon 1955
Inferência remota para modelos de destaque semântico 1956
Configurações do ML Commons não compatíveis 1957
Plug-in de estrutura de fluxo 1958
Criação de conectores de ML no Service OpenSearch 1958
Configurar permissões do 1966
Security Analytics 1968
Componentes e conceitos de Security Analytics 1968
Tipos de log 1969
Detectores 1969
Regras 1969
Conclusões 1969
Alertas 1969
Explorando o Security Analytics 1970
Configurar permissões do 1972
Solução de problemas 1974
Esse erro de índice não existe 1974
Observabilidade 1975
O fluxo de trabalho de observabilidade no Service OpenSearch 1975
OpenSearch UI e OpenSearch painéis 1977
Logs 1978
OpenSearch UI e espaço de trabalho de observabilidade 1979
Consultando seus registros usando PPL 1980
Consultando seus registros usando a IA 1980
Consultando seus registros usando SQL 1981
Consultando seus registros usando DQL 1981
Painéis e alertas para registros 1981
Rastreamentos 1982
Processamento e ingestão de dados de rastreamento 1983
OpenSearch UI e espaço de trabalho de observabilidade 1979
Recursos principais 1983
Práticas recomendadas 1987
Monitoramento e alertas 1987

Configurar CloudWatch alarmes 1987
Habilitar a publicação de logs 1988
Estratégia de fragmentação 1988
Determinar as contagens de fragmentos e de nós de dados 1989
Evitar distorções de armazenamento 1990
Estabilidade 1990
Mantenha-se atualizado com OpenSearch 1990
Melhore a performance do snapshot 1991
Habilite nós principais dedicados 1991
Implantar em diversas zonas de disponibilidade 1992
Controlar o fluxo de ingestão e o armazenamento em buffer 1992
Criar mapeamentos para workloads de pesquisa 1993
Usar modelos de índice 1993
Gerenciar índices com o Index State Management 1995
Remover índices não utilizados 1995
Usar vários domínios para alta disponibilidade 1995
desempenho 1996
Otimizar o tamanho e a compactação de solicitações em massa 1996
Reduzir o tamanho das respostas de solicitações em massa 1996
Ajustar os intervalos de atualização 1997
Habilitar o Auto-Tune 1997
Segurança 1997
Habilite o controle de acesso detalhado 1997
Implantar domínios em uma VPC 1998
Aplicar uma política de acesso restritiva 1998
Habilite a criptografia em repouso 1998
Ativar node-to-node criptografia 1999
Monitor com AWS Security Hub CSPM 1999
Otimização de custos 1999
Use os tipos de instâncias de última geração 1999
Usar os volumes gp3 do Amazon EBS 1999
Uso UltraWarm e armazenamento refrigerado para dados de registro de séries temporais 2000
Revisar as recomendações para instâncias reservadas 2000
CloudWatch Alarmes recomendados 2001
Outros alarmes que você pode considerar 2006
Dimensionamento de domínios 2008

Cálculo de requisitos de armazenamento ... 2009
Como escolher o número de fragmentos ... 2011
Escolha dos tipos de instância e testes ... 2012
Escala de petabytes ... 2014
Nodos de coordenação dedicados ... 2016
Quando usar nós coordenadores dedicados ... 2016
Arquitetura e comportamento ... 2017
Requisitos e limitações ... 2017
Provisionar nós coordenadores dedicados ... 2018
Práticas recomendadas ... 2018
Nós principais dedicados ... 2020
Como escolher o número de nós principais dedicados ... 2021
Escolher tipos de instâncias para nós principais dedicados ... 2023
Referência geral ... 2025
Tipos de instâncias compatíveis ... 2025
Tipos de instâncias da geração atual ... 2026
Tipos de instância da geração anterior ... 2042
Recursos por versão do mecanismo ... 2053
Plug-ins por versão do mecanismo ... 2059
Plug-ins opcionais ... 2063
Operações compatíveis ... 2064
Diferenças notáveis de API ... 2064
Cotas ... 2120
Cotas de armazenamento de nós quentes ... 2120
Número de nós de dados por zona de disponibilidade ... 2023
Limite total de nós por família de instâncias ... 2121
Limites de tamanhos de volume do EBS ... 2122
Limites de rede ... 2135
Cotas de tamanhos de fragmentos ... 2145
Cotas de número de fragmentos ... 2146
Limites dos processos Java ... 2146
Limites das políticas de domínio ... 2146
Instâncias reservadas ... 2147
Compra de instâncias reservadas (console) ... 2147
Compra de instâncias reservadas (AWS CLI) ... 2148
Comprando instâncias reservadas (AWS SDKs) ... 2151

Verificação dos custos ... 2153
Otimização de custos ... 2153
Otimização de custos para clusters gerenciados do Amazon OpenSearch Service ... 2153
Otimização de custos para Amazon OpenSearch Service Serverless ... 2158
Outros recursos compatíveis ... 2163
Tutoriais ... 2165
Criar e pesquisar documentos ... 2165
Pré-requisitos ... 2165
Adicionar um documento a um índice ... 2166
Criação gerada automaticamente IDs ... 2167
Atualizar um documento com um comando POST ... 2168
Executar ações em massa ... 2169
Pesquisando documentos ... 2170
Recursos relacionados ... 2172
Migrando para o serviço OpenSearch ... 2172
Obter e carregar do snapshot ... 2172
Criar um domínio ... 2174
Conceder permissões para o bucket do S3 ... 2175
Restaure o snapshot ... 2177
Criação de uma aplicação de pesquisa ... 2180
Pré-requisitos ... 2181
Etapa 1: Indexar dados de exemplo ... 2181
Etapa 2: criar e implantar a função do Lambda ... 2182
Etapa 3: Criar a API no Gateway da API ... 2185
Etapa 4: (opcional) modificar a política de acesso ao domínio ... 2187
Mapeamento da função do Lambda (se estiver usando um controle de acesso minucioso) 2190
Etapa 5: Testar a aplicação Web ... 2191
Próximas etapas ... 2193
Visualização de chamadas de suporte ... 2194
Etapa 1: Configurar os pré-requisitos ... 2195
Etapa 2: Copiar código de exemplo ... 2196
(Opcional) Etapa 3: Indexar dados de exemplo ... 2200
Etapa 4: Analisar e visualizar seus dados ... 2202
Etapa 5: Limpar recursos e próximas etapas ... 2206
Renomeação OpenSearch do Amazon Service ... 2208
Nova versão de API ... 2208

Tipos de instâncias renomeados 2209
Alterações na política de acesso 2209
Políticas do IAM 2209
Políticas de SCP 2209
Novos tipos de recursos 2210
Kibana renomeado para Painéis OpenSearch 2211
Métricas renomeadas CloudWatch 2212
Abra o console do Billing and Cost Management. 2213
Novo formato dos eventos 2214
O que não mudou? 2214
Comece: atualize seus domínios para 1.x OpenSearch 2214
Solução de problemas 2216
Não consigo acessar os OpenSearch painéis 2216
Não é possível acessar o domínio da VPC 2216
Cluster no estado somente leitura 2216
Status de cluster vermelho 2218
Correção automática de clusters vermelhos 2219
Recuperação de uma carga contínua de processamento pesado 2220
Status de cluster amarelo 2222
ClusterBlockException 2222
Falta de espaço de armazenamento disponível 2223
Alta pressão da memória da JVM 2223
Erro ao migrar para multi-AZ com modo de espera 2224
Criação de um índice, modelo de índice ou política do ISM durante a migração de domínios sem espera para domínios com modo de espera 1974
Número incorreto de cópias de dados 2224
JVM OutOfMemoryError 2224
Nós de cluster com falha 2225
Limite máximo de fragmentos excedido 2226
Domínio paralisado no estado de processamento 2226
O saldo de intermitência do EBS está baixo 2227
A métrica do EBS aumenta durante o redimensionamento de volume 2227
Não é possível habilitar logs de auditoria 2228
Não é possível fechar o índice 2228
Verificações de licenças do cliente 2228
Controle de utilização de solicitações 2229

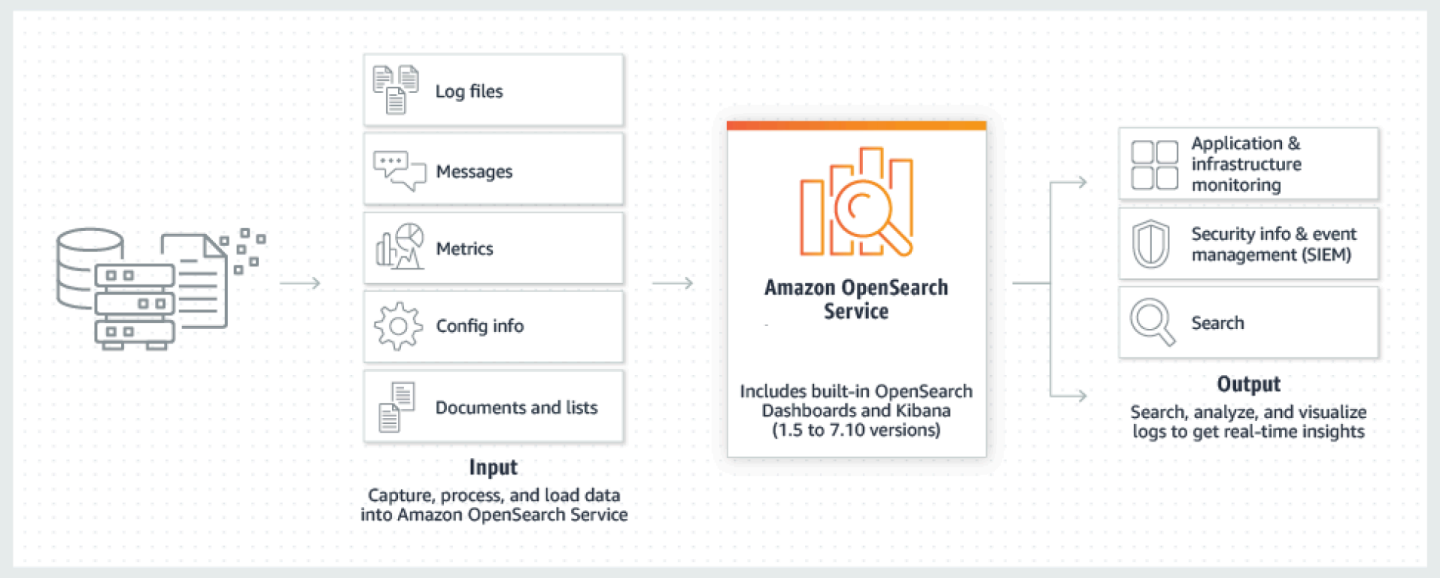
Não é possível executar o SSH no nó 2229
Erro de snapshot “Not Valid for the Object's Storage Class” (Inválido para a classe de armazenamento do objeto) 2229
Cabeçalho de host inválido 2229
Tipo de instância M3 inválido 2230
As consultas quentes param de funcionar após a ativação UltraWarm 2230
Não é possível reverter para a versão anterior após a atualização. 2230
Precisa de um resumo dos domínios para todos Regiões da AWS 2231
Erro do navegador ao usar OpenSearch painéis 2231
Distorção de armazenamento e de fragmentos do nó 2232
Distorção de armazenamento e de fragmentos de índices 2233
Operação não autorizada após a seleção do acesso via VPC 2233
Preso no carregamento após a criação do domínio da VPC 2234
Solicitações negadas para a OpenSearch API 2234
Não é possível conectar via Alpine Linux 2235
Muitas solicitações de pesquisa de contrapressão 2235
Erro de certificado ao usar o SDK 2236
Ocorre falha na instalação do plug-in personalizado devido a incompatibilidade entre versões 2237
Histórico do documento 2239
Atualizações anteriores 2309
AWS Glossário 2313
..... mmcccxiv

O que é o Amazon OpenSearch Service?

O Amazon OpenSearch Service é um serviço gerenciado que facilita a implantação, a operação e a escalabilidade de OpenSearch clusters na AWS nuvem. Um domínio OpenSearch de serviço é sinônimo de um OpenSearch cluster. Domínios são clusters com configurações, tipos de instância, contagens de instâncias e recursos de armazenamento especificados por você. O Amazon OpenSearch Service suporta OpenSearch e lega o Elasticsearch OSS (até a versão 7.10, a versão final de código aberto do software). Ao criar um domínio, você tem a opção de escolher qual mecanismo de pesquisa deseja usar.

OpenSearché um mecanismo de pesquisa e análise totalmente de código aberto para casos de uso como análise de registros, monitoramento de aplicativos em tempo real e análise de fluxo de cliques. Para obter mais informações, consulte a documentação do OpenSearch .

O Amazon OpenSearch Service provisiona todos os recursos do seu OpenSearch cluster e o executa. Ele também detecta e substitui automaticamente os nós de OpenSearch serviço com falha, reduzindo a sobrecarga associada às infraestruturas autogerenciadas. Você pode dimensionar seu cluster com uma única chamada de API ou alguns cliques no console.



Para começar a usar o OpenSearch Service, você cria um domínio OpenSearch Service, que é equivalente a um OpenSearch cluster. Cada instância do EC2 no cluster atua como um nó OpenSearch de serviço.

Você pode usar o console OpenSearch de serviço para instalar e configurar um domínio em minutos. Se preferir o acesso programático, você pode usar o AWS CLI, o ou o AWS SDKsTerraform.

Características do Amazon OpenSearch Service

OpenSearch O serviço inclui os seguintes recursos:

Dimensionar

- Várias configurações de CPU, memória e capacidade de armazenamento conhecidas como tipos de instância, incluindo instâncias do Graviton mais econômicas.
- Compatível com até 1002 nós de dados
- Até 25 PB de armazenamento conectado
- Armazenamento a frio UltraWarme econômico para dados somente para leitura

Segurança

- AWS Identity and Access Management Controle de acesso (IAM)
- Integração fácil à Amazon VPC e aos grupos de segurança da VPC
- Criptografia de dados em repouso e node-to-node criptografia
- Amazon Cognito, HTTP basic ou autenticação SAML para painéis OpenSearch
- Segurança no nível do índice, no nível do documento e no nível do campo
- Logs de auditoria
- Multilocação do Dashboards

Estabilidade

- Vários locais geográficos para os recursos, conhecidos como regiões e zonas de disponibilidade
- A alocação de nós em duas ou três zonas de disponibilidade na mesma região da AWS , recurso conhecido como Multi-AZ
- Nós principais dedicados para descarregar tarefas de gerenciamento de cluster
- Instantâneos automatizados para fazer backup e restaurar domínios OpenSearch de serviço

Flexibilidade

- Suporte SQL para a integração com aplicativos de business intelligence (BI)
- Pacotes personalizados para melhorar os resultados da pesquisa

Integração com serviços populares

- Visualização de dados usando painéis OpenSearch
- Integração com a Amazon CloudWatch para monitorar métricas OpenSearch de domínio do serviço e definir alarmes
- Integração com chamadas de API de configuração AWS CloudTrail para auditoria para domínios OpenSearch de serviço
- Integração com Amazon S3, Amazon Kinesis e Amazon DynamoDB para carregar dados de streaming no Serviço OpenSearch
- Alertas do Amazon SNS quando os dados excedem determinados limites

Quando usar OpenSearch versus Amazon OpenSearch Service

Use a tabela a seguir para ajudá-lo a decidir se o Amazon OpenSearch Service provisionado ou autogerenciado OpenSearch é a escolha correta para você.

OpenSearch	OpenSearch Serviço Amazon
• Sua organização está disposta e tem pessoal habilitado a monitorar e manter manualmente clusters autoprovisionados. • Você quer ter controle total do código no nível da compilação. • Sua organização prefere, ou usa exclusivamente, software de código aberto. • Você tem uma estratégia multinuvem, o que exige tecnologias não específicas de fornecedor. • Sua equipe tem capacidade de resolver qualquer problema crítico de produção. • Você quer ter a flexibilidade de usar, modificar e estender o produto como quiser.	• Você não quer gerenciar, monitorar e manter a infraestrutura manualmente. • Você quer ter maneiras simples de gerenciar os custos crescentes de analytics, distribuindo os dados entre diversos níveis de armazenamento, aproveitando a durabilidade e o baixo custo do Amazon S3. • Você quer aproveitar as integrações com outros, Serviços da AWS como DynamoDB, Amazon DocumentDB (com compatibilidade com MongoDB), IAM e. CloudWatch CloudFormation • Você quer acesso fácil à assistência fornecida Suporte para manutenção preventiva e durante problemas de produção. • Você quer aproveitar atributos como autorrecuperação, manutenção proativa, resiliência e backups.

OpenSearch	OpenSearch Serviço Amazon
• Você quer ter acesso imediato a novos recursos assim que forem lançados.	

Versões compatíveis do Elasticsearch e OpenSearch

OpenSearch O serviço oferece suporte às seguintes versões do OpenSearch:

- 3,5, 3,3, 3,1, 2,19, 2,17, 2,13, 2,11, 2,9, 2,7, 2,5, 2,3, 1,3, 1,2, 1,1 e 1,0

OpenSearch O serviço oferece suporte às seguintes versões do Elasticsearch legado:

- 7.10, 7.9, 7.8, 7.7, 7.4, 7.1, 6.8, 6.7, 6.5, 6.4, 6.3, 6.2, 6.0, 5.6, 5.5, 5.3, 5.1, 2.3 e 1.5

Recomendamos atualizar para a OpenSearch versão mais recente disponível para obter o melhor uso do OpenSearch Serviço, em termos de preço-desempenho, riqueza de recursos e melhorias de segurança.

Suporte padrão e estendido

AWS fornece correções de bugs e atualizações de segurança para versões com suporte padrão. Para versões com suporte estendido, AWS oferece correções de segurança críticas por pelo menos 12 meses após o término do suporte padrão, com uma taxa fixa por Hora de Instância Normalizada (NIH). A NIH é baseada no tamanho da instância e nas horas de uso.

As taxas de suporte estendido são aplicadas automaticamente quando um domínio executa uma versão que não é mais coberta pelo suporte padrão. Para evitar essas cobranças, atualize para uma versão compatível.

As tabelas a seguir mostram o fim do cronograma de suporte OpenSearch e as versões legadas do Elasticsearch.

OpenSearch O serviço oferece suporte a várias versões OpenSearch e versões legadas do Elasticsearch de código aberto. Já publicamos as datas de fim do suporte padrão e estendido para algumas versões. Recomendamos que você atualize para a OpenSearch versão mais recente disponível para obter o melhor uso do OpenSearch Serviço em termos de preço-desempenho,

riqueza de recursos e melhorias de segurança. As tabelas a seguir fornecem listas do Elasticsearch e das OpenSearch versões e seus cronogramas de suporte.

O cronograma de fim de suporte para as versões do Elasticsearch é o seguinte:

Versão do software	Fim do suporte padrão	Fim do suporte estendido
Elasticsearch versões 1.5 e 2.3	7 de novembro de 2025	7 de novembro de 2026
Elasticsearch versões 5.1 a 5.5	7 de novembro de 2025	7 de novembro de 2026
Elasticsearch versão 5.6	7 de novembro de 2025	7 de novembro de 2028
Elasticsearch versões 6.0 a 6.7	7 de novembro de 2025	7 de novembro de 2026
Elasticsearch versão 6.8	Não anunciado	Não anunciado
Elasticsearch versões 7.1 a 7.8	7 de novembro de 2025	7 de novembro de 2026
Elasticsearch versão 7.9	Não anunciado	Não anunciado

Versão do software	Fim do suporte padrão	Fim do suporte estendido
Elasticsearch versão 7.10	Não anunciado	Não anunciado

O cronograma de fim do suporte para OpenSearch as versões é o seguinte:

Versão do software	Fim do suporte padrão	Fim do suporte estendido
OpenSearch versões 1.0 a 1.2	7 de novembro de 2025	7 de novembro de 2026
OpenSearch versões 1.3	Não anunciado	Não anunciado
OpenSearch versões 2.3 a 2.9	7 de novembro de 2025	7 de novembro de 2026
OpenSearch versões 2.11 e versões superiores	Não anunciado	Não anunciado

Suporte padrão e suporte estendido do OpenSearch Elasticsearch

AWS fornece correções de bugs e atualizações de segurança regulares para as versões cobertas pelo Standard Support. Para versões incluídas no Suporte Estendido, a AWS fornece correções de segurança críticas por um período de pelo menos 12 meses após o término do suporte padrão, mediante uma taxa adicional fixa por cada hora de instância normalizada (NIH). A NIH é calculada

como um fator do tamanho da instância (por exemplo, média, grande) e do número de horas da instância (consulte a seção de cálculo de tarifas de suporte estendido abaixo para ver um exemplo). Tarifas de suporte estendido se aplicam automaticamente quando um domínio executa uma versão cujo suporte padrão terminou. Você pode atualizar para uma versão recente ainda coberta pelo suporte padrão para evitar os custos do suporte estendido. Para saber mais sobre os custos do suporte estendido, consulte a página de preços. Para obter informações gerais sobre o suporte estendido, consulte Perguntas frequentes sobre suporte estendido.

Cálculo de custos do suporte estendido

Aos domínios que executam versões sob suporte estendido será cobrada uma Hora de fee/ Normalized Instância (NIH) adicional fixa, por exemplo, 0,0065 USD na região Leste dos EUA (Norte da Virgínia). A NIH é calculada como um fator do tamanho da instância (por exemplo, média, grande) e do número de horas da instância. Por exemplo, se executar uma instância m7g.medium.search por 24 horas na região Leste dos EUA (Norte da Virgínia), com preço de USD 0,068/hora de instância (sob demanda), você normalmente pagará USD 1,632 (0,068 x 24). Se executar uma versão com suporte estendido, você pagará um adicional de USD 0,0065/NIH, que é calculado como USD 0,0065 x 24 (número de horas de instância) x 2 (fator de normalização de tamanho; 2 para instâncias de médio porte), o que equivale a USD 0,312 de suporte estendido por 24 horas. O valor total que você pagará por 24 horas será a soma do custo de uso da instância padrão e do custo do suporte estendido, que é de USD 1,944 (USD 1,632 + USD 0,312). A tabela abaixo mostra o fator de normalização para vários tamanhos de instância no OpenSearch Service.

Tamanho da instância	Fator de normalização	
nano	0,25	
micro	0,5	
pequeno	1	
médio	2	
grande	4	
xlarge	8	

Tamanho da instância	Fator de normalização	
2xlarge	16	
4xlarge	32	
8xlarge	64	
9xlarge	72	
10xlarge	80	
12xlarge	96	
16xlarge	128	
18xlarge	144	
24xlarge	192	
32xlarge	256	

Preços do Amazon OpenSearch Service

Pelo OpenSearch Serviço, você paga por cada hora de uso de uma instância do EC2 e pelo tamanho cumulativo de qualquer volume de armazenamento do EBS anexado às suas instâncias. Taxas padrão AWS de transferência de dados também se aplicam.

No entanto, existem algumas exceções notáveis de transferência de dados. Se um domínio usa várias zonas de disponibilidade, o OpenSearch serviço não cobra pelo tráfego entre as zonas de disponibilidade. Uma transferência significativa de dados ocorre dentro de um domínio durante a alocação e o rebalanceamento de fragmentos. OpenSearch Não pague medidores nem contas para esse tráfego. Da mesma forma, o OpenSearch Serviço não cobra pela transferência de dados entre UltraWarm/cold nodes e o Amazon S3.

Para obter detalhes completos sobre preços, consulte os preços OpenSearch do Amazon Service. Para obter informações sobre encargos incorridos durante as alterações de configuração, consulte the section called “Cobranças para alterações de configuração”.

Serviços relacionados

OpenSearch O serviço geralmente é usado com os seguintes serviços:

Amazon CloudWatch

OpenSearch Os domínios de serviço enviam métricas automaticamente para CloudWatch que você possa monitorar a integridade e o desempenho do domínio. Para obter mais informações, consulte Monitorando métricas de OpenSearch cluster com a Amazon CloudWatch.

CloudWatch Os troncos também podem ir na outra direção. Você pode configurar o CloudWatch Logs para transmitir dados ao OpenSearch Serviço para análise. Para saber mais, consulte the section called “Carregando dados de streaming da Amazon CloudWatch”.

AWS CloudTrail

Use AWS CloudTrail para obter um histórico das chamadas da API de configuração do OpenSearch serviço e dos eventos relacionados à sua conta. Para obter mais informações, consulte Monitorando chamadas OpenSearch de API do Amazon Service com AWS CloudTrail.

Amazon Kinesis

O Kinesis é um serviço totalmente gerenciado para processamento em tempo real de dados de streaming em altíssima escala. Para saber mais, consulte the section called “Carregamento dados de transmissão do Amazon Kinesis Data Streams” e the section called “Carregamento de dados de transmissão do Amazon Data Firehose”.

Amazon S3

O Amazon Simple Storage Service (Amazon S3) fornece armazenamento para a Internet. Esse guia oferece código de exemplo do Lambda para integração com o Amazon S3. Para obter mais informações, consulte the section called “Carregamento de dados de transmissão do Amazon S3”.

AWS IAM

AWS Identity and Access Management (IAM) é um serviço da web que você pode usar para gerenciar o acesso aos seus domínios OpenSearch de serviço. Para obter mais informações, consulte the section called “Gerenciamento de Identidade e Acesso”.

AWS Lambda

AWS Lambda é um serviço de computação que permite executar código sem provisionar ou gerenciar servidores. Esse guia fornece código de exemplo do Lambda para transmitir dados do DynamoDB, Amazon S3 e Kinesis Para saber mais, consulte the section called “Carregando dados de streaming no OpenSearch Serviço”.

Amazon DynamoDB

O Amazon DynamoDB é um serviço de banco de dados NoSQL totalmente gerenciado que fornece uma performance rápida e previsível com escalabilidade integrada. Para saber mais sobre streaming de dados para o OpenSearch Serviço, consultethe section called “Carregamento de dados de transmissão do Amazon DynamoDB”.

Amazon Quick

Você pode visualizar dados do OpenSearch Serviço usando painéis rápidos. Para obter mais informações, consulte Usando o Amazon OpenSearch Service com o Quick no Guia rápido do usuário.

> **Note**
>
> OpenSearch inclui determinados códigos Elasticsearch licenciados pela Apache da Elasticsearch B.V. e outros códigos-fonte. O Elasticsearch B.V. não é a fonte desse outro código-fonte. ELASTICSEARCH é uma marca registrada da Elasticsearch B.V.

Configurando o Amazon OpenSearch Service

Conceder permissões

Em ambientes de produção, recomendamos que você use políticas mais refinadas. Para saber mais sobre o gerenciamento de acesso, consulte Gerenciamento de acesso para AWS recursos no Guia do usuário do IAM.

Para conceder acesso, adicione as permissões aos seus usuários, grupos ou perfis:

- Usuários e grupos em Centro de Identidade do AWS IAM:

  Crie um conjunto de permissões. Siga as instruções em Criação de um conjunto de permissões no Guia do usuário do Centro de Identidade do AWS IAM .
- Usuários gerenciados no IAM com provedor de identidades:

  Crie um perfil para a federação de identidades. Siga as instruções em Criando um perfil para um provedor de identidades de terceiros (federação) no Guia do Usuário do IAM.
- Usuários do IAM:
  - Crie um perfil que seu usuário possa assumir. Siga as instruções em Criação de um perfil para um usuário do IAM no Guia do usuário do IAM.
  - (Não recomendado) Vincule uma política diretamente a um usuário ou adicione um usuário a um grupo de usuários. Siga as instruções em Adição de permissões a um usuário (console) no Guia do usuário do IAM.

Conceder acesso programático

Os usuários precisam de acesso programático se quiserem interagir com pessoas AWS fora do Console de gerenciamento da AWS. A forma de conceder acesso programático depende do tipo de usuário que está acessando AWS.

Para conceder acesso programático aos usuários, selecione uma das seguintes opções:

Qual usuário precisa de acesso programático?	Para	Por
IAM	(Recomendado) Use as credenciais do console como credenciais temporárias para assinar solicitações programáticas para o AWS CLI, AWS SDKs ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para o AWS CLI, consulte Login para desenvolvimento AWS local no Guia AWS Command Line Interface do usuário. • Para AWS SDKs isso, consulte Login para desenvolvimento AWS local no Guia de referência de ferramentas AWS SDKs e ferramentas.
Identidade da força de trabalho (Usuários gerenciados no Centro de Identidade do IAM)	Use credenciais temporárias para assinar solicitações programáticas para o AWS CLI AWS SDKs, ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para o AWS CLI, consulte Configurando o AWS CLI para uso Centro de Identidade do AWS IAM no Guia do AWS Command Line Interface usuário. • Para AWS SDKs, ferramentas e AWS APIs, consulte a autenticação do IAM Identity Center no Guia de referência de ferramentas AWS SDKs e ferramentas.
IAM	Use credenciais temporárias para assinar solicitações programáticas para o AWS	Siga as instruções em Como usar credenciais temporárias

Qual usuário precisa de acesso programático?	Para	Por
	CLI AWS SDKs, ou. AWS APIs	com AWS recursos no Guia do usuário do IAM.
IAM	(Não recomendado) Use credenciais de longo prazo para assinar solicitaç ões programáticas para o AWS CLI, AWS SDKs, ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para isso AWS CLI, consulte Autenticação usando credenciais de usuário do IAM no Guia do AWS Command Line Interface usuário. • Para ferramentas AWS SDKs e ferramentas, consulte Autenticar usando credenciais de longo prazo no Guia de referência de ferramentas AWS SDKs e ferramentas. • Para isso AWS APIs, consulte Gerenciamento de chaves de acesso para usuários do IAM no Guia do usuário do IAM.

Instale e configure o AWS CLI

Se quiser usar o OpenSearch Serviço APIs, você deve instalar a versão mais recente do AWS Command Line Interface (AWS CLI). Você não precisa AWS CLI usar o OpenSearch Service do console e pode começar sem a CLI seguindo as etapas em. Começando a usar o Amazon OpenSearch Service

Para configurar o AWS CLI

1. Para instalar a versão mais recente do AWS CLI para macOS, Linux ou Windows, consulte Instalando ou atualizando a versão mais recente do. AWS CLI
2. Para configurar AWS CLI e proteger seu acesso ao Serviço Serviços da AWS, incluindo o OpenSearch Serviço, consulte Configuração rápida com `aws configure`.
3. Para verificar a configuração, digite o seguinte DataBrew comando no prompt de comando.

   ```
   aws opensearch help
   ```

   AWS CLI os comandos usam o padrão Região da AWS da sua configuração, a menos que você o defina com um parâmetro ou um perfil. Para definir seu Região da AWS com um parâmetro, você pode adicionar o `--region` parâmetro a cada comando.

   Para definir seu Região da AWS com um perfil, primeiro adicione um perfil nomeado no `~/.aws/config` arquivo ou no `%UserProfile%/.aws/config` arquivo (para Microsoft Windows). Siga as etapas em Perfis nomeados para a AWS CLI. Em seguida, defina suas Região da AWS e outras configurações com um comando semelhante ao do exemplo a seguir.

   ```
   [profile opensearch]
   aws_access_key_id = ACCESS-KEY-ID-OF-IAM-USER
   aws_secret_access_key = SECRET-ACCESS-KEY-ID-OF-IAM-USER
   region = us-east-1
   output = text
   ```

Abra o console do .

A maioria dos tópicos orientados ao console nesta seção começa no console de serviço. OpenSearch Se você ainda não estiver conectado ao seu Conta da AWS, faça login, abra o console de OpenSearch serviço e vá para a próxima seção para continuar começando a usar o OpenSearch Serviço.

Começando a usar o Amazon OpenSearch Service

Para começar, cadastre-se em uma Conta da AWS, se ainda não tiver uma. Depois de configurar uma conta, conclua o tutorial de introdução do Amazon OpenSearch Service. Enquanto você se informa sobre o serviço, consulte os tópicos introdutórios a seguir se precisar de mais informações:

- Crie um domínio.
- Dimensione o domínio de forma apropriada para sua workload.
- Controle o acesso ao seu domínio usando uma política de acesso ao domínio ou um controle de acesso refinado.
- Indexe dados manualmente ou de outros AWS serviços.
- Use OpenSearch painéis para pesquisar seus dados e criar visualizações.
- Saiba mais sobre opções mais avançadas para criar um domínio. Para saber mais, consulte Criação e gerenciamento de domínios.
- Descubra como gerenciar os índices em seu domínio. Para obter mais informações, consulte Gerenciamento de Índices.
- Experimente um dos tutoriais para trabalhar com o Amazon OpenSearch Service. Para obter mais informações, consulte Tutoriais.

Para obter informações sobre como migrar para o OpenSearch Service a partir de um OpenSearch cluster autogerenciado, consulte. the section called “Migrando para o serviço OpenSearch ”

Para obter informações mais detalhadas, consulte Criação e gerenciamento de domínios e outros tópicos neste guia. Para obter informações sobre como migrar para o OpenSearch Service a partir de um OpenSearch cluster autogerenciado, consulte. the section called “Migrando para o serviço OpenSearch ”

Você pode concluir as etapas a seguir usando o console OpenSearch de serviço AWS CLI, o ou o AWS SDK. Para obter informações sobre como instalar e configurar o AWS CLI, consulte o Guia AWS Command Line Interface do usuário.

Crie um domínio do Amazon OpenSearch Service

> **Important**
>
> Este é um tutorial conciso para configurar um domínio de teste do Amazon OpenSearch Service. Não use esse processo para criar domínios de produção. Para obter uma versão abrangente do mesmo processo, consulte Criação e gerenciamento de domínios.

Um domínio OpenSearch de serviço é sinônimo de um OpenSearch cluster. Domínios são clusters com configurações, tipos de instância, contagens de instâncias e recursos de armazenamento especificados por você. Você pode criar um domínio de OpenSearch serviço usando o console AWS CLI, o ou AWS SDKs o.

Para criar um domínio OpenSearch de serviço usando o console

1. Acesse https://aws.amazon.come escolha Entrar no console.
2. Em Analytics, escolha Amazon OpenSearch Service.
3. Escolha Criar domínio.
4. Informe um nome para o domínio. Os exemplos neste tutorial usam o nome movies.
5. Como método de criação de domínio, escolha Criação padrão.

   > **Note**
   >
   > Para configurar rapidamente um domínio de produção com as melhores práticas, você pode escolher Criação fácil. Para fins de desenvolvimento e teste deste tutorial, usaremos a Criação padrão.

6. Para modelos, escolha dev/teste.
7. Para a opção de implantação, escolha Domínio com modo de espera.
8. Em Versão, escolha a versão mais recente.
9. Por enquanto, ignore as seções Nós de dados, Armazenamento de dados quentes e mornos, Nós principais dedicados, Configuração de snapshot e Endpoint personalizado.
10. Para simplificar este tutorial, use um domínio de acesso público. Sob Rede, selecione Acesso público.

11. Nas configurações de controle de acesso detalhado, mantenha a caixa de seleção Habilitar o controle de acesso refinado. Selecione Criar usuário primário e forneça um nome de usuário e senha.
12. Por enquanto, ignore as seções Autenticação SAML e Autenticação do Amazon Cognito.
13. Para Política de acesso), escolha Use somente controle de acesso refinado. Neste tutorial, o controle de acesso refinado processa a autenticação, não a política de acesso ao domínio.
14. Ignore o restante das configurações e escolha Criar. Os novos domínios normalmente levam de 15 a 30 minutos para inicializar, mas podem demorar mais dependendo da configuração. Após a inicialização do domínio, selecione-o para abrir o painel de configuração. Anote o endpoint do domínio em Informações gerais (p. ex., `https://search-`*`my-domain`*`.us-east-1.es.amazonaws.com`), você vai usá-lo na próxima etapa.

Próximo: Carregar dados em um domínio OpenSearch de serviço para indexação

Faça upload de dados para o Amazon OpenSearch Service para indexação

> **Important**
>
> Este é um tutorial conciso para fazer o upload de uma pequena quantidade de dados de teste para o Amazon OpenSearch Service. Para saber mais sobre como carregar dados em um domínio de produção, consulte Indexação de dados.

Você pode carregar dados para um domínio de OpenSearch serviço usando a linha de comando ou a maioria das linguagens de programação.

Os exemplos de solicitações a seguir usam curl, um cliente HTTP muito comum, para proporcionar agilidade e conveniência. Os clientes como o curl não podem executar a assinatura de solicitações exigida se as políticas de acesso especificam usuários ou funções do IAM. Para concluir esse processo com êxito, você deverá usar o controle de acesso refinado com um nome de usuário primário e uma senha, conforme configurados na Etapa 1.

Você pode instalar o curl no Windows e usá-lo no prompt de comando, mas recomendamos usar uma ferramenta como Cygwin ou o Windows Subsystem for Linux. O macOS e a maioria das distribuições do Linux já vêm com curl pré-instalado.

Opção 1: Carregar um único documento

Execute o comando a seguir para adicionar um único documento ao domínio movies:

```
curl -XPUT -u 'master-user:master-user-password' 'domain-endpoint/movies/_doc/1' -d
 '{"director": "Burton, Tim", "genre": ["Comedy","Sci-Fi"], "year": 1996, "actor":
 ["Jack Nicholson","Pierce Brosnan","Sarah Jessica Parker"], "title": "Mars Attacks!"}'
 -H 'Content-Type: application/json'
```

No comando, forneça o nome do usuário e a senha que você criou na Etapa 1.

Para obter uma explicação detalhada desse comando e de como fazer solicitações assinadas ao OpenSearch Service, consulteIndexação de dados.

Opção 2: carregar vários documentos

Para carregar um arquivo JSON que contém vários documentos em um domínio de OpenSearch serviço

1. Crie um arquivo local chamado `bulk_movies.json`. Copie e cole o seguinte conteúdo no arquivo, adicionando uma nova linha no final:

```
{ "index" : { "_index": "movies", "_id" : "2" } }
{"director": "Frankenheimer, John", "genre": ["Drama", "Mystery", "Thriller",
 "Crime"], "year": 1962, "actor": ["Lansbury, Angela", "Sinatra, Frank", "Leigh,
 Janet", "Harvey, Laurence", "Silva, Henry", "Frees, Paul", "Gregory, James",
 "Bissell, Whit", "McGiver, John", "Parrish, Leslie", "Edwards, James", "Flowers,
 Bess", "Dhiegh, Khigh", "Payne, Julie", "Kleeb, Helen", "Gray, Joe", "Nalder,
 Reggie", "Stevens, Bert", "Masters, Michael", "Lowell, Tom"], "title": "The
 Manchurian Candidate"}
{ "index" : { "_index": "movies", "_id" : "3" } }
{"director": "Baird, Stuart", "genre": ["Action", "Crime", "Thriller"], "year":
 1998, "actor": ["Downey Jr., Robert", "Jones, Tommy Lee", "Snipes, Wesley",
 "Pantoliano, Joe", "Jacob, Ir\u00e8ne", "Nelligan, Kate", "Roebuck, Daniel",
 "Malahide, Patrick", "Richardson, LaTanya", "Wood, Tom", "Kosik, Thomas",
 "Stellate, Nick", "Minkoff, Robert", "Brown, Spitfire", "Foster, Reese",
 "Spielbauer, Bruce", "Mukherji, Kevin", "Cray, Ed", "Fordham, David", "Jett,
 Charlie"], "title": "U.S. Marshals"}
{ "index" : { "_index": "movies", "_id" : "4" } }
{"director": "Ray, Nicholas", "genre": ["Drama", "Romance"], "year": 1955, "actor":
 ["Hopper, Dennis", "Wood, Natalie", "Dean, James", "Mineo, Sal", "Backus, Jim",
 "Platt, Edward", "Ray, Nicholas", "Hopper, William", "Allen, Corey", "Birch,
```

```
 Paul", "Hudson, Rochelle", "Doran, Ann", "Hicks, Chuck", "Leigh, Nelson",
 "Williams, Robert", "Wessel, Dick", "Bryar, Paul", "Sessions, Almira", "McMahon,
 David", "Peters Jr., House"], "title": "Rebel Without a Cause"}
```

2. Execute o comando a seguir no diretório local em que o arquivo está armazenado para carregar para o domínio movies:

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/movies/_bulk' --
data-binary @bulk_movies.json -H 'Content-Type: application/x-ndjson'
```

Para saber mais sobre o formato de arquivo em massa, consulte Indexação de dados.

Próximo: Pesquisar documentos

Pesquise documentos no Amazon OpenSearch Service

Para pesquisar documentos em um domínio do Amazon OpenSearch Service, use a API OpenSearch de pesquisa. Como alternativa, você pode usar OpenSearch painéis para pesquisar documentos no domínio.

Para pesquisar documentos via linha de comando

Execute o comando a seguir para realizar uma pesquisa no domínio movies usando a palavra mars:

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/movies/_search?
q=mars&pretty=true'
```

Se você usou dados em massa na página anterior, tente pesquisar rebeldes.

Você verá uma resposta semelhante à seguinte:

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
```

```
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 0.2876821,
    "hits" : [
      {
        "_index" : "movies",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 0.2876821,
        "_source" : {
          "director" : "Burton, Tim",
          "genre" : [
            "Comedy",
            "Sci-Fi"
          ],
          "year" : 1996,
          "actor" : [
            "Jack Nicholson",
            "Pierce Brosnan",
            "Sarah Jessica Parker"
          ],
          "title" : "Mars Attacks!"
        }
      }
    ]
  }
}
```

Pesquise documentos usando OpenSearch painéis

OpenSearch O Dashboards é uma ferramenta popular de visualização de código aberto projetada para trabalhar com. OpenSearch Ele fornece uma interface de usuário útil para você pesquisar e monitorar seus índices.

Para pesquisar documentos de um domínio OpenSearch de serviço usando painéis

1. Navegue até o URL dos OpenSearch painéis do seu domínio. Você pode encontrar o URL no painel do domínio no console OpenSearch de serviço. O URL segue este formato:

   ```
   domain-endpoint/_dashboards/
   ```

2. Faça login usando o nome de usuário principal e a respectiva senha.
3. Para usar o Dashboards, é necessário criar pelo menos um padrão de índice. O Dashboards usa esses padrões para identificar quais índices você deseja analisar. Abra o menu esquerdo de navegação, escolha Gerenciamento de pilhas, escolha Padrões de índice e, em seguida, escolha Criar padrão de índice. Para este tutorial, insira movies.
4. Escolha Próxima etapa e, em seguida, Criar padrão de índice. Depois que o padrão é criado, você pode visualizar os vários campos do documento, como `actor` e `director`.
5. Volte para a página Padrões de índice e verifique se `movies` está definido como o valor padrão. Caso não esteja, selecione o padrão e escolha o ícone de estrela para torná-lo o valor padrão.
6. Para começar a pesquisar seus dados, abra novamente o menu esquerdo de navegação e escolha Descobrir.
7. Na barra de pesquisa, insira mars se você carregou um único documento, ou rebel se você carregou vários documentos. Em seguida, pressione Enter. Você pode tentar pesquisar outros termos, como nomes de atores ou diretores.

Próximo: Excluir um domínio

Excluir um domínio do Amazon OpenSearch Service

Como o domínio movies deste tutorial é usado apenas para fins de teste, você deverá excluí-lo quando terminar os testes para evitar cobranças.

Para excluir um domínio de OpenSearch serviço do console

1. Faça login no console do Amazon OpenSearch Service.
2. Sob Domínios, selecione o domínio movies (filmes).
3. Escolha Excluir e confirme a exclusão.

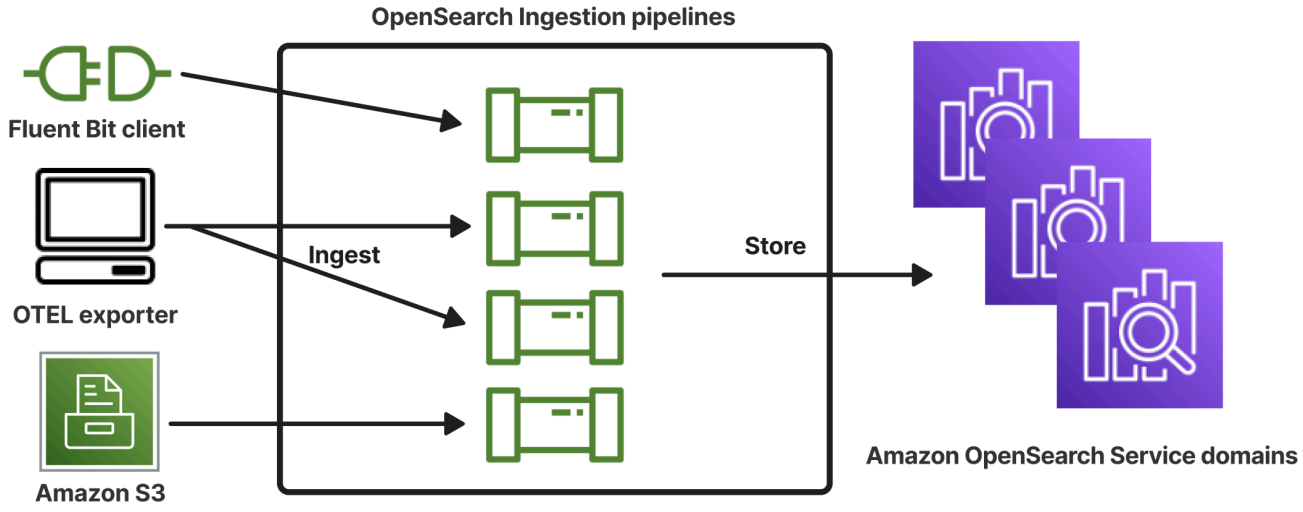
Visão geral da OpenSearch ingestão da Amazon

O Amazon OpenSearch Ingestion é um coletor de dados totalmente gerenciado e sem servidor que transmite registros, métricas e dados de rastreamento em tempo real para domínios do Amazon OpenSearch Service e coleções sem servidor. OpenSearch

Com a OpenSearch ingestão, você não precisa mais de ferramentas de terceiros, como Logstash ou Jaeger, para ingerir dados. Você configura seus produtores de dados para enviar dados para o OpenSearch Ingestion, e ele os entrega automaticamente ao seu domínio ou coleção especificado. Você também pode transformar os dados antes da entrega.

Como a OpenSearch ingestão é feita sem servidor, você não precisa gerenciar a infraestrutura, corrigir o software ou escalar clusters manualmente. Você pode provisionar pipelines de ingestão diretamente no Console de gerenciamento da AWS, e o OpenSearch Ingestion cuida do resto.

Como componente do Amazon OpenSearch Service, o OpenSearch Inestion é desenvolvido pelo Data Prepper, um coletor de dados de código aberto que filtra, enriquece, transforma, normaliza e agrega dados para análise e visualização posteriores.



Conceitos-chave na Amazon OpenSearch Ingestion

Antes de começar a usar o OpenSearch Ingestion, é útil entender esses conceitos-chave.

Pipeline

Do ponto de vista da OpenSearch ingestão, um pipeline se refere a um único coletor de dados provisionado que você cria no Service. OpenSearch Pense nisso como o arquivo de configuração

YAML completo, que inclui um ou mais subpipelines. Para ver as etapas para criar um pipeline de ingestão, consulte the section called “Como criar pipelines”.

Subpipeline

Você define subpipelines em um arquivo de configuração YAML. Cada subpipeline é uma combinação de uma fonte, um buffer, zero ou mais processadores e um ou mais coletores. Você pode definir vários subpipelines em um único arquivo YAML, cada um com fontes, processadores e coletores exclusivos. Para ajudar no monitoramento com CloudWatch e outros serviços, recomendamos que você especifique um nome de pipeline que seja diferente de todos os seus subpipelines.

Você pode agrupar vários subpipelines em um único arquivo YAML, de forma que a origem de um subpipeline seja outro subpipeline e seu coletor seja um terceiro subpipeline. Para ver um exemplo, consulte the section called “OpenTelemetry Colecionador”.

Fonte

O componente de entrada de um subpipeline. Ele define o mecanismo pelo qual um pipeline consome registros. A fonte pode consumir eventos recebendo-os por HTTPS ou lendo em endpoints externos, como o Amazon S3. Existem dois tipos de fontes: baseadas em push e baseadas em pull. Fontes baseadas em push, como HTTP e OTel registros, transmitem registros para endpoints de ingestão. Fontes baseadas em pull, como OTel trace e S3, extraem dados da fonte.

Processadores

Unidades de processamento intermediárias que podem filtrar, transformar e enriquecer registros no formato desejado antes de publicá-los no coletor. O processador é um componente opcional de um pipeline. Se você não definir um processador, os registros serão publicados no formato definido na fonte. Você pode usar mais de um processador. Um pipeline executa os processadores na ordem em que são definidos.

Sink

O componente de saída de um subpipeline. Ele define um ou mais destinos nos quais um subpipeline publica registros. OpenSearch A ingestão oferece suporte a domínios OpenSearch de serviço como coletores. Ele também é compatível com subtubulações como coletores. Isso significa que você pode agrupar vários subpipelines em um único pipeline de OpenSearch ingestão (arquivo YAML). OpenSearch Clusters autogerenciados não são suportados como coletores.

Buffer

A parte de um processador que atua como a camada entre a fonte e o coletor. Você não pode configurar um buffer no seu pipeline manualmente. OpenSearch A ingestão usa uma configuração de buffer padrão.

Rota

A parte de um processador que permite que os autores do pipeline enviem somente eventos que correspondam a determinadas condições para diferentes coletores.

Uma definição de subpipeline válida deve conter uma fonte e um coletor. Para saber mais sobre cada um desses elementos do pipeline, consulte a referência de configuração.

Benefícios da OpenSearch ingestão da Amazon

OpenSearch A ingestão tem os seguintes benefícios principais:

- Elimina a necessidade de gerenciar manualmente um pipeline autoprovisionado.
- Dimensiona automaticamente seus pipelines com base nos limites de capacidade definidos por você.
- Mantém seu pipeline atualizado com correções de segurança e bugs.
- Oferece a opção de conectar pipelines à sua nuvem privada virtual (VPC) para uma camada adicional de segurança.
- Permite que você pare e inicie pipelines para controlar os custos.
- Fornece esquemas de configuração de pipeline para casos de uso populares para ajudar você a começar a trabalhar com mais rapidez.
- Permite que você interaja programaticamente com seus pipelines por meio dos vários AWS SDKs e da OpenSearch API de ingestão.
- Oferece suporte ao monitoramento de desempenho na Amazon CloudWatch e ao registro de erros no CloudWatch Logs.

Limitações da OpenSearch ingestão da Amazon

OpenSearch A ingestão tem as seguintes limitações:

- Você só pode ingerir dados em domínios que executam OpenSearch 1.0 ou posterior, ou Elasticsearch 6.8 ou posterior. Se você estiver usando a fonte de OTel rastreamento, recomendamos usar o Elasticsearch 7.9 ou posterior para poder usar o OpenSearch plug-in Dashboards.
- Se um pipeline estiver gravando em um domínio de OpenSearch serviço que esteja dentro de uma VPC, o pipeline deverá ser criado da Região da AWS mesma forma que o domínio.
- Você pode configurar uma única fonte de dados dentro de uma definição de pipeline.
- Você não pode especificar OpenSearch clusters autogerenciados como coletores.
- Não é possível especificar um endpoint personalizado como coletor. Você ainda pode gravar em um domínio que tenha endpoints personalizados habilitados, mas deve especificar seu endpoint padrão.
- Você não pode especificar recursos em regiões opcionais como fontes ou coletores.
- Há algumas restrições nos parâmetros que você pode incluir em uma configuração de pipeline. Para saber mais, consulte the section called “Requisitos e restrições de configuração”.

Versões do Data Prepper compatíveis

OpenSearch Atualmente, o Inestion é compatível com as seguintes versões principais do Data Prepper:

- 2.x

Ao criar um pipeline, use ao editor de código, use a opção de `version` necessária para especificar a versão principal do Data Prepper a ser usada. Por exemplo,`version: "2"`. OpenSearch A ingestão recupera a versão secundária mais recente compatível dessa versão principal e provisiona o pipeline com essa versão.

Se você não usa o editor de código para criar seu pipeline, o OpenSearch Ingestion provisiona automaticamente seu pipeline com a versão mais recente compatível.

Atualmente, o OpenSearch Ingestion provisiona pipelines com a versão 2.7 do Data Prepper. Para saber mais, consulte as notas de lançamento da versão 2.7. Nem todas as versões secundárias de uma versão principal específica são suportadas pelo OpenSearch Ingestion.

Quando você atualiza a configuração de um pipeline, se houver suporte para uma nova versão secundária do Data Prepper, o OpenSearch Ingestion atualiza automaticamente o pipeline para

a versão secundária mais recente compatível da versão principal especificada na configuração do pipeline. Por exemplo, você pode ter `version: "2"` em sua configuração de pipeline, e a OpenSearch Ingestion inicialmente provisionou o pipeline com a versão 2.6.0. Quando o suporte para a versão 2.7.0 é adicionado e você faz uma alteração na configuração do pipeline, o OpenSearch Ingestion atualiza o pipeline para a versão 2.7.0. Esse processo mantém seu pipeline atualizado com as últimas correções de bugs e melhorias de desempenho. OpenSearch A ingestão não pode atualizar a versão principal do seu pipeline, a menos que você altere manualmente a `version` opção na configuração do pipeline. Para obter mais informações, consulte the section called “Atualizar pipelines”.

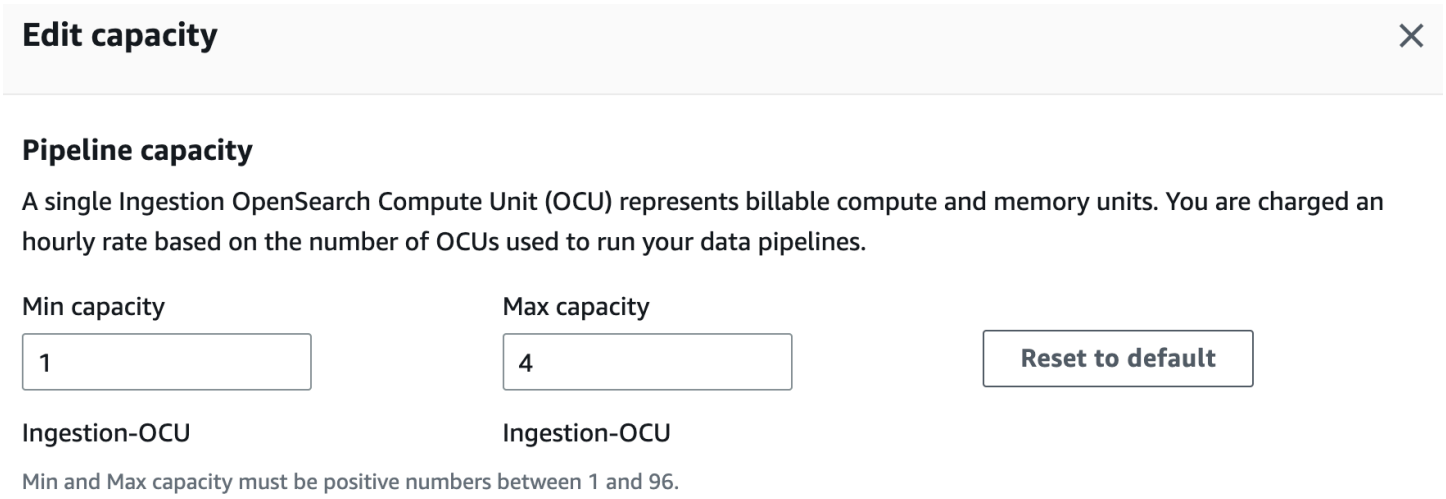
Escalando pipelines na Amazon Ingestion OpenSearch

OpenSearch A ingestão escala automaticamente a capacidade do pipeline com base nas unidades OpenSearch computacionais de ingestão mínimas e máximas especificadas (ingestão). OCUs Isso elimina a necessidade de provisionamento e gerenciamento manuais.

Cada OCU de ingestão é uma combinação de aproximadamente 15 GiB de memória e 2 v. CPUs Você pode especificar os valores mínimo e máximo de OCU para um pipeline, e o OpenSearch Ingestion escala automaticamente a capacidade do pipeline com base nesses limites.

Você especifica os seguintes valores criar um pipeline:

- Capacidade mínima — O pipeline pode reduzir a capacidade até esse número de ingestão OCUs. A capacidade mínima especificada também é a capacidade inicial de uma pipeline.
- Capacidade máxima — O pipeline pode aumentar a capacidade até esse número de ingestão OCUs.



Assegure-se de garantir que a capacidade máxima do pipeline seja alta o suficiente para lidar com picos da workload, e a capacidade mínima seja baixa o suficiente para minimizar os custos quando o pipeline não estiver ocupado. Com base nas suas configurações, o OpenSearch Ingestion escala automaticamente o número de ingestão do seu pipeline OCUs para processar a carga de trabalho de ingestão. Em qualquer momento específico, você é cobrado somente pela ingestão OCUs que está sendo usada ativamente pelo seu funil.

A capacidade alocada para o pipeline de OpenSearch ingestão aumenta e diminui com base nos requisitos de processamento do pipeline e na carga gerada pelo aplicativo cliente. Quando a capacidade é restrita, o OpenSearch Ingestion aumenta alocando mais unidades de computação (GiB de memória). Quando seu pipeline está processando cargas de trabalho menores ou não processando nenhum dado, ele pode ser reduzido até a ingestão OCUs mínima configurada.

Você pode especificar um mínimo de 1 OCU de ingestão, um máximo de 96 ingestão OCUs para pipelines sem estado e um máximo de 48 ingestão para pipelines com estado. OCUs Recomendamos um mínimo de pelo menos 2 ingestões OCUs para fontes baseadas em push. Quando o buffer persistente está ativado, você pode especificar no mínimo 2 e no máximo 384 Ingestão. OCUs

Com um pipeline de log padrão com uma única fonte, um padrão Grok simples e um coletor, cada unidade computacional pode suportar até 2 MiB por segundo. Para pipelines de log mais complexos com vários processadores, cada unidade computacional pode suportar menos carga de ingestão. Com base na capacidade do pipeline e na utilização de recursos, o processo de escalabilidade OpenSearch de ingestão entra em ação.

Para garantir a alta disponibilidade, a ingestão OCUs é distribuída entre as zonas de disponibilidade (AZs). O número de AZs depende da capacidade mínima que você especificar.

Por exemplo, se você especificar um mínimo de 2 unidades de computação, a ingestão OCUs que está em uso a qualquer momento será distribuída uniformemente em 2. AZs Se você especificar um mínimo de 3 ou mais unidades de computação, a ingestão OCUs será distribuída uniformemente em 3. AZs Recomendamos que você provisione pelo menos duas ingestões OCUs para garantir 99,9% de disponibilidade para seus pipelines de ingestão.

Você não é cobrado pela ingestão OCUs quando um funil está nos `Create failed` estados`Creating`,`Deleting`, e. `Stopped`

Para obter instruções sobre como definir e recuperar as configurações de capacidade de um pipeline, consulte the section called “Como criar pipelines”.

OpenSearch Preços de ingestão

Em qualquer momento específico, você paga apenas pelo número de ingestão OCUs alocado a um pipeline, independentemente de haver dados fluindo pelo pipeline. OpenSearch A ingestão acomoda imediatamente suas cargas de trabalho, aumentando ou diminuindo a capacidade do pipeline com base no uso.

Para obter detalhes completos sobre preços, consulte os preços OpenSearch do Amazon Service.

Suportado Regiões da AWS

OpenSearch A ingestão está disponível em um subconjunto Regiões da AWS desse OpenSearch serviço disponível em. Para obter uma lista das regiões suportadas, consulte os endpoints e cotas do Amazon OpenSearch Service no. Referência geral da AWS

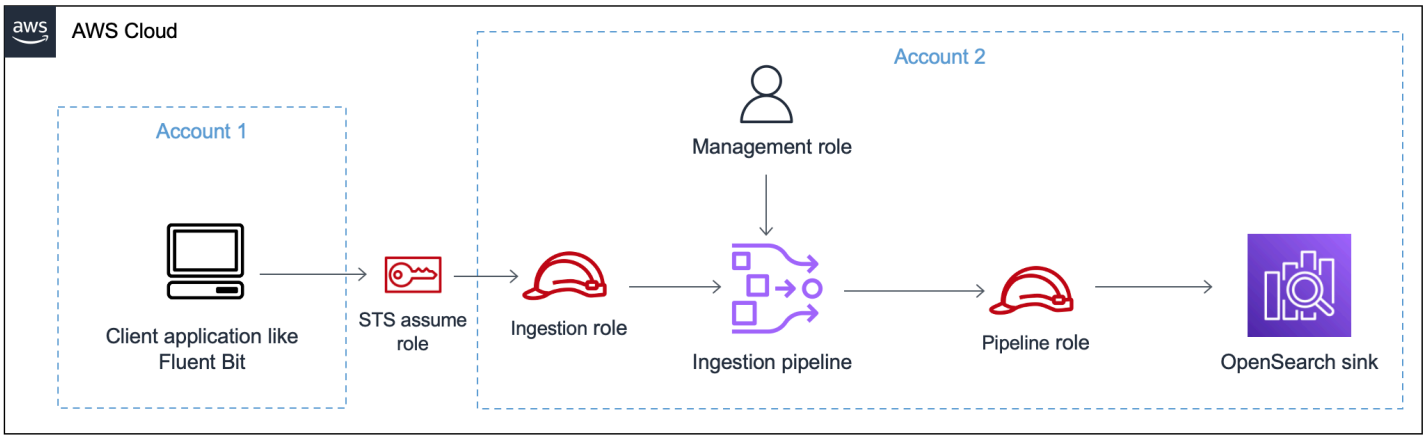
Configurando funções e usuários na Amazon OpenSearch Ingestion

O Amazon OpenSearch Ingestion usa uma variedade de modelos de permissões e funções do IAM para permitir que os aplicativos de origem gravem em pipelines e para permitir que os pipelines gravem em sumidouros. Antes de começar a ingerir dados, você precisa criar um ou mais perfis do IAM com permissões específicas com base no seu caso de uso.

No mínimo, os seguintes perfis são necessários para configurar um pipeline bem-sucedido.

Name (Nome)	Description
Perfis do pipeline	O perfil de pipeline fornece as permissões necessárias para que um pipeline leia na origem e escreva no domínio ou no destino da coleção. Você pode criar manualmente a função do pipeline ou fazer com que o OpenSearch Inestion a crie para você.
Perfil de ingestão	O perfil de ingestão contém a permissão `osis:Ingest` para o recurso de pipeline. Essa permissão permite que fontes baseadas em push consumam dados em um pipeline.

A imagem a seguir demonstra uma configuração típica de pipeline, em que uma fonte de dados, como Amazon S3 ou Fluent Bit, está gravando em um pipeline em uma conta diferente. Nesse caso, o cliente precisa assumir o perfil de ingestão para acessar o pipeline. Para saber mais, consulte the section called “Ingestão entre contas”.



Para obter um guia de configuração simples, consulte the section called “Tutorial: ingerir dados em um domínio”.

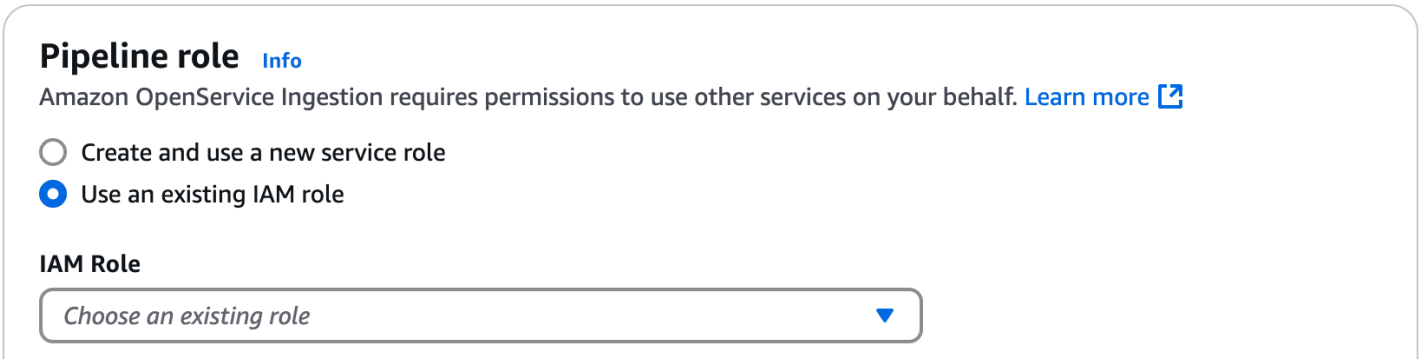
Tópicos

- the section called “Perfis do pipeline”
- the section called “Perfil de ingestão”
- the section called “Ingestão entre contas”

Perfis do pipeline

Um pipeline precisa de certas permissões ler na origem e gravar no coletor. Essas permissões dependem do aplicativo cliente ou do aplicativo AWS service (Serviço da AWS) que está gravando no pipeline e se o coletor é um domínio de OpenSearch serviço, uma coleção OpenSearch sem servidor ou o Amazon S3. Além disso, um pipeline pode precisar de permissões para fisicamente extrair por pull os dados da aplicação de origem (se a origem for um plug-in baseado em pull) e permissões para gravar em uma fila de mensagens não entregues do S3, se habilitada.

Ao criar um pipeline, você tem a opção de especificar uma função existente do IAM que você criou manualmente ou fazer com que o OpenSearch Ingestion crie automaticamente a função do pipeline com base na fonte e no coletor que você selecionou. A imagem a seguir mostra como especificar o perfil do pipeline no Console de gerenciamento da AWS.



Tópicos

- Automatizar a criação de perfil de pipeline
- Criar o perfil de pipeline manualmente

Automatizar a criação de perfil de pipeline

Você pode escolher que o OpenSearch Inestion crie a função de pipeline para você. Ele identifica automaticamente quais permissões o perfil requer com base na origem e nos coletores configurados. Ele cria um perfil do IAM com o prefixo `OpenSearchIngestion-` e com o sufixo que você insere. Por exemplo, se você inserir `PipelineRole` como sufixo, o OpenSearch Ingestion cria uma função chamada. `OpenSearchIngestion-PipelineRole`

A criação automática do perfil de pipeline simplifica o processo de configuração e reduz a probabilidade de erros de configuração. Automatizando a criação de perfis, você pode evitar a atribuição manual de permissões, garantindo que as políticas corretas sejam aplicadas sem correr

o risco de configurações de segurança incorretas. Isso também economiza tempo e melhora a conformidade de segurança aplicando as práticas recomendadas e, ao mesmo tempo, garantindo a consistência entre várias implantações do pipeline.

Você só pode fazer com que o OpenSearch Ingestion crie automaticamente a função do pipeline no Console de gerenciamento da AWS. Se você estiver usando a AWS CLI API de OpenSearch ingestão ou uma das SDKs, deverá especificar uma função de pipeline criada manualmente.

Para que o OpenSearch Inestion crie a função para você, selecione Criar e usar uma nova função de serviço.

> **Important**
>
> Você ainda precisa modificar manualmente a política de acesso ao domínio ou à coleção para conceder acesso ao perfil do pipeline. Para domínios que usam um controle de acesso refinado, você também deve mapear o perfil do pipeline para um perfil de backend. Você pode realizar essas etapas antes ou depois de criar o pipeline.
> Para obter instruções detalhadas, consulte os seguintes tópicos:
>
> - Configurar o acesso a dados para o domínio
> - Configurar dados e acesso à rede para a coleção

Criar o perfil de pipeline manualmente

Talvez você prefira criar manualmente o perfil do pipeline se precisar de mais controle sobre as permissões para atender aos requisitos específicos de segurança ou conformidade. A criação manual permite que você personalize as perfis de acordo com a infraestrutura existente ou com as estratégias de gerenciamento de acesso. Você também pode escolher a configuração manual para integrar a função a outra Serviços da AWS ou garantir que ela esteja alinhada às suas necessidades operacionais exclusivas.

Para escolher um perfil de pipeline criado manualmente, selecione Usar um perfil do IAM existente e escolha um perfil existente. O perfil deve ter todas as permissões necessárias para receber dados da origem selecionada e gravar no coletor selecionado. As seções a seguir descrevem como criar manualmente um perfil de pipeline.

Tópicos

- Permissões para ler uma origem

- Permissões para gravar em um coletor de domínios
- Permissões para gravar em um coletor de coleções
- Permissões para gravar no Amazon S3 ou em uma fila de mensagens não entregues

Permissões para ler uma origem

Um pipeline OpenSearch de ingestão precisa de permissão para ler e receber dados da fonte especificada. Por exemplo, para uma origem do Amazon DynamoDB, ele precisa de permissões como `dynamodb:DescribeTable` e `dynamodb:DescribeStream`. Para exemplos de políticas de acesso à função de pipeline para fontes comuns, como Amazon S3, Fluent Bit e OpenTelemetry Collector, consulte. the section called “Integração de pipelines”

Permissões para gravar em um coletor de domínios

Um pipeline OpenSearch de ingestão precisa de permissão para gravar em um domínio OpenSearch de serviço configurado como coletor. Essas permissões incluem a capacidade de descrever o domínio e enviar solicitações HTTP para ele. Essas permissões são as mesmas para domínios públicos e domínios de VPC. Para obter instruções para criar um perfil de pipeline e especificá-lo na política de acesso ao domínio, consulte Permitir que pipelines acessem domínios.

Permissões para gravar em um coletor de coleções

Um pipeline OpenSearch de ingestão precisa de permissão para gravar em uma coleção OpenSearch Serverless configurada como coletor. Essas permissões incluem a capacidade de descrever a coleção e enviar solicitações HTTP para ela.

Primeiro, certifique-se de que sua política de acesso ao perfil do pipeline conceda as permissões necessárias. Em seguida, inclua esse perfil em uma política de acesso a dados e forneça permissões para criar índices, atualizar índices, descrever índices e escrever documentos na coleção. Para obter instruções sobre como concluir cada uma dessas etapas, consulte Como permitir que os pipelines acessem as coleções.

Permissões para gravar no Amazon S3 ou em uma fila de mensagens não entregues

Se você especificar o Amazon S3 como destino do coletor do pipeline ou se habilitar uma fila de mensagens não entregues (DLQ), o perfil do pipeline deverá permitir que ele acesse o bucket do S3 especificado como destino.

Anexe uma política de permissões separada ao perfil do pipeline que fornece acesso à DLQ. No mínimo, o perfil deve ter permissão para a ação `S3:PutObject` no recurso de bucket:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "WriteToS3DLQ",
      "Effect": "Allow",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::my-dlq-bucket/*"
    }
  ]
}
```

Perfil de ingestão

A função de ingestão é uma função do IAM que permite que serviços externos interajam com segurança e enviem dados para um OpenSearch pipeline de ingestão. Para origens baseadas em push, como o Amazon Security Lake, esse perfil deve conceder permissões para enviar dados ao pipeline por push, incluindo `osis:Ingest`. Para fontes baseadas em pull, como o Amazon S3, a função deve OpenSearch permitir que a Ingestion a assuma e acesse os dados com as permissões necessárias.

Tópicos

- Perfil de ingestão para origens baseadas em push
- Perfil de ingestão para origens baseadas em pull
- Ingestão entre contas

Perfil de ingestão para origens baseadas em push

Para origens baseadas em push, os dados são enviados por push de outro serviço, como o Amazon Security Lake ou o Amazon DynamoDB, para o pipeline de ingestão. Nesse cenário, o perfil de ingestão precisa, no mínimo, da permissão `osis:Ingest` para interagir com o pipeline.

A seguinte política de acesso do IAM demonstra como conceder essa permissão ao perfil de ingestão:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "osis:Ingest"
      ],
      "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/pipeline-name/*"
    }
  ]
}
```

Perfil de ingestão para origens baseadas em pull

Para fontes baseadas em pull, o pipeline de OpenSearch ingestão extrai ou busca ativamente dados de uma fonte externa, como o Amazon S3. Nesse caso, o pipeline deve assumir um perfil de pipeline do IAM que conceda as permissões necessárias para acessar a fonte de dados. Nesses cenários, perfil de ingestão é sinônimo de perfil de pipeline.

A função deve incluir uma relação de confiança que permita que a OpenSearch Inestion a assuma e permissões específicas para a fonte de dados. Para obter mais informações, consulte the section called “Permissões para ler uma origem”.

Ingestão entre contas

Talvez seja necessário ingerir dados em um pipeline de outro Conta da AWS, como uma conta de aplicativo. Para configurar a ingestão entre contas, defina uma perfil de ingestão na mesma conta do pipeline e estabeleça uma relação de confiança entre o perfil de ingestão e a conta do aplicativo:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
     "Effect": "Allow",
     "Principal": {
```

```
        "AWS": "arn:aws:iam::444455556666:root"
      },
     "Action": "sts:AssumeRole"
  }]
}
```

Em seguida, configure seu aplicativo para assumir o perfil de ingestão. A conta do aplicativo deve conceder AssumeRolepermissões à função do aplicativo para a função de ingestão na conta do pipeline.

Para obter etapas detalhadas e exemplos de políticas do IAM, consulte the section called "Concessão de acesso de ingestão entre contas".

Concedendo acesso aos pipelines OpenSearch do Amazon Ingestion aos domínios

Um pipeline OpenSearch de ingestão da Amazon precisa de permissão para gravar no domínio do OpenSearch serviço que está configurado como seu coletor. Para fornecer acesso, você configura uma função AWS Identity and Access Management (IAM) com uma política de permissões restritiva que limita o acesso ao domínio para o qual um pipeline está enviando dados. Por exemplo, talvez você queira limitar um pipeline de ingestão somente ao domínio e aos índices necessários para ser compatível com seu caso de uso.

> **Important**
>
> Você pode escolher criar manualmente a função do pipeline ou fazer com que o OpenSearch Inestion a crie para você durante a criação do pipeline. Se você escolher a criação automática de funções, o OpenSearch Ingestion adicionará todas as permissões necessárias à política de acesso à função do pipeline com base na fonte e no coletor que você escolher. Ele cria um perfil de pipeline no IAM com o prefixo `OpenSearchIngestion-` e o sufixo que você insere. Para obter mais informações, consulte the section called "Perfis do pipeline". Se você fizer com que o OpenSearch Ingestion crie a função de pipeline para você, ainda precisará incluir a função na política de acesso ao domínio e mapeá-la para uma função de back-end (se o domínio usar controle de acesso refinado), antes ou depois de criar o pipeline. Veja instruções na etapa 2.

Tópicos

- Etapa 1: Criar a função de pipeline
- Etapa 2: configurar o acesso a dados para o domínio

Etapa 1: Criar a função de pipeline

O perfil deve ter uma política de permissões anexada que permita enviar dados para o coletor de domínios. Ele também deve ter uma relação de confiança que permita que o OpenSearch Inestion assuma a função. Para obter instruções de como associar uma política gerenciada a uma função, consulte Adição de permissões de identidade do IAM no Manual do usuário do IAM.

O seguinte exemplo de política demonstra o privilégio mínimo que você pode fornecer em um perfil de pipeline para que ele grave em um único domínio:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:*:111122223333:domain/*"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:*:111122223333:domain/domain-name/*"
        }
    ]
}
```

Se planeja reutilizar a função para gravar em vários domínios, você pode tornar a política mais ampla substituindo o nome do domínio por um caractere curinga (*).

A função deve ter a seguinte relação de confiança, o que permite que o OpenSearch Inestion assuma a função do pipeline:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Etapa 2: configurar o acesso a dados para o domínio

Para que um pipeline grave dados em um domínio o domínio deve ter uma política de acesso em nível de domínio que permita que o perfil de pipeline o acesse.

O seguinte exemplo de política de acesso ao domínio permite que o perfil de pipeline denominado `pipeline-role` grave dados no domínio denominado `ingestion-domain`:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/pipeline-role"
            },
            "Action": [
                "es:DescribeDomain",
                "es:ESHttp*"
            ],
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
        }
    ]
```

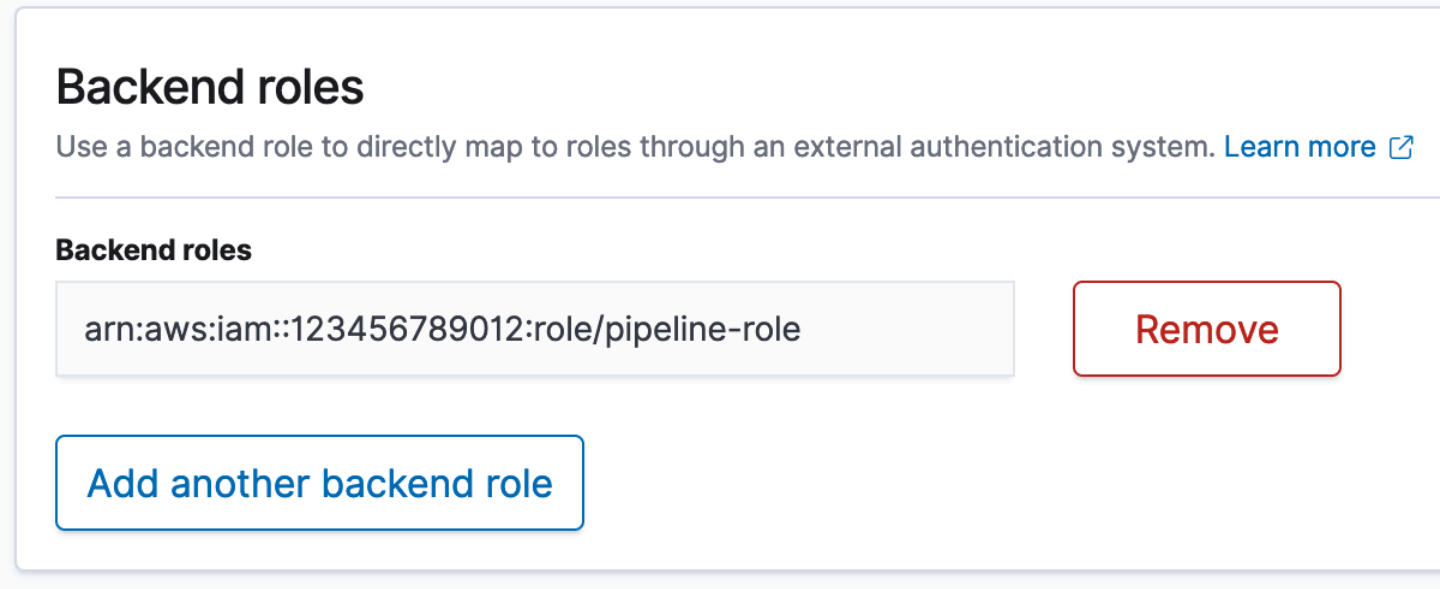
```
}
```

Mapear o perfil de pipeline (somente para domínios que usam o controle de acesso refinado)

Se seu domínio usa controle de acesso refinado para autenticação, há etapas adicionais que você precisa seguir para fornecer acesso ao pipeline a um domínio. As etapas variam de acordo com a configuração do seu domínio:

- Cenário 1: função de mestre e função de pipeline diferentes — Se você estiver usando um Amazon Resource Name (ARN) do IAM como usuário principal e ele é diferente da função de pipeline, você precisa mapear a função de pipeline para a função de OpenSearch `all_access` back-end. Isso adiciona o perfil do pipeline como um usuário-mestre adicional. Para saber mais, consulte Usuários primários adicionais.
- Cenário 2: Usuário principal no banco de dados de usuário interno — Se seu domínio usa um usuário mestre no banco de dados de usuário interno e autenticação básica HTTP para OpenSearch painéis, você não pode passar o nome de usuário e a senha principais diretamente para a configuração do pipeline. Em vez disso, mapeie a função do pipeline para a função de OpenSearch `all_access` back-end. Isso adiciona o perfil do pipeline como um usuário-mestre adicional. Para saber mais, consulte Usuários primários adicionais.
- Cenário 3: perfil do mestre e perfil do pipeline iguais (pouco comum): se você estiver usando um ARN do IAM como usuário-mestre e for o mesmo ARN que você está usando como perfil do pipeline, não será necessário realizar nenhuma ação adicional. O pipeline tem as permissões necessárias para gravar no domínio. Esse cenário é pouco comum porque a maioria dos ambientes usa um perfil de administrador ou algum outro perfil como perfil de mestre.

A imagem a seguir mostra como mapear a função do pipeline para uma função de backend:



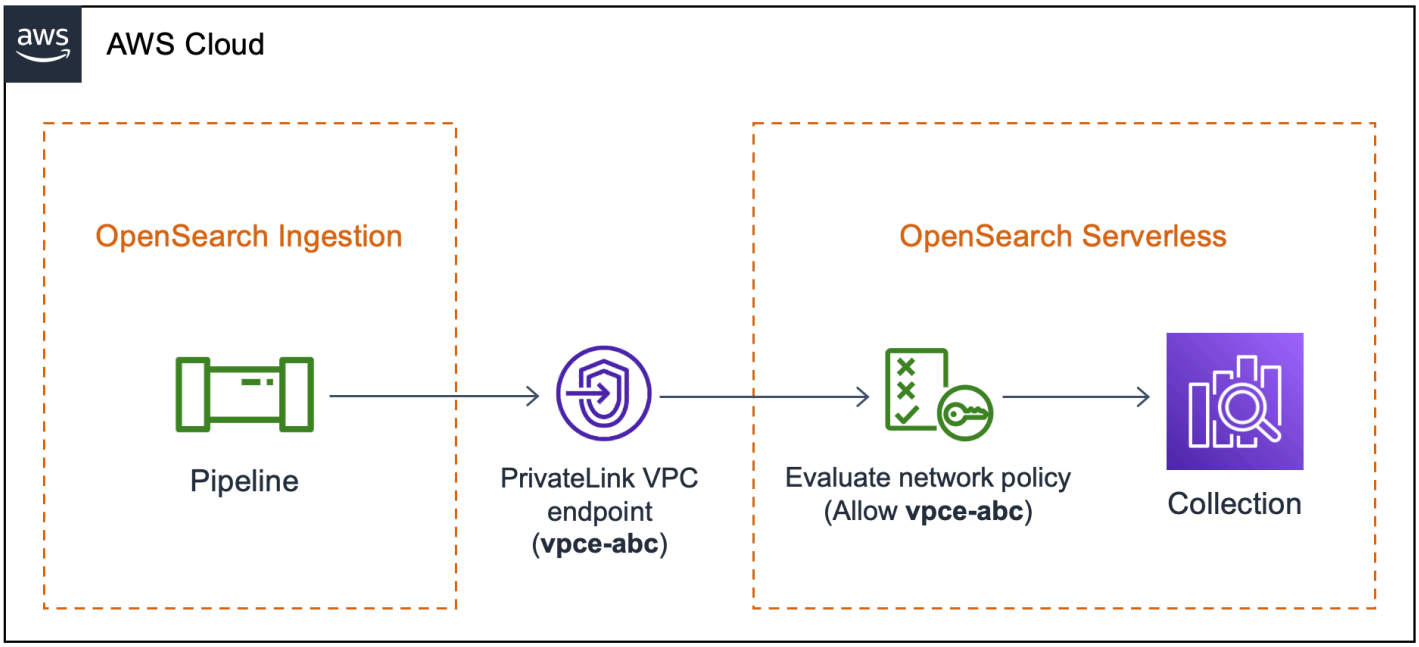
Concedendo aos pipelines do Amazon OpenSearch Ingestion acesso às coleções

Um pipeline OpenSearch de ingestão da Amazon pode gravar em uma coleção pública OpenSearch sem servidor ou coleção VPC. Para fornecer acesso à coleção, você configura uma função de pipeline AWS Identity and Access Management (IAM) com uma política de permissões que concede acesso à coleção. O pipeline assume essa função para assinar solicitações no coletor de coleta OpenSearch Serverless.

> **Important**
>
> Você pode escolher criar manualmente a função do pipeline ou fazer com que o OpenSearch Inestion a crie para você durante a criação do pipeline. Se você escolher a criação automática de funções, o OpenSearch Ingestion adicionará todas as permissões necessárias à política de acesso à função do pipeline com base na fonte e no coletor que você escolher. Ele cria um perfil de pipeline no IAM com o prefixo `OpenSearchIngestion-` e o sufixo que você insere. Para obter mais informações, consulte the section called “Perfis do pipeline”. Se você fizer com que o OpenSearch Ingestion crie a função de pipeline para você, ainda precisará incluir a função na política de acesso a dados da coleção, antes ou depois de criar o pipeline. Veja instruções na etapa 2.

Durante a criação do pipeline, o OpenSearch Ingestion cria uma AWS PrivateLink conexão entre o pipeline e a coleção OpenSearch Serverless. Todo o tráfego do pipeline passa por esse endpoint da

VPC e é roteado para a coleção. Para acessar a coleção, o endpoint deve ter acesso à coleção por meio de uma política de acesso à rede.



Tópicos

- Etapa 1: Criar a função de pipeline
- Etapa 2: configurar o acesso a dados e à rede para a coleção

Etapa 1: Criar a função de pipeline

O perfil do pipleline deve ter uma política de permissões anexada que permita que ele envie dados para o coletor de coleções. Ele também deve ter uma relação de confiança que permita que o OpenSearch Inestion assuma a função. Para obter instruções de como associar uma política gerenciada a uma função, consulte Adição de permissões de identidade do IAM no Manual do usuário do IAM.

O seguinte exemplo de política demonstra o privilégio mínimo que você pode fornecer em uma política de acesso de perfil de pipeline para gravar em coleções:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:BatchGetCollection",
                "aoss:CreateSecurityPolicy",
                "aoss:GetSecurityPolicy",
                "aoss:UpdateSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

A função deve ter a seguinte relação de confiança, que permita que a OpenSearch Inestion a assuma:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "osis-pipelines.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Etapa 2: configurar o acesso a dados e à rede para a coleção

Crie uma coleção OpenSearch Serverless com as seguintes configurações. Para obter instruções sobre como criar uma coleção, consulte the section called “Criação de coleções”.

Política de acesso a dados

Crie uma política de acesso a dados para a coleção que conceda as permissões necessárias para o perfil do pipeline. Por exemplo:

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/collection-name/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::account-id:role/pipeline-role"
    ],
    "Description": "Pipeline role access"
  }
]
```

> **Note**
>
> No elemento `Principal`, especifique o nome do recurso da Amazon (ARN) do perfil do pipeline.

Política de acesso à rede

Cada coleção que você cria no OpenSearch Serverless tem pelo menos uma política de acesso à rede associada a ela. As políticas de acesso à rede determinam se a coleção é acessível pela internet a partir de redes públicas ou se deve ser acessada de forma privada. Para saber mais sobre políticas de rede, consulte the section called “Acesso à rede”.

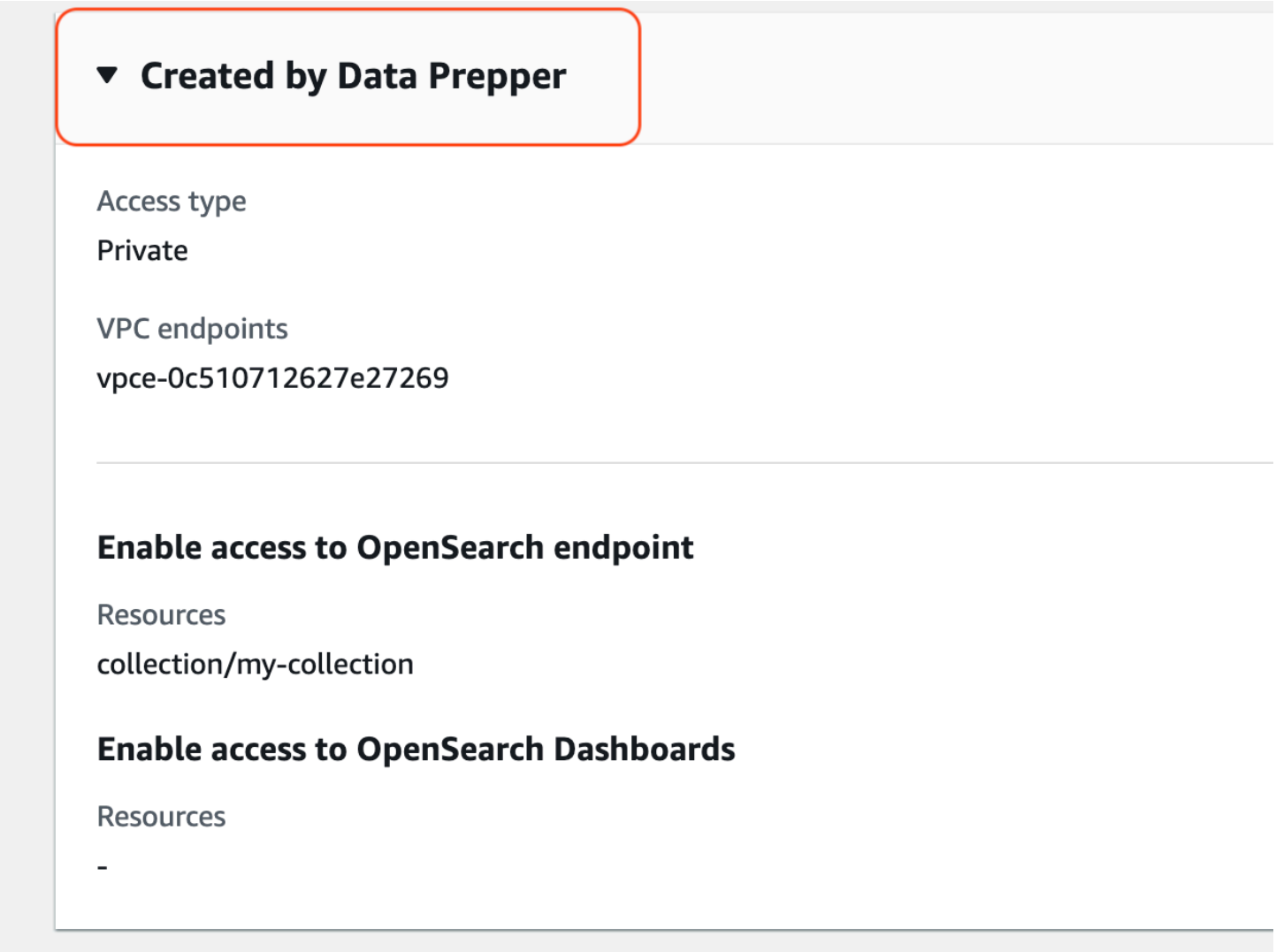
Em uma política de acesso à rede, você só pode especificar VPC endpoints OpenSearch gerenciados sem servidor. Para obter mais informações, consulte the section called “Endpoint VPC do plano de dados”. No entanto, para que o pipeline seja gravado na coleção, a política também deve conceder acesso ao VPC endpoint que o OpenSearch Ingestion cria automaticamente entre o pipeline e a coleção. Portanto, se você escolher uma coleção OpenSearch sem servidor como coletor de destino para um pipeline, deverá inserir o nome da política de rede associada no campo Nome da política de rede.

Durante a criação do pipeline, o OpenSearch Inestion verifica a existência da política de rede especificada. Se não existir, o OpenSearch Ingestion o cria. Se ela existir, o OpenSearch Ingestion a atualizará adicionando uma nova regra a ela. A regra concede acesso ao endpoint da VPC que conecta o pipeline e a coleção.

Por exemplo:

```
{
   "Rules":[
      {
         "Resource":[
            "collection/my-collection"
         ],
         "ResourceType":"collection"
      }
   ],
   "SourceVPCEs":[
      "vpce-0c510712627e27269" # The ID of the VPC endpoint that OpenSearch Ingestion
 creates between the pipeline and collection
   ],
   "Description":"Created by Data Prepper"
}
```

No console, todas as regras que o OpenSearch Inestion adiciona às suas políticas de rede são denominadas Created by Data Prepper:



> **Note**
>
> Em geral, uma regra que especifique o acesso público para uma coleção substitui uma regra que especifique o acesso privado. Portanto, se a política já tinha acesso público configurado, essa nova regra adicionada pelo OpenSearch Ingestion não altera, na verdade, o comportamento da política. Para obter mais informações, consulte the section called "Precedência das políticas".

Se você interromper ou excluir o pipeline, o OpenSearch Ingestion excluirá o VPC endpoint entre o pipeline e a coleção. Ele também modifica a política de rede para remover o endpoint da VPC da lista de endpoints permitidos. Se você reiniciar o pipeline, ele recriará o endpoint da VPC e reatualizará a política de rede com o ID do endpoint.

Começando a usar o Amazon OpenSearch Ingestion

O Amazon OpenSearch Ingestion suporta a ingestão de dados em domínios de OpenSearch serviços gerenciados e OpenSearch coleções sem servidor. Os tutoriais a seguir orientam você nas etapas básicas para colocar um pipeline em funcionamento.

O primeiro tutorial mostra como usar o Amazon OpenSearch Ingestion para configurar um pipeline simples e ingerir dados em um domínio do Amazon OpenSearch Service.

O segundo tutorial mostra como usar o Amazon OpenSearch Ingestion para configurar um pipeline simples e ingerir dados em uma coleção Amazon OpenSearch Serverless.

> **Note**
>
> A criação de pipeline falhará se você não configurar as permissões corretas. Consulte the section called “Configurar funções e usuários” para entender melhor as funções necessárias antes de criar um pipeline.

Tópicos

- Tutorial: Ingestão de dados em um domínio usando o Amazon OpenSearch Ingestion
- Tutorial: Ingestão de dados em uma coleção usando o Amazon OpenSearch Ingestion

Tutorial: Ingestão de dados em um domínio usando o Amazon OpenSearch Ingestion

Este tutorial mostra como usar o Amazon OpenSearch Ingestion para configurar um pipeline simples e ingerir dados em um domínio do Amazon OpenSearch Service. Um pipeline é um recurso que o OpenSearch Ingestion provisiona e gerencia. Você pode usar um pipeline para filtrar, enriquecer, transformar, normalizar e agregar dados para análises e visualizações posteriores no Service. OpenSearch

Este tutorial orienta você pelas etapas básicas de como conseguir montar um pipeline rapidamente. Para obter instruções mais abrangentes, consulte the section called “Como criar pipelines”.

Você concluirá as seguintes etapas neste tutorial:

1. Crie um domínio.

2. Crie um pipeline.
3. Ingira alguns dados de amostra.

Neste tutorial, você vai criar os recursos a seguir:

- Um domínio denominado `ingestion-domain` no qual o pipeline grava
- Um pipeline chamado `ingestion-pipeline`

Permissões obrigatórias

Para fazer este tutorial, o usuário ou o perfil deve ter uma política baseada em identidade anexada com as permissões mínimas a seguir. Essas permissões autorizam você criar um perfil de pipeline e anexar uma política (`iam:Create*` e `iam:Attach*`), criar ou modificar um domínio (`es:*`) e trabalhar com pipelines (`osis:*`).

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "es:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
         ]
```

```
      }
   ]
}
```

Etapa 1: Criar a função de pipeline

Primeiro, crie uma função que o pipeline assumirá para acessar o coletor OpenSearch de domínio do serviço. Neste tutorial, você incluirá esse perfil posteriormente na configuração do pipeline.

Para criar a função de pipeline

1. Abra o AWS Identity and Access Management console em https://console.aws.amazon.com/iamv2/.
2. Escolha Políticas e, depois, Criar política.
3. Neste tutorial, você consumirá dados em um domínio chamado `ingestion-domain`, que você criará na próxima etapa. Selecione JSON e cole a política a seguir no editor. Substitua *`your-account-id`* pelo ID da sua conta e modifique a região, se necessário.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-
domain"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-
domain/*"
        }
    ]
}
```

Se quiser gravar dados em um domínio existente, `ingestion-domain` substitua pelo nome do seu domínio.

> **Note**
>
> Para simplificar, neste tutorial, usaremos uma política de acesso ampla. Em ambientes de produção, no entanto, recomendamos que você aplique uma política de acesso mais restritiva à sua função de pipeline. Para obter um exemplo de política que fornece as permissões mínimas necessárias, consulte the section called “Concedendo acesso aos pipelines aos domínios”.

4. Escolha Próximo, então Próximo, e nomeie sua política pipeline-policy.
5. Escolha Criar política.
6. Depois, crie um perfil e anexe a política à ele. Selecione Funções e, depois, Criar função.
7. Escolha Política de confiança personalizada e cole a política a seguir no editor:

   JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

8. Escolha Próximo. Em seguida, pesquise e selecione pipeline-policy (que você acabou de criar).
9. Escolha Avançar e nomeie a função PipelineRole.
10. Selecione Criar função.

Lembre-se do nome do recurso da Amazon (ARN) do perfil (por exemplo, `arn:aws:iam::`*`your-account-id`*`:role/PipelineRole`). Você precisará dele quando criar seu pipeline.

Etapa 2: Criar um domínio

Primeiro, crie um domínio denominado `ingestion-domain` no qual ingerir dados.

Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa e crie um domínio que atenda aos seguintes requisitos:

- Está executando OpenSearch 1.0 ou posterior, ou Elasticsearch 7.4 ou posterior
- Usa o acesso público
- Não use controle de acesso detalhado.

> **Note**
>
> Esses requisitos têm como objetivo garantir a simplicidade deste tutorial. Em ambientes de produção, você pode configurar um domínio com acesso à VPC and/or usando controle de acesso refinado. Para usar controle de acesso refinado, consulte Mapear a função do pipeline.

O domínio deve ter uma política de acesso que conceda permissão à função `OpenSearchIngestion-PipelineRole` do IAM, que o OpenSearch Serviço criará para você na próxima etapa. O pipeline assumirá esse perfil para enviar dados para o coletor de domínios.

Assegure-se de que o domínio tenha a seguinte política de acesso em nível de domínio, que concede ao perfil de pipeline acesso ao domínio. Substitua a região e a ID da conta com seus dados:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
      },
```

```
      "Action": "es:*",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/ingestion-domain/*"
    }
  ]
}
```

Para saber mais sobre a criação de políticas de acesso em nível de domínio, consulte the section called “Políticas baseadas em recursos”.

Se você já tiver um domínio criado, modifique sua política de acesso existente para fornecer as permissões acima para a `OpenSearchIngestion-PipelineRole`.

Etapa 3: Criar um pipeline

Agora que você tem um domínio, pode criar um pipeline.

Para criar um pipeline

1. No console do Amazon OpenSearch Service, escolha Pipelines no painel de navegação esquerdo.
2. Selecione Criar pipeline.
3. Selecione o pipeline Em branco e escolha Selecionar esquema.
4. Neste tutorial, criaremos um subpipeline simples que usa o plug-in Origem HTTP. O plug-in aceita dados de log em formato de matriz JSON. Vamos especificar um único domínio OpenSearch de serviço como coletor e ingerir todos os dados no `application_logs` índice.

   No menu Origem, escolha HTTP. Em Caminho, insira /logs.
5. Para simplificar neste tutorial, configuraremos o acesso público do pipeline. Em Opções de rede de origem, escolha Acesso público. Para saber mais sobre como configurar VPC, consulte the section called “Como configurar o acesso à VPC para pipelines”.
6. Escolha Próximo.
7. Em Processador, insira Data e escolha Adicionar.
8. Habilite A partir da hora do recebimento. Deixe todas as outras configurações com seus valores padrão.
9. Escolha Próximo.
10. Configure os detalhes do coletor. Para tipo OpenSearch de recurso, escolha Cluster gerenciado. Em seguida, escolha o domínio do OpenSearch serviço que você criou na seção anterior.

Em Nome do índice, insira application_logs. OpenSearch A ingestão cria automaticamente esse índice no domínio, caso ele ainda não exista.

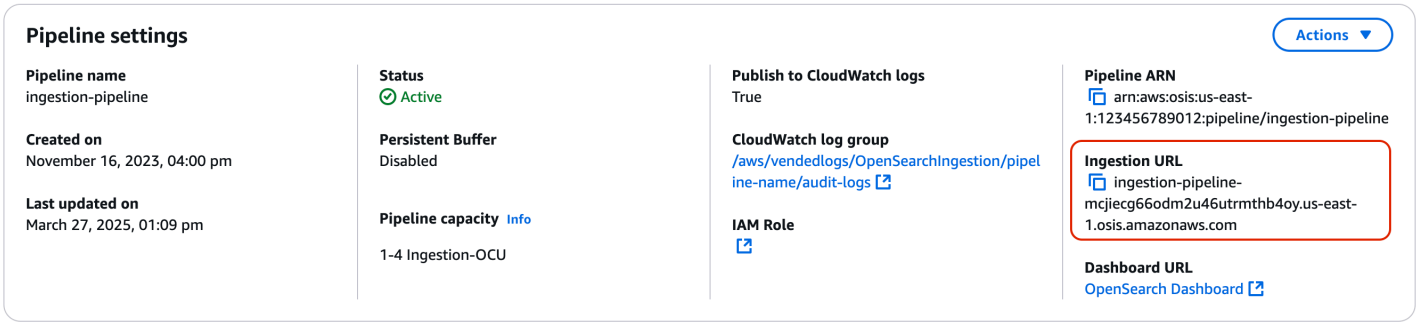
11. Escolha Próximo.
12. Nomeie o pipeline de ingestion-pipeline. Deixe todas as configurações de capacidade com seus valores padrão.
13. Em Perfil do pipeline, selecione Criar e usar um novo perfil de serviço. O perfil do pipeline fornece as permissões necessárias para um pipeline gravar no coletor de domínios e ler em origens baseadas em pull. Ao selecionar essa opção, você permite que o OpenSearch Inestion crie a função para você, em vez de criá-la manualmente no IAM. Para obter mais informações, consulte the section called “Configurar funções e usuários”.
14. Em Sufixo do nome da função de serviço, insira PipelineRole. No IAM, o perfil terá o formato `arn:aws:iam::`*`your-account-id`*`:role/OpenSearchIngestion-`**`PipelineRole`**.
15. Escolha Próximo. Revise sua configuração do pipeline e escolha Criar pipeline. O pipeline leva de 5 a 10 minutos para se tornar ativo.

Etapa 4: ingestão de dados de exemplo

Quando o status do pipeline é `Active`, você pode começar a ingerir dados nele. Você deve assinar todas as solicitações HTTP no pipeline usando o Signature Version 4. Use uma ferramenta HTTP, como o Postman ou awscurl, para enviar alguns dados para o pipeline. Assim como acontece com a indexação de dados diretamente em um domínio, a ingestão de dados em um pipeline sempre exige um perfil do IAM ou uma chave de acesso e chave secreta do IAM.

> **Note**
>
> A entidade principal responsável pela assinatura da solicitação deve ter a permissão `osis:Ingest` do IAM.

Primeiro, obtenha o URL de ingestão na página Configurações do Pipeline:



Em seguida, faça a ingestão de alguns dados de exemplo. A solicitação a seguir usa awscurl para enviar um único arquivo de log para o pipeline:

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Você obterá uma resposta `200 OK`. Se você receber um erro de autenticação, pode ser porque está ingerindo dados de uma conta diferente daquela em que o pipeline está. Consulte the section called "Corrigindo problemas de permissão".

Agora, consulte o índice `application_logs` para garantir que sua entrada de log tenha sido ingerida com sucesso:

```
awscurl --service es --region us-east-1 \
     -X GET \
     https://search-ingestion-domain.us-east-1.es.amazonaws.com/application_logs/
_search | json_pp
```

Resposta de exemplo:

```
{
   "took":984,
   "timed_out":false,
   "_shards":{
      "total":1,
      "successful":5,
```

```
      "skipped":0,
      "failed":0
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"application_logs",
            "_type":"_doc",
            "_id":"z6VY_IMBRpceX-DU6V4O",
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2022-10-21T21:00:25.502Z"
            }
         }
      ]
   }
}
```


Corrigindo problemas de permissão

Se você seguiu as etapas do tutorial e ainda vê erros de autenticação ao tentar ingerir dados, talvez seja porque a função que está gravando em um pipeline é Conta da AWS diferente do próprio pipeline. Nesse caso, você precisa criar e assumir uma função que permita especificamente a ingestão de dados. Para instruções, consulte the section called “Concessão de acesso de ingestão entre contas”.

Recursos relacionados

Este tutorial apresentou um caso de uso simples de ingestão de um único documento via HTTP. Em cenários de produção, você configurará seus aplicativos cliente (como Fluent Bit, Kubernetes ou OpenTelemetry Collector) para enviar dados para um ou mais pipelines. Seus pipelines provavelmente serão mais complexos do que o exemplo simples deste tutorial.

Para começar a configurar seus clientes e ingerir dados, consulte os seguintes recursos:

- Criação e gerenciamento de pipelines
- Configurando seus clientes para enviar dados para OpenSearch o Inestion
- Documentação do Data Prepper

Tutorial: Ingestão de dados em uma coleção usando o Amazon OpenSearch Ingestion

Este tutorial mostra como usar o Amazon OpenSearch Ingestion para configurar um pipeline simples e ingerir dados em uma coleção Amazon OpenSearch Serverless. Um pipeline é um recurso que o OpenSearch Ingestion provisiona e gerencia. Você pode usar um pipeline para filtrar, enriquecer, transformar, normalizar e agregar dados para análises e visualizações posteriores no Service. OpenSearch

Para ver um tutorial que demonstra como ingerir dados em um domínio de OpenSearch serviço provisionado, consulte. the section called “Tutorial: ingerir dados em um domínio”

Você concluirá as seguintes etapas neste tutorial:

1. Crie uma coleção.
2. Crie um pipeline.
3. Ingira alguns dados de amostra.

Neste tutorial, você vai criar os recursos a seguir:

- Um coleção chamada `ingestion-collection` no qual o pipeline fará a gravação
- Um pipeline chamado `ingestion-pipeline-serverless`

Permissões obrigatórias

Para fazer este tutorial, o usuário ou o perfil deve ter uma política baseada em identidade anexada com as permissões mínimas a seguir. Essas permissões autorizam você criar um perfil de pipeline e anexar uma política (`iam:Create*` e `iam:Attach*`), criar ou modificar uma coleção (`aoss:*`) e trabalhar com pipelines (`osis:*`).

Além disso, várias permissões do IAM são necessárias para criar automaticamente a função do pipeline e passá-la para o OpenSearch Ingestion para que ele possa gravar dados na coleção.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "aoss:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
         ]
      }
   ]
}
```

Etapa 1: criar uma coleção

Primeiro, crie uma coleção na qual ingerir dados. Daremos o nome da coleção de `ingestion-collection`.

1. Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Escolha Coleções no painel de navegação à esquerda e escolha Criar coleção.

3. Nomeie a coleção ingestion-collection.
4. Em Segurança, escolha Criação padrão.
5. Em Configurações de acesso à rede, altere o tipo de acesso para Público .
6. Mantenha todas as outras configurações em seus valores padrão e escolha Próximo.
7. Agora, configure uma política de acesso aos dados para a coleção. Desmarque a opção Atender automaticamente às configurações da política de acesso.
8. Para Método de definição, escolha JSON e cole a seguinte política no editor. Essa política faz duas coisas:

   - Permite que o perfil de pipeline faça gravações na coleção.
   - Permite que você leia a coleção. Posteriormente, depois de ingerir alguns dados de amostra no pipeline, você consultará a coleção para garantir que os dados foram ingeridos e gravados com sucesso no índice.

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/ingestion-collection/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
          "aoss:ReadDocument",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole",
      "arn:aws:iam::your-account-id:role/Admin"
    ],
    "Description": "Rule 1"
  }
]
```

9. Modifique os `Principal` elementos para incluir seu Conta da AWS ID. Como segunda entidade principal, especifique um usuário ou perfil que você possa usar para consultar a coleção mais tarde.
10. Escolha Próximo. Nomeie a política de acesso pipeline-collection-accesse escolha Avançar novamente.
11. Reveja sua configuração da coleção e escolha Enviar.

Etapa 2: criar um pipeline

Agora que você tem uma coleção, pode criar um pipeline.

Para criar um pipeline

1. No console do Amazon OpenSearch Service, escolha Pipelines no painel de navegação esquerdo.
2. Selecione Criar pipeline.
3. Selecione o pipeline Em branco e escolha Selecionar esquema.
4. Neste tutorial, criaremos um subpipeline simples que usa o plug-in Origem HTTP. O plug-in aceita dados de log em formato de matriz JSON. Vamos especificar uma única coleção OpenSearch Serverless como coletor e ingerir todos os dados no índice. `my_logs`

   No menu Origem, escolha HTTP. Em Caminho, insira /logs.
5. Para simplificar neste tutorial, configuraremos o acesso público do pipeline. Em Opções de rede de origem, escolha Acesso público. Para saber mais sobre como configurar VPC, consulte the section called “Como configurar o acesso à VPC para pipelines”.
6. Escolha Próximo.
7. Em Processador, insira Data e escolha Adicionar.
8. Habilite A partir da hora do recebimento. Deixe todas as outras configurações com seus valores padrão.
9. Escolha Próximo.
10. Configure os detalhes do coletor. Para tipo de OpenSearch recurso, escolha Coleção (sem servidor). Em seguida, escolha a coleção de OpenSearch serviços que você criou na seção anterior.

    Deixe o nome padrão a política de rede. Em Nome do índice, insira my_logs. OpenSearch A ingestão cria automaticamente esse índice na coleção, caso ele ainda não exista.

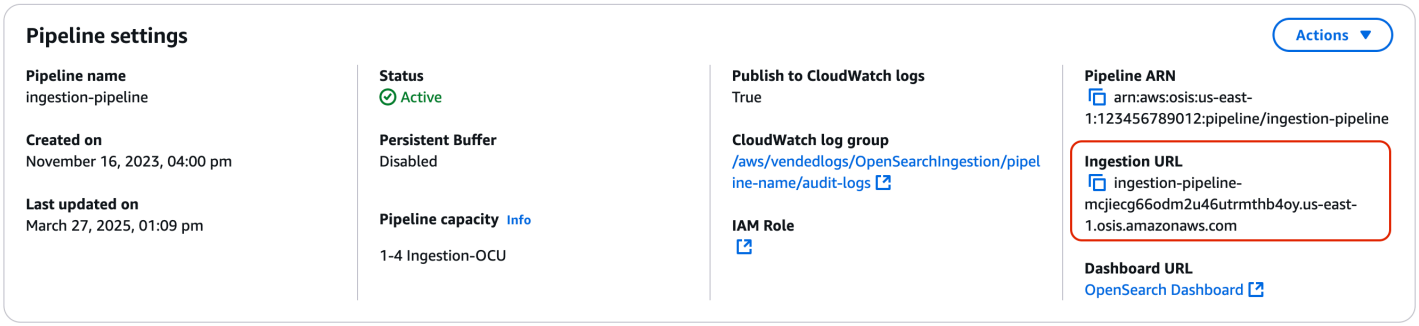
11. Escolha Próximo.
12. Dê um nome ao gasoduto ingestion-pipeline-serverless. Deixe todas as configurações de capacidade com seus valores padrão.
13. Em Perfil do pipeline, selecione Criar e usar um novo perfil de serviço. O perfil do pipeline fornece as permissões necessárias para um pipeline gravar no coletor de coleções e ler de origens baseadas em pull. Ao selecionar essa opção, você permite que o OpenSearch Inestion crie a função para você, em vez de criá-la manualmente no IAM. Para obter mais informações, consulte the section called “Configurar funções e usuários”.
14. Em Sufixo do nome da função de serviço, insira PipelineRole. No IAM, o perfil terá o formato `arn:aws:iam::`*`your-account-id`*`:role/OpenSearchIngestion-`**`PipelineRole`**.
15. Escolha Próximo. Revise sua configuração do pipeline e escolha Criar pipeline. O pipeline leva de 5 a 10 minutos para se tornar ativo.

Etapa 3: ingerir alguns dados de amostra

Quando o status do pipeline é `Active`, você pode começar a ingerir dados nele. Você deve assinar todas as solicitações HTTP no pipeline usando o Signature Version 4. Use uma ferramenta HTTP, como o Postman ou awscurl, para enviar alguns dados para o pipeline. Assim como acontece com a indexação de dados diretamente em uma coleção, a ingestão de dados em um pipeline sempre requer um perfil do IAM, uma chave de acesso do IAM e uma chave secreta.

> **Note**
>
> A entidade principal responsável pela assinatura da solicitação deve ter a permissão `osis:Ingest` do IAM.

Primeiro, obtenha o URL de ingestão na página Configurações do Pipeline:



Depois, envie alguns dados de exemplo para o caminho de ingestão. O exemplo de solicitação a seguir usa awscurl para enviar um único arquivo de log para o pipeline:

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Você obterá uma resposta `200 OK`.

Agora, consulte o índice `my_logs` para garantir que a entrada do log tenha sido ingerida com sucesso:

```
awscurl --service aoss --region us-east-1 \
     -X GET \
     https://collection-id.us-east-1.aoss.amazonaws.com/my_logs/_search | json_pp
```

Resposta de exemplo:

```
{
   "took":348,
   "timed_out":false,
   "_shards":{
      "total":0,
      "successful":0,
      "skipped":0,
      "failed":0
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"my_logs",
            "_id":"1%3A0%3ARJgDvIcBTy5m12xrKE-y",
```

```
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2023-04-26T05:22:16.204Z"
            }
         }
      ]
   }
}
```


Recursos relacionados

Este tutorial apresentou um caso de uso simples de ingestão de um único documento via HTTP. Em cenários de produção, você configurará seus aplicativos cliente (como Fluent Bit, Kubernetes ou OpenTelemetry Collector) para enviar dados para um ou mais pipelines. Seus pipelines provavelmente serão mais complexos do que o exemplo simples deste tutorial.

Para começar a configurar seus clientes e ingerir dados, consulte os seguintes recursos:

- Criação e gerenciamento de pipelines
- Configurando seus clientes para enviar dados para OpenSearch o Inestion
- Documentação do Data Prepper

Visão geral dos recursos do pipeline no Amazon OpenSearch Ingestion

O Amazon OpenSearch Ingestion provisiona pipelines, que consistem em uma fonte, um buffer, zero ou mais processadores e um ou mais coletores. Os pipelines de ingestão são alimentados pelo Data Prepper como mecanismo de dados. Para obter uma visão geral de vários componentes de um pipeline, consulte the section called “Principais conceitos”.

As seções a seguir fornecem uma visão geral de alguns dos recursos mais usados no Amazon OpenSearch Ingestion.

> **Note**
>
> Esta não é uma lista completa de atributos disponíveis para pipelines. Para obter uma documentação abrangente de todas as funcionalidades disponíveis do pipeline, consulte a documentação do Data Prepper. Observe que o OpenSearch Inestion impõe algumas restrições aos plug-ins e às opções que você pode usar. Para obter mais informações, consulte the section called “Plug-ins e opções compatíveis”.

Tópicos

- Armazenamento em buffer persistente
- Dividindo
- Encadeamento
- Filas de mensagens não entregues
- Gerenciamento de índices
- End-to-end reconhecimento
- Pressão oposta da origem

Armazenamento em buffer persistente

Um buffer persistente armazena os dados em um buffer baseado em disco entre várias zonas de disponibilidade para aumentar a durabilidade dos dados. Você pode usar o buffer persistente para ingerir dados de todas as origens baseadas em push compatíveis sem a necessidade de configurar um buffer independente. Essas fontes incluem HTTP e OpenTelemetry para registros, rastreamentos e métricas. Para habilitar buffer persistente, escolha Habilitar buffer persistente ao criar ou atualizar um pipeline. Para obter mais informações, consulte the section called “Como criar pipelines”.

OpenSearch A ingestão determina dinamicamente o número de unidades a OCUs serem usadas para armazenamento em buffer persistente, considerando a fonte de dados, as transformações de streaming e o destino do coletor. Como ele aloca parte OCUs do armazenamento em buffer, talvez seja necessário aumentar o mínimo e o máximo OCUs para manter a mesma taxa de transferência de ingestão. Os pipelines retêm os dados no buffer por até 72 horas.

Se você habilitar buffer persistente para um pipeline, os tamanhos máximos padrão de carga útil de solicitação são os seguintes:

- Fontes HTTP: 10 MB
- OpenTelemetry fontes — 4 MB

Para origens HTTP, você pode aumentar o tamanho máximo da carga útil de solicitação para 20 MB. O tamanho da carga útil de solicitação inclui toda a solicitação HTTP, que normalmente contém vários eventos. Cada evento não pode exceder 3,5 MB.

Pipelines com armazenamento em buffer persistente habilitado dividem as unidades de pipeline configuradas entre unidades de computação e unidades de buffer. Se um pipeline usa um processador com uso intensivo de CPU, como grok, chave-valor ou string dividida, ele aloca as unidades na proporção de 1:1. buffer-to-compute Caso contrário, ele os alocará em uma proporção de 3:1, sempre favorecendo as unidades de computação.

Por exemplo:

- Pipeline com grok e máximo de 2 unidades: 1 unidade de computação e 1 unidade de buffer
- Pipeline com grok e máximo de 5 unidades: 3 unidades de computação e 2 unidades de buffer
- Pipeline sem processador e máximo de 2 unidades: 1 unidade de computação e 1 unidade de buffer
- Pipeline sem processador e máximo de 4 unidades: 1 unidade de computação e 3 unidades de buffer
- Pipeline com grok e máximo de 5 unidades: 2 unidades de computação e 3 unidades de buffer

Por padrão, os pipelines usam an Chave pertencente à AWS para criptografar dados do buffer. Esses pipelines não precisam de nenhuma permissão adicional para o perfil de pipeline.

Como alternativa, é possível especificar uma chave gerenciada pelo cliente e adicionar as seguintes permissões do IAM ao perfil do pipeline:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "KeyAccess",
            "Effect": "Allow",
```

```
            "Action": [
              "kms:Decrypt",
              "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "arn:aws:kms:us-
east-1:111122223333:key/ASIAIOSFODNN7EXAMPLE"
        }
    ]
}
```

Para saber mais, consulte Chaves mestras do cliente (CMKs) no AWS Key Management Service Guia do desenvolvedor.

> **Note**
>
> Se você desabilitar o armazenamento em buffer persistente, seu pipeline começará a ser executado inteiramente com armazenamento em buffer na memória.

Dividindo

Você pode configurar um pipeline de OpenSearch ingestão para dividir os eventos recebidos em um subpipeline, permitindo que você execute diferentes tipos de processamento no mesmo evento de entrada.

O exemplo de pipeline a seguir divide os eventos recebidos em dois subpipelines. Cada subpipeline usa seu próprio processador para enriquecer e manipular os dados e, em seguida, envia os dados para índices diferentes. OpenSearch

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  sink:
    - pipeline:
        name: "logs_enriched_one_pipeline"
    - pipeline:
        name: "logs_enriched_two_pipeline"
```

```
logs_enriched_one_pipeline:
  source:
    pipeline:
      name: "log-pipeline"
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
        index: "enriched_one_logs"

logs_enriched_two_pipeline:
  source:
    pipeline:
      name: "log-pipeline"
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_two_logs"
```

Encadeamento

Você pode encadear vários subpipelines para realizar o processamento e o enriquecimento de dados em partes. Em outras palavras, você pode enriquecer um evento de entrada com determinados recursos de processamento em um subpipeline, enviá-lo para outro subpipeline para enriquecimento adicional com um processador diferente e, finalmente, enviá-lo para o coletor. OpenSearch

No exemplo a seguir, o `log_pipeline` subpipeline enriquece um evento de log de entrada com um conjunto de processadores e, em seguida, envia o evento para um índice chamado. OpenSearch `enriched_logs` O pipeline envia o mesmo evento para o `log_advanced_pipeline` subpipeline, que o processa e o envia para um OpenSearch índice diferente chamado`enriched_advanced_logs`.

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  processor:
    ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_logs"
    - pipeline:
        name: "log_advanced_pipeline"

log_advanced_pipeline:
  source:
    pipeline:
      name: "log-pipeline"
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_advanced_logs"
```

Filas de mensagens não entregues

As filas de letras mortas (DLQs) são destinos para eventos que um pipeline não consegue gravar em um coletor. Em OpenSearch Ingestão, você deve especificar um bucket do Amazon S3 com permissões de gravação apropriadas para ser usado como DLQ. Você pode adicionar uma configuração de DLQ a cada coletor em um pipeline. Quando um pipeline encontra erros de gravação, ele cria objetos DLQ no bucket S3 configurado. Os objetos DLQ existem em um arquivo JSON como uma matriz de eventos com falha.

Um pipeline grava eventos na DLQ quando uma das condições a seguir é atendida:

- A contagem máxima de novas tentativas do OpenSearch coletor foi esgotada. OpenSearch A ingestão requer um mínimo de 16 para essa configuração.
- O coletor está rejeitando eventos devido a uma condição de erro.

Configuração

Para configurar uma fila de mensagens não entregues para um sub-pipeline, escolha Habilitar DLQ do S3 ao configurar o destino do coletor. Depois, especifique as configurações necessárias para a fila. Para saber mais, consulte Configuration na documentação da Data Prepper DLQ.

Os arquivos gravados nessa DLQ do S3 seguem o seguinte padrão de nomenclatura:

```
dlq-v${version}-${pipelineName}-${pluginId}-${timestampIso8601}-${uniqueId}
```

Para obter instruções para configurar manualmente o perfil do pipeline para permitir o acesso ao bucket do S3 no qual o DLQ grava, consulte the section called “Permissões para gravar no Amazon S3 ou em uma fila de mensagens não entregues”.

Exemplo

Considere o seguinte exemplo de arquivo DLQ:

```
dlq-v2-apache-log-pipeline-opensearch-2023-04-05T15:26:19.152938Z-e7eb675a-
f558-4048-8566-dac15a4f8343
```

Aqui está um exemplo de dados que não foram gravados no coletor e foram enviados ao bucket DLQ S3 para análise posterior:

```
Record_0
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index    "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
```

```
log      "sample log"
timestamp     "2023-04-14T10:36:01.070Z"

Record_1
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index               "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
log                 "another sample log"
timestamp           "2023-04-14T10:36:01.071Z"
```

Gerenciamento de índices

O Amazon OpenSearch Ingestion tem muitos recursos de gerenciamento de índices, incluindo os seguintes.

Criar índices

Você pode especificar um nome de índice em um coletor de pipeline e o OpenSearch Ingestion cria o índice ao provisionar o pipeline. Se um índice já existir, o pipeline o usará para indexar eventos recebidos. Se você parar e reiniciar um pipeline ou atualizar sua configuração YAML, o pipeline tentará criar novos índices, caso eles ainda não existam. Um pipeline nunca pode excluir um índice.

Os coletores de exemplo a seguir criam dois índices quando o pipeline é provisionado:

```
sink:
  - opensearch:
      index: apache_logs
  - opensearch:
      index: nginx_logs
```

Geração de nomes e padrões de índice

Você pode gerar nomes de índices dinâmicos usando variáveis dos campos de eventos recebidos. Na configuração do coletor, use o formato `string${}` para sinalizar a interpolação de strings e use um ponteiro JSON para extrair campos de eventos. As opções para `index_type` são `custom`

oumanagement_disabled. Como o index_type padrão é custom para OpenSearch domínios e management_disabled coleções OpenSearch sem servidor, ele pode ser deixado sem definição.

Por exemplo, o pipeline a seguir seleciona o campo metadataType dos eventos recebidos para gerar nomes de índice.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}"
```

A configuração a seguir continua gerando um novo índice a cada dia ou a cada hora.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"

pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd.HH}"
```

O nome do índice também pode ser uma string simples com um padrão de data e hora como sufixo, como my-index-%{yyyy.MM.dd}. Quando o coletor envia dados para OpenSearch, ele substitui o padrão de data e hora pela hora UTC e cria um novo índice para cada dia, como. my-index-2022.01.25 Para obter mais informações, consulte a DateTimeFormatteraula.

Esse nome de índice também pode ser uma string formatada (com ou sem um sufixo de padrão de data e hora), como my-${index}-name. Quando o coletor envia dados para OpenSearch, ele substitui a "${index}" parte pelo valor no evento que está sendo processado. Se o formato for "${index1/index2/index3}", ele substituirá o campo index1/index2/index3 por seu valor no evento.

Gerando documento IDs

Um pipeline pode gerar uma ID de documento ao indexar OpenSearch documentos em. Ele pode inferir esses documentos a IDs partir dos campos dos eventos recebidos.

Este exemplo usa o campo `uuid` de um evento recebido para gerar um ID do documento.

```
pipeline:
  ...
  sink:
    opensearch:
      index_type: custom
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"
      "document_id": "uuid"
```

No exemplo a seguir, o processador Adicionar entradas mescla os campos `uuid` e `other_field` do evento recebido para gerar um ID do documento.

A `create` ação garante que documentos idênticos IDs não sejam substituídos. O pipeline elimina documentos duplicados sem nenhuma nova tentativa ou evento de DLQ. Essa é uma expectativa razoável para autores de pipelines que usam essa ação, pois o objetivo é evitar a atualização de documentos existentes.

```
pipeline:
  ...
  processor:
   - add_entries:
      entries:
        - key: "my_doc_id_field"
          format: "${uuid}-${other_field}"
  sink:
    - opensearch:
       ...
       action: "create"
       document_id: "my_doc_id"
```

Talvez você queira definir o ID do documento de um evento como um campo de um subobjeto. No exemplo a seguir, o plug-in OpenSearch sink usa o subobjeto `info/id` para gerar uma ID de documento.

```
sink:
  - opensearch:
       ...
       document_id: info/id
```

Dado o evento a seguir, o pipeline gerará um documento com o campo `_id` definido como `json001`:

```
{
   "fieldA":"arbitrary value",
   "info":{
      "id":"json001",
      "fieldA":"xyz",
      "fieldB":"def"
   }
}
```

Gerando roteamento IDs

Você pode usar a `routing_field` opção no plug-in de OpenSearch coletor para definir o valor de uma propriedade de roteamento de documentos (`_routing`) como um valor de um evento de entrada.

O roteamento é compatível com a sintaxe de ponteiro do JSON, portanto, campos aninhados também estão disponíveis, e não apenas campos de nível superior.

```
sink:
  - opensearch:
       ...
       routing_field: metadata/id
       document_id: id
```

Dado o evento a seguir, o plug-in gerará um documento com o campo `_routing` definido como `abcd`:

```
{
   "id":"123",
   "metadata":{
      "id":"abcd",
      "fieldA":"valueA"
   },
   "fieldB":"valueB"
}
```

Para obter instruções sobre como criar modelos de índice que os pipelines podem usar durante a criação do índice, consulte Modelos de índice.

End-to-end reconhecimento

OpenSearch A ingestão garante a durabilidade e a confiabilidade dos dados, rastreando sua entrega da origem aos sumidouros em pipelines sem estado usando reconhecimento. end-to-end

> **Note**
>
> Atualmente, somente o plug-in de origem do S3 oferece suporte à end-to-end confirmação.

Com a end-to-end confirmação, o plug-in de origem do pipeline cria um conjunto de confirmações para monitorar um lote de eventos. Ele recebe uma confirmação positiva quando esses eventos são enviados com sucesso para seus coletores ou uma confirmação negativa quando nenhum dos eventos pôde ser enviado para seus coletores.

No caso de um evento negativo ou falha de um componente do pipeline, ou se uma fonte não receber uma confirmação, a fonte atinge o tempo limite e toma as medidas necessárias, como tentar novamente ou registrar a falha. Se o pipeline tiver vários coletores ou vários subpipelines configurados, as confirmações em nível de evento serão enviadas somente após o evento ser enviado para todos os coletores em todos os subpipelines. Se um coletor tiver uma DLQ configurada, as end-to-end confirmações também rastrearão eventos gravados na DLQ.

Para ativar a end-to-end confirmação, expanda Opções adicionais na configuração de origem do Amazon S3 e escolha end-to-end Habilitar confirmação de mensagem.

Pressão oposta da origem

Um pipeline pode sofrer contrapressão quando está ocupado processando dados ou se seus sumidouros estão temporariamente inativos ou lentos para ingerir dados. OpenSearch A ingestão tem maneiras diferentes de lidar com a contrapressão, dependendo do plug-in de origem que um pipeline está usando.

Origem HTTP

Os pipelines que usam o plug-in de origem HTTP lidam com a pressão oposta de maneira diferente, dependendo de qual componente do pipeline está congestionado:

- Buffers: quando os buffers estão cheios, o pipeline começa a retornar o status HTTP `REQUEST_TIMEOUT` com o código de erro 408 de volta ao endpoint de origem. À medida que os buffers são liberados, o pipeline começa a processar eventos HTTP novamente.

- Threads de origem: quando todos os threads de origem HTTP estão ocupados executando solicitações e o tamanho da fila de solicitações não processadas excede o número máximo permitido de solicitações, o pipeline começa a retornar o status HTTP `TOO_MANY_REQUESTS` com o código de erro 429 de volta ao endpoint de origem. Quando a fila de solicitações fica abaixo do tamanho máximo permitido, o pipeline começa a processar as solicitações novamente.

OTel fonte

Quando os buffers estão cheios para pipelines que usam OpenTelemetry fontes (OTel registros, OTel métricas e OTel rastreamento), o pipeline começa a retornar o status HTTP `REQUEST_TIMEOUT` com o código de erro 408 para o endpoint de origem. À medida que os buffers são liberados, o pipeline começa a processar eventos novamente.

Origem do S3

Quando os buffers estão cheios para pipelines com uma origem do S3, os pipelines param de processar notificações SQS. À medida que os buffers são liberados, os pipelines começam a processar as notificações novamente.

Se um coletor estiver inativo ou não conseguir ingerir dados e a end-to-end confirmação for ativada para a origem, o pipeline interromperá o processamento das notificações do SQS até receber uma confirmação bem-sucedida de todos os coletores.

Criação de pipelines OpenSearch de ingestão da Amazon

Um pipeline é o mecanismo que o Amazon OpenSearch Ingeston usa para mover dados da fonte (de onde vêm os dados) para o coletor (para onde vão os dados). Na OpenSearch ingestão, o coletor sempre será um único domínio do Amazon OpenSearch Service, enquanto a fonte de seus dados pode ser clientes como Amazon S3, Fluent Bit ou Collector. OpenTelemetry

Para obter mais informações, consulte Pipelines na OpenSearch documentação.

Tópicos

- Pré-requisitos e perfil do IAM necessário
- Permissões obrigatórias do IAM
- Como especificar a versão do pipeline
- Como especificar o caminho de ingestão

- Como criar pipelines
- Acompanhar o status da criação do pipeline
- Trabalhar com esquemas

Pré-requisitos e perfil do IAM necessário

Para criar um pipeline OpenSearch de ingestão, você deve ter os seguintes recursos:

- Uma função do IAM, chamada de função de pipeline, que o OpenSearch Ingestion assume para gravar no coletor. Você pode criar essa função com antecedência ou fazer com que o OpenSearch Inestion a crie automaticamente enquanto você cria o pipeline.
- Um domínio OpenSearch de serviço ou coleção OpenSearch sem servidor para atuar como coletor. Se você estiver gravando em um domínio, ele deverá estar executando a OpenSearch versão 1.0 ou posterior, ou o Elasticsearch 7.4 ou posterior. O coletor deve ter uma política de acesso que conceda as permissões apropriadas à sua perfil de pipeline do IAM.

Para obter instruções sobre como criar esses recursos, consulte os tópicos a seguir:

- the section called “Concedendo acesso aos pipelines aos domínios”
- the section called “Conceder aos pipelines acesso às coleções”

> **Note**
>
> Se você estiver escrevendo para um domínio que usa controle de acesso detalhado, há etapas extras que você precisa concluir. Consulte the section called “Mapear o perfil de pipeline (somente para domínios que usam o controle de acesso refinado)”.

Permissões obrigatórias do IAM

OpenSearch A ingestão usa as seguintes permissões do IAM para criar pipelines:

- `osis:CreatePipeline` – crie um pipeline.
- `osis:ValidatePipeline` – verifica se a configuração do pipeline é válida.
- `iam:CreateRolee iam:AttachPolicy` — Faça com que o OpenSearch Ingestion crie automaticamente a função de pipeline para você.

- `iam:PassRole`— passe a função do pipeline para o OpenSearch Ingestion para que ele possa gravar dados no domínio. Essa permissão deverá ser o recurso de perfil do pipeline ou simplesmente `*` se você planejar usar perfis diferentes em cada pipeline.

Por exemplo, a política a seguir concede permissão para criar um pipeline:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:CreatePipeline",
            "osis:ListPipelineBlueprints",
            "osis:ValidatePipeline"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::111122223333:role/pipeline-role"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachRolePolicy",
            "iam:PassRole"
         ]
      }
   ]
}
```

OpenSearch A ingestão também inclui uma permissão chamada`osis:Ingest`, que é necessária para enviar solicitações assinadas ao pipeline usando o Signature Version 4. Para obter mais informações, consulte the section called “Criação de uma função de ingestão”.

> **Note**
>
> Além disso, o primeiro usuário a criar um pipeline em uma conta precisa ter permissões para a ação `iam:CreateServiceLinkedRole`. Para saber mais, consulte Recurso de perfil de pipeline.

Para obter mais informações sobre cada permissão, consulte Ações, recursos e chaves de condição para OpenSearch ingestão na Referência de autorização de serviço.

Como especificar a versão do pipeline

Ao configurar um pipeline usando o editor de configuração. você deve especificar a versão principal do Data Prepper que o pipeline executará. Para especificar a versão, inclua a opção `version` na configuração do pipeline:

```
version: "2"
log-pipeline:
  source:
    ...
```

Quando você escolhe Criar, a OpenSearch ingestão determina a última versão secundária disponível da versão principal especificada e provisiona o pipeline com essa versão. Por exemplo, se você especificar `version: "2"` e a versão mais recente compatível do Data Prepper for 2.1.1, o OpenSearch Ingestion provisionará seu pipeline com a versão 2.1.1. Não exibimos publicamente a versão secundária que seu pipeline está executando.

Para atualizar seu pipeline quando uma nova versão principal do Data Prepper estiver disponível, edite a configuração do pipeline e especifique a nova versão. Você não pode fazer o downgrade de um pipeline para uma versão anterior.

> **Note**
>
> OpenSearch O Inestion não oferece suporte imediato às novas versões do Data Prepper assim que elas são lançadas. Haverá algum atraso entre o momento em que uma nova versão estará disponível publicamente e o momento em que ela será suportada no OpenSearch Ingestion. Além disso, o OpenSearch Inestion pode explicitamente não oferecer

suporte total a determinadas versões principais ou secundárias. Para obter uma lista abrangente, consulte the section called “Versões do Data Prepper compatíveis”.

Sempre que você fizer uma alteração no pipeline que inicia uma blue/green implantação, o OpenSearch Ingestion pode atualizá-la para a versão secundária mais recente da versão principal que está atualmente configurada para o pipeline. Para obter mais informações, consultethe section called “Implantações azul/verde para atualizações de pipeline”. OpenSearch A ingestão não pode alterar a versão principal do seu pipeline, a menos que você atualize explicitamente a `version` opção na configuração do pipeline.

Como especificar o caminho de ingestão

Para fontes baseadas em pull, como OTel rastreamento e OTel métricas, a OpenSearch ingestão requer a `path` opção adicional na configuração da fonte. O caminho é uma string como `/log/ingest`, que representa o caminho do URI para ingestão. Esse caminho define o URI que você usa para enviar dados para o pipeline.

Por exemplo, digamos que você especifique o seguinte caminho para um pipeline com uma origem HTTP:



Ao ingerir dados no pipeline, você deve especificar o seguinte endpoint na configuração do seu cliente:. `https://pipeline-name-abc123.us-west-2.osis.amazonaws.com/my/test_path`

O caminho deve começar com uma barra (/) e pode conter os caracteres especiais '-', '_', ' . 'e'/', bem como o placeholder `${pipelineName}`. Se você usar `${pipelineName}` (como/ `${pipelineName}/test_path`), o OpenSearch Inestion substituirá a variável pelo nome do subpipeline associado.

Como criar pipelines

Esta seção descreve como criar pipelines OpenSearch de ingestão usando o console OpenSearch de serviço e o. AWS CLI

Console

Para criar um pipeline, faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home e escolha Create pipeline.

Selecione um pipeline em branco ou escolha um esquema de configuração. Os esquemas incluem um pipeline pré-configurado para vários casos de uso comuns. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Escolha Selecionar esquema.

Configuração da fonte

1. Se você estiver começando com um pipeline em branco, selecione uma origem no menu suspenso. As fontes disponíveis podem incluir outras Serviços da AWS fontes ou HTTP. OpenTelemetry Para obter mais informações, consulte the section called “Integração de pipelines”.
2. Dependendo da origem escolhida, defina configurações adicionais para a origem. Por exemplo, para usar o Amazon S3 como origem, você deve especificar a URL da fila do Amazon SQS das mensagens recebidas pelo pipeline. Para obter uma lista dos plug-ins de origem compatíveis e os links para sua documentação, consulte the section called “Plug-ins e opções compatíveis”.
3. Para algumas origens, você deve especificar as Opções de rede de origem. Escolha entre Acesso de VPC ou Acesso público. Se você selecionar Acesso público, vá para a próxima etapa. Se você escolher Acesso à VPC, defina as seguintes configurações:

Configuração	Description
Gerenciamento de endpoints	Escolha se você mesmo quer criar seus endpoints de nuvem privada virtual (VPC) ou deixar que o OpenSearch Ingestion os crie para você. O gerenciamento de endpoints é padronizado para endpoints gerenciados pelo Ingestion. OpenSearch
VPC	Escolha o ID da VPC que você deseja usar. A VPC e o pipeline devem estar na mesma Região da AWS.

Configuração	Description
Sub-redes	Escolha uma ou mais sub-redes. OpenSearch O serviço colocará um endpoint VPC e interfaces de rede elástica nas sub-redes.
Grupos de segurança	Escolha um ou mais grupos de segurança de VPC que permitam que o aplicativo necessário alcance o pipeline de OpenSearch ingestão nas portas (80 ou 443) e protocolos (HTTP ou HTTPs) expostos pelo pipeline.
Opções de anexo de VPC	Se sua origem for um endpoint autogerenciado, conecte seu pipeline a uma VPC. Escolha uma das opções de CIDR padrão fornecidas ou use um CIDR personalizado.

Para saber mais, consulte the section called “Como configurar o acesso à VPC para pipelines”.

4. Escolha Próximo.

Configurar processador

Adicione um ou mais processadores ao pipeline. Os processadores são componentes dentro de um subpipeline que permitem filtrar, transformar e enriquecer eventos antes de publicar registros no coletor de domínios ou coleções. Para obter uma lista dos processadores compatíveis e os links para sua documentação, consulte the section called “Plug-ins e opções compatíveis”.

Você pode selecionar Ações e adicionar o seguinte:

- Roteamento condicional: encaminha eventos para diferentes coletores com base em condições específicas. Para saber mais, consulte Roteamento condicional.
- Subpipeline: cada subpipeline é uma combinação de uma origem, zero ou mais processadores e um coletor. Apenas um único subpipeline pode ter uma origem externa. Todos os outros devem ter origens que sejam outros subpipelines dentro da configuração geral do pipeline. Um único arquivo de configuração de pipeline pode conter de 1 a 10 subpipelines.

Escolha Próximo.

Configurar coletor

Selecione o destino onde o pipeline publica registros. Cada subpipeline deve conter pelo menos um coletor. É possível adicionar até dez coletores a um pipeline.

Para OpenSearch coletores, configure os seguintes campos:

Configuração	Description
Tamanho da política de rede (Somente coletores com tecnologia sem servidor)	Se você selecionou uma coleção OpenSearch sem servidor, insira um nome de política de rede. OpenSearch A ingestão cria a política, se ela não existir, ou a atualiza com uma regra que concede acesso ao VPC endpoint que conecta o pipeline e a coleção. Para obter mais informações, consulte the section called “Conceder aos pipelines acesso às coleções”.
Nome do índice	O nome do índice para o qual o pipeline envia dados. OpenSearch A ingestão cria esse índice se ele ainda não existir.
Opções de mapeamento de índice	Escolha como o pipeline armazena e indexa os documentos e seus campos no OpenSearch coletor. Se você selecionar Mapeamento dinâmico, OpenSearch adicionará campos automaticamente ao indexar um documento . Se você selecionar Personalizar mapeamento, insira um modelo de mapeamento de índice. Para saber mais, consulte Modelos de índice.
Habilitar DLQ	Configure uma fila de mensagens não entregues (DLQ) do Amazon S3 para o pipeline. Para saber mais, consulte the section called “Filas de mensagens não entregues”.
Configurações adicionais	Configure opções avançadas para o OpenSearch coletor. Para saber mais, consulte Configuration options na documentação da Data Prepper DLQ.

Para adicionar um coletor do Amazon S3, escolha Adicionar coletor e Amazon S3. Para saber mais, consulte the section called “Amazon S3 como destino”.

Escolha Próximo.

Configurar pipeline

Defina as seguintes configurações adicionais de pipeline:

Configuração	Description
Nome do pipeline	Insira um nome exclusivo para o pipeline.
Buffer persistente	Um buffer persistente armazena os dados em um buffer baseado em disco em várias zonas de disponibilidade. Para saber mais, consulte the section called “Armazenamento em buffer persistente”. Se você habilitar o buffer persistente, selecione a chave do AWS Key Management Service para criptografar os dados do buffer.
Capacidade do pipeline	A capacidade mínima e máxima do pipeline, em Unidades de OpenSearch computação de ingestão ()OCUs. Para obter mais informações, consulte the section called “Pipelines de escalabilidade”.
Perfis do pipeline	O perfil do IAM que fornece as permissões necessárias para o pipeline gravar no destino e ler origens baseadas em pull. Você mesmo pode criar a função ou fazer com que o OpenSearch Inestion a crie para você com base no caso de uso selecionado. Para obter mais informações, consulte the section called “Configurar funções e usuários”.
Tags	Adicione uma ou mais tags ao pipeline. Para saber mais, consulte the section called “Uso de tags com pipelines”.
Opções de publicação de log	Habilite a publicação de registros do pipeline no Amazon CloudWatch Logs. Recomendamos que você habilite a publicação de logs para poder soluciona r problemas de pipeline com mais facilidade. Para saber mais, consulte the section called “Monitoramento dos logs de pipeline”.

Escolha Avançar, revise a configuração do pipeline e escolha Criar pipeline.

OpenSearch A ingestão executa um processo assíncrono para criar o pipeline. Quando o status do pipeline for `Active`, você pode começar a ingerir dados.

AWS CLI

O comando create-pipeline. aceita a configuração do pipeline como uma string ou como um arquivo .yaml ou .json. Se você fornecer a configuração como uma string, cada nova linha deverá ser escapada com \n. Por exemplo, `"log-pipeline:\n source:\n http:\n processor:\n - grok:\n ...`.

O exemplo de comando a seguir cria um pipeline com a seguinte configuração:

- Mínimo de 4 de ingestão OCUs, máximo de 10 de ingestão OCUs
- Provisionado em uma nuvem privada virtual (VPC)
- Publicação de logs habilitada

```
aws osis create-pipeline \
  --pipeline-name my-pipeline \
  --min-units 4 \
  --max-units 10 \
  --log-publishing-options
 IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="MyLogGroup"} \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml" \
  --pipeline-role-arn  arn:aws:iam::1234456789012:role/pipeline-role
```

OpenSearch A ingestão executa um processo assíncrono para criar o pipeline. Quando o status do pipeline for `Active`, você pode começar a ingerir dados. Para verificar o status do pipeline, use o GetPipelinecomando.

OpenSearch API de ingestão

Para criar um pipeline OpenSearch de ingestão usando a API OpenSearch de ingestão, chame a CreatePipelineoperação.

Depois que seu pipeline for criado com sucesso, você poderá configurar seu cliente e começar a ingerir dados em seu domínio OpenSearch de serviço. Para obter mais informações, consulte the section called “Integração de pipelines”.

Acompanhar o status da criação do pipeline

Você pode acompanhar o status de um pipeline à medida que o OpenSearch Inestion o provisiona e o prepara para ingerir dados.

Console

Depois de criar inicialmente um pipeline, ele passa por vários estágios à medida que o OpenSearch Inestion o prepara para ingerir dados. Para visualizar os vários estágios da criação do pipeline, escolha o nome do pipeline para ver sua página Configurações do pipeline. Em Status, escolha Exibir detalhes.

Um pipeline passa pelos seguintes estágios antes de estar disponível para ingestão de dados:

- Validação: valida a configuração do pipeline. Quando esse estágio estiver concluído, todas as validações serão bem-sucedidas.
- Criação de um ambiente: prepara e provisiona recursos Quando esse estágio estiver concluído, o novo ambiente de pipeline será criado.
- Implantação do pipeline: implanta o pipeline. Quando esse estágio estiver concluído, o pipeline foi implantado com sucesso.
- Verificação da integridade do pipeline: verifica a integridade da pipeline. Quando esse estágio estiver concluído, todas as verificações de integridade serão aprovadas.
- Habilitação de tráfego: permite que o pipeline consuma dados. Quando este estágio for concluído, você pode começar a ingerir dados no pipeline.

CLI

Use o get-pipeline-change-progresscomando para verificar o status de um pipeline. A AWS CLI solicitação a seguir verifica o status de um pipeline chamado`my-pipeline`:

```
aws osis get-pipeline-change-progress \
    --pipeline-name my-pipeline
```

Resposta:

```
{
   "ChangeProgressStatuses": {
      "ChangeProgressStages": [
         {
```

```
            "Description": "Validating pipeline configuration",
            "LastUpdated": 1.671055851E9,
            "Name": "VALIDATION",
            "Status": "PENDING"
         }
      ],
      "StartTime": 1.671055851E9,
      "Status": "PROCESSING",
      "TotalNumberOfStages": 5
   }
}
```

OpenSearch API de ingestão

Para acompanhar o status da criação do pipeline usando a API OpenSearch de ingestão, chame a GetPipelineChangeProgressoperação.

Trabalhar com esquemas

Em vez de criar uma definição de pipeline do zero, você pode usar esquemas de configuração, que são modelos pré-configurados para cenários comuns de ingestão, como Trace Analytics ou logs do Apache. Os esquemas de configuração ajudam você a provisionar pipelines facilmente, sem precisar criar uma configuração do zero.

Console

Como usar um esquema de pipeline

1. Faça login no console de OpenSearch ingestão em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Selecione Criar pipeline.
3. Selecione um esquema na lista de casos de uso e escolha Selecionar esquema. A configuração do pipeline é preenchida com um subpipeline para o caso de uso selecionado.

   O esquema do pipeline não é válido no estado em que se encontra. Você precisa especificar configurações adicionais dependendo da origem selecionada.

CLI

Para obter uma lista de todos os blueprints disponíveis usando o AWS CLI, envie uma list-pipeline-blueprintssolicitação.

```
aws osis list-pipeline-blueprints
```

A solicitação retorna uma lista com todos os esquemas disponíveis.

Para obter informações mais detalhadas sobre um blueprint específico, use o get-pipeline-blueprintcomando:

```
aws osis get-pipeline-blueprint --blueprint-name AWS-ApacheLogPipeline
```

Essa solicitação retorna o conteúdo do esquema do pipeline de log do Apache:

```
{
   "Blueprint":{
      "PipelineConfigurationBody":"###\n  # Limitations: https://docs.aws.amazon.com/
opensearch-service/latest/ingestion/ingestion.html#ingestion-limitations\n###\n###\n
 # apache-log-pipeline:\n    # This pipeline receives logs via http (e.g. FluentBit),
 extracts important values from the logs by matching\n    # the value in the 'log' key
 against the grok common Apache log pattern. The grokked logs are then sent\n    # to
 OpenSearch to an index named 'logs'\n###\n\nversion: \"2\"\napache-log-pipeline:\n
 source:\n    http:\n      # Provide the path for ingestion. ${pipelineName} will be
 replaced with pipeline name configured for this pipeline.\n      # In this case it
 would be \"/apache-log-pipeline/logs\". This will be the FluentBit output URI value.
\n      path: \"/${pipelineName}/logs\"\n  processor:\n    - grok:\n        match:\n
         log: [ \"%{COMMONAPACHELOG_DATATYPED}\" ]\n  sink:\n    - opensearch:\n
   # Provide an AWS OpenSearch Service domain endpoint\n        # hosts: [ \"https://
search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-east-1.es.amazonaws.com\" ]\n        aws:
\n          # Provide the region of the domain.\n          # region: \"us-east-1\"\n
        # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection\n          # serverless: true\n        index: \"logs\"\n        # Enable
 the S3 DLQ to capture any failed requests in an S3 bucket\n        # dlq:\n
  # s3:\n            # Provide an S3 bucket\n            # bucket: \"your-dlq-bucket-
name\"\n            # Provide a key path prefix for the failed requests\n
  # key_path_prefix: \"${pipelineName}/logs/dlq\"\n            # Provide the region
 of the bucket.\n            # region: \"us-east-1\"\n            # Provide a Role
 ARN with access to the bucket. This role should have a trust relationship with osis-
pipelines.amazonaws.com\n"
      "BlueprintName":"AWS-ApacheLogPipeline"
   }
}
```

OpenSearch API de ingestão

Para obter informações sobre esquemas de pipeline usando a API de OpenSearch ingestão, use as operações ListPipelineBlueprintse. GetPipelineBlueprint

Visualizando pipelines OpenSearch de ingestão da Amazon

Você pode ver os detalhes sobre um pipeline de OpenSearch ingestão da Amazon usando a API Console de gerenciamento da AWS AWS CLI, a ou a API OpenSearch de ingestão.

Console

Para visualizar um pipeline

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. (Opcional) Para visualizar pipelines com um status específico, escolha Qualquer status e selecione um status para filtrar.

   Um pipeline pode ter os seguintes status:

   - `Active`: o pipeline está ativo e pronto para ingerir dados.
   - `Creating`: o pipeline está sendo criado.
   - `Updating`: o pipeline está sendo atualizado.
   - `Deleting`: o pipeline está sendo excluído.
   - `Create failed`: o pipeline não pôde ser criado.
   - `Update failed`: o pipeline não pôde ser atualizado.
   - `Stop failed`: não foi possível parar o pipeline.
   - `Start failed`: o pipeline não pôde ser iniciado.
   - `Stopping`: o pipeline está sendo interrompido.
   - `Stopped`: o pipeline está parado e pode ser reiniciado a qualquer momento.
   - `Starting`: o pipeline está sendo iniciado.

Você não é cobrado pela ingestão OCUs quando um funil está nos `Create failed` estados`Creating`,`Deleting`, e. `Stopped`

CLI

Para visualizar pipelines usando o AWS CLI, envie uma solicitação list-pipelines:

```
aws osis list-pipelines
```

A solicitação retorna uma lista de todos os pipelines existentes:

```
{
    "NextToken": null,
    "Pipelines": [
        {,
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 4,
            "MinUnits": 2,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/log-pipeline",
            "PipelineName": "log-pipeline",
            "Status": "ACTIVE",
            "StatusReason": {
                "Description": "The pipeline is ready to ingest data."
            }
        },
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 2,
            "MinUnits": 8,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/another-
pipeline",
            "PipelineName": "another-pipeline",
            "Status": "CREATING",
            "StatusReason": {
                "Description": "The pipeline is being created. It is not able to ingest
 data."
            }
        }
    ]
}
```

Para obter informações sobre um único pipeline, use o comando get-pipeline:

```
aws osis get-pipeline --pipeline-name "my-pipeline"
```

A solicitação retorna informações de configuração para o pipeline especificado:

```
{
    "Pipeline": {
        "PipelineName": "my-pipeline",
        "PipelineArn": "arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline",
        "MinUnits": 9,
        "MaxUnits": 10,
        "Status": "ACTIVE",
        "StatusReason": {
            "Description": "The pipeline is ready to ingest data."
        },
        "PipelineConfigurationBody": "log-pipeline:\n source:\n http:\n processor:\n
 - grok:\n match:\nlog: [ '%{COMMONAPACHELOG}' ]\n - date:\n from_time_received: true
\n destination: \"@timestamp\"\n  sink:\n - opensearch:\n hosts: [ \"https://search-
mdp-performance-test-duxkb4qnycd63rpy6svmvyvfpi.us-east-1.es.amazonaws.com\" ]\n index:
 \"apache_logs\"\n aws_sts_role_arn: \"arn:aws:iam::123456789012:role/my-domain-role
\"\n  aws_region: \"us-east-1\"\n  aws_sigv4: true",,
        "CreatedAt": "2022-10-01T15:28:05+00:00",
        "LastUpdatedAt": "2022-10-21T21:41:08+00:00",
        "IngestEndpointUrls": [
            "my-pipeline-123456789012.us-east-1.osis.amazonaws.com"
        ]
    }
}
```

OpenSearch API de ingestão

Para visualizar os pipelines de OpenSearch ingestão usando a API OpenSearch de ingestão, chame as operações e. ListPipelinesGetPipeline

Atualização dos pipelines OpenSearch de ingestão da Amazon

Você pode atualizar os pipelines OpenSearch de ingestão da Amazon usando a API Console de gerenciamento da AWS AWS CLI, a ou a API de OpenSearch ingestão. OpenSearch A ingestão inicia uma blue/green implantação quando você atualiza a configuração de um pipeline. Para obter mais informações, consulte the section called “Implantações azul/verde para atualizações de pipeline”.

Tópicos

- Considerações

- Permissões obrigatórias
- Atualizar pipelines
- Implantações azul/verde para atualizações de pipeline

Considerações

Considere o seguinte ao atualizar um pipeline:

- Não é possível atualizar o nome nem as configurações de rede.
- Se o pipeline gravar em um coletor de domínio da VPC, você não pode voltar e alterar o coletor para um domínio de VPC diferente após a criação do pipeline. Você deve excluir e recriar o pipeline com o novo coletor. Você ainda pode mudar o coletor de um domínio da VPC para um domínio público, de um domínio público para um domínio VPC ou de um domínio público para outro domínio público.
- Você pode alternar o coletor do pipeline a qualquer momento entre um domínio OpenSearch de serviço público e uma OpenSearch coleção sem servidor.
- Quando você atualiza a configuração de origem, processador ou coletor de um pipeline, o OpenSearch Ingestion inicia uma blue/green implantação. Para obter mais informações, consulte the section called “Implantações azul/verde para atualizações de pipeline”.
- Quando você atualiza a configuração de origem, processador ou coletor de um pipeline, o OpenSearch Ingestion atualiza automaticamente seu pipeline para a versão secundária mais recente compatível da versão principal do Data Prepper que o pipeline está executando. Esse processo mantém seu pipeline atualizado com as últimas correções de bugs e melhorias de desempenho.
- Você ainda pode fazer atualizações no seu pipeline quando ele estiver parado.

Permissões obrigatórias

OpenSearch A ingestão usa as seguintes permissões do IAM para atualizar os pipelines:

- `osis:UpdatePipeline` – atualizar um pipeline.
- `osis:ValidatePipeline` – verifica se a configuração do pipeline é válida.
- `iam:PassRole`— passe a função do pipeline para o OpenSearch Ingestion para que ele possa gravar dados no domínio. Essa permissão será necessária apenas se você estiver atualizando a

configuração de pipeline, não se estiver modificando outras configurações, como publicação de log ou limites de capacidade.

Por exemplo, a política a seguir concede permissão para atualizar um pipeline:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "osis:UpdatePipeline",
                "osis:ValidatePipeline"
            ]
        },
        {
            "Resource": [
                "arn:aws:iam::111122223333:role/pipeline-role"
            ],
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ]
        }
    ]
}
```

Atualizar pipelines

Você pode atualizar os pipelines OpenSearch de ingestão da Amazon usando a API Console de gerenciamento da AWS AWS CLI, a ou a API de OpenSearch ingestão.

Console

Como atualizar um pipeline

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Escolhe um pipeline para abrir suas configurações. Em seguida, escolha uma das opções de Editar.
3. Quando terminar de fazer as alterações, selecione Salvar.

CLI

Para atualizar um pipeline usando o AWS CLI, envie uma solicitação update-pipeline. O exemplo de solicitação a seguir carrega um novo arquivo de configuração e atualiza os valores de capacidade mínima e máxima:

```
aws osis update-pipeline \
  --pipeline-name "my-pipeline" \
  --pipline-configuration-body "file://new-pipeline-config.yaml" \
  --min-units 11 \
  --max-units 18
```

OpenSearch API de ingestão

Para atualizar um pipeline OpenSearch de ingestão usando a API OpenSearch de ingestão, chame a UpdatePipelineoperação.

Implantações azul/verde para atualizações de pipeline

OpenSearch A ingestão inicia um processo de implantação azul/verde quando você atualiza a configuração de um pipeline.

Blue/green refers to the practice of creating a new environment for pipeline updates and routing traffic to the new environment after those updates are complete. The practice minimizes downtime and maintains the original environment in the event that deployment to the new environment is unsuccessful. Blue/greenas implantações em si não têm nenhum impacto no desempenho, mas o desempenho pode mudar se a configuração do pipeline mudar de uma forma que altere o desempenho.

OpenSearch A ingestão bloqueia o escalonamento automático blue/green durante as implantações. Você continua sendo cobrado somente pelo tráfego do pipeline antigo até que ele seja redirecionado para o novo pipeline. Depois que o tráfego for redirecionado, você será cobrado apenas pelo novo pipeline. Você nunca será cobrado por dois pipelines simultaneamente.

Quando você atualiza a configuração de origem, processador ou coletor de um pipeline, o OpenSearch Ingestion pode atualizar automaticamente seu pipeline para a versão secundária mais recente compatível da versão principal que o pipeline está executando. Por exemplo, você pode ter `version: "2"` em sua configuração de pipeline, e a OpenSearch Ingestion inicialmente provisionou o pipeline com a versão 2.1.0. Quando o suporte para a versão 2.1.1 é adicionado e você faz uma alteração na configuração do pipeline, o OpenSearch Ingestion atualiza seu pipeline para a versão 2.1.1.

Esse processo mantém seu pipeline atualizado com as últimas correções de bugs e melhorias de desempenho. OpenSearch A ingestão não pode atualizar a versão principal do seu pipeline, a menos que você altere manualmente a `version` opção na configuração do pipeline.

Gerenciando os custos do pipeline OpenSearch de ingestão da Amazon

Você pode iniciar e interromper os pipelines de ingestão no Amazon OpenSearch Ingestion para controlar o fluxo de dados com base nas suas necessidades. Parar um pipeline interrompe o processamento de dados e, ao mesmo tempo, preserva as configurações, para que você possa reiniciá-lo sem reconfigurá-lo. Isso pode ajudar a otimizar custos, gerenciar o uso de recursos ou solucionar problemas. Quando você interrompe um pipeline, a OpenSearch ingestão não processa os dados recebidos, mas os dados ingeridos anteriormente permanecem disponíveis no. OpenSearch

Iniciar e parar simplifica os processos de configuração e destruição dos pipelines que você usa em desenvolvimento, teste ou atividades afins que não exigem disponibilidade contínua. Enquanto seu pipeline estiver interrompido, você não será cobrado por nenhuma hora da OCU de ingestão. Você ainda pode atualizar pipelines interrompidos, e eles recebem atualizações automáticas de versões secundárias e patches de segurança.

Parar e iniciar um pipeline resultará no reprocessamento de todos os dados desde o início para pipelines baseados em pull (DDB, S3, DocDB, etc). Quando você interrompe um pipeline, todos os VPC endpoints gerenciados por serviços criados pelo pipeline são removidos. Para pipelines com VPC endpoints autogerenciados, você deve recriar o VPC endpoint em sua conta

ao reiniciar o pipeline. Para obter mais informações, consulte the section called “Endpoints da VPC autogerenciados”.

> **Note**
>
> Se o pipeline tiver excesso de capacidade, mas precisar permanecer operacional, considere ajustar seus limites máximos de capacidade em vez de pará-lo e reiniciá-lo. Isso pode ajudar a gerenciar os custos e, ao mesmo tempo, garantir que o pipeline continue processando os dados com eficiência. Consulte mais detalhes em the section called “Pipelines de escalabilidade”.

Os tópicos a seguir explicam como iniciar e interromper pipelines usando a API Console de gerenciamento da AWS AWS CLI, e OpenSearch Ingestion.

Tópicos

- Interrompendo um pipeline OpenSearch de ingestão da Amazon
- Iniciando um pipeline de OpenSearch ingestão da Amazon

Interrompendo um pipeline OpenSearch de ingestão da Amazon

Para usar um pipeline de OpenSearch ingestão ou realizar a administração, você sempre começa com um pipeline ativo, depois interrompe o pipeline e, em seguida, inicia o pipeline novamente. Enquanto seu pipeline estiver interrompido, você não precisará pagar por nenhuma hora da OCU de ingestão.

Console

Para interromper um pipeline

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Escolha um funil. Você pode executar a operação de interrupção nesta página ou navegar até a página de detalhes do pipeline de banco de dados que você deseja interromper.
3. Em Ações, escolha Parar pipeline.

   Se um pipeline não puder ser interrompido e iniciado, a ação Stop pipeline não estará disponível.

AWS CLI

Para interromper um pipeline usando o AWS CLI, chame o comando stop-pipeline com os seguintes parâmetros:

- `--pipeline-name` – nome do pipeline.

Example

```
aws osis stop-pipeline --pipeline-name my-pipeline
```

OpenSearch API de ingestão

Para interromper um pipeline usando a API OpenSearch de ingestão, chame a StopPipelineoperação com o seguinte parâmetro:

- `PipelineName` – nome do pipeline.

Iniciando um pipeline de OpenSearch ingestão da Amazon

Você sempre inicia um pipeline de OpenSearch ingestão começando com um pipeline que já está parado. O pipeline mantém suas configurações, como limites de capacidade, configurações de rede e opções de publicação de logs.

A reinicialização de um pipeline normalmente leva vários minutos.

Console

Para iniciar um pipeline

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Escolha um funil. É possível executar a operação de início nesta página ou navegar até a página de detalhes do pipeline que você deseja iniciar.
3. Em Ações, escolha Iniciar pipeline.

AWS CLI

Para iniciar um pipeline usando o AWS CLI, chame o comando start-pipeline com os seguintes parâmetros:

- `--pipeline-name` – nome do pipeline.

Example

```
aws osis start-pipeline --pipeline-name my-pipeline
```

OpenSearch API de ingestão

Para iniciar um pipeline OpenSearch de ingestão usando a API OpenSearch de ingestão, chame a StartPipelineoperação com o seguinte parâmetro:

- `PipelineName` – nome do pipeline.

Excluindo pipelines de OpenSearch ingestão da Amazon

Você pode excluir um pipeline de OpenSearch ingestão da Amazon usando a API Console de gerenciamento da AWS AWS CLI, a ou a API OpenSearch de ingestão. Você não pode excluir um pipeline quando tem um status de `Creating` ou `Updating`.

Console

Para excluir um pipeline

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Selecione o pipeline que você deseja excluir e escolha Ações, Excluir.
3. Confirme a exclusão e escolha Excluir.

CLI

Para excluir um pipeline usando o AWS CLI, envie uma solicitação de exclusão do pipeline:

```
aws osis delete-pipeline --pipeline-name "my-pipeline"
```

OpenSearch API de ingestão

Para excluir um pipeline OpenSearch de ingestão usando a API OpenSearch de ingestão, chame a DeletePipelineoperação com o seguinte parâmetro:

- `PipelineName` – nome do pipeline.

Plugins e opções compatíveis para pipelines OpenSearch de ingestão da Amazon

O Amazon OpenSearch Ingestion oferece suporte a um subconjunto de fontes, processadores e coletores dentro do Data Prepper de código OpenSearch aberto. Além disso, há algumas restrições que o OpenSearch Inestion impõe às opções disponíveis para cada plug-in compatível. As seções a seguir descrevem os plug-ins e as opções associadas compatíveis com o OpenSearch Inestion.

> **Note**
>
> OpenSearch A ingestão não oferece suporte a nenhum plug-in de buffer porque configura automaticamente um buffer padrão. Você receberá um erro de validação se incluir um buffer na configuração do pipeline.

Tópicos

- Plug-ins compatíveis
- Processadores sem estado x processadores com estado
- Requisitos e restrições de configuração

Plug-ins compatíveis

OpenSearch O Inestion é compatível com os seguintes plug-ins do Data Prepper:

Sources (Origens):

- DocumentDB
- DynamoDB

- HTTP
- Kafka
- Kinesis
- OpenSearch
- OTel logs
- OTel métricas
- OTel traço
- S3

Processadores:

- Adicionar entradas
- Aggregate
- Detector de anomalias
- AWS Lambda
- Converter tipo de entrada
- Copiar valores
- CSV
- Data
- Atraso
- Descomprimir
- Excluir entradas
- Dissecar
- Descarte eventos
- Achatar
- IP geográfico
- Grok
- Valor da chave
- Lista para mapa
- String em minúsculas

- Mapear para a lista
- Mudar evento (série de processadores)
- Mudar string (série de processadores)
- Obfuscate (Ofuscar)
- OTel métricas
- OTel grupo de rastreamento
- OTel traço
- Analisar Ion
- Parse JSON (Analisar JSON)
- Analisar XML
- Renomear chaves
- Selecionar entradas
- Mapa de serviço
- Dividir evento
- Dividir string
- Conversor de strings
- Substituir string
- Rastrear o remetente entre os pares
- Translate
- Cortar string
- Truncar
- String em maiúsculas
- Agente de usuário
- Gravar JSON

Coletores:

- OpenSearch(compatível com OpenSearch Service, OpenSearch Serverless e Elasticsearch 6.8 ou posterior)
- S3

Codecs Sink:

- Avro
- NDJSON
- JSON
- Parquet

Processadores sem estado x processadores com estado

Os processadores sem estado realizam operações como transformações e filtragem, enquanto os processadores com estado realizam operações como agregações, que lembram o resultado da execução anterior. OpenSearch A ingestão suporta os processadores com estado Aggregate e Service-MAP. Todos os outros processadores compatíveis são sem estado.

Para pipelines que contêm somente processadores sem estado, o limite máximo de capacidade é 96 de ingestão. OCUs Se um pipeline contiver algum processador com estado, o limite máximo de capacidade será 48 de ingestão OCUs. No entanto, se um pipeline tiver o buffer persistente ativado, ele poderá ter no máximo 384 Ingestão OCUs com apenas processadores sem estado ou 192 Ingestão OCUs se contiver algum processador com estado. Para obter mais informações, consulte the section called “Pipelines de escalabilidade”.

End-to-end o reconhecimento só é suportado para processadores sem estado. Para obter mais informações, consulte the section called “End-to-end reconhecimento”.

Requisitos e restrições de configuração

A menos que especificado de outra forma abaixo, todas as opções descritas na referência de configuração do Data Prepper para os plug-ins compatíveis listados acima são permitidas nos pipelines OpenSearch de ingestão. As seções a seguir explicam as restrições que o OpenSearch Inestion impõe a determinadas opções de plug-in.

> **Note**
>
> OpenSearch A ingestão não oferece suporte a nenhum plug-in de buffer porque configura automaticamente um buffer padrão. Você receberá um erro de validação se incluir um buffer na configuração do pipeline.

Muitas opções são configuradas e gerenciadas internamente pelo OpenSearch Inestion, como e. `authentication` `acm_certificate_arn` Outras opções, como `thread_count` e `request_timeout`, sofrem impactos no desempenho se alteradas manualmente. Portanto, esses valores são definidos internamente para garantir o desempenho ideal de seus pipelines.

Por fim, algumas opções não podem ser passadas para a OpenSearch Ingestão, como `ism_policy_file` `esink_template`, porque são arquivos locais quando executados no Data Prepper de código aberto. Não oferece suporte a esses valores.

Tópicos

- Opções gerais de pipeline
- Processador Grok
- Origem HTTP
- OpenSearch pia
- OTel fonte de métricas, fonte de OTel rastreamento e origem OTel de registros
- OTel processador de grupos de rastreamento
- OTel processador de rastreamento
- Processador de mapas de serviços
- Origem do S3

Opções gerais de pipeline

As seguintes opções gerais de pipeline são definidas pela OpenSearch Ingestão e não são suportadas nas configurações de pipeline:

- `workers`
- `delay`

Processador Grok

As seguintes opções do processador Grok não são compatíveis:

- `patterns_directories`
- `patterns_files_glob`

Origem HTTP

O plug-in de origem HTTP tem os seguintes requisitos e restrições:

- A opção `path` é obrigatória. O caminho é uma string como `/log/ingest`, que representa o caminho do URI para ingestão de logs. Esse caminho define o URI que você usa para enviar dados para o pipeline. Por exemplo, .`https://log-pipeline.us-west-2.osis.amazonaws.com`*`/log/ingest`* O caminho deve começar com uma barra (/) e pode conter os caracteres especiais '-', '_', ' . 'e'/', bem como o placeholder `${pipelineName}`.
- As seguintes opções de origem HTTP são definidas pelo OpenSearch Inestion e não são compatíveis com as configurações de pipeline:
  - `port`
  - `ssl`
  - `ssl_key_file`
  - `ssl_certificate_file`
  - `aws_region`
  - `authentication`
  - `unauthenticated_health_check`
  - `use_acm_certificate_for_ssl`
  - `thread_count`
  - `request_timeout`
  - `max_connection_count`
  - `max_pending_requests`
  - `health_check_service`
  - `acm_private_key_password`
  - `acm_certificate_timeout_millis`
  - `acm_certificate_arn`

OpenSearch pia

O plug-in OpenSearchsink tem os seguintes requisitos e limitações.

- A opção `aws` é obrigatória e deve conter as opções a seguir.
  - `sts_role_arn`

  - `region`
  - `hosts`
  - `serverless`(se o coletor for uma OpenSearch coleção sem servidor)
- A opção `sts_role_arn` deve apontar para a mesma função para cada coletor em um arquivo de definição YAML.
- A `hosts` opção deve especificar um endpoint OpenSearch de domínio de serviço ou um endpoint de coleta OpenSearch sem servidor. Você não pode especificar um endpoint personalizado para um domínio; ele deve ser o endpoint padrão.
- Se a opção `hosts` for um endpoint de coleta de tecnologia sem servidor, você deverá definir a opção `serverless` como `true`. Além disso, se o arquivo de definição YAML contiver a opção `index_type`, ela deverá ser definida como `management_disabled`, caso contrário, a validação falhará.
- As seguintes opções não são compatíveis:
  - `username`
  - `password`
  - `cert`
  - `proxy`
  - `dlq_file`: se quiser transferir eventos com falha para uma fila de mensagens não entregues (DLQ), você deve usar a opção `dlq` e especificar um bucket do S3.
  - `ism_policy_file`
  - `socket_timeout`
  - `template_file`
  - `insecure`

OTel fonte de métricas, fonte de OTel rastreamento e origem OTel de registros

Os plug-ins de origem de OTel métricas, origem de OTel rastreamento e origem de OTel registros têm os seguintes requisitos e limitações:

- A opção `path` é obrigatória. O caminho é uma string como `/log/ingest`, que representa o caminho do URI para ingestão de logs. Esse caminho define o URI que você usa para enviar dados para o pipeline. Por exemplo, .`https://log-pipeline.us-`

`west-2.osis.amazonaws.com/log/ingest` O caminho deve começar com uma barra (/) e pode conter os caracteres especiais '-', '_', ' . 'e'/', bem como o placeholder `${pipelineName}`.

- As opções a seguir são definidas pelo OpenSearch Ingestion e não são compatíveis com as configurações de pipeline:
  - `port`
  - `ssl`
  - `sslKeyFile`
  - `sslKeyCertChainFile`
  - `authentication`
  - `unauthenticated_health_check`
  - `useAcmCertForSSL`
  - `unframed_requests`
  - `proto_reflection_service`
  - `thread_count`
  - `request_timeout`
  - `max_connection_count`
  - `acmPrivateKeyPassword`
  - `acmCertIssueTimeOutMillis`
  - `health_check_service`
  - `acmCertificateArn`
  - `awsRegion`

OTel processador de grupos de rastreamento

O processador do grupo de OTel rastreamento tem os seguintes requisitos e limitações:

- A opção `aws` é obrigatória e deve conter as opções a seguir.
  - `sts_role_arn`
  - `region`
  - `hosts`
- A `sts_role_arn` opção especifica a mesma função do pipeline que você especifica na configuração do OpenSearch coletor.

- As opções `username`, `password`, `cert` e `insecure` não são compatíveis.
- A opção `aws_sigv4` é obrigatória e deve ser definida como verdadeira.
- A `serverless` opção dentro do plug-in do OpenSearch coletor não é suportada. Atualmente, o processador de grupos de rastreamento da Otel não funciona com coleções sem OpenSearch servidor.
- O número de processadores `otel_trace_group` dentro do corpo de configuração do pipeline não pode exceder 8.

OTel processador de rastreamento

O processador de OTel rastreamento tem os seguintes requisitos e limitações:

- O valor da opção `trace_flush_interval` não pode exceder 300 segundos.

Processador de mapas de serviços

O processador Service-map (Mapa de serviços) apresenta os seguintes requisitos e limitações:

- O valor da opção `window_duration` não pode exceder 300 segundos.

Origem do S3

O plug-in de origem do S3 tem os seguintes requisitos e limitações:

- A opção `aws` é obrigatória e deve conter as opções `region` e `sts_role_arn`.
- O valor da opção `records_to_accumulate` não pode exceder 200.
- O valor da opção `maximum_messages` não pode exceder 10.
- Se especificada, a opção `disable_bucket_ownership_validation` deve ser definida como falsa.
- Se especificada, a opção `input_serialization` deve ser definida como `parquet`.

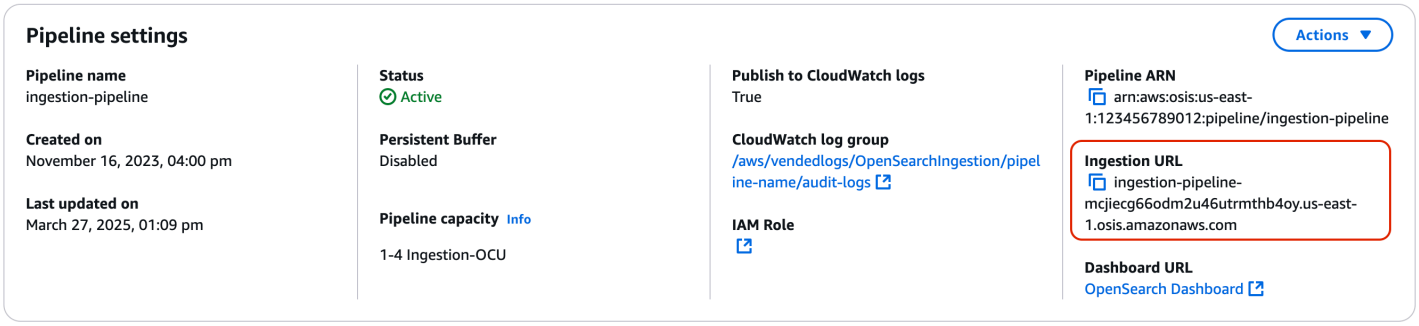
Integração dos pipelines OpenSearch de ingestão da Amazon com outros serviços e aplicativos

Para ingerir dados com sucesso em um pipeline de OpenSearch ingestão da Amazon, você deve configurar seu aplicativo cliente (a fonte) para enviar dados para o endpoint do pipeline. Sua fonte pode ser clientes como os registros do Fluent Bit, o OpenTelemetry Collector ou um simples bucket S3. A configuração exata é diferente para cada cliente.

As diferenças importantes durante a configuração da fonte (em comparação com o envio de dados diretamente para um domínio de OpenSearch serviço ou coleção OpenSearch sem servidor) são o nome do AWS serviço (`osis`) e o endpoint do host, que deve ser o endpoint do pipeline.

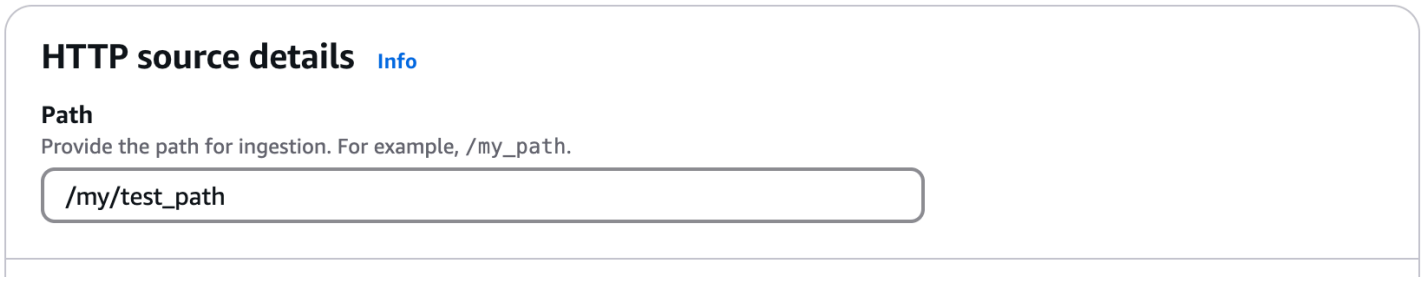
Criar o endpoint de ingestão

Para ingerir dados em um pipeline, envie-os para o endpoint de ingestão. Para localizar a URL de ingestão, navegue até a página Configurações do pipeline e copie a URL de ingestão.



Para criar o endpoint de ingestão completo para fontes baseadas em pull, como OTel rastreamento e OTel métricas, adicione o caminho de ingestão da configuração do pipeline ao URL de ingestão.

Por exemplo, digamos que a configuração do pipeline tem o seguinte caminho de ingestão:



O endpoint de ingestão completo, que você especifica na configuração do seu cliente, terá o seguinte formato: `https://ingestion-pipeline-abcdefg.us-east-1.osis.amazonaws.com/my/test_path`.

Criação de uma função de ingestão

Todas as solicitações de OpenSearch ingestão devem ser assinadas com o Signature versão 4. No mínimo, a função que assina a solicitação deve receber permissão para a `osis:Ingest` ação, o que permite que ela envie dados para um pipeline OpenSearch de ingestão.

Por exemplo, a política a seguir AWS Identity and Access Management (IAM) permite que a função correspondente envie dados para um único pipeline:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "osis:Ingest",
      "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/pipeline-name"
    }
  ]
}
```

> **Note**
>
> Para usar a função em todos os pipelines, substitua o ARN no elemento `Resource` por um caractere curinga (*).

Concessão de acesso de ingestão entre contas

> **Note**
>
> Você só pode fornecer acesso de ingestão entre contas para pipelines públicos, não para pipelines de VPC.

Talvez seja necessário ingerir dados em um pipeline de outro Conta da AWS, como uma conta que hospeda seu aplicativo de origem. Se a entidade principal que está gravando em um pipeline estiver em uma conta diferente do próprio pipeline, você precisará configurar a entidade principal para confiar em outro perfil do IAM para ingerir dados no pipeline.

Como configurar permissões de ingestão entre contas

1. Crie a função de ingestão com `osis:Ingest` permissão (descrita na seção anterior) dentro da Conta da AWS mesma função do pipeline. Para obter instruções, consulte Como criar perfis do IAM.
2. Vincule uma política de confiança à função de ingestão que permita que uma entidade principal em outra conta a assuma:

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [{
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root"
         },
        "Action": "sts:AssumeRole"
     }]
   }
   ```

3. Na outra conta, configure seu aplicativo cliente (por exemplo, Fluent Bit) para assumir a função de ingestão. Para que isso funcione, a conta do aplicativo deve conceder permissões ao usuário ou à função do aplicativo para assumir a função de ingestão.

   O exemplo a seguir de política baseada em identidade permite que a entidade principal anexada assuma o `ingestion-role` a partir da conta do pipeline:

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
   ```

```
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::111122223333:role/ingestion-role"
    }
  ]
}
```

O aplicativo cliente pode então usar a AssumeRoleoperação para assumir `ingestion-role` e ingerir dados no pipeline associado.

Usando um pipeline de OpenSearch ingestão com os serviços da Atlassian

Você pode usar os plug-ins de origem do Atlassian Jira e do Confluence para ingerir dados dos serviços da Atlassian em seu pipeline de ingestão. OpenSearch Essas integrações permitem que você crie uma base de conhecimentos unificada pesquisável sincronizando projetos do Jira e espaços do Confluence inteiros, mantendo a relevância em tempo real por meio de monitoramento contínuo e sincronização automática das atualizações.

Integrating with Jira

Transforme sua experiência com o Jira com poderosos recursos de pesquisa contextual integrando seu conteúdo do Jira em. OpenSearch O plugin de origem Atlassian Jira da Data Prepper permite criar uma base de conhecimentos unificada e pesquisável sincronizando projetos inteiros do Jira, mantendo a relevância em tempo real por meio de monitoramento contínuo e sincronização automática de atualizações. Essa integração permite a sincronização de dados com opções flexíveis de filtragem por projetos, tipos de problemas e status específicos, garantindo que somente as informações necessárias sejam importadas.

Para garantir uma conectividade segura e confiável, o plug-in oferece suporte a vários métodos de autenticação, incluindo autenticação e OAuth2 autenticação básicas de chave de API, com a segurança adicional de gerenciar credenciais usando um segredo armazenado AWS Secrets Manager. Ele também conta com renovação automática de tokens para acesso ininterrupto, garantindo operação contínua. Baseada na API da Atlassian versão 2, essa integração permite que as equipes desbloqueiem informações valiosas de seus dados do Jira por meio dos recursos avançados OpenSearch de pesquisa.

Integrating with Confluence

Melhore os recursos de colaboração e gerenciamento de conhecimento da sua equipe integrando o conteúdo do Atlassian Confluence por OpenSearch meio do plug-in de origem Confluence

do Data Prepper. Essa integração permite criar um repositório coletivo de conhecimentos, centralizado e pesquisável, melhorando a descoberta de informações e a produtividade da equipe. Ao sincronizar o conteúdo do Confluence e monitorar continuamente as atualizações, o plug-in garante que seu OpenSearch índice permaneça up-to-date abrangente.

A integração oferece opções flexíveis de filtragem, permitindo que você importe seletivamente conteúdo de espaços ou tipos de página específicos, adaptando o conteúdo sincronizado às necessidades da sua organização. O plug-in oferece suporte à chave básica de API e aos métodos de OAuth2 autenticação, com a opção de gerenciar credenciais com segurança por meio de. AWS Secrets Manager O atributo de renovação automática de tokens do plug-in garante acesso ininterrupto e operação sem falhas. Baseada na API Confluence da Atlassian, essa integração permite que as equipes aproveitem os recursos de pesquisa avançada OpenSearch do Confluence em todo o conteúdo do Confluence, aprimorando a acessibilidade e a utilização das informações na organização.

Tópicos

- Pré-requisitos
- Configurar um perfil de pipeline
- Configuração do pipeline de conector do Jira
- Configuração do pipeline de conector do Confluence
- Consistência de dados
- Limitações
- Métricas CloudWatch para conectores Atlassian
- Conectando um pipeline de OpenSearch ingestão da Amazon ao Atlassian Jira ou ao Confluence usando 2.0 OAuth

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, conclua as seguintes etapas:

1. Prepare as credenciais para seu site Jira escolhendo uma das seguintes opções. OpenSearch A ingestão requer apenas `ReadOnly` autorização para o conteúdo.
   a. Opção 1, chave de API: faça login na sua conta da Atlassian e use as informações do seguinte tópico para gerar a chave de API:

- Gerenciar tokens de API para a sua conta da Atlassian

b. Opção 2: OAuth2 — Faça login na sua conta Atlassian e use as informações em. the section called “Conectando um pipeline de OpenSearch ingestão da Amazon ao Atlassian Jira ou ao Confluence usando 2.0 OAuth”

2. Crie um segredo no AWS Secrets Manager para armazenar as credenciais criadas na etapa anterior. Faça as seguintes escolhas ao seguir o procedimento:
   - Em Tipo de segredo, escolha Outro tipo de segredo.
   - Em Pares de chave/valor, crie os seguintes pares, dependendo do tipo de autorização selecionado:

   API key

   ```
   {
      "username": user-name-usualy-email-id,
      "password": api-key
   }
   ```

   OAuth 2.0

   ```
   {
      "clientId": client-id
      "clientSecret": client-secret
      "accessKey": access-key
      "refreshKey": refresh-key
   }
   ```

   Depois de criar o segredo, copie o nome do recurso da Amazon (ARN) do segredo. Você o incluirá na política de permissões de perfis de pipeline.

Configurar um perfil de pipeline

O perfil passado no pipeline deve ter a seguinte política anexada para leitura e gravação no segredo criado na seção de pré-requisitos.

JSON

```
{
    "Version":"2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "SecretReadWrite",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetResourcePolicy",
                "secretsmanager:GetSecretValue",
                "secretsmanager:DescribeSecret",
                "secretsmanager:PutSecretValue",
                "secretsmanager:ListSecretVersionIds"
            ],
            "Resource": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name-random-6-characters"
        }
    ]
}
```

O perfil também deve ter uma política anexada para acessar e gravar no coletor escolhido. Por exemplo, se você escolher OpenSearch como coletor, a política será semelhante à seguinte:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "OpenSearchWritePolicy",
            "Effect": "Allow",
            "Action": "aoss:*",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        }
    ]
}
```

Configuração do pipeline de conector do Jira

É possível usar um esquema pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Substitua *placeholder values* por suas próprias informações.

```
version: "2"
extension:
  aws:
    secrets:
      jira-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-jira-pipeline:
  source:
    jira:
      # We only support one host url for now
      hosts: ["jira-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your jira account
        basic:
          username: ${{aws_secrets:jira-account-credentials:username}}
          password: ${{aws_secrets:jira-account-credentials:password}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the below link to generate these keys.
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require, write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:jira-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:jira-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:jira-account-credentials:accessToken}}
          # refresh_token: ${{aws_secrets:jira-account-credentials:refreshToken}}
      filter:
        project:
          key:
            include:
              # This is not project name.
              # It is an alphanumeric project key that you can find under project
 details in Jira.
              - "project-key"
```

```
              - "project-key"
            # exclude:
              # - "project-key"
              # - "project-key"
        issue_type:
          include:
            - "issue-type"
            # - "Story"
            # - "Bug"
            # - "Task"
         # exclude:
             # - "Epic"
        status:
          include:
            - "ticket-status"
            # - "To Do"
            # - "In Progress"
            # - "Done"
         # exclude:
           # - "Backlog"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
        index: "index_${getMetadata(\"project\")}"
        # Ensure adding unique document id which is the unique ticket id in this case
        document_id: '${/id}'
        aws:
          # Provide a Role ARN with access to the domain. This role should have a trust
 relationship with osis-pipelines.amazonaws.com
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
```

```
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
compression setting is changed in the domain.
        # See the section called “Compactação de solicitações HTTP”
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the region of the bucket.
            region: "us-east-1"
            # Provide a Role ARN with access to the bucket. This role should have a
trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Chave para atributos na origem do Jira:

1. hosts: a nuvem do Jira ou a URL on-premises. Geralmente, é assim `https://your-domain-name.atlassian.net/`.
2. acknowledgments: para garantir a entrega dos dados ao coletor.
3. authentication: descreve como você deseja que o pipeline acesse a instância do Jira. Escolha `Basic` ou `0Auth2` e especifique os atributos de chave correspondentes que fazem referência às chaves em seu AWS segredo.
4. filter: esta seção ajuda você a selecionar qual porção dos dados do Jira deve ser extraída e sincronizada.
   a. project: liste as chaves dos projetos que você deseja sincronizar na seção `include`. Caso contrário, liste os projetos que você deseja excluir na seção `exclude`. Nunca forneça mais de uma das opções de inclusão ou exclusão.
   b. issue_type: os tipos de problemas específicos que você deseja sincronizar. Siga o padrão similar `include` ou `exclude` que atenda às suas necessidades. Observe que os anexos aparecerão como links-âncora para o anexo original, mas o conteúdo do anexo não será extraído.
   c. status: filtro de status específico que você deseja aplicar para a consulta de extração de dados. Se você especificar `include`, somente os tickets com esses status serão sincronizados.

Se você especificar `exclude`, todos os tíquetes, exceto os que tiverem os status excluídos listados, serão sincronizados.

Configuração do pipeline de conector do Confluence

É possível usar um esquema pré-configurado do Atlassian Confluence para criar esse pipeline. Para saber mais, consulte the section called “Trabalhar com esquemas”.

```
version: "2"
extension:
  aws:
    secrets:
      confluence-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-confluence-pipeline:
  source:
    confluence:
      # We currently support only one host URL.
      hosts: ["confluence-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your Confluence account
        basic:
          username: ${{aws_secrets:confluence-account-credentials:confluenceId}}
          password: ${{aws_secrets:confluence-account-
credentials:confluenceCredential}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the following link to generate these keys:
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:confluence-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:confluence-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:confluence-account-credentials:accessToken}}
```

```
          # refresh_token: ${{aws_secrets:confluence-account-credentials:refreshToken}}
      filter:
        space:
          key:
            include:
              # This is not space name.
              # It is a space key that you can find under space details in Confluence.
              - "space key"
              - "space key"
             # exclude:
               #  - "space key"
               #  - "space key"
        page_type:
          include:
            - "content type"
            # - "page"
            # - "blogpost"
            # - "comment"
         # exclude:
            # - "attachment"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
         index: "index_${getMetadata(\"space\")}"
        # Ensure adding unique document id which is the unique ticket ID in this case.
        document_id: '${/id}'
        aws:
          # Provide the Amazon Resource Name (ARN) for a role with access to the
 domain. This role should have a trust relationship with osis-pipelines.amazonaws.com.
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the Region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection.
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
```

```
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
 compression setting is changed in the domain.
        # For more information, see the section called “Compactação de solicitações
 HTTP”.
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
 Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the Rregion of the bucket.
            region: "us-east-1"
            # Provide the Amazon Resource Name (ARN) for a role with access to the
 bucket. This role should have a trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Principais atributos na origem do Confluence:

1. hosts: a nuvem do Confluence ou a URL on-premises. Geralmente, é similar a `https://your-domain-name.atlassian.net/`
2. acknowledgments: para garantir a entrega dos dados ao coletor.
3. authentication: descreve como você deseja que o pipeline acesse a instância do Confluence. Escolha `Basic` ou `0Auth2` e especifique os atributos de chave correspondentes que fazem referência às chaves em seu AWS segredo.
4. filter: esta seção ajuda você a selecionar qual porção dos dados do Confluence deve ser extraída e sincronizada.
   a. space: liste as chaves dos espaços que você deseja sincronizar na seção `include`. Caso contrário, liste os espaços que você deseja excluir na seção `exclude`. Nunca forneça mais de uma das opções de inclusão ou exclusão.
   b. page_type: tipos de página específicos (como página, postagem de blog ou anexos) que você deseja sincronizar. Siga o padrão similar `include` ou `exclude` que atenda às suas necessidades. Observe que os anexos aparecerão como links-âncora para o anexo original, mas o conteúdo do anexo não será extraído.

Consistência de dados

Com base nos filtros especificados no pipeline YAML, os projetos (ou espaços) selecionados serão extraídos uma vez e totalmente sincronizados com o coletor de destino. Depois, o monitoramento contínuo das alterações capturará as alterações à medida que elas ocorrerem e atualizará os dados no coletor. Uma exceção é que o monitoramento de alterações sincroniza apenas as ações `create` e `update`, não as ações `delete`.

Limitações

- As ações de exclusão de usuário não serão sincronizadas. Os dados, uma vez gravados no coletor, ali permanecerão. As atualizações substituirão o conteúdo existente por novas alterações se o mapeamento de ID for especificado nas configurações do coletor.
- As instâncias locais que usam versões mais antigas do software Atlassian que não oferecem suporte ao seguinte não APIs são compatíveis com essa fonte:
  - API search versão 3 do Jira
    - `rest/api/3/search`
    - `rest/api/3/issue`
  - Confluence
    - `wiki/rest/api/content/search`
    - `wiki/rest/api/content`
    - `wiki/rest/api/settings/systemInfo`

Métricas CloudWatch para conectores Atlassian

Tipo: métricas do conector do Jira

Fonte	Métrica	Tipo de métrica
acknowledgementSetSuccesses.contar	Contador	Se as confirmações estiverem habilitadas, essa métrica fornecerá o número de tickets sincronizados com sucesso.
acknowledgementSet	Contador	Se as confirmações estiverem habilitadas, essa métrica fornecerá o número de

Fonte	Métrica	Tipo de métrica
Failures.contar		tickets em que houve falha de sincronização.
crawlingTime.avg	Timer	O tempo necessário para fazer o crawling por todas as novas alterações.
ticketFetchLatency.avg	Timer	A média de latência da API de busca de ticket.
ticketFetchLatency.máximo	Timer	O máximo de latência da API de busca de ticket.
ticketsRequested.count	Contador	Número de solicitações de busca de tíquete feitas.
ticketRequestedFailed.contar	Contador	Número de solicitações de busca de tíquete em que houve falha.
ticketRequestedSuccess.contar	Contador	Número de solicitações de busca de tíquete bem-sucedidas.
searchCallLatency.avg	Timer	Média de latência de chamada de API search.
searchCallLatency.máximo	Timer	Máximo de latência de chamada de API search.

Fonte	Métrica	Tipo de métrica
searchResultsFound.contar	Contador	Número de itens encontrados em uma determinada chamada search.
searchRequestFailed.contar	Contador	Contagem de falhas de chamada de API search.
AuthFailures.count	Contador	Contagem de falhas de autenticação.

Tipo: métricas do conector do Confluence

Fonte	Métrica	Tipo de métrica
acknowledgementSetSuccesses.contar	Contador	Se as confirmações estiverem habilitadas, essa métrica fornecerá o número de páginas sincronizadas com sucesso.
acknowledgementSetFailures.contar	Contador	Se as confirmações estiverem habilitadas, essa métrica fornecerá o número de páginas em que houve falha na sincronização.
crawlingTime.avg	Timer	O tempo necessário para fazer o crawling por todas as novas alterações.
pageFetchLatency.avg	Timer	Latência da API de busca de conteúdo (média).
pageFetchLatency.máximo	Timer	Latência da API de busca de conteúdo (máxima).

Fonte	Métrica	Tipo de métrica
pagesRequested.count	Contador	Número de invocações da API de busca de conteúdo.
pageRequestFailed.contar	Contador	Número de solicitações da API de busca de conteúdo em que houve falha.
pageRequestedSuccess.contar	Contador	Número de solicitações da API de busca de conteúdo bem-sucedidas.
searchCallLatency.avg	Timer	Média de latência de chamada de API search.
searchCallLatency.máximo	Timer	Latência máxima de chamada de API search.
searchResultsFound.contar	Contador	Número de itens encontrados em uma determinada chamada search.
searchRequestsFailed.contar	Contador	Contagem de falhas de chamada de API search.
AuthFailures.count	Contador	Contagem de falhas de autenticação.

Conectando um pipeline de OpenSearch ingestão da Amazon ao Atlassian Jira ou ao Confluence usando 2.0 OAuth

Use as informações deste tópico para ajudá-lo a configurar e conectar um pipeline de OpenSearch ingestão da Amazon a uma conta do Jira ou do Confluence usando a autenticação 2.0. OAuth

Execute essa tarefa quando estiver concluindo o processo the section called “Pré-requisitos” para usar um pipeline de OpenSearch ingestão com os serviços da Atlassian, mas opte por não usar as credenciais da chave de API.

Tópicos

- Crie um aplicativo de integração OAuth 2.0
- Gerar e atualizar um token de acesso ao Atlassian Developer

Crie um aplicativo de integração OAuth 2.0

Use o procedimento a seguir para ajudá-lo a criar um aplicativo de integração OAuth 2.0 no site do Atlassian Developer.

Para criar um aplicativo de integração OAuth 2.0

1. Faça login na sua conta de desenvolvedor da Atlassian em https://developer.atlassian.com/console/ myapps/.
2. Escolha Criar, integração OAuth 2.0.
3. Em Name, insira um nome para identificar a finalidade da aplicação.
4. Selecione I agree to be bound by Atlassian's developer terms e selecione Create.
5. No painel de navegação à esquerda, escolha Authorization e depois Add.
6. Em Callback URL, insira qualquer URL, como **`https://www.amazon.com`** ou **`https://www.example.com`**, e escolha Save changes.
7. No painel de navegação à esquerda, escolha a página Permissions e, na linha da API do Jira, escolha Add, depois escolha Configure, selecione todas as permissões Classic Scopes Read (lista fornecida abaixo) e selecione Save
8. Escolha a guia Granular scopes e, em seguida, escolha Edit Scopes para abrir a caixa de diálogo Edit Jira API.
9. Selecione as permissões para o plug-in de origem que você está usando:

   Jira

```
read:audit-log:jira
read:issue:jira
read:issue-meta:jira
read:attachment:jira
read:comment:jira
```

```
read:comment.property:jira
read:field:jira
read:field.default-value:jira
read:field.option:jira
read:field-configuration-scheme:jira
read:field-configuration:jira
read:issue-link:jira
read:issue-link-type:jira
read:issue-link-type:jira
read:issue.remote-link:jira
read:issue.property:jira
read:resolution:jira
read:issue-details:jira
read:issue-type:jira
read:issue-worklog:jira
read:issue-field-values:jira
read:issue.changelog:jira
read:issue.transition:jira
read:issue.vote:jira
read:jira-expressions:jira
```

Confluence

```
read:content:confluence
read:content-details:confluence
read:space-details:confluence
read:audit-log:confluence
read:page:confluence
read:blogpost:confluence
read:custom-content:confluence
read:comment:confluence
read:space:confluence
read:space.property:confluence
read:space.setting:confluence
read:content.property:confluence
read:content.metadata:confluence
read:task:confluence
read:whiteboard:confluence
read:app-data:confluence
manage:confluence-configuration
```

10. Escolha Salvar.

Para obter informações relacionadas, consulte Implementação OAuth 2.0 (3LO) e Determinação dos escopos necessários para uma operação no site de desenvolvedores da Atlassian.

Gerar e atualizar um token de acesso ao Atlassian Developer

Use o procedimento a seguir para ajudar você a gerar e atualizar um token de acesso ao Atlassian Developer no site do Atlassian Developer.

Para gerar e atualizar um token de acesso do Jira

1. Faça login na sua conta de desenvolvedor da Atlassian em https://developer.atlassian.com/console/ myapps/.
2. Selecione a aplicação que você criou em the section called “Crie um aplicativo de integração OAuth 2.0”.
3. No menu de navegação à esquerda, escolha Authorization.
4. Copie o valor granular da URL de autorização da API do Atlassian da parte inferior da página e cole-o no editor de texto de sua preferência.

   O formato do ARN é o seguinte:

   ```
   https://auth.atlassian.com/authorize?
   audience=api.atlassian.com
   &client_id=YOUR_CLIENT_ID
   &scope=REQUESTED_SCOPE%20REQUESTED_SCOPE_TWO
   &redirect_uri=https://YOUR_APP_CALLBACK_URL
   &state=YOUR_USER_BOUND_VALUE
   &response_type=code
   &prompt=consent
   ```

5. Em `state=YOUR_USER_BOUND_VALUE`, altere o valor do parâmetro à sua escolha, como state=“**`sample_text`**”.

   Para saber mais, consulte What is the state parameter used for? no site do Atlassian Developer.
6. Observe que a seção `scope` lista os escopos granulares que você selecionou em uma tarefa anterior. Por exemplo: `scope=read%3Ajira-work%20read%3Ajira-user%20offline_access`

   `offline_access` indica que você deseja gerar um `refresh_token`.
7. Abra uma janela do navegador e insira a URL de autorização que você copiou na barra de endereço da janela do navegador.

8. Quando a página de destino se abrir, verifique se as informações estão corretas e escolha Accept para ser redirecionado para página inicial do Jira ou do Confluence.
9. Depois que a página inicial for carregada, copie a URL dessa página. Ele contém o código de autorização da aplicação. Você precisará desse código para gerar o token de acesso. Toda a seção depois de `code=` é o código de autorização.
10. Use o seguinte comando cURL para gerar o token de acesso. Substitua *`placeholder values`* por suas próprias informações.

> **Tip**
>
> Você também pode usar um serviço de terceiros, como o Postman.

```
curl --request POST --url 'https://auth.atlassian.com/oauth/token' \
--header 'Content-Type: application/json' \
--data '{"grant_type": "authorization_code",
"client_id": "YOUR_CLIENT_ID",
"client_secret": "YOUR_CLIENT_SECRET",
"code": "AUTHORIZATION_CODE",
"redirect_uri": "YOUR_CALLBACK_URL"}'
```

A resposta a esse comando inclui os valores de `access_code` e `refresh_token`.

Usando um pipeline OpenSearch de ingestão com o Amazon Aurora

Você pode usar um pipeline de OpenSearch ingestão com o Amazon Aurora para exportar dados existentes e transmitir alterações (como criar, atualizar e excluir) para domínios e coleções do OpenSearch Amazon Service. O pipeline OpenSearch de ingestão incorpora a infraestrutura de captura de dados de alteração (CDC) para fornecer uma forma de alta escala e baixa latência de transmitir dados continuamente do Amazon Aurora. O Aurora MySQL e o Aurora PostgreSQL são compatíveis.

Há duas maneiras de usar o Amazon Aurora como origem para processar dados: com e sem um snapshot inicial completo. Um snapshot inicial completo é um snapshot das tabelas especificadas, e ele é exportado para o Amazon S3. A partir daí, um pipeline de OpenSearch ingestão o envia para um índice em um domínio ou o particiona em vários índices em um domínio. Para manter os dados no Amazon Aurora e OpenSearch consistentes, o pipeline sincroniza todos os eventos de criação,

atualização e exclusão nas tabelas dos clusters do Amazon Aurora com os documentos salvos no índice ou índices. OpenSearch

Quando você usa um snapshot inicial completo, seu pipeline de OpenSearch ingestão primeiro ingere o snapshot e depois começa a ler os dados dos fluxos de alterações do Amazon Aurora. Eventualmente, ele recupera e mantém a consistência de dados quase em tempo real entre o Amazon Aurora e. OpenSearch

Você também pode usar a integração de OpenSearch ingestão com o Amazon Aurora para rastrear alterações na captura de dados e ingerir todas as atualizações no Aurora para. OpenSearch Escolha essa opção se você já tiver um snapshot completo de algum outro mecanismo ou se quiser apenas capturar todas as alterações nos dados do cluster do Amazon Aurora.

Ao escolher essa opção, você precisa configurar o registro em log binário para o Aurora MySQL ou configurar a replicação lógica para o AuroraPostgreSQL no cluster.

Tópicos

- Aurora MySQL
- Aurora PostgreSQL

Aurora MySQL

Conclua as etapas a seguir para configurar um pipeline de OpenSearch ingestão com o Amazon Aurora para o Aurora MySQL.

Tópicos

- Pré-requisitos do Aurora MySQL
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline
- Consistência de dados
- Mapear tipo de dados
- Limitações
- CloudWatch Alarmes recomendados

Pré-requisitos do Aurora MySQL

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um grupo de parâmetros do cluster do banco de dados Aurora no Amazon Aurora para configurar o registro em log binário.

   ```
   aurora_enhanced_binlog=1
   binlog_backup=0
   binlog_format=ROW
   binlog_replication_globaldb=0
   binlog_row_image=full
   binlog_row_metadata=full
   ```

   Além disso, certifique-se de que o parâmetro `binlog_transaction_compression` não esteja definido como `ON` e de que o parâmetro `binlog_row_value_options` não esteja definido como `PARTIAL_JSON`.

2. Selecione ou crie um cluster do banco de dados Aurora MySQL e associe o grupo de parâmetros criado na etapa anterior ao cluster do banco de dados.
3. Configure a retenção de logs binários como 24 horas ou mais.
4. Configure a autenticação de nome de usuário e senha no cluster do Amazon Aurora usando o gerenciamento de senhas com o Aurora e o AWS Secrets Manager. Você também pode criar uma username/password combinação criando um segredo do Secrets Manager.
5. Se você usar o recurso de snapshot inicial completo, crie uma função AWS KMS key e uma do IAM para exportar dados do Amazon Aurora para o Amazon S3.

   O perfil do IAM também deve ter a seguinte política de permissão:

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Sid": "ExportPolicy",
               "Effect": "Allow",
               "Action": [
                   "s3:PutObject*",
                   "s3:ListBucket",
                   "s3:GetObject*",
                   "s3:DeleteObject*",
                   "s3:GetBucketLocation"
               ],
               "Resource": [
   ```

```
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

O perfil também deve ter a seguinte relação de confiança:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

6. Selecione ou crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.
7. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster de banco de dados Amazon Aurora em seu domínio ou coleção.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos de pipeline do Amazon Aurora, configure o perfil do pipeline para usar na configuração do pipeline. Adicione também ao perfil as seguintes permissões para a origem do Amazon Aurora:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
```

```
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowDescribeClusters",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBClusters"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id"
    ]
    },
    {
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBClusterSnapshots",
    "rds:CreateDBClusterSnapshot",
    "rds:AddTagsToResource"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
    "arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
    ]
```

```
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
    "arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
```

```
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Etapa 2: Criar o pipeline

Configure um pipeline de OpenSearch ingestão semelhante ao seguinte. O exemplo de pipeline especifica um cluster do Amazon Aurora como origem.

```
version: "2"
aurora-mysql-pipeline:
  source:
    rds:
      db_identifier: "cluster-id"
      engine: aurora-mysql
      database: "database-name"
      tables:
        include:
          - "table1"
          - "table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
```

```
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

Você pode usar um esquema pré-configurado do Amazon Aurora para criar esse pipeline. Para saber mais, consulte Trabalhar com esquemas.

Para usar o Amazon Aurora como origem, você precisa configurar o acesso à VPC para o pipeline. A VPC escolhida deve ser a mesma VPC que a origem do Amazon Aurora usa. Depois, selecione uma ou mais sub-redes e um ou mais grupos de segurança da VPC. Observe que o pipeline precisa de acesso de rede a um banco de dados MySQL do Aurora, por isso, você também deve se certificar de que o cluster do Aurora esteja configurado com um grupo de segurança da VPC que permita tráfego de entrada do grupo de segurança da VPC do pipeline para a porta do banco de dados. Para saber mais, consulte Controlar o acesso com grupos de segurança.

Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline à sua VPC para usar o Amazon Aurora como fonte. Para fazer isso, encontre a seção Configuração de rede, marque a caixa de seleção Anexar à VPC e escolha seu

CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.

Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre a OpenSearch ingestão e o Amazon Aurora, certifique-se de que o CIDR do Amazon Aurora VPC seja diferente do CIDR para ingestão. OpenSearch

Para saber mais, consulte Configurar o acesso à VPC para um pipeline.

Consistência de dados

O pipeline garante a consistência dos dados pesquisando ou recebendo continuamente alterações do cluster Amazon Aurora e atualizando os documentos correspondentes no OpenSearch índice.

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção. Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento, portanto, você deve omiti-lo `document_id: "${getMetadata(\"primary_key\")}"` na configuração do coletor do pipeline.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de modo que cada alteração de dados no Amazon Aurora seja reconciliada com as alterações correspondentes do documento. OpenSearch

Mapear tipo de dados

OpenSearch O pipeline de ingestão mapeia os tipos de dados do MySQL em representações que são adequadas OpenSearch para o consumo de domínios ou coleções de serviços. Se nenhum modelo de mapeamento estiver definido em OpenSearch, determina OpenSearch automaticamente os tipos de campo com mapeamento dinâmico com base no primeiro documento enviado. Você também pode definir explicitamente os tipos de campo que funcionam melhor para você por OpenSearch meio de um modelo de mapeamento.

A tabela abaixo lista os tipos de dados do MySQL e os tipos de OpenSearch campo correspondentes. A coluna Tipo de OpenSearch campo padrão mostra o tipo de campo correspondente OpenSearch se nenhum mapeamento explícito for definido. Nesse caso, determina

OpenSearch automaticamente os tipos de campo com mapeamento dinâmico. A coluna Tipo de OpenSearch campo recomendado é o tipo de campo correspondente que é recomendado especificar explicitamente em um modelo de mapeamento. Esses tipos de campo estão mais alinhados com os tipos de dados no MySQL e geralmente podem permitir melhores recursos de pesquisa disponíveis no. OpenSearch

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomenda do
BIGINT	longo	longo
BIGINT UNSIGNED	longo	unsigned long
BIT	longo	byte, short, integer ou long dependendo do número de bits
DECIMAL	texto	double ou keyword
DOUBLE	flutuação	double
FLOAT	flutuação	flutuação
INT	longo	integer
INT UNSIGNED	longo	longo
MEDIUMIN	longo	integer
MEDIUMIN UNSIGNED	longo	integer
NUMERIC	texto	double ou keyword
SMALLINT	longo	curto
SMALLINT UNSIGNED	longo	integer

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
TINYINT	longo	byte
TINYINT UNSIGNED	longo	curto
BINARY	texto	binary
BLOB	texto	binary
CHAR	texto	texto
ENUM	texto	palavra-chave
LONGBLOB	texto	binary
LONGTEXT	texto	texto
MEDIUMBLOB	texto	binary
MEDIUMTEXT	texto	texto
SET	texto	palavra-chave
TEXT	texto	texto
TINYBLOB	texto	binary
TINYTEXT	texto	texto
VARBINARY	texto	binary
VARCHAR	texto	texto
DATE	long (em milissegundos epoch)	date

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
DATETIME	long (em milissegundos epoch)	date
TIME	long (em milissegundos epoch)	date
TIMESTAM	long (em milissegundos epoch)	date
YEAR	long (em milissegundos epoch)	date
GEOMETR	text (em formato WKT)	geo_shape
GEOMETR OLLECTIO	text (em formato WKT)	geo_shape
LINESTRIN G	text (em formato WKT)	geo_shape
MULTILINE STRING	text (em formato WKT)	geo_shape
MULTIPOIN T	text (em formato WKT)	geo_shape
MULTIPOL GON	text (em formato WKT)	geo_shape
POINT	text (em formato WKT)	geo_point ou geo_shape
POLYGON	text (em formato WKT)	geo_shape
JSON	texto	objeto

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos não funcionarem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para o Aurora MySQL:

- A integração é compatível com apenas um banco de dados MySQL por pipeline.
- Atualmente, a integração não oferece suporte à ingestão de dados entre regiões; seu cluster OpenSearch e domínio do Amazon Aurora devem estar no mesmo. Região da AWS
- Atualmente, a integração não oferece suporte à ingestão de dados entre contas; seu cluster do Amazon Aurora OpenSearch e seu pipeline de ingestão devem estar no mesmo. Conta da AWS
- Certifique-se de que a autenticação esteja habilitada no cluster do Amazon Aurora usando o Secrets Manager, que é o único mecanismo de autenticação compatível.
- A configuração existente do pipeline não pode ser atualizada para ingerir dados de um banco de dados diferente ou de and/or uma tabela diferente. Para atualizar o nome do banco de dados e/ou tabela de um pipeline, você deve parar o pipeline e reiniciá-lo com uma configuração atualizada ou criar um novo pipeline.
- Instruções em Data Definition Language (DDL) não são compatíveis em geral. A consistência de dados não será mantida se:
  - As chaves primárias são alteradas (add/delete/rename).
  - As tabelas forem eliminadas/truncadas.
  - Os nomes das colunas ou os tipos de dados forem alterados.
- Se as tabelas MySQL a serem sincronizadas não tiverem chaves primárias definidas, a consistência de dados não será garantida. Você precisará definir a `document_id` opção personalizada na configuração do OpenSearch coletor corretamente para poder updates/deletes sincronizar com OpenSearch.
- Referências de chave estrangeira com ações de exclusão em cascata não são suportadas e podem resultar em inconsistência de dados entre o Aurora MySQL e. OpenSearch
- Versões compatíveis: Aurora MySQL versão 3.05.2 e acima.

CloudWatch Alarmes recomendados

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo, e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

Métrica	Description
pipeline-name .RDS. Credenciais alteradas	Essa métrica indica com que frequência AWS os segredos são alternados.
pipeline-name .rds. executorRefreshErrors	Essa métrica indica falhas na atualização de AWS segredos.
pipeline-name .rds. exportRecordsTotal	Essa métrica indica o número de registros exportados do Amazon Aurora.
pipeline-name .rds. exportRecordsProcessed	Essa métrica indica o número de registros processados pelo pipeline OpenSearch de ingestão.
pipeline-name .rds. exportRecordProcessingErros	Essa métrica indica o número de erros de processamento em um pipeline OpenSearch de ingestão durante a leitura dos dados de um cluster do Amazon Aurora.
pipeline-name .rds.	Essa métrica indica o número total de registros de exportação processados com êxito.

Métrica	Description
exportRecordsSuccessTotal	
pipeline-name .rds.exportRecordsFailedTotal	Essa métrica indica o número total de registros de exportação com falha no processamento.
pipeline-name .rds.bytesRecebidos	Essa métrica indica o número total de bytes recebidos por um pipeline OpenSearch de ingestão.
pipeline-name .rds.BytesProcessados	Essa métrica indica o número total de bytes processados por um pipeline OpenSearch de ingestão.
pipeline-name .rds.streamRecordsSuccessTotal	Essa métrica indica o número de registros processados com êxito a partir do fluxo.
pipeline-name .rds.streamRecordsFailedTotal	Essa métrica indica o número total de registros com falha no processamento do fluxo.

Aurora PostgreSQL

Conclua as etapas a seguir para configurar um pipeline de OpenSearch ingestão com o Amazon Aurora para o Aurora PostgreSQL.

Tópicos

- Pré-requisitos do Aurora PostgreSQL
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline

- Consistência de dados
- Mapear tipo de dados
- Limitações
- CloudWatch Alarmes recomendados

Pré-requisitos do Aurora PostgreSQL

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um grupo personalizado de parâmetros de cluster do banco de dados no Amazon Aurora para configurar a replicação lógica.

```
rds.logical_replication=1
    aurora.enhanced_logical_replication=1
    aurora.logical_replication_backup=0
    aurora.logical_replication_globaldb=0
```

2. Selecione ou crie um cluster do banco de dados Aurora PostgreSQL e associe o grupo de parâmetros criado na etapa 1 ao cluster do banco de dados.
3. Configure a autenticação de nome de usuário e senha no cluster do Amazon Aurora usando o gerenciamento de senhas com o Aurora e o AWS Secrets Manager. Você também pode criar uma username/password combinação criando um segredo do Secrets Manager.
4. Se você usar o recurso de snapshot inicial completo, crie uma função AWS KMS key e uma do IAM para exportar dados do Amazon Aurora para o Amazon S3.

   O perfil do IAM também deve ter a seguinte política de permissão:

   JSON

```
{
        "Version":"2012-10-17",
        "Statement": [
            {
                "Sid": "ExportPolicy",
                "Effect": "Allow",
                "Action": [
                    "s3:PutObject*",
                    "s3:ListBucket",
                    "s3:GetObject*",
```

```
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

O perfil também deve ter a seguinte relação de confiança:

JSON

```
{
        "Version":"2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Principal": {
                    "Service": "export.rds.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ]
    }
```

5. Selecione ou crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.
6. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster de banco de dados Amazon Aurora em seu domínio ou coleção.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos de pipeline do Amazon Aurora, configure o perfil do pipeline para usar na configuração do pipeline. Adicione também ao perfil as seguintes permissões para a origem do Amazon Aurora:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
```

```
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowDescribeClusters",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBClusters"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id"
    ]
    },
    {
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBClusterSnapshots",
    "rds:CreateDBClusterSnapshot",
    "rds:AddTagsToResource"
    ],
```

```
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
    "arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:cluster:DB-id",
    "arn:aws:rds:us-east-2:111122223333:cluster-snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
```

```
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Etapa 2: Criar o pipeline

Configure um pipeline OpenSearch de ingestão como o seguinte, que especifica o cluster Aurora PostgreSQL como origem.

```
version: "2"
aurora-postgres-pipeline:
  source:
    rds:
      db_identifier: "cluster-id"
      engine: aurora-postgresql
      database: "database-name"
      tables:
        include:
          - "schema1.table1"
          - "schema2.table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
```

```
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

> **Note**
>
> Você pode usar um esquema pré-configurado do Amazon Aurora para criar esse pipeline. Para saber mais, consulte Trabalhar com esquemas.

Para usar o Amazon Aurora como origem, você precisa configurar o acesso à VPC para o pipeline. A VPC escolhida deve ser a mesma VPC que a origem do Amazon Aurora usa. Depois, selecione uma ou mais sub-redes e um ou mais grupos de segurança da VPC. Observe que o pipeline precisa de acesso de rede a um banco de dados MySQL do Aurora, por isso, você também deve se certificar de que o cluster do Aurora esteja configurado com um grupo de segurança da VPC que permita tráfego

de entrada do grupo de segurança da VPC do pipeline para a porta do banco de dados. Para saber mais, consulte Controlar o acesso com grupos de segurança.

Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline à sua VPC para usar o Amazon Aurora como fonte. Para fazer isso, encontre a seção Configuração de rede, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.

Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre a OpenSearch ingestão e o Amazon Aurora, certifique-se de que o CIDR do Amazon Aurora VPC seja diferente do CIDR para ingestão. OpenSearch

Para saber mais, consulte Configurar o acesso à VPC para um pipeline.

Consistência de dados

O pipeline garante a consistência dos dados pesquisando ou recebendo continuamente alterações do cluster Amazon Aurora e atualizando os documentos correspondentes no OpenSearch índice.

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção. Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento, portanto, você deve omiti-lo `document_id: "${getMetadata(\"primary_key\")}"` na configuração do coletor do pipeline.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de modo que cada alteração de dados no Amazon Aurora seja reconciliada com as alterações correspondentes do documento. OpenSearch

Mapear tipo de dados

OpenSearch O pipeline de ingestão mapeia os tipos de dados do Aurora PostgreSQL para representações que são OpenSearch adequadas para o consumo de domínios ou coleções de serviços. Se nenhum modelo de mapeamento estiver definido em OpenSearch, determine

OpenSearch automaticamente os tipos de campo com um mapeamento dinâmico baseado no primeiro documento enviado. Você também pode definir explicitamente os tipos de campo que funcionam melhor para você por OpenSearch meio de um modelo de mapeamento.

A tabela abaixo lista os tipos de dados do Aurora PostgreSQL e os tipos de campo correspondentes. OpenSearch A coluna Tipo de OpenSearch campo padrão mostra o tipo de campo correspondente OpenSearch se nenhum mapeamento explícito for definido. Nesse caso, determina OpenSearch automaticamente os tipos de campo com mapeamento dinâmico. A coluna Tipo de OpenSearch campo recomendado é o tipo de campo recomendado correspondente a ser especificado explicitamente em um modelo de mapeamento. Esses tipos de campo estão mais alinhados com os tipos de dados no Aurora PostgreSQL e geralmente podem permitir melhores recursos de pesquisa disponíveis no. OpenSearch

Tipo de dado do Aurora PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
smallint	longo	curto
integer	longo	integer
bigint	longo	longo
decimal	texto	double ou keyword
numeric[(p, s)]	texto	double ou keyword
real	flutuação	flutuação
double precision	flutuação	double
smallserial	longo	curto
serial	longo	integer

Tipo de dado do Aurora PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
bigserial	longo	longo
money	objeto	objeto
caractere variável(n)	texto	texto
varchar(n)	texto	texto
character(n)	texto	texto
char(n)	texto	texto
bpchar(n)	texto	texto
bpchar	texto	texto
texto	texto	texto
enum	texto	texto
bytea	texto	binary
timestamp [(p)] [sem fuso horário]	long (em milissegundos epoch)	date

Tipo de dado do Aurora PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
timestamp [(p)] com fuso horário	long (em milissegundos epoch)	date
date	long (em milissegundos epoch)	date
hora [(p)] [sem fuso horário]	long (em milissegundos epoch)	date
hora [(p)] com fuso horário	long (em milissegundos epoch)	date
interval [fields] [(p)]	texto (ISO8601 formato)	texto
booleano	booleano	booleano
point	text (em formato WKT)	geo_shape
linha	text (em formato WKT)	geo_shape
lseg	text (em formato WKT)	geo_shape
box	text (em formato WKT)	geo_shape
caminho	text (em formato WKT)	geo_shape

Tipo de dado do Aurora PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
polígono	text (em formato WKT)	geo_shape
circle	objeto	objeto
cidr	texto	texto
inet	texto	texto
macaddr	texto	texto
macaddr8	texto	texto
bit(n)	longo	byte, short, integer, ou long (dependendo do número de bits)
bit variável (n)	longo	byte, short, integer, ou long (dependendo do número de bits)
json	objeto	objeto
jsonb	objeto	objeto
jsonpath	texto	texto

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos falharem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para o Aurora PostgreSQL:

- A integração é compatível com apenas um banco de dados Aurora PostgreSQL por pipeline.
- Atualmente, a integração não oferece suporte à ingestão de dados entre regiões; seu cluster OpenSearch e domínio do Amazon Aurora devem estar no mesmo. Região da AWS
- Atualmente, a integração não oferece suporte à ingestão de dados entre contas; seu cluster do Amazon Aurora OpenSearch e seu pipeline de ingestão devem estar no mesmo. Conta da AWS
- Certifique-se de que a autenticação esteja habilitada no cluster do Amazon Aurora usando o AWS Secrets Manager, que é o único mecanismo de autenticação compatível.
- A configuração existente do pipeline não pode ser atualizada para ingerir dados de um banco de dados diferente ou de and/or uma tabela diferente. Para atualizar o nome do banco de dados e/ou tabela de um pipeline, você deve parar o pipeline e reiniciá-lo com uma configuração atualizada ou criar um novo pipeline.
- Instruções em Data Definition Language (DDL) não são compatíveis em geral. A consistência de dados não será mantida se:
  - As chaves primárias são alteradas (add/delete/rename).
  - As tabelas forem eliminadas/truncadas.
  - Os nomes das colunas ou os tipos de dados forem alterados.
- Se as tabelas Aurora PostgreSQL a serem sincronizadas não tiverem chaves primárias definidas, a consistência de dados não será garantida. Você precisará definir a `document_id` opção personalizada OpenSearch e a configuração do coletor corretamente para poder updates/deletes sincronizar com OpenSearch.
- Versões compatíveis: Aurora PostgreSQL versão 16.4 e acima.

CloudWatch Alarmes recomendados

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo, e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

Métrica	Description
pipeline-name .RDS. Credenciais alteradas	Essa métrica indica com que frequência AWS os segredos são alternados.
pipeline-name .rds. executorRefreshErrors	Essa métrica indica falhas na atualização de AWS segredos.
pipeline-name .rds. exportRecordsTotal	Essa métrica indica o número de registros exportados do Amazon Aurora.
pipeline-name .rds. exportRecordsProcessed	Essa métrica indica o número de registros processados pelo pipeline OpenSearch de ingestão.
pipeline-name .rds. exportRecordProcessingErros	Essa métrica indica o número de erros de processamento em um pipeline OpenSearch de ingestão durante a leitura dos dados de um cluster do Amazon Aurora.
pipeline-name .rds. exportRecordsSuccessTotal	Essa métrica indica o número total de registros de exportação processados com êxito.
pipeline-name .rds. exportRecordsFailedTotal	Essa métrica indica o número total de registros de exportação com falha no processamento.

Métrica	Description
pipeline-name .rds.bytes recebidos	Essa métrica indica o número total de bytes recebidos por um pipeline OpenSearch de ingestão.
pipeline-name .rds.Bytes processados	Essa métrica indica o número total de bytes processados por um pipeline OpenSearch de ingestão.
pipeline-name .rds. streamRec ordsSuccessTotal	Essa métrica indica o número de registros processados com êxito a partir do fluxo.
pipeline-name .rds. streamRecordsFaile dTotal	Essa métrica indica o número total de registros com falha no processam ento do fluxo.

Usando um pipeline de OpenSearch ingestão com o Amazon DynamoDB

Você pode usar o plug-in do DynamoDB para transmitir eventos de tabela, como criações, atualizações e exclusões, para domínios do OpenSearch Amazon Service e coleções do Amazon Serverless. OpenSearch O pipeline usa captura de dados alterados (CDC) para transmissão contínua em grande escala e baixa latência.

Você pode processar dados do DynamoDB com ou sem um snapshot inicial completo.

- Com um snapshot completo — o DynamoDB point-in-time usa a recuperação (PITR) para criar um backup e carregá-lo no Amazon S3. OpenSearch Em seguida, a ingestão indexa o instantâneo em um ou vários índices. OpenSearch Para manter a consistência, o pipeline sincroniza todas as alterações do DynamoDB com o. OpenSearch Essa opção exige que você habilite PITR e DynamoDB Streams.
- Sem um snapshot — a OpenSearch ingestão transmite somente novos eventos do DynamoDB. Escolha essa opção se você já tiver um snapshot ou precisar de transmissão contínua em tempo real sem dados históricos. Essa opção exige que você habilite apenas o DynamoDB Streams.

Para obter mais informações, consulte Integração do DynamoDB Zero-ETL com o OpenSearch Amazon Service no Guia do desenvolvedor.Amazon DynamoDB

Tópicos

- Pré-requisitos
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline
- Consistência de dados
- Mapear tipo de dados
- Limitações
- CloudWatch Alarmes recomendados para o DynamoDB

Pré-requisitos

Para configurar o pipeline, você precisa ter uma tabela do DynamoDB com o DynamoDB Streams habilitado. Seu fluxo deve usar o tipo de visualização de fluxo `NEW_IMAGE`. No entanto, os pipelines de OpenSearch ingestão também podem transmitir eventos `NEW_AND_OLD_IMAGES` se esse tipo de visualização de fluxo for adequado ao seu caso de uso.

Se você estiver usando instantâneos, também deverá ativar a point-in-time recuperação em sua tabela. Para obter mais informações, consulte Criar uma tabela, Habilitar a point-in-time recuperação e Habilitar um stream no Amazon DynamoDB Developer Guide.

Etapa 1: configurar a função do pipeline

Depois de configurar a tabela do DynamoDB, defina o perfil de pipeline que você deseja usar na configuração do pipeline e adicione as seguintes permissões do DynamoDB nesse perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowRunExportJob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeTable",
```

```
                "dynamodb:DescribeContinuousBackups",
                "dynamodb:ExportTableToPointInTime"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table"
            ]
        },
        {
            "Sid": "allowCheckExportjob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeExport"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table/export/*"
            ]
        },
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeStream",
                "dynamodb:GetRecords",
                "dynamodb:GetShardIterator"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-east-1:111122223333:table/my-table/stream/*"
            ]
        },
        {
            "Sid": "allowReadAndWriteToS3ForExport",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/export-folder/*"
            ]
        }
    ]
```

```
}
```

Você também pode usar uma chave gerenciada pelo AWS KMS cliente para criptografar os arquivos de dados de exportação. Para descriptografar os objetos exportados, especifique `s3_sse_kms_key_id` para o ID da chave na configuração de exportação do pipeline, com o seguinte formato: `arn:aws:kms:`*`region`*`:`*`account-id`*`:key/`*`my-key-id`*. A política a seguir inclui as permissões necessárias para usar uma chave gerenciada pelo cliente:

```
{
    "Sid": "allowUseOfCustomManagedKey",
    "Effect": "Allow",
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": arn:aws:kms:region:account-id:key/my-key-id
}
```

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline OpenSearch de ingestão como o seguinte, que especifica o DynamoDB como origem. Essa amostra de pipeline ingere dados de `table-a` com o snapshot de PITR, seguido por eventos do DynamoDB Streams. Uma posição inicial de `LATEST` indica que o pipeline deve ler os dados mais recentes do DynamoDB Streams.

```
version: "2"
cdc-pipeline:
  source:
    dynamodb:
      tables:
      - table_arn: "arn:aws:dynamodb:region:account-id:table/table-a"
        export:
          s3_bucket: "my-bucket"
          s3_prefix: "export/"
        stream:
          start_position: "LATEST"
      aws:
        region: "us-east-1"
  sink:
  - opensearch:
```

```
      hosts: ["https://search-mydomain.region.es.amazonaws.com"]
      index: "${getMetadata(\"table-name\")}"
      index_type: custom
      normalize_index: true
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
```

Você pode usar um esquema do DynamoDB pré-configurado para criar esse pipeline. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Consistência de dados

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção.

Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de forma que cada alteração de dados no DynamoDB seja reconciliada com as alterações correspondentes no documento. OpenSearch

Mapear tipo de dados

OpenSearch O serviço mapeia dinamicamente os tipos de dados em cada documento recebido para o tipo de dados correspondente no DynamoDB. A tabela a seguir mostra como o OpenSearch Service mapeia automaticamente vários tipos de dados.

Tipo de dados	OpenSearch	DynamoDB
Número	OpenSearch mapeia automaticamente os dados numéricos. Se o número for um número inteiro, OpenSearch mapeie-	O DynamoDB é compatível com números.

Tipo de dados	OpenSearch	DynamoDB
	o como um valor longo. Se o número for fracionário, ele será OpenSearch mapeado como um valor flutuante. OpenSearch mapeia dinamicamente vários atributos com base no primeiro documento enviado. Se houver uma combinação de tipos de dados para o mesmo atributo no DynamoDB, como um número inteiro e um fracionário, o mapeamento poderá falhar. Por exemplo, se seu primeiro documento tiver um atributo que seja um número inteiro e um documento posterior tiver o mesmo atributo de um número fracionár io, OpenSearch não conseguirá ingerir o segundo documento. Nesses casos, é necessário fornecer um modelo de mapeamento explícito, como o seguinte: `{ "template": { "mappings": { "properties": { "MixedNumberAttribute": { "type": "float" } } } } }` Se precisar de precisão dupla, use o mapeamento de campo do tipo string. Não há nenhum tipo numérico equivalen	

Tipo de dados	OpenSearch	DynamoDB
	te que suporte 38 dígitos de precisão em. OpenSearch	
Number set	OpenSearch mapeia automaticamente um conjunto de números em uma matriz de valores longos ou valores flutuantes. Assim como os números escalares, isso depende de o primeiro número ingerido ser um número inteiro ou fracionár io. É possível fornecer mapeamentos para conjuntos de números da mesma maneira que você mapeia strings escalares.	O DynamoDB oferece suporte a tipos que representam conjuntos de números.
String	OpenSearch mapeia automaticamente valores de string como texto. Em algumas situações, como valores enumerados, é possível mapear para o tipo de palavra-chave. O exemplo a seguir mostra como mapear um atributo do DynamoDB `PartType` nomeado para uma palavra-c have. OpenSearch `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	O DynamoDB é compatível com strings.

Tipo de dados	OpenSearch	DynamoDB
String set	OpenSearch mapeia automaticamente um conjunto de strings em uma matriz de strings. É possível fornecer mapeamentos para conjuntos de strings da mesma maneira que você mapeia strings escalares.	O DynamoDB oferece suporte a tipos que representam conjuntos de strings.
Binário	OpenSearch mapeia automaticamente dados binários como texto. Você pode fornecer um mapeamento para escrevê-los como campos binários OpenSearch. O exemplo a seguir mostra como mapear um atributo do DynamoDB `ImageData` nomeado para OpenSearch um campo binário. `{ "template": { "mappings": { "properties": { "ImageData": { "type": "binary" } } } } }`	O DynamoDB oferece suporte a atributos de tipo binário.
Binary Set	OpenSearch mapeia automaticamente um conjunto binário em uma matriz de dados binários como texto. É possível fornecer mapeamentos para conjuntos de números da mesma maneira que você mapeia binários escalares.	O DynamoDB oferece suporte a tipos que representam conjuntos de valores binários.

Tipo de dados	OpenSearch	DynamoDB
Booleano	OpenSearch mapeia um tipo booleano do DynamoDB em um tipo booleano. OpenSearch	O DynamoDB é compatível com atributos do tipo booliano.
Null	OpenSearch pode ingerir documento s com o tipo nulo do DynamoDB. Ele salva o valor como um valor nulo no documento. Não há mapeamento para esse tipo, e esse campo não é indexado nem pesquisável. Se o mesmo nome de atributo for usado para um tipo nulo e depois for alterado para um tipo diferente, como string, OpenSearch criará um mapeamento dinâmico para o primeiro valor não nulo. Os valores subsequentes ainda podem ser valores nulos do DynamoDB.	O DynamoDB oferece suporte a atributos de tipo nulo.

Tipo de dados	OpenSearch	DynamoDB
Mapa	OpenSearch mapeia os atributos do mapa do DynamoDB para campos aninhados. Os mesmos mapeamentos são aplicáveis a um campo aninhado. O exemplo a seguir mapeia uma string em um campo aninhado para um tipo de palavra-chave em OpenSearch: `{ "template": { "mappings": { "properties": { "AdditionalDescriptions": { "properties": { "PartType": { "type": "keyword" } } } } } } }`	O DynamoDB oferece suporte a atributos de tipo de mapa.

Tipo de dados	OpenSearch	DynamoDB
Lista	OpenSearch fornece resultados diferentes para as listas do DynamoDB, dependendo do que está na lista. Quando uma lista contém todos os mesmos tipos de tipos escalares (por exemplo, uma lista de todas as cadeias de caracteres), a lista é OpenSearc h ingerida como uma matriz desse tipo. Isso funciona para os tipos string, número, booliano e null. As restrições para cada um desses tipos são iguais às restrições para um escalar do mesmo tipo. Também é possível fornecer mapeament os para listas de mapas usando o mesmo mapeamento que você usaria para um mapa. Você não pode fornecer uma lista de tipos mistos.	O DynamoDB oferece suporte para atributos de tipo de lista.

Tipo de dados	OpenSearch	DynamoDB
Defina	OpenSearch fornece resultados diferentes para conjuntos do DynamoDB, dependendo do que está no conjunto. Quando um conjunto contém todos os mesmos tipos de tipos escalares (por exemplo, um conjunto de todas as cadeias de caracteres), ele OpenSearch ingere o conjunto como uma matriz desse tipo. Isso funciona para os tipos string, número, booliano e null. As restrições para cada um desses tipos são iguais às restrições para um escalar do mesmo tipo. Também é possível fornecer mapeamentos para conjuntos de mapas usando o mesmo mapeamento que você usaria para um mapa. Você não pode fornecer um conjunto de tipos mistos.	O DynamoDB oferece suporte a tipos que representam conjuntos.

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos falharem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para o DynamoDB:

- Atualmente, a integração de OpenSearch ingestão com o DynamoDB não oferece suporte à ingestão entre regiões. Sua tabela do DynamoDB OpenSearch e seu pipeline de ingestão devem estar no mesmo lugar. Região da AWS
- Sua tabela do DynamoDB OpenSearch e seu pipeline de ingestão devem estar no mesmo lugar. Conta da AWS
- Um pipeline OpenSearch de ingestão suporta somente uma tabela do DynamoDB como origem.
- O DynamoDB Streams apenas armazena dados em log por até 24 horas. Se a ingestão de um snapshot inicial de uma tabela grande levar 24 horas ou mais, haverá uma certa perda inicial de dados. Para mitigar essa perda de dados, estime o tamanho da tabela e configure as unidades computacionais apropriadas dos pipelines de OpenSearch ingestão.

CloudWatch Alarmes recomendados para o DynamoDB

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar você a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

Métrica	Description
`dynamodb-pipeline.BlockingBuffer.bufferUsage.value`	Indica quanto do buffer está sendo utilizado.
`dynamodb-pipeline.dynamodb.activeExportS3ObjectConsumers.value`	Mostra o número total de pessoas OCUs que estão processando ativamente objetos do Amazon S3 para a exportação.
`dynamodb-pipeline.dynamodb.bytesProcessed.count`	Contagem de bytes processados a partir da fonte do DynamoDB.

Métrica	Description
dynamodb-pipeline.dynamodb.changeEventsProcessed.count	Número de eventos de alteração processados no fluxo do DynamoDB.
dynamodb-pipeline.dynamodb.changeEventsProcessingErrors.count	Número de erros de eventos de alteração processados no DynamoDB.
dynamodb-pipeline.dynamodb.exportJobFailure.count	Número de tentativas de envio de trabalhos de exportação que falharam.
dynamodb-pipeline.dynamodb.exportJobSuccess.count	Número de trabalhos de exportação que foram enviados com sucesso.
dynamodb-pipeline.dynamodb.exportRecordsProcessed.count	Número total de registros processados a partir da exportação.
dynamodb-pipeline.dynamodb.exportRecordsTotal.count	Número total de registros exportados do DynamoDB, essencial para acompanhar volumes de exportação de dados.
dynamodb-pipeline.dynamodb.exportS3ObjectsProcessed.count	Número total de arquivos de dados de exportação que foram processados com sucesso no Amazon S3.
dynamodb-pipeline.opensearch.bulkBadRequestErrors.count	Contagem de erros durante solicitações em massa devido a uma solicitação malformada.
dynamodb-pipeline.opensearch.bulkRequestLatency.avg	Latência média para solicitações de gravação em massa feitas para OpenSearch.
dynamodb-pipeline.opensearch.bulkRequestNotFoundErrors.count	Número de solicitações em massa que falharam porque os dados de destino não puderam ser encontrados.
dynamodb-pipeline.opensearch.bulkRequestNumberOfRetries.count	Número de novas tentativas por pipelines OpenSearch de ingestão para gravar o cluster. OpenSearch

Métrica	Description
dynamodb-pipeline.opensearch.bulkRequestSizeBytes.sum	Tamanho total em bytes de todas as solicitações em massa feitas para OpenSearch.
dynamodb-pipeline.opensearch.documentErrors.count	Número de erros ao enviar documentos para OpenSearch. Os documentos que causam os erros serão enviados para a DLQ.
dynamodb-pipeline.opensearch.documentsSuccess.count	Número de documentos gravados com sucesso em um OpenSearch cluster ou coleção.
dynamodb-pipeline.opensearch.documentsSuccessFirstAttempt.count	Número de documentos indexados com sucesso OpenSearch na primeira tentativa.
dynamodb-pipeline.opensearch.documentsVersionConflictErrors.count	Contagem de erros devido a conflitos de versão em documentos durante o processamento.
dynamodb-pipeline.opensearch.PipelineLatency.avg	Latência média do pipeline de OpenSearch ingestão para processar os dados lendo da origem até a gravação no destino.
dynamodb-pipeline.opensearch.PipelineLatency.max	Latência máxima do pipeline de OpenSearch ingestão para processar os dados lendo da origem até a gravação no destino.
dynamodb-pipeline.opensearch.recordsIn.count	Contagem de registros ingeridos com sucesso. OpenSearch Essa métrica é essencial para rastrear o volume de dados sendo processados e armazenados.
dynamodb-pipeline.opensearch.s3.dlqS3RecordsFailed.count	Número de registros que falharam na gravação na DLQ.
dynamodb-pipeline.opensearch.s3.dlqS3RecordsSuccess.count	Número de registros gravados no DLQ.

Métrica	Description
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.count`	Contagem de medidas de latência para solicitações à fila de mensagens mortas do Amazon S3.
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.sum`	Latência total para todas as solicitações para a fila de mensagens mortas do Amazon S3
`dynamodb-pipeline.opensearch.s3.dlqS3RequestSizeBytes.sum`	Tamanho total em bytes de todas as solicitações feitas na fila de mensagens mortas do Amazon S3.
`dynamodb-pipeline.recordsProcessed.count`	Número total de registros processados no pipeline, uma métrica fundamental para a produtividade geral.
`dynamodb.changeEventsProcessed.count`	Nenhum registro está sendo coletado dos fluxos do DynamoDB. Isso pode ser devido a nenhuma atividade na tabela, a uma exportação em andamento ou a um problema no acesso aos streams do DynamoDB.
`dynamodb.exportJobFailure.count`	A tentativa de acionar uma exportação para o S3 falhou.
`dynamodb-pipeline.opensearch.bulkRequestInvalidInputErrors.count`	Contagem de erros de solicitação em massa OpenSearch devido à entrada inválida, crucial para monitorar a qualidade dos dados e problemas operacionais.

Métrica	Description
`opensearch.EndToEndLatency.avg`	A latência de ponta a ponta é maior do que a desejada para leitura de streams do DynamoDB. Isso pode ser devido a um OpenSearch cluster subdimensionado ou a uma capacidade máxima de OCU do pipeline que é muito baixa para a taxa de transferê ncia da WCU na tabela do DynamoDB. Essa latência de ponta a ponta será alta após uma exportação e deverá diminuir com o tempo, à medida que se adapta aos streams mais recentes do DynamoDB.

Usando um pipeline OpenSearch de ingestão com o Amazon DocumentDB

Você pode usar o plug-in DocumentDB para transmitir alterações de documentos, como criações, atualizações e exclusões, para o Amazon Service. OpenSearch O pipeline é compatível com captura de dados alterados (CDC), se disponível, ou pesquisa de API para transmissão contínua em grande escala e baixa latência.

Você pode processar dados com ou sem um snapshot inicial completo. Um snapshot completo captura uma coleção inteira do Amazon DocumentDB e a carrega no Amazon S3. O pipeline então envia os dados para um ou mais OpenSearch índices. Depois de ingerir o snapshot, o pipeline sincroniza as alterações em andamento para manter a consistência e, em algum tempo, recupera a diferença e faz atualizações quase em tempo real.

Se você já tiver um snapshot completo de outra origem ou só precisar processar eventos novos, poderá fazer transmissão contínua sem um snapshot. Nesse caso, o pipeline lê diretamente dos fluxos de alterações do Amazon DocumentDB sem um carregamento inicial em massa.

Se você habilitar a transmissão contínua, deverá habilitar um fluxo de alterações na coleção do Amazon DocumentDB. Porém, se você executar apenas um carregamento ou exportação completa, não será necessário um fluxo de alterações.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster Amazon DocumentDB com permissão para ler dados seguindo as etapas em Criar um cluster Amazon DocumentDB no Guia do desenvolvedor do Amazon DocumentDB. Se você usar a infraestrutura de CDC, configure o cluster do Amazon DocumentDB para publicar fluxos de alterações.
2. Habilitar o TLS no cluster do Amazon DocumentDB.
3. Configure um CIDR VPC de um espaço de endereço privado para uso com Ingestão. OpenSearch
4. Configure a autenticação em seu cluster Amazon DocumentDB com. AWS Secrets Manager Habilite o rodízio de segredos seguindo as etapas em Rodízio de sanhas automático para o Amazon DocumentDB. Para saber mais, consulte Acesso ao banco de dados usando controle de acesso baseado em perfil e Segurança no Amazon DocumentDB.
5. Se você usar um fluxo de alterações para assinar as alterações de dados em sua coleção do Amazon DocumentDB, evite a perda de dados estendendo o período de retenção para até 7 dias usando o parâmetro `change_stream_log_retention_duration`. Os eventos de fluxos de alterações são armazenados por 3 horas, por padrão, após a gravação do evento, o que não é tempo suficiente para grandes coleções. Para modificar o período de retenção do fluxo de alterações, consulte Modificar a duração da retenção de fluxos de alterações.
6. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor. Para obter mais informações, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
7. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster Amazon DocumentDB em seu domínio ou coleção.

   O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
```

```
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline do Amazon DocumentDB, defina o perfil de pipeline que você deseja usar na configuração do pipeline e adicione as seguintes permissões do pipeline do Amazon DocumentDB nesse perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowS3ListObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket"
            ],
            "Condition": {
                "StringLike": {
                    "s3:prefix": "s3-prefix/*"
                }
            }
        },
        {
            "Sid": "allowReadAndWriteToS3ForExportStream",
```

```
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket/s3-prefix/*"
            ]
        },
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": [
                "arn:aws:secretsmanager:us-east-1:111122223333:secret:secret-
name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:111122223333:network-interface/*",
                "arn:aws:ec2:*:111122223333:subnet/*",
                "arn:aws:ec2:*:111122223333:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
```

```
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/OSISManaged": "true"
                }
            }
        }
    ]
}
```

Você deve fornecer as permissões do Amazon EC2 acima sobre a função do IAM que você usa para criar o pipeline de OpenSearch ingestão, pois o pipeline usa essas permissões para criar e excluir uma interface de rede em sua VPC. O pipeline pode acessar o cluster do Amazon DocumentDB somente por meio dessa interface de rede.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que especifica o Amazon DocumentDB como origem. Observe que, para preencher o nome do índice, a função `getMetadata` usa *`documentdb_collection`* como chave de metadados. Se quiser usar um nome de índice diferente sem o método `getMetadata`, você pode usar a configuração `index: "`*`my_index_name`*`"`.

```
version: "2"
documentdb-pipeline:
  source:
    documentdb:
      acknowledgments: true
      host: "https://docdb-cluster-id.us-east-1.docdb.amazonaws.com"
      port: 27017
```

```
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
      aws:
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "path" #optional path for storing the temporary data
      collections:
        - collection: "dbname.collection"
          export: true
          stream: true
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      index: "${getMetadata(\"documentdb_collection\")}"
      index_type: custom
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-docdb-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Você pode usar um esquema pré-configurado do Amazon DocumentDB para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline à sua VPC para usar o Amazon DocumentDB como fonte. Para fazer isso, encontre a seção Opções de rede de origem, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.

Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre OpenSearch ingestão e Amazon DocumentDB, certifique-se de que o CIDR de VPC do Amazon DocumentDB seja diferente do CIDR para ingestão. OpenSearch

Para saber mais, consulte Configurar o acesso à VPC para um pipeline.

Consistência de dados

O pipeline garante a consistência dos dados pesquisando continuamente ou recebendo alterações do cluster Amazon DocumentDB e atualizando os documentos correspondentes no OpenSearch índice.

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção.

Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento, portanto, você deve omiti-lo `document_id: "${getMetadata(\"primary_key\")}"` na configuração do coletor do pipeline.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de modo que cada alteração de dados no Amazon DocumentDB seja reconciliada com as alterações correspondentes no documento. OpenSearch

Mapear tipo de dados

OpenSearch O serviço mapeia dinamicamente os tipos de dados em cada documento recebido para o tipo de dados correspondente no Amazon DocumentDB. A tabela a seguir mostra como o OpenSearch Service mapeia automaticamente vários tipos de dados.

Tipo de dados	OpenSearch	Amazon DocumentDB
Inteiro	OpenSearch mapeia automatic amente os valores inteiros do Amazon DocumentDB para números inteiros. OpenSearch OpenSearch mapeia dinamicamente o campo com base no primeiro documento enviado. Se houver uma combinação de	O Amazon DocumentDB oferece suporte a números inteiros.

Tipo de dados	OpenSearch	Amazon DocumentDB
	tipos de dados para o mesmo atributo no Amazon DocumentDB, o mapeamento automático poderá falhar. Por exemplo, se o primeiro documento tiver um atributo longo e um documento posterior tiver esse mesmo atributo como número inteiro, OpenSearch não conseguirá ingerir o segundo documento . Nesses casos, é necessário fornecer um modelo de mapeamento explícito que escolhe o tipo de número mais flexível, como o seguinte: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	

Tipo de dados	OpenSearch	Amazon DocumentDB
Longo	OpenSearch mapeia automaticamente valores longos do Amazon DocumentDB para OpenSearch longos. OpenSearch mapeia dinamicamente o campo com base no primeiro documento enviado. Se houver uma combinação de tipos de dados para o mesmo atributo no Amazon DocumentDB, o mapeamento automático poderá falhar. Por exemplo, se o primeiro documento tiver um atributo longo e um documento posterior tiver esse mesmo atributo como número inteiro, OpenSearch não conseguirá ingerir o segundo documento . Nesses casos, é necessário fornecer um modelo de mapeamento explícito que escolhe o tipo de número mais flexível, como o seguinte: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	O Amazon DocumentDB oferece suporte a valores longos.

Tipo de dados	OpenSearch	Amazon DocumentDB
String	OpenSearch mapeia automaticamente valores de string como texto. Em algumas situações, como valores enumerados, é possível mapear para o tipo de palavra-chave. O exemplo a seguir mostra como mapear um atributo do Amazon DocumentDB nomeado `PartType` para uma OpenSearch palavra-chave. `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	O Amazon DocumentDB oferece suporte a strings.

Tipo de dados	OpenSearch	Amazon DocumentDB
Duplo	OpenSearch mapeia automatic amente os valores duplos do Amazon DocumentDB para OpenSearch duplos. OpenSearch mapeia dinamicamente o campo com base no primeiro documento enviado. Se houver uma combinação de tipos de dados para o mesmo atributo no Amazon DocumentDB, o mapeamento automático poderá falhar. Por exemplo, se o primeiro documento tiver um atributo longo e um documento posterior tiver esse mesmo atributo como número inteiro, OpenSearch não conseguirá ingerir o segundo documento . Nesses casos, é necessário fornecer um modelo de mapeamento explícito que escolhe o tipo de número mais flexível, como o seguinte: `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	O Amazon DocumentDB oferece suporte a valores duplos.

Tipo de dados	OpenSearch	Amazon DocumentDB
Data	Por padrão, a data é mapeada para um número inteiro em OpenSearch. Você pode definir um modelo de mapeament o personalizado para mapear uma data até uma OpenSearch data. `{ "template": { "mappings": { "properties": { "myDateField": { "type": "date", "format": "epoch_second" } } } } }`	O Amazon DocumentDB oferece suporte a datas.

Tipo de dados	OpenSearch	Amazon DocumentDB
Timestamp	Por padrão, o timestamp é mapeado para um número inteiro em. OpenSearch Você pode definir um modelo de mapeamento personalizado para mapear uma data até uma OpenSearch data. `{ "template": { "mappings": { "properties": { "myTimestampField": { "type": "date", "format": "epoch_second" } } } } }`	O Amazon DocumentDB oferece suporte a carimbos de data/hora.
Booleano	OpenSearch mapeia um tipo booleano do Amazon DocumentDB em um OpenSearch tipo booleano.	O Amazon DocumentDB oferece suporte a atributos do tipo booleano.

Tipo de dados	OpenSearch	Amazon DocumentDB
Decimal	OpenSearch mapeia os atributos de mapas do Amazon DocumentDB para campos aninhados. Os mesmos mapeamentos são aplicáveis a um campo aninhado. O exemplo a seguir mapeia uma string em um campo aninhado para um tipo de palavra-chave em OpenSearch: `{ "template": { "mappings": { "properties": { "myDecimalField": { "type": "double" } } } } }` Com esse mapeamento personalizado, você pode consultar e agregar o campo com precisão de dois níveis. O valor original mantém a precisão total na `_source` propriedade do OpenSearc h documento. Sem esse mapeamento, OpenSearch usa texto por padrão.	O Amazon DocumentDB oferece suporte a números decimais.
Expressão Regular	O tipo regex cria campos aninhados . Entre eles estão *<myFieldN ame>* `.pattern` e *<myFieldN ame>* `.options`.	O Amazon DocumentDB oferece suporte a expressões regulares.

Tipo de dados	OpenSearch	Amazon DocumentDB
Dados binários	OpenSearch mapeia automatic amente os dados binários do Amazon DocumentDB para OpenSearch texto. Você pode fornecer um mapeamento para escrevê-los como campos binários OpenSearch. O exemplo a seguir mostra como mapear um campo Amazon DocumentD B nomeado `imageData` para um campo OpenSearch binário. `{ "template": { "mappings": { "properties": { "imageData": { "type": "binary" } } } } }`	O Amazon DocumentDB oferece suporte a campos de dados binários.
ObjectId	Campos com um tipo de ID de objeto são mapeados para campos de OpenSearch texto. O valor será a representação em string do objectId.	O Amazon DocumentDB oferece suporte a objectIds.

Tipo de dados	OpenSearch	Amazon DocumentDB
Null	OpenSearch pode ingerir documentos com o tipo nulo Amazon DocumentDB. Ele salva o valor como um valor nulo no documento. Não há mapeamento para esse tipo, e esse campo não é indexado nem pesquisável. Se o mesmo nome de atributo for usado para um tipo nulo e depois for alterado para um tipo diferente, como string, OpenSearch criará um mapeament o dinâmico para o primeiro valor não nulo. Os valores subsequentes ainda podem ser valores nulos do Amazon DocumentDB.	O Amazon DocumentDB oferece suporte a campos do tipo nulo.
Não definido	OpenSearch pode ingerir documento s com o tipo indefinido do Amazon DocumentDB. Ele salva o valor como um valor nulo no documento. Não há mapeamento para esse tipo, e esse campo não é indexado nem pesquisável. Se o mesmo nome de campo for usado para um tipo indefinido e depois mudar para um tipo diferente, como string, OpenSearch cria um mapeamento dinâmico para o primeiro valor não indefinido. Os valores subsequentes ainda podem ser valores indefinidos do Amazon DocumentDB.	O Amazon DocumentDB oferece suporte a campos do tipo indefinido.

Tipo de dados	OpenSearch	Amazon DocumentDB
MinKey	OpenSearch pode ingerir documento s com o tipo Amazon DocumentDB MinKey. Ele salva o valor como um valor nulo no documento. Não há mapeament o para esse tipo, e esse campo não é indexado nem pesquisável. Se o mesmo nome de campo for usado para um tipo MinKey e depois for alterado para um tipo diferente, como string, OpenSearch criará um mapeamento dinâmico para o primeiro valor que não seja MinKey. Os valores subsequentes ainda podem ser valores minKey do Amazon DocumentDB.	O Amazon DocumentDB oferece suporte a campos do tipo minKey.
MaxKey	OpenSearch pode ingerir documento s com o tipo Amazon DocumentD B MaxKey. Ele salva o valor como um valor nulo no documento. Não há mapeamento para esse tipo, e esse campo não é indexado nem pesquisável. Se o mesmo nome de campo for usado para um tipo MaxKey e depois for alterado para um tipo diferente, como string, OpenSearch criará um mapeamento dinâmico para o primeiro valor que não seja MaxKey. Os valores subsequentes ainda podem ser valores maxKey do Amazon DocumentDB.	O Amazon DocumentDB oferece suporte a campos do tipo maxKey.

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos falharem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para o Amazon DocumentDB:

- Atualmente, a integração de OpenSearch ingestão com o Amazon DocumentDB não oferece suporte à ingestão entre regiões. Seu cluster Amazon DocumentDB e seu pipeline OpenSearch de ingestão devem estar no mesmo. Região da AWS
- Atualmente, a integração de OpenSearch ingestão com o Amazon DocumentDB não oferece suporte à ingestão entre contas. Seu cluster Amazon DocumentDB e seu pipeline OpenSearch de ingestão devem estar no mesmo. Conta da AWS
- Um pipeline OpenSearch de ingestão suporta somente um cluster Amazon DocumentDB como origem.
- A integração do OpenSearch Inestion com o Amazon DocumentDB oferece suporte específico a clusters baseados em instâncias do Amazon DocumentDB. Ele não é compatível com clusters elásticos do Amazon DocumentDB.
- A integração do OpenSearch Inestion só é compatível com AWS Secrets Manager um mecanismo de autenticação para seu cluster Amazon DocumentDB.
- Você não pode atualizar a configuração existente do pipeline para ingerir dados de um banco de dados ou coleção diferente. Em vez disso, você deve criar um novo pipeline.

CloudWatch Alarmes recomendados

Para obter o melhor desempenho, recomendamos que você use os seguintes CloudWatch alarmes ao criar um pipeline de OpenSearch ingestão para acessar um cluster do Amazon DocumentDB como fonte.

CloudWatch Alarme	Description
<pipeline-name> .documentdb.Credenciais alteradas	Essa métrica indica com que frequência AWS os segredos são alternados.
<pipeline-name> .documentdb. executorR efreshErrors	Essa métrica indica falhas na atualização de segredos da AWS .
<pipeline-name> .documentdb. exportRec ordsTotal	Essa métrica indica o número de registros exportados do Amazon DocumentDB.
<pipeline-name> .documentdb. exportRec ordsProcessed	Essa métrica indica o número de registros processados pelo pipeline OpenSearch de ingestão.
<pipeline-name> .documentdb. exportRec ordProcessingErros	Essa métrica indica o número de erros de processamento em um pipeline OpenSearch de ingestão durante a leitura dos dados de um cluster do Amazon DocumentDB.
<pipeline-name> .documentdb. exportRec ordsSuccessTotal	Essa métrica indica o número total de registros de exportação processados com êxito.
<pipeline-name> .documentdb. exportRec ordsFailedTotal	Essa métrica indica o número total de registros de exportação com falha no processamento.
<pipeline-name> .documentdb.bytes recebidos	Essa métrica indica o número total de bytes recebidos por um pipeline OpenSearch de ingestão.
<pipeline-name> .documentdb.bytes processados	Essa métrica indica o número total de bytes processados por um pipeline OpenSearch de ingestão.
<pipeline-name> .documentdb. exportPar titionQueryTotal	Essa métrica indica o total da partição de exportação.
<pipeline-name> .documentdb. streamRec ordsSuccessTotal	Essa métrica indica o número de registros processados com êxito a partir do fluxo.

CloudWatch Alarme	Description
<pipeline-name> .documentdb. streamRecordsFailedTotal	Essa métrica indica o número total de registros com falha no processamento do fluxo.

Usando um pipeline de OpenSearch ingestão com o Confluent Cloud Kafka

Você pode usar um pipeline de OpenSearch ingestão para transmitir dados dos clusters do Confluent Cloud Kafka para domínios do OpenSearch Amazon Service e coleções sem servidor. OpenSearch OpenSearch A ingestão suporta configurações de rede pública e privada para o streaming de dados dos clusters do Confluent Cloud Kafka para domínios ou coleções gerenciados por Service ou Serverless. OpenSearch OpenSearch

Conectividade com clusters do Kafka públicos do Confluent Cloud

Você pode usar pipelines de OpenSearch ingestão para migrar dados de um cluster do Confluent Cloud Kafka com uma configuração pública, o que significa que o nome DNS do domínio pode ser resolvido publicamente. Para fazer isso, configure um pipeline de OpenSearch ingestão com o cluster Kafka público do Confluent Cloud como origem e OpenSearch Service ou OpenSearch Serverless como destino. Isso processa seus dados de streaming de um cluster de origem autogerenciado para um domínio ou AWS coleção de destino gerenciado.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster de clusters do Confluent Cloud Kafka atuando como fonte. O cluster deve conter os dados que você deseja ingerir no OpenSearch Service.
2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para obter mais informações, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
3. Configure a autenticação em seu cluster do Confluent Cloud Kafka com o AWS Secrets Manager. Habilite a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline de cluster do Confluent Cloud Kafka, configure a função do pipeline que você deseja usar na configuração do pipeline e adicione permissão para gravar em um domínio de OpenSearch serviço ou coleção OpenSearch sem servidor, bem como permissão para ler segredos do Secrets Manager.

A permissão a seguir é necessária para gerenciar a interface de rede:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
                "arn:aws:ec2:us-east-1:111122223333:subnet/*",
                "arn:aws:ec2:us-east-1:111122223333:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:subnet/*"
        },
        {
            "Effect": "Allow",
            "Action": [ "ec2:CreateTags" ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
            "Condition": {
               "StringEquals": { "aws:RequestTag/OSISManaged": "true" }
            }
        }
    ]
}
```

A seguir está a permissão necessária para ler segredos do AWS Secrets Manager serviço:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": ["secretsmanager:GetSecretValue"],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:,secret-name"]
        }
    ]
}
```

As seguintes permissões são necessárias para gravar em um domínio do Amazon OpenSearch Service:

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que especifica seu Confluent Cloud Kafka como fonte.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de um cluster de origem do Confluent Kafka para uma coleção de VPC sem servidor. OpenSearch Forneça uma política de acesso à rede na configuração do pipeline. Você pode usar um registro de esquema do Confluent para definir um esquema do Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Conectividade com clusters do Confluent Cloud Kafka em uma VPC

Você também pode usar pipelines OpenSearch de ingestão para migrar dados de um cluster do Confluent Cloud Kafka executado em uma VPC. Para fazer isso, configure um pipeline de OpenSearch ingestão com um cluster do Confluent Cloud Kafka como origem e OpenSearch serviço ou OpenSearch sem servidor como destino. Isso processa seus dados de streaming de um cluster de origem do Confluent Cloud Kafka para um domínio ou coleção de destino gerenciado pela AWS.

OpenSearch A ingestão é compatível com clusters do Confluent Cloud Kafka configurados em todos os modos de rede compatíveis no Confluent. Os seguintes modos de configuração de rede são suportados como fonte na OpenSearch Ingestão:

- AWS Emparelhamento de VPC
- AWS PrivateLink para clusters dedicados
- AWS PrivateLink para clusters corporativos
- AWS Transit Gateway

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster do Confluent Cloud Kafka com uma configuração de rede VPC que contenha os dados que você deseja ingerir no Service. OpenSearch
2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para saber mais, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
3. Configure a autenticação em seu cluster do Confluent Cloud Kafka com o AWS Secrets Manager. Habilite a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Obtenha o ID da VPC que tem acesso ao cluster do Confluent Cloud Kafka. Escolha o CIDR da VPC a ser usado pela ingestão. OpenSearch

> **Note**
>
> Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline de OpenSearch ingestão à sua VPC para usar o cluster Confluent Cloud Kafka. Para fazer isso, encontre a seção Configuração de

rede, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.
Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre OpenSearch ingestão e autogerenciamento OpenSearch, certifique-se de que o CIDR autogerenciado da OpenSearch VPC seja diferente do CIDR para ingestão. OpenSearch

5. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

> **Note**
>
> Se usar AWS PrivateLink para conectar seu Confluent Cloud Kafka, precisará configurar as Opções de DHCP da VPC. Os nomes de host DNS e a resolução do DNS devem estar habilitados.
> Especificamente, use os seguintes valores de conjunto de opções:
> Clusters corporativos:
>
> ```
> domain-name: aws.private.confluent.cloud
> domain-name-servers: AmazonProvidedDNS
> ```
>
> Clusters dedicados:
>
> ```
> domain-name: aws.confluent.cloud
> domain-name-servers: AmazonProvidedDNS
> ```
>
> Essa alteração garante que a resolução de DNS para o PrivateLink endpoint do Confluent funcione corretamente na VPC.

O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline, configure o perfil de pipeline que você deseja usar na configuração do pipeline e adicione as seguintes permissões nesse perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
```

```
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
```

```
        }
    ]
}
```

Você deve fornecer as permissões do Amazon EC2 acima sobre a função do IAM que você usa para criar o pipeline de OpenSearch ingestão, pois o pipeline usa essas permissões para criar e excluir uma interface de rede em sua VPC. O pipeline só pode acessar o cluster do Kafka por meio dessa interface de rede.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline OpenSearch de ingestão como o seguinte, que especifica o Kafka como a origem.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de um cluster de origem do Confluent Kafka para uma coleção de VPC sem servidor. OpenSearch Forneça uma política de acesso à rede na configuração do pipeline. Você pode usar um registro de esquema do Confluent para definir um esquema do Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
```

```
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-2"
```

Usando um pipeline OpenSearch de ingestão com Amazon Managed Streaming for Apache Kafka

Você pode usar o plug-in Kafka para ingerir dados do Amazon Managed Streaming for Apache Kafka (Amazon MSK) em seu pipeline de ingestão. OpenSearch Com o Amazon MSK, você pode criar e executar aplicativos que usam o Apache Kafka para processar dados em streaming. OpenSearch A ingestão é usada AWS PrivateLink para se conectar ao Amazon MSK. Você pode ingerir dados dos clusters do Amazon MSK e do Amazon MSK Serverless. A única diferença entre os dois processos são as etapas de pré-requisito que você deve seguir antes de configurar seu pipeline.

Tópicos

- Pré-requisitos do Amazon MSK provisionados
- Pré-requisitos do Amazon MSK Serverless
- Etapa 1: configurar o perfil de pipeline
- Etapa 2: Criar o pipeline
- Etapa 3: (Opcional) Usar o Registro do AWS Glue Esquema
- Etapa 4: (Opcional) Configurar as unidades computacionais recomendadas (OCUs) para o pipeline do Amazon MSK

Pré-requisitos do Amazon MSK provisionados

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster provisionado do Amazon MSK seguindo as etapas em Criar um cluster no Guia do desenvolvedor do Amazon Managed Streaming para Apache Kafka. Para o tipo de corretor, escolha qualquer opção, exceto `t3` os tipos, pois eles não são compatíveis com a OpenSearch ingestão.
2. Depois que o cluster tiver um status Ativo, siga as etapas em Ativar a conectividade de várias VPCs.
3. Siga as etapas em Anexar uma política de cluster ao cluster MSK para anexar uma das políticas a seguir, dependendo se o cluster e o pipeline estão na mesma Conta da AWS. Essa política permite que o OpenSearch Inestion crie uma AWS PrivateLink conexão com seu cluster Amazon MSK e leia dados de tópicos do Kafka. Lembre-se de atualizar o `resource` com seu próprio ARN.

   As políticas a seguir se aplicam quando o cluster e o pipeline estão na mesma Conta da AWS:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
```

```
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Se seu cluster Amazon MSK estiver em um pipeline Conta da AWS diferente do seu pipeline, anexe a seguinte política em vez disso. Observe que o acesso entre contas é possível somente com clusters provisionados do Amazon MSK e não com clusters do Amazon MSK Serverless. O ARN do AWS `principal` deve ser o ARN da mesma função de pipeline que você fornece à configuração do pipeline:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
```

```
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "kafka-cluster:*",
        "kafka:*"
      ],
      "Resource": [
        "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-name/cluster-id",
        "arn:aws:kafka:us-east-1:111122223333:topic/cluster-name/cluster-id/*",
        "arn:aws:kafka:us-east-1:111122223333:group/cluster-name/*"
      ]
    }
  ]
}
```

4. Crie um tópico do Kafka seguindo as etapas em Criar um tópico. Certifique-se de que *BootstrapServerString* seja um dos bootstrap de endpoint privado (VPC única). URLs O valor de `--replication-factor` deve ser 2 ou 3, com base no número de zonas que seu cluster do Amazon MSK tem. O valor de `--partitions` deve ser pelo menos `10`.
5. Produza e consuma dados seguindo as etapas em Produzir e consumir dados. Novamente, verifique se esse *BootstrapServerString* é um dos seus bootstrap de endpoint privado (VPC única). URLs

Pré-requisitos do Amazon MSK Serverless

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster do Amazon MSK Serverless seguindo as etapas em Criar um cluster do MSK Serverless no Guia do desenvolvedor do Amazon Managed Streaming para Apache Kafka.
2. Depois que o cluster tiver um status Ativo, siga as etapas em Anexar uma política de cluster ao cluster do MSK para anexar a política a seguir. Lembre-se de atualizar o `resource` com seu próprio ARN.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Essa política permite que o OpenSearch Inestion crie uma AWS PrivateLink conexão com seu cluster Amazon MSK Serverless e leia dados de tópicos do Kafka. Essa política se aplica quando seu cluster e pipeline estão no mesmo lugar, o que deve ser verdade Conta da AWS, pois o Amazon MSK Serverless não oferece suporte ao acesso entre contas.

3. Crie um tópico do Kafka seguindo as etapas em Criar um tópico. Certifique-se de que *`BootstrapServerString`* seja um dos seus bootstrap URLs IAM de Simple Authentication and Security Layer (SASL). O valor de `--replication-factor` deve ser 2 ou 3, com base no

número de zonas que seu cluster do Amazon MSK Serverless tem. O valor de `--partitions` deve ser pelo menos `10`.

4. Produza e consuma dados seguindo as etapas em Produzir e consumir dados. Novamente, certifique-se de que esse *`BootstrapServerString`* seja um dos seus bootstrap URLs do IAM Simple Authentication and Security Layer (SASL).

Etapa 1: configurar o perfil de pipeline

Depois de configurar seu cluster provisionado ou sem servidor do Amazon MSK, adicione as seguintes permissões do Kafka na função do pipeline que você deseja usar na configuração do pipeline:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:Connect",
                "kafka-cluster:AlterCluster",
                "kafka-cluster:DescribeCluster",
                "kafka:DescribeClusterV2",
                "kafka:GetBootstrapBrokers"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:cluster/cluster-
name/cluster-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:*Topic*",
                "kafka-cluster:ReadData"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:topic/cluster-name/cluster-
id/topic-name"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:AlterGroup",
                "kafka-cluster:DescribeGroup"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:111122223333:group/cluster-name/*"
            ]
        }
    ]
}
```

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que especifica o Kafka como fonte:

```
version: "2"
log-pipeline:
  source:
    kafka:
      acknowledgements: true
      topics:
      - name: "topic-name"
        group_id: "grouplambd-id"
      aws:
        msk:
          arn: "arn:aws:kafka:region:account-id:cluster/cluster-name/cluster-id"
        region: "us-west-2"
  processor:
  - grok:
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
```

```
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index_name"
      aws_region: "region"
      aws_sigv4: true
```

Você pode usar um esquema do Amazon MSK pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Etapa 3: (Opcional) Usar o Registro do AWS Glue Esquema

Ao usar o OpenSearch Inestion com o Amazon MSK, você pode usar o formato de dados AVRO para esquemas hospedados no Schema Registry. AWS Glue Com o registro de esquema do AWS Glue, você pode descobrir, controlar e evoluir centralmente esquemas de fluxo de dados.

Para usar essa opção, habilite o esquema `type` na configuração do seu pipeline:

```
schema:
  type: "aws_glue"
```

Você também deve AWS Glue fornecer permissões de acesso de leitura em sua função de funil. Você pode usar a política AWS gerenciada chamada AWSGlueSchemaRegistryReadonlyAccess. Além disso, seu registro deve estar na mesma Conta da AWS região do pipeline OpenSearch de ingestão.

Etapa 4: (Opcional) Configurar as unidades computacionais recomendadas (OCUs) para o pipeline do Amazon MSK

Cada unidade computacional tem um consumidor por tópico. Os corretores equilibram as partições entre esses consumidores para um determinado tópico. No entanto, quando o número de partições é maior que o número de consumidores, o Amazon MSK hospeda várias partições em cada consumidor. OpenSearch A ingestão tem escalonamento automático integrado para aumentar ou diminuir a escala com base no uso da CPU ou no número de registros pendentes no pipeline.

Para um desempenho ideal, distribua suas partições em várias unidades de computação para processamento paralelo. Se os tópicos tiverem um grande número de partições (por exemplo, mais de 96, que é o máximo OCUs por pipeline), recomendamos que você configure um pipeline com OCUs 1—96. Isso ocorre porque ele será escalado automaticamente conforme necessário. Se um tópico tiver um número baixo de partições (por exemplo, menos de 96), mantenha o máximo de unidades computacionais igual ao número de partições.

Quando um pipeline tiver mais de um tópico, escolha o tópico com o maior número de partições como referência para configurar o máximo de unidades computacionais. Ao adicionar outro pipeline com um novo conjunto de OCUs ao mesmo tópico e grupo de consumidores, você pode escalar a taxa de transferência quase linearmente.

Usando um pipeline OpenSearch de ingestão com o Amazon RDS

Você pode usar um pipeline de OpenSearch ingestão com o Amazon RDS para exportar dados existentes e transmitir alterações (como criar, atualizar e excluir) para domínios e coleções do Amazon OpenSearch Service. O pipeline OpenSearch de ingestão incorpora a infraestrutura de captura de dados de alteração (CDC) para fornecer uma forma de alta escala e baixa latência de transmitir dados continuamente do Amazon RDS. O RDS para MySQL e o RDS para PostgreSQL são compatíveis.

Há duas maneiras de usar o Amazon RDS como origem para processar dados: com e sem um snapshot inicial completo. Um snapshot inicial completo é um snapshot das tabelas especificadas, e ele é exportado para o Amazon S3. A partir daí, um pipeline de OpenSearch ingestão o envia para um índice em um domínio ou o particiona em vários índices em um domínio. Para manter os dados no Amazon RDS e OpenSearch consistentes, o pipeline sincroniza todos os eventos de criação, atualização e exclusão nas tabelas nas instâncias do Amazon RDS com os documentos salvos no OpenSearch índice ou índices.

Quando você usa um snapshot inicial completo, seu pipeline de OpenSearch ingestão primeiro ingere o snapshot e depois começa a ler os dados dos fluxos de alteração do Amazon RDS. Eventualmente, ele recupera e mantém a consistência de dados quase em tempo real entre o Amazon RDS e. OpenSearch

Você também pode usar a integração de OpenSearch ingestão com o Amazon RDS para rastrear a captura de dados de alterações e ingerir todas as atualizações no Aurora para. OpenSearch Escolha essa opção se você já tiver um snapshot completo de algum outro mecanismo ou se quiser apenas capturar todas as alterações nos dados em uma instância do Amazon RDS.

Ao escolher essa opção, você precisa configurar o registro em log binário do Amazon RDS para MySQL ou configurar a replicação lógica para a instância do banco de dados do Amazon RDS para PostgresSQL.

Tópicos

- RDS para MySQL
- RDS para PostgreSQL

RDS para MySQL

Conclua as etapas a seguir para configurar um pipeline OpenSearch de ingestão com o Amazon RDS for RDS for MySQL.

Tópicos

- Pré-requisitos do RDS para MySQL
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline
- Consistência de dados
- Mapear tipo de dados
- Limitações
- CloudWatch Alarmes recomendados

Pré-requisitos do RDS para MySQL

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um grupo personalizado de parâmetros de banco de dados no Amazon RDS para configurar o registro em log binário e definir os parâmetros a seguir.

   ```
   binlog_format=ROW
   binlog_row_image=full
   binlog_row_metadata=FULL
   ```

   Além disso, certifique-se de que o parâmetro `binlog_row_value_options` não seja definido como `PARTIAL_JSON`.

   Para saber mais, consulte Configurar o registro em log binário do RDS for MySQL.
2. Selecione ou crie uma instância do banco de dados do RDS para MySQL e associe o grupo de parâmetros criado na etapa anterior à instância do banco de dados.
3. Verifique se os backups automatizados estão habilitados no banco de dados. Para saber mais, consulte Habilitando backups automatizados.
4. Configure a retenção de registros em log binário com tempo suficiente para que a replicação ocorra, por exemplo, 24 horas. Para saber mais, consulte Definir e exibir a configuração de logs binários no Guia do usuário do Amazon RDS.

5. Configure a autenticação de nome de usuário e senha na instância do Amazon RDS usando o gerenciamento de senhas com o Amazon RDS e o AWS Secrets Manager. Você também pode criar uma username/password combinação criando um segredo do Secrets Manager.
6. Se você usar o recurso de snapshot inicial completo, crie uma função AWS KMS key e uma do IAM para exportar dados do Amazon RDS para o Amazon S3.

   O perfil do IAM também deve ter a seguinte política de permissão:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

   O perfil também deve ter a seguinte relação de confiança:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
```

```
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

7. Selecione ou crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.
8. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados da sua instância de banco de dados Amazon RDS em seu domínio ou coleção.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos de pipeline do Amazon RDS, configure o perfil do pipeline para usar na configuração do pipeline. Adicione também as seguintes permissões para a origem do Amazon RDS ao perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
```

```
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
```

```
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBSnapshots",
    "rds:CreateDBSnapshot",
    "rds:AddTagsToResource"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:DB-id",
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
```

```
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Etapa 2: Criar o pipeline

Configure um pipeline de OpenSearch ingestão semelhante ao seguinte. O exemplo de pipeline especifica uma instância do Amazon RDS como a origem.

```
version: "2"
rds-mysql-pipeline:
  source:
    rds:
```

```
      db_identifier: "instance-id"
      engine: mysql
      database: "database-name"
      tables:
        include:
          - "table1"
          - "table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

Você pode usar um esquema pré-configurado do Amazon RDS para criar esse pipeline. Para saber mais, consulte Trabalhar com esquemas.

Para usar o Amazon Aurora como origem, você precisa configurar o acesso à VPC para o pipeline. A VPC escolhida deve ser a mesma VPC que a origem do Amazon Aurora usa. Depois, selecione uma ou mais sub-redes e um ou mais grupos de segurança da VPC. Observe que o pipeline precisa de acesso de rede a um banco de dados MySQL do Aurora, por isso, você também deve se certificar de que o cluster do Aurora esteja configurado com um grupo de segurança da VPC que permita tráfego de entrada do grupo de segurança da VPC do pipeline para a porta do banco de dados. Para saber mais, consulte Controlar o acesso com grupos de segurança.

Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline à sua VPC para usar o Amazon Aurora como fonte. Para fazer isso, encontre a seção Configuração de rede, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.

Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre OpenSearch Ingestão e Amazon RDS, certifique-se de que o CIDR VPC do Amazon RDS seja diferente do CIDR para Ingestão. OpenSearch

Para saber mais, consulte Configurar o acesso à VPC para um pipeline.

Consistência de dados

O pipeline garante a consistência dos dados pesquisando continuamente ou recebendo alterações da instância do Amazon RDS e atualizando os documentos correspondentes no OpenSearch índice.

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção. Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento, portanto, você deve omiti-lo `document_id: "${getMetadata(\"primary_key\")}"` na configuração do coletor do pipeline.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de modo que cada alteração de dados no Amazon RDS seja reconciliada com as alterações correspondentes no documento. OpenSearch

Mapear tipo de dados

OpenSearch O pipeline de ingestão mapeia os tipos de dados do MySQL em representações que são adequadas OpenSearch para o consumo de domínios ou coleções de serviços. Se nenhum modelo de mapeamento estiver definido em OpenSearch, determina OpenSearch automaticamente os tipos de campo com mapeamento dinâmico com base no primeiro documento enviado. Você também pode definir explicitamente os tipos de campo que funcionam melhor para você por OpenSearch meio de um modelo de mapeamento.

A tabela abaixo lista os tipos de dados do MySQL e os tipos de OpenSearch campo correspondentes. A coluna Tipo de OpenSearch campo padrão mostra o tipo de campo correspondente OpenSearch se nenhum mapeamento explícito for definido. Nesse caso, determina OpenSearch automaticamente os tipos de campo com mapeamento dinâmico. A coluna Tipo de OpenSearch campo recomendado é o tipo de campo correspondente que é recomendado especificar explicitamente em um modelo de mapeamento. Esses tipos de campo estão mais alinhados com os tipos de dados no MySQL e geralmente podem permitir melhores recursos de pesquisa disponíveis no. OpenSearch

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
BIGINT	longo	longo
BIGINT UNSIGNED	longo	unsigned long
BIT	longo	byte, short, integer ou long dependendo do número de bits
DECIMAL	texto	double ou keyword
DOUBLE	flutuação	double
FLOAT	flutuação	flutuação
INT	longo	integer

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
INT UNSIGNED	longo	longo
MEDIUMINT	longo	integer
MEDIUMINT UNSIGNED	longo	integer
NUMERIC	texto	double ou keyword
SMALLINT	longo	curto
SMALLINT UNSIGNED	longo	integer
TINYINT	longo	byte
TINYINT UNSIGNED	longo	curto
BINARY	texto	binary
BLOB	texto	binary
CHAR	texto	texto
ENUM	texto	palavra-chave
LONGBLOB	texto	binary
LONGTEXT	texto	texto
MEDIUMBLOB	texto	binary
MEDIUMTEXT	texto	texto

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
SET	texto	palavra-chave
TEXT	texto	texto
TINYBLOB	texto	binary
TINYTEXT	texto	texto
VARBINAR	texto	binary
VARCHAR	texto	texto
DATE	long (em milissegundos epoch)	date
DATETIME	long (em milissegundos epoch)	date
TIME	long (em milissegundos epoch)	date
TIMESTAM	long (em milissegundos epoch)	date
YEAR	long (em milissegundos epoch)	date
GEOMETR	text (em formato WKT)	geo_shape
GEOMETR OLLECTIO	text (em formato WKT)	geo_shape
LINESTRIN G	text (em formato WKT)	geo_shape
MULTILINE STRING	text (em formato WKT)	geo_shape
MULTIPOIN T	text (em formato WKT)	geo_shape

Tipo de dados MySQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
MULTIPOLYGON	text (em formato WKT)	geo_shape
POINT	text (em formato WKT)	geo_point ou geo_shape
POLYGON	text (em formato WKT)	geo_shape
JSON	texto	objeto

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos não funcionarem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para RDS for MySQL:

- A integração é compatível com apenas um banco de dados MySQL por pipeline.
- Atualmente, a integração não oferece suporte à ingestão de dados entre regiões; sua instância e OpenSearch domínio do Amazon RDS devem estar nos mesmos. Região da AWS
- Atualmente, a integração não oferece suporte à ingestão de dados entre contas; sua instância do Amazon RDS e seu pipeline de OpenSearch ingestão devem estar no mesmo nível. Conta da AWS
- Certifique-se de que a autenticação esteja habilitada na instância do Amazon RDS usando o Secrets Manager, que é o único mecanismo de autenticação compatível.
- A configuração existente do pipeline não pode ser atualizada para ingerir dados de um banco de dados diferente ou de and/or uma tabela diferente. Para atualizar o banco de dados e/ou o nome da tabela de um pipeline, você deve criar um novo pipeline.

- Instruções em Data Definition Language (DDL) não são compatíveis em geral. A consistência de dados não será mantida se:
  - As chaves primárias são alteradas (add/delete/rename).
  - As tabelas forem eliminadas/truncadas.
  - Os nomes das colunas ou os tipos de dados forem alterados.
- Se as tabelas MySQL a serem sincronizadas não tiverem chaves primárias definidas, a consistência de dados não será garantida. Você precisará definir a `document_id` opção personalizada na configuração do OpenSearch coletor corretamente para poder updates/deletes sincronizar com OpenSearch.
- Referências de chave estrangeira com ações de exclusão em cascata não são suportadas e podem resultar em inconsistência de dados entre o RDS for MySQL e. OpenSearch
- Clusters de banco de dados de zona de multidisponibilidade do Amazon RDS não são compatíveis.
- Versões compatíveis: Aurora MySQL versão 8.0 e acima.

CloudWatch Alarmes recomendados

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo, e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

Métrica	Description
pipeline-name .RDS. Credenciais alteradas	Essa métrica indica com que frequência AWS os segredos são alternados.
pipeline-name .rds. executorRefreshErrors	Essa métrica indica falhas na atualização de AWS segredos.

Métrica	Description
pipeline-name .rds.exportRecordsTotal	Essa métrica indica o número de registros exportados do Amazon Aurora.
pipeline-name .rds.exportRecordsProcessed	Essa métrica indica o número de registros processados pelo pipeline OpenSearch de ingestão.
pipeline-name .rds.exportRecordProcessingErros	Essa métrica indica o número de erros de processamento em um pipeline OpenSearch de ingestão durante a leitura dos dados de um cluster do Amazon Aurora.
pipeline-name .rds.exportRecordsSuccessTotal	Essa métrica indica o número total de registros de exportação processados com êxito.
pipeline-name .rds.exportRecordsFailedTotal	Essa métrica indica o número total de registros de exportação com falha no processamento.
pipeline-name .rds.bytesrecebidos	Essa métrica indica o número total de bytes recebidos por um pipeline OpenSearch de ingestão.
pipeline-name .rds.Bytesprocessados	Essa métrica indica o número total de bytes processados por um pipeline OpenSearch de ingestão.
pipeline-name .rds.streamRecordsSuccessTotal	Essa métrica indica o número de registros processados com êxito a partir do fluxo.

Métrica	Description
pipeline-name .rds.streamRecordsFailedTotal	Essa métrica indica o número total de registros com falha no processamento do fluxo.

RDS para PostgreSQL

Conclua as etapas a seguir para configurar um pipeline OpenSearch de ingestão com o Amazon RDS for RDS for PostgreSQL.

Tópicos

- Pré-requisitos do RDS para PostgreSQL
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline
- Consistência de dados
- Mapear tipo de dados
- Limitações
- CloudWatch Alarmes recomendados

Pré-requisitos do RDS para PostgreSQL

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um grupo personalizado de parâmetros de banco de dados no Amazon RDS para configurar a replicação lógica.

   ```
   rds.logical_replication=1
   ```

   Para saber mais, consulte Executar replicação lógica para o Amazon RDS para PostgreSQL.
2. Selecione ou crie uma instância do banco de dados do RDS para PostgreSQL e associe o grupo de parâmetros criado na etapa 1 à instância do banco de dados.

3. Configure a autenticação de nome de usuário e senha na instância do Amazon RDS usando o gerenciamento de senhas com o Aurora e o AWS Secrets Manager. Você também pode criar uma username/password combinação criando um segredo do Secrets Manager.
4. Se você usar o recurso de snapshot inicial completo, crie uma função AWS KMS key e uma do IAM para exportar dados do Amazon RDS para o Amazon S3.

   O perfil do IAM também deve ter a seguinte política de permissão:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket-used-in-pipeline",
                "arn:aws:s3:::s3-bucket-used-in-pipeline/*"
            ]
        }
    ]
}
```

   O perfil também deve ter a seguinte relação de confiança:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "export.rds.amazonaws.com"
```

```
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

5. Selecione ou crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.
6. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados da sua instância de banco de dados Amazon RDS em seu domínio ou coleção.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos de pipeline do Amazon RDS, configure o perfil do pipeline para usar na configuração do pipeline. Adicione também as seguintes permissões para a origem do Amazon RDS ao perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
    "Sid": "allowReadingFromS3Buckets",
    "Effect": "Allow",
    "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:PutObject"
    ],
    "Resource": [
    "arn:aws:s3:::s3_bucket",
    "arn:aws:s3:::s3_bucket/*"
    ]
    },
    {
    "Sid": "allowNetworkInterfacesActions",
```

```
    "Effect": "Allow",
    "Action": [
    "ec2:AttachNetworkInterface",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DetachNetworkInterface",
    "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": [
    "arn:aws:ec2:*:111122223333:network-interface/*",
    "arn:aws:ec2:*:111122223333:subnet/*",
    "arn:aws:ec2:*:111122223333:security-group/*"
    ]
    },
    {
    "Sid": "allowDescribeEC2",
    "Effect": "Allow",
    "Action": [
    "ec2:Describe*"
    ],
    "Resource": "*"
    },
    {
    "Sid": "allowTagCreation",
    "Effect": "Allow",
    "Action": [
    "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:111122223333:network-interface/*",
    "Condition": {
    "StringEquals": {
    "aws:RequestTag/OSISManaged": "true"
    }
    }
    },
    {
    "Sid": "AllowDescribeInstances",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBInstances"
    ],
    "Resource": [
```

```
    "arn:aws:rds:us-east-2:111122223333:db:*"
    ]
    },
    {
    "Sid": "AllowSnapshots",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeDBSnapshots",
    "rds:CreateDBSnapshot",
    "rds:AddTagsToResource"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:db:DB-id",
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowExport",
    "Effect": "Allow",
    "Action": [
    "rds:StartExportTask"
    ],
    "Resource": [
    "arn:aws:rds:us-east-2:111122223333:snapshot:DB-id*"
    ]
    },
    {
    "Sid": "AllowDescribeExports",
    "Effect": "Allow",
    "Action": [
    "rds:DescribeExportTasks"
    ],
    "Resource": "*",
    "Condition": {
    "StringEquals": {
    "aws:RequestedRegion": "us-east-2",
    "aws:ResourceAccount": "111122223333"
    }
    }
    },
    {
    "Sid": "AllowAccessToKmsForExport",
    "Effect": "Allow",
    "Action": [
```

```
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:DescribeKey",
    "kms:RetireGrant",
    "kms:CreateGrant",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
    ],
    "Resource": [
    "arn:aws:kms:us-east-2:111122223333:key/export-key-id"
    ]
    },
    {
    "Sid": "AllowPassingExportRole",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
    "arn:aws:iam::111122223333:role/export-role"
    ]
    },
    {
    "Sid": "SecretsManagerReadAccess",
    "Effect": "Allow",
    "Action": [
    "secretsmanager:GetSecretValue"
    ],
    "Resource": [
    "arn:aws:secretsmanager:*:111122223333:secret:*"
    ]
    }
    ]
    }
```

Etapa 2: Criar o pipeline

Configure um pipeline de OpenSearch ingestão como o seguinte, que especifica uma instância do RDS para PostgreSQL como origem.

```
version: "2"
rds-postgres-pipeline:
  source:
    rds:
```

```
      db_identifier: "instance-id"
      engine: postgresql
      database: "database-name"
      tables:
        include:
          - "schema1.table1"
          - "schema2.table2"
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "prefix-name"
      export:
        kms_key_id: "kms-key-id"
        iam_role_arn: "export-role-arn"
      stream: true
      aws:
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        region: "us-east-1"
      authentication:
        username: ${{aws_secrets:secret:username}}
        password: ${{aws_secrets:secret:password}}
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
        index: "${getMetadata(\"table_name\")}"
        index_type: custom
        document_id: "${getMetadata(\"primary_key\")}"
        action: "${getMetadata(\"opensearch_action\")}"
        document_version: "${getMetadata(\"document_version\")}"
        document_version_type: "external"
        aws:
          sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "rds-secret-id"
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::account-id:role/pipeline-role"
        refresh_interval: PT1H
```

> **Note**
>
> Você pode usar um esquema pré-configurado do Amazon RDS para criar esse pipeline. Para saber mais, consulte Trabalhar com esquemas.

Para usar o Amazon Aurora como origem, você precisa configurar o acesso à VPC para o pipeline. A VPC escolhida deve ser a mesma VPC que a origem do Amazon Aurora usa. Depois, selecione uma ou mais sub-redes e um ou mais grupos de segurança da VPC. Observe que o pipeline precisa de acesso de rede a um banco de dados MySQL do Aurora, por isso, você também deve se certificar de que o cluster do Aurora esteja configurado com um grupo de segurança da VPC que permita tráfego de entrada do grupo de segurança da VPC do pipeline para a porta do banco de dados. Para saber mais, consulte Controlar o acesso com grupos de segurança.

Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline à sua VPC para usar o Amazon Aurora como fonte. Para fazer isso, encontre a seção Configuração de rede, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.

Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre a OpenSearch ingestão e o Amazon RDS, certifique-se de que o CIDR do Amazon Aurora VPC seja diferente do CIDR para ingestão. OpenSearch

Para saber mais, consulte Configurar o acesso à VPC para um pipeline.

Consistência de dados

O pipeline garante a consistência dos dados pesquisando continuamente ou recebendo alterações da instância do Amazon RDS e atualizando os documentos correspondentes no OpenSearch índice.

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando um pipeline lê snapshots ou fluxos, ele cria partições dinamicamente para processamento paralelo. O pipeline marca uma partição como concluída quando ela recebe uma confirmação após a ingestão de todos os registros no OpenSearch domínio ou na coleção. Se quiser fazer a ingestão em uma coleção de pesquisa OpenSearch sem servidor, você pode gerar uma ID de documento no pipeline. Se você quiser fazer a ingestão em uma coleção de séries temporais OpenSearch sem servidor, observe que o pipeline não gera uma ID de documento, portanto, você

deve omiti-lo `document_id: "${getMetadata(\"primary_key\")}"` na configuração do coletor do pipeline.

Um pipeline OpenSearch de ingestão também mapeia as ações de eventos recebidos em ações de indexação em massa correspondentes para ajudar a ingerir documentos. Isso mantém os dados consistentes, de modo que cada alteração de dados no Amazon RDS seja reconciliada com as alterações correspondentes no documento. OpenSearch

Mapear tipo de dados

OpenSearch O pipeline de ingestão mapeia os tipos de dados do PostgreSQL em representações que são adequadas OpenSearch para o consumo de domínios ou coleções de serviços. Se nenhum modelo de mapeamento estiver definido em OpenSearch, determine OpenSearch automaticamente os tipos de campo com um mapeamento dinâmico baseado no primeiro documento enviado. Você também pode definir explicitamente os tipos de campo que funcionam melhor para você por OpenSearch meio de um modelo de mapeamento.

A tabela abaixo lista os tipos de dados do RDS para PostgreSQL e os tipos de campo correspondentes. OpenSearch A coluna Tipo de OpenSearch campo padrão mostra o tipo de campo correspondente OpenSearch se nenhum mapeamento explícito for definido. Nesse caso, determina OpenSearch automaticamente os tipos de campo com mapeamento dinâmico. A coluna Tipo de OpenSearch campo recomendado é o tipo de campo recomendado correspondente a ser especificado explicitamente em um modelo de mapeamento. Esses tipos de campo estão mais alinhados com os tipos de dados no RDS para PostgreSQL e geralmente podem permitir melhores recursos de pesquisa disponíveis no. OpenSearch

Tipo de dado do RDS para PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
smallint	longo	curto
integer	longo	integer
bigint	longo	longo
decimal	texto	double ou keyword

Tipo de dado do RDS para PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
numeric[(p, s)]	texto	double ou keyword
real	flutuação	flutuação
double precision	flutuação	double
smallserial	longo	curto
serial	longo	integer
bigserial	longo	longo
money	objeto	objeto
caractere variável(n)	texto	texto
varchar(n)	texto	texto
character(n)	texto	texto
char(n)	texto	texto
bpchar(n)	texto	texto
bpchar	texto	texto
texto	texto	texto

Tipo de dado do RDS para PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
enum	texto	texto
bytea	texto	binary
timestamp [(p)] [sem fuso horário]	long (em milissegundos epoch)	date
timestamp [(p)] com fuso horário	long (em milissegundos epoch)	date
date	long (em milissegundos epoch)	date
hora [(p)] [sem fuso horário]	long (em milissegundos epoch)	date
hora [(p)] com fuso horário	long (em milissegundos epoch)	date
interval [fields] [(p)]	texto (ISO8601 formato)	texto
booleano	booleano	booleano

Tipo de dado do RDS para PostgreSQL	Tipo de OpenSearch campo padrão	Tipo de OpenSearch campo recomendado
point	text (em formato WKT)	geo_shape
linha	text (em formato WKT)	geo_shape
lseg	text (em formato WKT)	geo_shape
box	text (em formato WKT)	geo_shape
caminho	text (em formato WKT)	geo_shape
polígono	text (em formato WKT)	geo_shape
circle	objeto	objeto
cidr	texto	texto
inet	texto	texto
macaddr	texto	texto
macaddr8	texto	texto
bit(n)	longo	byte, short, integer, ou long (dependendo do número de bits)
bit variável (n)	longo	byte, short, integer, ou long (dependendo do número de bits)
json	objeto	objeto
jsonb	objeto	objeto
jsonpath	texto	texto

Recomendamos que você configure a fila de mensagens mortas (DLQ) em seu pipeline de ingestão. OpenSearch Se você configurou a fila, o OpenSearch Service envia todos os documentos com falha que não podem ser ingeridos devido a falhas de mapeamento dinâmico para a fila.

Se os mapeamentos automáticos falharem, será possível usar `template_type` e `template_content` na configuração do pipeline para definir regras de mapeamento explícitas. Como alternativa, é possível criar modelos de mapeamento diretamente no seu domínio de pesquisa ou na sua coleção antes de iniciar o pipeline.

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para RDS para PostgreSQL:

- A integração é compatível com apenas um banco de dados PostgreSQL por pipeline.
- Atualmente, a integração não oferece suporte à ingestão de dados entre regiões; sua instância e OpenSearch domínio do Amazon RDS devem estar nos mesmos. Região da AWS
- Atualmente, a integração não oferece suporte à ingestão de dados entre contas; sua instância do Amazon RDS e seu pipeline de OpenSearch ingestão devem estar no mesmo nível. Conta da AWS
- Certifique-se de que a instância do Amazon RDS tenha a autenticação habilitada usando AWS Secrets Manager, que é o único mecanismo de autenticação compatível.
- A configuração existente do pipeline não pode ser atualizada para ingerir dados de um banco de dados diferente ou de and/or uma tabela diferente. Para atualizar o nome do banco de dados e/ou tabela de um pipeline, você deve parar o pipeline e reiniciá-lo com uma configuração atualizada ou criar um novo pipeline.
- Instruções em Data Definition Language (DDL) não são compatíveis em geral. A consistência de dados não será mantida se:
  - As chaves primárias são alteradas (add/delete/rename).
  - As tabelas forem eliminadas/truncadas.
  - Os nomes das colunas ou os tipos de dados forem alterados.
- Se as tabelas PostgreSQL a serem sincronizadas não tiverem chaves primárias definidas, a consistência de dados não será garantida. Você precisará definir a `document_id` opção personalizada OpenSearch e a configuração do coletor corretamente para poder updates/deletes sincronizar com OpenSearch.
- Clusters do banco de dados multi-AZ do RDS não são compatíveis.

- Versões compatíveis: PostgreSQL 16 e acima.

CloudWatch Alarmes recomendados

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo, e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

Métrica	Description
pipeline-name .RDS. Credenciais alteradas	Essa métrica indica com que frequência AWS os segredos são alternados.
pipeline-name .rds. executorRefreshErrors	Essa métrica indica falhas na atualização de AWS segredos.
pipeline-name .rds. exportRecordsTotal	Essa métrica indica o número de registros exportados do Amazon Aurora.
pipeline-name .rds. exportRecordsProcessed	Essa métrica indica o número de registros processados pelo pipeline OpenSearch de ingestão.
pipeline-name .rds. exportRecordProcessingErros	Essa métrica indica o número de erros de processamento em um pipeline OpenSearch de ingestão durante a leitura dos dados de um cluster do Amazon Aurora.

Métrica	Description
pipeline-name .rds.exportRecordsSuccessTotal	Essa métrica indica o número total de registros de exportação processados com êxito.
pipeline-name .rds.exportRecordsFailedTotal	Essa métrica indica o número total de registros de exportação com falha no processamento.
pipeline-name .rds.bytesRecebidos	Essa métrica indica o número total de bytes recebidos por um pipeline OpenSearch de ingestão.
pipeline-name .rds.BytesProcessados	Essa métrica indica o número total de bytes processados por um pipeline OpenSearch de ingestão.
pipeline-name .rds.streamRecordsSuccessTotal	Essa métrica indica o número de registros processados com êxito a partir do fluxo.
pipeline-name .rds.streamRecordsFailedTotal	Essa métrica indica o número total de registros com falha no processamento do fluxo.

Usando um pipeline OpenSearch de ingestão com o Amazon S3

Com OpenSearch a ingestão, você pode usar o Amazon S3 como origem ou destino. Ao usar o Amazon S3 como fonte, você envia dados para um pipeline de OpenSearch ingestão. Ao usar o Amazon S3 como destino, você grava dados de um pipeline de OpenSearch ingestão em um ou mais buckets do S3.

Tópicos

- Amazon S3 como origem
- Amazon S3 como destino
- Amazon S3 entre contas como origem

Amazon S3 como origem

Há duas maneiras de usar o Amazon S3 como fonte para processar dados: com o processamento do S3-SQS e com escaneamentos agendados.

Use o processamento S3-SQS quando precisar escanear arquivos quase em tempo real depois que eles forem gravados no S3. Você pode configurar buckets do Amazon S3 para gerar um evento sempre que um objeto for armazenado ou modificado dentro do bucket. Use uma verificação agendada única ou recorrente para processar dados em lote em um bucket do S3.

Tópicos

- Pré-requisitos
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline

Pré-requisitos

Para usar o Amazon S3 como fonte de um pipeline de OpenSearch ingestão para uma verificação programada ou processamento do S3-SQS, primeiro crie um bucket do S3.

> **Note**
>
> Se o bucket do S3 usado como fonte no pipeline de OpenSearch ingestão estiver em outro Conta da AWS, você também precisará habilitar as permissões de leitura entre contas no bucket. Isso permite que o pipeline leia e processe os dados. Para habilitar permissões entre contas, consulte Bucket owner granting cross-account bucket permissions (Conceder permissões de bucket entre contas como proprietário do bucket) no Guia do usuário do Amazon S3.
>
> Se seus buckets do S3 estiverem em várias contas, use um mapa `bucket_owners`. Para ver um exemplo, consulte Acesso ao S3 entre contas na OpenSearch documentação.

Para configurar o processamento do S3-SQS, você também precisa executar as seguintes etapas:

1. Como criar uma fila do Amazon SQS.
2. Ative as notificações de eventos no bucket do S3 com a fila SQS como destino.

Etapa 1: configurar a função do pipeline

Ao contrário de outros plug-ins de origem que enviam dados para um pipeline, o plug-in de origem do S3 tem uma arquitetura baseada em leitura na qual o pipeline extrai dados da fonte.

Portanto, para que um pipeline seja lido do S3, você deve especificar uma função na configuração de origem do S3 do pipeline que tenha acesso ao bucket do S3 e à fila do Amazon SQS. O pipeline assumirá essa função para ler os dados da fila.

> **Note**
>
> A função que você especifica na configuração de origem do S3 deve ser a função do pipeline. Portanto, sua função de pipeline deve conter duas políticas de permissões separadas: uma para gravar em um coletor e outra para extrair da origem do S3. Você deve usar o mesmo `sts_role_arn` em todos os componentes do pipeline.

O exemplo de política a seguir mostra as permissões necessárias para usar o S3 como fonte:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action":[
          "s3:ListBucket",
          "s3:GetBucketLocation",
          "s3:GetObject"
       ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
    },
    {
       "Effect":"Allow",
       "Action":"s3:ListAllMyBuckets",
       "Resource":"arn:aws:s3:::*"
```

```
    },
    {
      "Effect": "Allow",
      "Action": [
        "sqs:DeleteMessage",
        "sqs:ReceiveMessage",
        "sqs:ChangeMessageVisibility"
      ],
      "Resource": "arn:aws:sqs:us-east-1:111122223333:MyS3EventSqsQueue"
    }
  ]
}
```

Você deve anexar essas permissões ao perfil do IAM especificado na opção `sts_role_arn` na configuração do plug-in de origem do S3:

```
version: "2"
source:
  s3:
    ...
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Etapa 2: Criar o pipeline

Depois de configurar suas permissões, você pode configurar um pipeline de OpenSearch ingestão, dependendo do seu caso de uso do Amazon S3.

Processamento do S3-SQS

Para configurar o processamento do S3-SQS, configure seu pipeline para especificar o S3 como origem e configure as notificações do Amazon SQS:

```
version: "2"
s3-pipeline:
  source:
```

```
    s3:
      notification_type: "sqs"
      codec:
        newline: null
      sqs:
        queue_url: "https://sqs.us-east-1amazonaws.com/account-id/ingestion-queue"
      compression: "none"
      aws:
        region: "region"
  processor:
  - grok:
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index-name"
      aws:
        region: "region"
```

Se você observar uma baixa utilização da CPU ao processar arquivos pequenos no Amazon S3, considere aumentar o throughput modificando o valor da opção `workers`. Para saber mais, consulte as opções de configuração do plug-in do S3.

Varredura agendada

Para configurar uma verificação agendada, configure seu pipeline com uma programação no nível da verificação que se aplique a todos os seus buckets do S3 ou no nível de bucket. Uma programação em nível de bucket ou uma configuração de intervalo de escaneamento sempre substitui uma configuração em nível de escaneamento.

Você pode configurar escaneamentos agendados com um escaneamento único, que é ideal para migração de dados, ou um escaneamento recorrente, que é ideal para processamento em lote.

Para configurar seu pipeline para ler a partir Amazon S3, use os esquemas pré-configurados do Amazon S3. Você pode editar a parte da `scan` da configuração do seu pipeline para atender às suas necessidades de agendamento. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Digitalização única

Uma varredura agendada única é executada uma vez. Na configuração do pipeline, você pode usar `start_time` e `end_time` para especificar quando deseja que os objetos no bucket sejam verificados. Como alternativa, você pode usar `range` para especificar o intervalo de tempo em relação ao horário atual em que você deseja que os objetos no bucket sejam digitalizados.

Por exemplo, um intervalo definido para PT4H verificar todos os arquivos criados nas últimas quatro horas. Para configurar uma varredura única para ser executada pela segunda vez, você deve parar e reiniciar o pipeline. Se você não tiver um intervalo configurado, também deverá atualizar os horários de início e término.

A configuração a seguir configura uma varredura única para todos os buckets e todos os objetos nesses buckets:

```
version: "2"
log-pipeline:
  source:
    s3:
      codec:
        csv:
      compression: "none"
      aws:
        region: "region"
      acknowledgments: true
      scan:
        buckets:
          - bucket:
              name: my-bucket
              filter:
                include_prefix:
                  - Objects1/
                exclude_suffix:
                  - .jpeg
                  - .png
          - bucket:
              name: my-bucket-2
              key_prefix:
                include:
                  - Objects2/
                exclude_suffix:
                  - .jpeg
```

```
                  - .png
      delete_s3_objects_on_read: false
  processor:
    - date:
        destination: "@timestamp"
        from_time_received: true
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index-name"
        aws:
          region: "region"
        dlq:
          s3:
            bucket: "dlq-bucket"
            region: "us-east-1"
```

A configuração a seguir configura uma varredura única para todos os buckets durante uma janela de tempo especificada. Isso significa que o S3 processa somente os objetos com horários de criação que se enquadram nessa janela.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

A configuração a seguir configura uma verificação única no nível de escaneamento e no nível do bucket. Os horários de início e término no nível do bucket substituem os horários de início e término no nível do escaneamento.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

A interrupção de um pipeline remove qualquer referência pré-existente de quais objetos foram verificados pelo pipeline antes da parada. Se uma única verificação de pipeline for interrompida, ele verificará novamente todos os objetos após seu início, mesmo que eles já tenham sido verificados. Se você precisar interromper uma única verificação de pipeline, é recomendável alterar sua janela de tempo antes de iniciar o pipeline novamente.

Se você precisar filtrar objetos por hora de início e hora de término, parar e iniciar seu pipeline é a única opção. Se você não precisar filtrar por hora de início e hora de término, poderá filtrar objetos por nome. Filtrar por nome não exige que você pare e inicie seu pipeline. Para fazer isso, use `include_prefix` e `exclude_suffix`.

Escaneamento recorrente

Uma verificação agendada recorrente executa uma varredura de seus buckets S3 especificados em intervalos regulares e agendados. Você só pode configurar esses intervalos no nível de escaneamento porque não há suporte para configurações individuais em nível de bucket.

Na configuração do pipeline, o `interval` especifica a frequência da verificação recorrente que pode ser de 30 segundos a 365 dias. A primeira dessas varreduras sempre ocorre quando você cria o pipeline. `count` define o número total de instâncias de verificação.

A configuração a seguir configura um escaneamento recorrente, com um atraso de 12 horas entre os escaneamentos:

```
scan:
  scheduling:
    interval: PT12H
    count: 4
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

Amazon S3 como destino

Para gravar dados de um pipeline de OpenSearch ingestão em um bucket do S3, use o blueprint pré-configurado do S3 para criar um pipeline com um coletor do S3. Esse pipeline direciona dados seletivos para um OpenSearch coletor e envia simultaneamente todos os dados para arquivamento no S3. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Ao criar seu coletor S3, você pode especificar sua formatação preferida a partir de uma variedade de codecs de coletor. Por exemplo, se você quiser gravar dados em formato de coluna, escolha o codec

Parquet ou Avro. Se você preferir um formato baseado em linhas, escolha JSON ou NDJSON. Para gravar dados no S3 em um esquema especificado, você também pode definir um esquema embutido nos codecs de coletor usando o formato Avro.

O exemplo a seguir define um esquema embutido em um coletor do S3:

```
- s3:
  codec:
    parquet:
      schema: >
        {
           "type" : "record",
           "namespace" : "org.vpcFlowLog.examples",
           "name" : "VpcFlowLog",
           "fields" : [
             { "name" : "version", "type" : "string"},
             { "name" : "srcport", "type": "int"},
             { "name" : "dstport", "type": "int"},
             { "name" : "start", "type": "int"},
             { "name" : "end", "type": "int"},
             { "name" : "protocol", "type": "int"},
             { "name" : "packets", "type": "int"},
             { "name" : "bytes", "type": "int"},
             { "name" : "action", "type": "string"},
             { "name" : "logStatus", "type" : "string"}
           ]
         }
```

Ao definir esse esquema, especifique um superconjunto de todas as chaves que podem estar presentes nos diferentes tipos de eventos que seu pipeline entrega a um coletor.

Por exemplo, se um evento tiver a possibilidade de uma chave faltar, adicione essa chave em seu esquema com um valor `null`. Declarações de valor nulo permitem que o esquema processe dados não uniformes (onde alguns eventos têm essas chaves e outros não). Quando os eventos recebidos têm essas chaves presentes, seus valores são gravados em coletores.

Essa definição de esquema atua como um filtro que só permite que chaves definidas sejam enviadas aos coletores e elimina chaves indefinidas dos eventos recebidos.

Você também pode usar `include_keys` e `exclude_keys` no seu coletor para filtrar dados que são roteados para outros coletores. Esses dois filtros são mutuamente exclusivos, então você só pode

usar um por vez em seu esquema. Além disso, não é possível usá-los em esquemas definidos pelo usuário.

Para criar pipelines com esses filtros, use o esquema do filtro de coletor pré-configurado. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Amazon S3 entre contas como origem

Você pode conceder acesso a várias contas com o Amazon S3 para que os pipelines de OpenSearch ingestão possam acessar buckets do S3 em outra conta como fonte. Para habilitar o acesso entre contas, consulte Bucket owner granting cross-account bucket permissions (Conceder permissões de bucket entre contas como proprietário do bucket) no Guia do usuário do Amazon S3. Depois de conceder acesso, certifique-se de que seu perfil no pipeline tenha as permissões necessárias.

Depois, você pode criar um pipeline usando `bucket_owners` para habilitar acesso entre contas a um bucket do Amazon S3 como origem:

```
s3-pipeline:
 source:
  s3:
   notification_type: "sqs"
   codec:
    csv:
     delimiter: ","
     quote_character: "\""
     detect_header: True
   sqs:
    queue_url: "https://sqs.ap-northeast-1.amazonaws.com/401447383613/test-s3-queue"
   bucket_owners:
    my-bucket-01: 123456789012
    my-bucket-02: 999999999999
   compression: "gzip"
```

Usando um pipeline OpenSearch de ingestão com o Amazon Security Lake

Você pode usar o plug-in de origem do S3 para ingerir dados do Amazon Security Lake em seu pipeline de OpenSearch ingestão. O Security Lake centraliza automaticamente os dados de segurança de AWS ambientes, ambientes locais e provedores de SaaS em um data lake específico. Você pode criar uma assinatura que replica os dados do Security Lake para o pipeline de

OpenSearch ingestão e, em seguida, os grava no domínio do OpenSearch Service ou na coleção OpenSearch Serverless.

Para configurar seu pipeline para ler a partir do Security Lake, use o esquema pré-configurado do Security Lake. O esquema inclui uma configuração padrão para ingerir arquivos de parquet do Open Cybersecurity Schema Framework (OCSF) do Security Lake. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Tópicos

- Usando um pipeline OpenSearch de ingestão com o Amazon Security Lake como fonte
- Usando um pipeline OpenSearch de ingestão com o Amazon Security Lake como coletor

Usando um pipeline OpenSearch de ingestão com o Amazon Security Lake como fonte

Você pode usar o plug-in de origem do Amazon S3 em seu pipeline de OpenSearch ingestão para ingerir dados do Amazon Security Lake. O Security Lake centraliza automaticamente os dados de segurança de ambientes da AWS , sistemas on-premises e provedores de SaaS em um data lake específico.

O Amazon Security Lake tem os seguintes atributos de metadados em um pipeline:

- `bucket_name`: o nome do bucket do Amazon S3 criado pelo Security Lake para armazenar dados de segurança.
- `path_prefix`: o nome da origem personalizada definido na política de perfil do IAM do Security Lake.
- `region`: Região da AWS Onde está localizado o bucket do Security Lake S3.
- `accountID`: O Conta da AWS ID no qual o Security Lake está ativado.
- `sts_role_arn`: o ARN do perfil do IAM a ser usado com o Security Lake.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

- Habilitar o Security Lake.
- Criar um assinante no Security Lake.
  - Escolha as fontes que você deseja ingerir em seu pipeline.

- Para credenciais de assinante, adicione o ID do Conta da AWS local em que você pretende criar o pipeline. Para o ID externo, especifique `OpenSearchIngestion-{accountid}`.
- Em Método de acesso a dados, escolha S3.
- Para Detalhes de notificação, escolha SQS queue.

Quando você cria um assinante, o Security Lake cria automaticamente duas políticas de permissões em linha: uma para o S3 e outra para SQS. As políticas têm o seguinte formato: `AmazonSecurityLake-amzn-s3-demo-bucket-S3` e `AmazonSecurityLake-AWS Demo-SQS`. Para permitir que seu pipeline acesse as origens de assinantes, você deve associar as permissões necessárias à sua função do pipeline.

Configurar o perfil do pipeline

Crie uma nova política de permissões no IAM que combine somente as permissões necessárias das duas políticas que o Security Lake criou automaticamente. O exemplo de política a seguir mostra o menor privilégio necessário para que um pipeline de OpenSearch ingestão leia dados de várias fontes do Security Lake:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "s3:GetObject"
         ],
         "Resource":[
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
LAMBDA_EXECUTION/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/S3_DATA/1.0/
*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
VPC_FLOW/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/ROUTE53/1.0/
*",
            "arn:aws:s3:::aws-security-data-lake-us-east-1-abcde/aws/
SH_FINDINGS/1.0/*"
         ]
```

```
      },
      {
         "Effect":"Allow",
         "Action":[
            "sqs:ReceiveMessage",
            "sqs:DeleteMessage"
         ],
         "Resource":[
            "arn:aws:sqs:us-east-1:111122223333:AmazonSecurityLake-abcde-Main-
Queue"
         ]
      }
   ]
}
```

> **Important**
>
> O Security Lake não gerencia a política de função do pipeline para você. Se você adicionar ou remover fontes da sua assinatura do Security Lake, deverá atualizar a política manualmente. O Security Lake cria partições para cada fonte de log, então você precisa adicionar ou remover manualmente as permissões na função de pipeline.

Você deve anexar essas permissões ao perfil do IAM que você especifica na opção `sts_role_arn` na configuração do plug-in de origem do S3, em `sqs`.

```
version: "2"
source:
  s3:
    ...
    sqs:
      queue_url: "https://sqs.us-east-1amazonaws.com/account-id/
AmazonSecurityLake-amzn-s3-demo-bucket-Main-Queue"
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Criar o pipeline

Depois de adicionar as permissões ao perfil do pipeline, use o esquema pré-configurado do Security Lake para criar o pipeline. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Você deve especificar a opção `queue_url` na configuração de origem do `s3`, que é o URL da fila do Amazon SQS para leitura. Para formatar o URL, localize o endpoint da assinatura na configuração do assinante e altere `arn:aws:` para `https://`. Por exemplo, .`https://sqs.`*`us-east-1`*`amazonaws.com/`*`account-id`*`/AmazonSecurityLake-`*`AWS Demo`*`-Main-Queue`

O `sts_role_arn` que você especifica na configuração de origem do S3 deve ser o ARN da função do pipeline.

Usando um pipeline OpenSearch de ingestão com o Amazon Security Lake como coletor

Use o plug-in de coletor do Amazon S3 na OpenSearch Ingestão para enviar dados de qualquer fonte compatível para o Amazon Security Lake. O Security Lake coleta e armazena dados de segurança de AWS ambientes locais e provedores de SaaS em um data lake dedicado.

Para configurar o pipeline para gravar dados de log no Security Lake, use o esquema pré-configurado de Logs de tráfego de firewall. O esquema inclui uma configuração padrão para recuperar logs de segurança brutos ou outros dados armazenados em um bucket do Amazon S3, processar os registros e normalizá-los. Em seguida, mapeia os dados para o Open Cybersecurity Schema Framework (OCSF) e envia os dados conformes com OCSF transformados para o Security Lake.

O pipeline tem os seguintes atributos de metadados:

- `bucket_name`: o nome do bucket do Amazon S3 criado pelo Security Lake para armazenar dados de segurança.
- `path_prefix`: o nome da origem personalizada definido na política de perfil do IAM do Security Lake.
- `region`: Região da AWS Onde está localizado o bucket do Security Lake S3.
- `accountID`: O Conta da AWS ID no qual o Security Lake está ativado.
- `sts_role_arn`: o ARN do perfil do IAM a ser usado com o Security Lake.

Pré-requisitos

Antes de criar um pipeline para enviar dados para o Security Lake, realize as seguintes etapas:

- Habilitar e configurar o Amazon Security Lake: configure o Amazon Security Lake para centralizar dados de segurança de várias origens. Para obter instruções, consulte Habilitar o Security Lake usando o console.

  Ao selecionar uma origem, escolha Ingerir origens específicas da AWS e selecione uma ou mais origens de log e eventos que você deseja ingerir.
- Configurar permissões: configure o perfil do pipeline com as permissões necessárias para gravar dados no Security Lake. Para saber mais, consulte Perfil do pipeline.

Criar o pipeline

Use o esquema pré-configurado do Security Lake para criar o pipeline. Para saber mais, consulte Usar ewsquemas para criar um pipeline.

Usando um pipeline de OpenSearch ingestão com o FluentBit

Esse exemplo de arquivo de configuração do Fluent Bit envia dados de log do Fluent Bit para um pipeline de OpenSearch ingestão. Para saber mais sobre a ingestão de dados de log, consulte Log Analytics na documentação do Data Prepper.

Observe o seguinte:

- O valor `host` deve ser o endpoint do seu pipeline. Por exemplo, *`.pipeline-endpoint.us-east-1`*`osis.amazonaws.com`
- O valor de `aws_service` deve ser `osis`.
- O `aws_role_arn` valor é o ARN da função do AWS IAM que o cliente deve assumir e usar para a autenticação Signature versão 4.

```
[INPUT]
  name                  tail
  refresh_interval      5
  path                  /var/log/test.log
  read_from_head        true

[OUTPUT]
```

```
  Name http
  Match *
  Host pipeline-endpoint.us-east-1osis.amazonaws.com
  Port 443
  URI /log/ingest
  Format json
  aws_auth true
  aws_region region
  aws_service osis
  aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  Log_Level trace
  tls On
```

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que tem HTTP como origem:

```
version: "2"
unaggregated-log-pipeline:
  source:
    http:
      path: "/log/ingest"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
    - grok:
        match:
          details:
            - "'%{NOTSPACE:http_method} %{NOTSPACE:http_uri}' %{NOTSPACE:protocol}"
            - "TLS%{NOTSPACE:tls_version} %{GREEDYDATA:encryption}"
            - "%{NUMBER:status_code:int} %{NUMBER:response_size:int}"
    - delete_entries:
        with_keys: ["details", "log"]

  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index_name"
        index_type: custom
        bulk_size: 20
```

```
        aws:
          region: "region"
```

Usando um pipeline de OpenSearch ingestão com o Fluentd

O Fluentd é um ecossistema de coleta de dados de código aberto que fornece SDKs diferentes linguagens e subprojetos, como o Fluent Bit. Esse exemplo de arquivo de configuração do Fluentd envia dados de log do Fluentd para um pipeline de ingestão. OpenSearch Para saber mais sobre a ingestão de dados de log, consulte Log Analytics na documentação do Data Prepper.

Observe o seguinte:

- O valor `endpoint` deve ser o endpoint do seu pipeline. Por exemplo, *.pipeline-endpoint.us-east-1*`osis.amazonaws.com/apache-log-pipeline/logs`
- O valor de `aws_service` deve ser `osis`.
- O `aws_role_arn` valor é o ARN da função do AWS IAM que o cliente deve assumir e usar para a autenticação Signature versão 4.

```
<source>
  @type tail
  path logs/sample.log
  path_key log
  tag apache
  <parse>
    @type none
  </parse>
</source>

<filter apache>
  @type record_transformer
  <record>
    log ${record["message"]}
  </record>
</filter>

<filter apache>
  @type record_transformer
  remove_keys message
</filter>
```

```
<match apache>
  @type http
  endpoint pipeline-endpoint.us-east-1osis.amazonaws.com/apache-log-pipeline/logs
  json_array true

  <auth>
    method aws_sigv4
    aws_service osis
    aws_region region
    aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  </auth>

  <format>
    @type json
  </format>

  <buffer>
    flush_interval 1s
  </buffer>
</match>
```

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que tem HTTP como origem:

```
version: "2"
apache-log-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index_name"
        aws_region: "region"
        aws_sigv4: true
```

Usando um pipeline OpenSearch de ingestão com inferência de lote offline de aprendizado de máquina

Os pipelines do Amazon OpenSearch Ingestion (OSI) oferecem suporte ao processamento de inferência em lote offline de aprendizado de máquina (ML) para enriquecer com eficiência grandes volumes de dados a baixo custo. Use inferência em lote offline sempre que tiver grandes conjuntos de dados que possam ser processados de modo assíncrono. A inferência de lotes offline funciona com o Amazon Bedrock e SageMaker os modelos. Esse recurso está disponível em todos os domínios Regiões da AWS compatíveis com OpenSearch Ingestão com o OpenSearch Service 2.17+.

> **Note**
>
> Para processamento de inferência em tempo real, use Conectores Amazon OpenSearch Service ML para plataformas de terceiros.

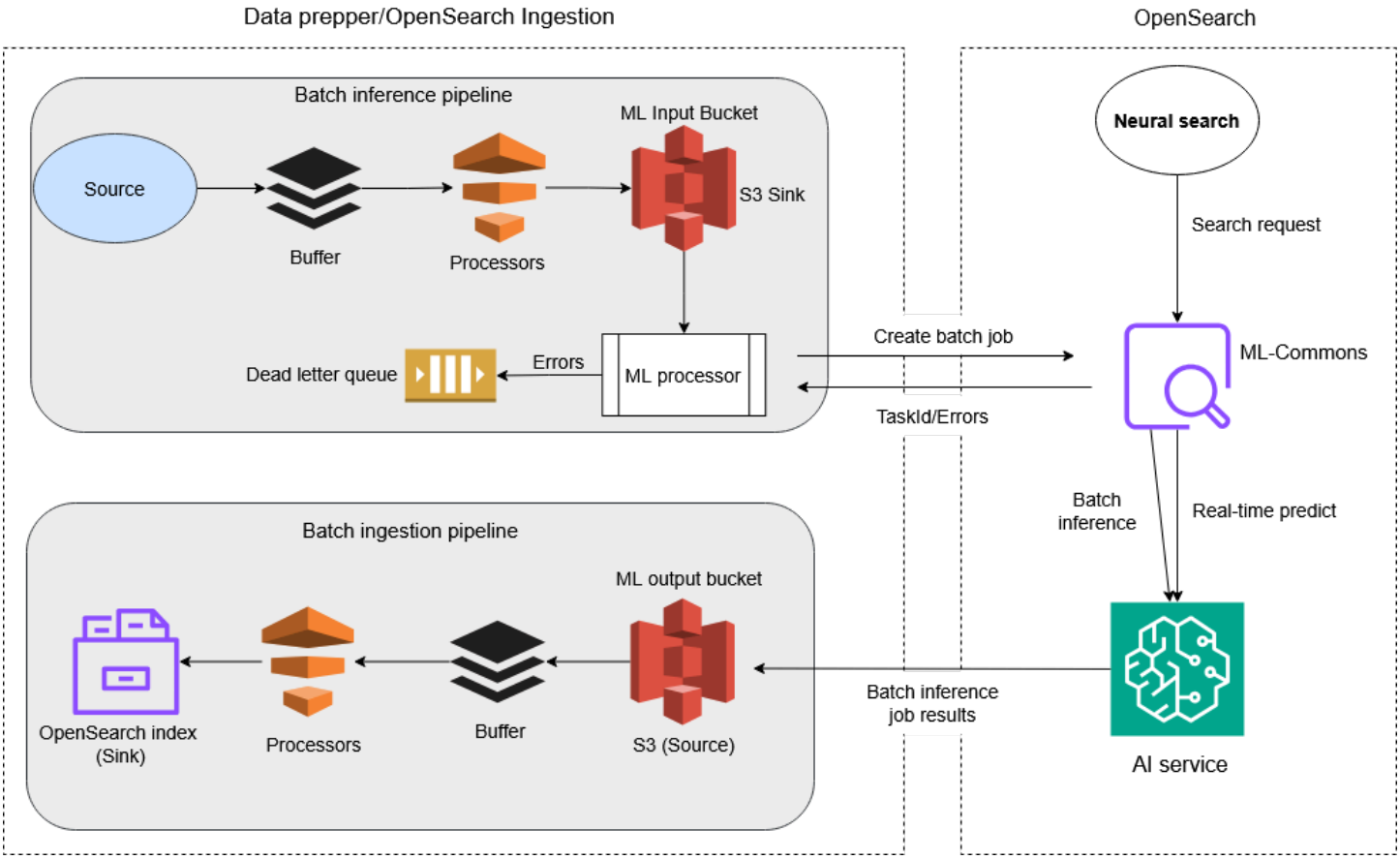
O processamento offline de inferência em lote aproveita um recurso OpenSearch chamado ML Commons. O ML Commons fornece algoritmos de ML por meio de chamadas de API de transporte e REST. Essas chamadas escolhem os nós e os recursos certos para cada solicitação de ML e monitoram as tarefas de ML para garantir o tempo de atividade. Desse modo, o ML Commons permite que você aproveite os algoritmos de ML de código aberto existentes e reduza o esforço necessário para desenvolver novos atributos de ML. Para obter mais informações sobre o ML Commons, consulte Aprendizado de máquina na OpenSearch documentação.org.

Como funciona

Você pode criar um pipeline de inferência em lote offline no OpenSearch Ingestion adicionando um processador de inferência de aprendizado de máquina a um pipeline. Esse processador permite que seu pipeline se conecte a serviços de IA, como SageMaker executar trabalhos de inferência em lote. Você pode configurar o processador para se conectar ao serviço de IA desejado por meio dos conectores de IA (compatíveis com batch_predict) em execução no domínio de destino.

OpenSearch A ingestão usa o `ml_inference` processador com o ML Commons para criar trabalhos de inferência em lote offline. Em seguida, o ML Commons usa a API batch_predict, que realiza inferência em grandes conjuntos de dados em um modo assíncrono offline usando um modelo implantado em servidores de modelos externos no Amazon Bedrock, Amazon, Cohere e OpenAI.

SageMaker O diagrama a seguir mostra um pipeline de OpenSearch ingestão que orquestra vários componentes para realizar esse processo de ponta a ponta:



Os componentes da tubulação funcionam da seguinte maneira:

Pipeline 1 (preparação e transformação de dados)*:

- Fonte: Os dados são escaneados de sua fonte externa suportada pelo OpenSearch Ingestion.
- Processadores de dados: os dados brutos são processados e transformados no formato correto para inferência em lote no serviço de IA integrado.
- S3 (coletor): os dados processados são armazenados em um bucket do Amazon S3 pronto para servir como entrada para a execução de trabalhos de inferência em lote no serviço de IA integrado.

Pipeline 2 (acionar batch_inference de ML):

- Origem: detecção automatizada de eventos do S3 de novos arquivos criados pela saída do pipeline 1.

- Processador de ML_inference: processador que gera inferências de ML por meio de um trabalho em lote assíncrono. Ele se conecta aos serviços de IA por meio do conector de IA configurado que é executado no domínio de destino.
- ID da tarefa: cada trabalho em lotes é associado a um ID de tarefa no ml-commons para rastreamento e gerenciamento.
- OpenSearch ML Commons: O ML Commons, que hospeda o modelo para pesquisa neural em tempo real, gerencia os conectores para servidores remotos de IA e serve APIs para inferência em lote e gerenciamento de tarefas.
- Serviços de IA: o OpenSearch ML Commons interage com serviços de IA como Amazon Bedrock e Amazon SageMaker para realizar inferência em lote nos dados, produzindo previsões ou insights. Os resultados são salvos de modo assíncrono em um arquivo separado do S3.

Pipeline 3 (ingestão em massa):

- S3 (origem): os resultados dos trabalhos em lotes são armazenados no S3, que é a origem desse pipeline.
- Processadores de transformação de dados: processamento e transformação adicionais são aplicados à saída da inferência em lote antes da ingestão. Isso garante que os dados sejam mapeados corretamente no OpenSearch índice.
- OpenSearch índice (Sink): os resultados processados são indexados OpenSearch para armazenamento, pesquisa e análise posterior.

> **Note**
>
> *O processo descrito pelo pipeline 1 é opcional. Se preferir, você pode pular esse processo e simplesmente carregar os dados preparados no coletor do S3 para criar trabalhos em lotes.

Sobre o processador de ml_inference

OpenSearch A ingestão usa uma integração especializada entre a fonte do S3 Scan e o processador de inferência de ML para processamento em lote. O S3 Scan opera no modo de metadados somente para coletar com eficiência informações do arquivo do S3 sem ler o conteúdo real do arquivo. O `ml_inference` processador usa o arquivo S3 URLs para coordenar com o ML Commons o processamento em lote. Esse design otimiza o fluxo de trabalho de inferência em lote, minimizando

a transferência desnecessária de dados durante a fase de verificação. Você define o processador de `ml_inference` usando parâmetros. Exemplo:

```
processor:
    - ml_inference:
        # The endpoint URL of your OpenSearch domain
        host: "https://AWS test-offlinebatch-123456789abcdefg.us-
west-2.es.amazonaws.com"

        # Type of inference operation:
        # - batch_predict: for batch processing
        # - predict: for real-time inference
        action_type: "batch_predict"

        # Remote ML model service provider (Amazon Bedrock or SageMaker)
        service_name: "bedrock"

        # Unique identifier for the ML model
        model_id: "AWS TestModelID123456789abcde"

        # S3 path where batch inference results will be stored
        output_path: "s3://amzn-s3-demo-bucket/"

        # Supports ISO_8601 notation strings like PT20.345S or PT15M
        # These settings control how long to keep your inputs in the processor for
 retry on throttling errors
        retry_time_window: "PT9M"

        # AWS configuration settings
        aws:
            # Região da AWS where the Lambda function is deployed
            region: "us-west-2"
            # IAM role ARN for Lambda function execution
            sts_role_arn: "arn:aws::iam::account_id:role/Admin"

        # Dead-letter queue settings for storing errors
        dlq:
          s3:
            region: us-west-2
            bucket: batch-inference-dlq
            key_path_prefix: bedrock-dlq
            sts_role_arn: arn:aws:iam::account_id:role/OSI-invoke-ml
```

```
        # Conditional expression that determines when to trigger the processor
        # In this case, only process when bucket matches "amzn-s3-demo-bucket"
        ml_when: /bucket == "amzn-s3-demo-bucket"
```

Melhorias na performance de ingestão usando o processador de ml_inference

O `ml_inference` processador OpenSearch de ingestão melhora significativamente o desempenho de ingestão de dados para pesquisas habilitadas para ML. O processador é ideal para casos de uso que exigem dados gerados pelo modelo de machine learning, incluindo pesquisa semântica, pesquisa multimodal, enriquecimento de documentos e compreensão de consultas. Na pesquisa semântica, o processador pode acelerar a criação e a ingestão de vetores de alta dimensão e grande volume por ordem de magnitude.

A capacidade de inferência em lote offline do processador oferece vantagens claras sobre a invocação de modelo em tempo real. Embora o processamento em tempo real exija um servidor de modelo ativo com limitações de capacidade, a inferência em lote escala dinamicamente os recursos de computação sob demanda e processa os dados em paralelo. Por exemplo, quando o pipeline de OpenSearch ingestão recebe um bilhão de solicitações de dados de origem, ele cria 100 arquivos S3 para entrada de inferência em lote de ML. Em seguida, o `ml_inference` processador inicia um trabalho SageMaker em lote usando 100 instâncias do `ml.m4.xlarge` Amazon Elastic Compute Cloud (Amazon EC2), concluindo a vetorização de um bilhão de solicitações em 14 horas — uma tarefa que seria praticamente impossível de realizar em tempo real.

Configurar o processador de ml_inference para ingerir solicitações de dados para uma pesquisa semântica

Os procedimentos a seguir orientam você no processo de instalação e configuração do `ml_inference` processador de OpenSearch ingestão para ingerir um bilhão de solicitações de dados para pesquisa semântica usando um modelo de incorporação de texto.

Tópicos

- Etapa 1: criar conectores e registrar modelos no OpenSearch
- Etapa 2: criar um pipeline OpenSearch de ingestão para inferência de lote offline de ML
- Etapa 3: preparar os dados para Ingestão
- Etapa 4: monitorar o trabalho de inferência em lote
- Etapa 5: executar a pesquisa

Etapa 1: criar conectores e registrar modelos no OpenSearch

Para o procedimento a seguir, use o ML Commons batch_inference_sagemaker_connector_blueprint para criar um conector e um modelo na Amazon. SageMaker Se você preferir usar modelos de OpenSearch CloudFormation integração, consulte (Procedimento alternativo) Etapa 1: criar conectores e modelos usando um modelo de CloudFormation integração mais adiante nesta seção.

Para criar conectores e registrar modelos no OpenSearch

1. Crie um modelo de ML da Deep Java Library (DJL) SageMaker para transformação em lote. Para ver outros modelos de DJL, consulte Semantic_search_with_cfn_template_for_SageMaker em: GitHub

```
POST https://api.sagemaker.us-east-1.amazonaws.com/CreateModel
{
   "ExecutionRoleArn": "arn:aws:iam::123456789012:role/aos_ml_invoke_sagemaker",
   "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
   "PrimaryContainer": {
      "Environment": {
         "SERVING_LOAD_MODELS" : "djl://ai.djl.huggingface.pytorch/sentence-
transformers/all-MiniLM-L6-v2"
      },
      "Image": "763104351884.dkr.ecr.us-east-1.amazonaws.com/djl-inference:0.29.0-
cpu-full"
   }
}
```

2. Crie um conector com `batch_predict` como novo tipo de `action` no campo `actions`:

```
POST /_plugins/_ml/connectors/_create
{
  "name": "DJL Sagemaker Connector: all-MiniLM-L6-v2",
  "version": "1",
  "description": "The connector to sagemaker embedding model all-MiniLM-L6-v2",
  "protocol": "aws_sigv4",
  "credential": {
  "roleArn": "arn:aws:iam::111122223333:role/SageMakerRole"
},
  "parameters": {
    "region": "us-east-1",
    "service_name": "sagemaker",
    "DataProcessing": {
```

```
      "InputFilter": "$.text",
      "JoinSource": "Input",
      "OutputFilter": "$"
    },
    "MaxConcurrentTransforms": 100,
    "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
    "TransformInput": {
      "ContentType": "application/json",
      "DataSource": {
        "S3DataSource": {
          "S3DataType": "S3Prefix",
          "S3Uri": "s3://offlinebatch/msmarcotests/"
        }
      },
      "SplitType": "Line"
    },
    "TransformJobName": "djl-batch-transform-1-billion",
    "TransformOutput": {
      "AssembleWith": "Line",
      "Accept": "application/json",
      "S3OutputPath": "s3://offlinebatch/msmarcotestsoutputs/"
    },
    "TransformResources": {
      "InstanceCount": 100,
      "InstanceType": "ml.m4.xlarge"
    },
    "BatchStrategy": "SingleRecord"
  },
  "actions": [
    {
      "action_type": "predict",
      "method": "POST",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://runtime.sagemaker.us-east-1.amazonaws.com/endpoints/
OpenSearch-sagemaker-060124023703/invocations",
      "request_body": "${parameters.input}",
      "pre_process_function": "connector.pre_process.default.embedding",
      "post_process_function": "connector.post_process.default.embedding"
    },
    {
      "action_type": "batch_predict",
      "method": "POST",
```

```
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/CreateTransformJob",
      "request_body": """{ "BatchStrategy": "${parameters.BatchStrategy}",
 "ModelName": "${parameters.ModelName}", "DataProcessing" :
 ${parameters.DataProcessing}, "MaxConcurrentTransforms":
 ${parameters.MaxConcurrentTransforms}, "TransformInput":
 ${parameters.TransformInput}, "TransformJobName" :
 "${parameters.TransformJobName}", "TransformOutput" :
 ${parameters.TransformOutput}, "TransformResources" :
 ${parameters.TransformResources}}"""
    },
    {
      "action_type": "batch_predict_status",
      "method": "GET",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/DescribeTransformJob",
      "request_body": """{ "TransformJobName" :
 "${parameters.TransformJobName}"}"""
    },
    {
      "action_type": "cancel_batch_predict",
      "method": "POST",
      "headers": {
        "content-type": "application/json"
      },
      "url": "https://api.sagemaker.us-east-1.amazonaws.com/StopTransformJob",
      "request_body": """{ "TransformJobName" :
 "${parameters.TransformJobName}"}"""
    }
  ]
}
```

3. Use o ID do conector retornado para registrar o SageMaker modelo:

```
POST /_plugins/_ml/models/_register
{
    "name": "SageMaker model for batch",
    "function_name": "remote",
    "description": "test model",
    "connector_id": "example123456789-abcde"
```

```
}
```

4. Invoque o modelo com o tipo de ação `batch_predict`:

```
POST /_plugins/_ml/models/teHr3JABBiEvs-eod7sn/_batch_predict
{
  "parameters": {
    "TransformJobName": "SM-offline-batch-transform"
  }
}
```

A resposta contém uma ID de tarefa para o trabalho em lote:

```
{
 "task_id": "exampleIDabdcefd_1234567",
 "status": "CREATED"
}
```

5. Verifique o status do trabalho em lote chamando a API Get Task usando o ID da tarefa:

```
GET /_plugins/_ml/tasks/exampleIDabdcefd_1234567
```

A resposta contém o status da tarefa:

```
{
  "model_id": "nyWbv5EB_tT1A82ZCu-e",
  "task_type": "BATCH_PREDICTION",
  "function_name": "REMOTE",
  "state": "RUNNING",
  "input_type": "REMOTE",
  "worker_node": [
    "WDZnIMcbTrGtnR4Lq9jPDw"
  ],
  "create_time": 1725496527958,
  "last_update_time": 1725496527958,
  "is_async": false,
  "remote_job": {
    "TransformResources": {
      "InstanceCount": 1,
      "InstanceType": "ml.c5.xlarge"
    },
    "ModelName": "DJL-Text-Embedding-Model-imageforjsonlines",
```

```
    "TransformOutput": {
      "Accept": "application/json",
      "AssembleWith": "Line",
      "KmsKeyId": "",
      "S3OutputPath": "s3://offlinebatch/output"
    },
    "CreationTime": 1725496531.935,
    "TransformInput": {
      "CompressionType": "None",
      "ContentType": "application/json",
      "DataSource": {
        "S3DataSource": {
          "S3DataType": "S3Prefix",
          "S3Uri": "s3://offlinebatch/sagemaker_djl_batch_input.json"
        }
      },
      "SplitType": "Line"
    },
    "TransformJobArn": "arn:aws:sagemaker:us-east-1:111122223333:transform-job/SM-
offline-batch-transform15",
    "TransformJobStatus": "InProgress",
    "BatchStrategy": "SingleRecord",
    "TransformJobName": "SM-offline-batch-transform15",
    "DataProcessing": {
      "InputFilter": "$.content",
      "JoinSource": "Input",
      "OutputFilter": "$"
    }
  }
}
```

(Procedimento alternativo) Etapa 1: criar conectores e modelos usando um modelo de CloudFormation integração

Se preferir, você pode usar AWS CloudFormation para criar automaticamente todos os SageMaker conectores e modelos Amazon necessários para inferência de ML. Essa abordagem simplifica a configuração usando um modelo pré-configurado disponível no console do Amazon OpenSearch Service. Para obter mais informações, consulte Usando CloudFormation para configurar a inferência remota para pesquisa semântica.

Para implantar uma CloudFormation pilha que cria todos os SageMaker conectores e modelos necessários

1. Abra o console do Amazon OpenSearch Service.
2. No painel de navegação, selecione Integrações.
3. No campo Pesquisar**`SageMaker`**, insira e escolha Integração com modelos de incorporação de texto por meio da Amazon SageMaker.
4. Escolha Configurar domínio e depois Configurar domínio da VPC ou Configurar domínio público.
5. Insira as informações nos campos do modelo. Em Habilitar inferência em lote offline, escolha verdadeiro para provisionar recursos para processamento em lote offline.
6. Escolha Criar para criar a CloudFormation pilha.
7. Depois que a pilha for criada, abra a guia Saídas e, no console do CloudFormation , localize o connector_id e o model_id. Você precisará desses valores mais tarde quando configurar o pipeline.

Etapa 2: criar um pipeline OpenSearch de ingestão para inferência de lote offline de ML

Use o exemplo a seguir para criar um pipeline OpenSearch de ingestão para inferência de lote offline de ML. Para obter mais informações sobre como criar um pipeline para OpenSearch ingestão, consulteCriação de pipelines OpenSearch de ingestão da Amazon.

Antes de começar

No exemplo a seguir, você especifica um ARN de perfil do IAM para o parâmetro `sts_role_arn`. Use o procedimento a seguir para verificar se essa função está mapeada para a função de back-end que tem acesso ao ml-commons em. OpenSearch

1. Navegue até o plug-in OpenSearch Dashboards do seu domínio OpenSearch de serviço. Você pode encontrar o endpoint do painel no painel do seu domínio no console de OpenSearch serviço.
2. No menu principal, escolha Segurança, Funções e selecione a função ml_full_access.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Em perfis de backend, adicione o ARN do perfil do Lambda que precisa de permissão para chamar o domínio. Aqui está um exemplo: arn:aws:iam: ::role/ *111122223333 lambda-role*
5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Exemplo para criar um pipeline OpenSearch de ingestão para inferência de lote offline de ML

```
version: '2'
extension:
  osis_configuration_metadata:
    builder_type: visual
sagemaker-batch-job-pipeline:
  source:
    s3:
      acknowledgments: true
      delete_s3_objects_on_read: false
      scan:
        buckets:
          - bucket:
              name: name
              data_selection: metadata_only
              filter:
                include_prefix:
                  - sagemaker/sagemaker_djl_batch_input
                exclude_suffix:
                  - .manifest
          - bucket:
              name: name
              data_selection: data_only
              filter:
                include_prefix:
                  - sagemaker/output/
        scheduling:
          interval: PT6M
      aws:
        region: name
      default_bucket_owner: account_ID
      codec:
        ndjson:
          include_empty_objects: false
      compression: none
      workers: '1'
  processor:
    - ml_inference:
        host: "https://search-AWStest-offlinebatch-123456789abcdef.us-
west-2.es.amazonaws.com"
        aws_sigv4: true
        action_type: "batch_predict"
        service_name: "sagemaker"
```

```
        model_id: "model_ID"
        output_path: "s3://AWStest-offlinebatch/sagemaker/output"
        aws:
          region: "us-west-2"
          sts_role_arn: "arn:aws:iam::account_ID:role/Admin"
        ml_when: /bucket == "AWStest-offlinebatch"
        dlq:
          s3:
            region: us-west-2
            bucket: batch-inference-dlq
            key_path_prefix: bedrock-dlq
            sts_role_arn: arn:aws:iam::account_ID:role/OSI-invoke-ml
    - copy_values:
        entries:
          - from_key: /text
            to_key: chapter
          - from_key: /SageMakerOutput
            to_key: chapter_embedding
          - delete_entries:
            with_keys:
          - text
          - SageMakerOutput
  sink:
    - opensearch:
        hosts: ["https://search-AWStest-offlinebatch-123456789abcdef.us-
west-2.es.amazonaws.com"]
        aws:
          serverless: false
          region: us-west-2
        routes:
          - ml-ingest-route
        index_type: custom
        index: test-nlp-index
  routes:
    - ml-ingest-route: /chapter != null and /title != null
```

Etapa 3: preparar os dados para Ingestão

Para preparar seus dados para o processamento de inferência em lote offline de ML, prepare você mesmo os dados usando suas próprias ferramentas ou processos ou use o OpenSearch Data Prepper. Verifique se os dados estão organizados no formato correto usando um pipeline para consumir os dados da fonte de dados ou criando um conjunto de dados de machine learning.

O exemplo a seguir usa o conjunto de dados MS MARCO, que inclui uma coleção de consultas reais de usuários para tarefas de processamento em linguagem natural. O conjunto de dados é estruturado no formato JSONL, em que cada linha representa uma solicitação enviada ao modelo de incorporação de ML:

```
{"_id": "1185869", "text": ")what was the immediate impact of the Paris Peace Treaties
 of 1947?", "metadata": {"world war 2"}}
{"_id": "1185868", "text": "_________ justice is designed to repair the harm to victim,
 the community and the offender caused by the offender criminal act. question 19
 options:", "metadata": {"law"}}
{"_id": "597651", "text": "what is amber", "metadata": {"nothing"}}
{"_id": "403613", "text": "is autoimmune hepatitis a bile acid synthesis disorder",
 "metadata": {"self immune"}}
...
```

Para testar usando o conjunto de dados MS MARCO, imagine um cenário no qual você monta um bilhão de solicitações de entrada distribuídas em 100 arquivos, cada um contendo 10 milhões de solicitações. Os arquivos seriam armazenados no Amazon S3 com o prefixo s3://_djl_batch_input/. offlinebatch/sagemaker/sagemaker O pipeline de OpenSearch ingestão digitalizaria esses 100 arquivos simultaneamente e iniciaria um trabalho SageMaker em lote com 100 trabalhadores para processamento paralelo, permitindo a vetorização e a ingestão eficientes de um bilhão de documentos. OpenSearch

Em ambientes de produção, você pode usar um pipeline de OpenSearch ingestão para gerar arquivos S3 para entrada de inferência em lote. O pipeline é compatível com a várias fontes de dados e opera de acordo com um cronograma para transformar continuamente os dados da fonte em arquivos do S3. Esses arquivos são então processados automaticamente pelos servidores de IA por meio de trabalhos em lote offline agendados, garantindo o processamento e a ingestão contínuos de dados.

Etapa 4: monitorar o trabalho de inferência em lote

Você pode monitorar os trabalhos de inferência em lote usando o SageMaker console ou o. AWS CLI Você também pode usar a API Get Task para monitorar trabalhos em lote:

```
GET /_plugins/_ml/tasks/_search
{
  "query": {
    "bool": {
      "filter": [
```

```
        {
          "term": {
            "state": "RUNNING"
          }
        }
      ]
    }
  },
  "_source": ["model_id", "state", "task_type", "create_time", "last_update_time"]
}
```

A API retorna uma lista de tarefas ativas de trabalho em lote:

```
{
  "took": 2,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 3,
      "relation": "eq"
    },
    "max_score": 0.0,
    "hits": [
      {
        "_index": ".plugins-ml-task",
        "_id": "nyWbv5EB_tT1A82ZCu-e",
        "_score": 0.0,
        "_source": {
          "model_id": "nyWbv5EB_tT1A82ZCu-e",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496527958,
          "last_update_time": 1725496527958
        }
      },
      {
        "_index": ".plugins-ml-task",
```

```
        "_id": "miKbv5EB_tT1A82ZCu-f",
        "_score": 0.0,
        "_source": {
          "model_id": "miKbv5EB_tT1A82ZCu-f",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496528123,
          "last_update_time": 1725496528123
        }
      },
      {
        "_index": ".plugins-ml-task",
        "_id": "kiLbv5EB_tT1A82ZCu-g",
        "_score": 0.0,
        "_source": {
          "model_id": "kiLbv5EB_tT1A82ZCu-g",
          "state": "RUNNING",
          "task_type": "BATCH_PREDICTION",
          "create_time": 1725496529456,
          "last_update_time": 1725496529456
        }
      }
    ]
  }
}
```

Etapa 5: executar a pesquisa

Depois de monitorar o trabalho de inferência em lote e confirmar sua conclusão, você pode executar vários tipos de pesquisas baseadas em IA, incluindo pesquisas semânticas, híbridas, conversacionais (com RAG), esparsas neurais e multimodais. Para obter mais informações sobre pesquisas de IA suportadas pelo OpenSearch Serviço, consulte Pesquisa de IA.

Para pesquisar vetores brutos, use o tipo de consulta `knn`, forneça a matriz `vector` como entrada e especifique o número de resultados retornados `k`:

```
GET /my-raw-vector-index/_search
{
  "query": {
    "knn": {
      "my_vector": {
        "vector": [0.1, 0.2, 0.3],
        "k": 2
```

```
      }
    }
  }
}
```

Para executar uma pesquisa baseada em IA, use o tipo de consulta `neural`. Especifique a `query_text` entrada, o modelo `model_id` de incorporação que você configurou no pipeline de OpenSearch ingestão e o `k` número de resultados retornados. Para excluir incorporações dos resultados de pesquisa, especifique o nome do campo de incorporação no parâmetro `_source.excludes`:

```
GET /my-ai-search-index/_search
{
  "_source": {
    "excludes": [
      "output_embedding"
    ]
  },
  "query": {
    "neural": {
      "output_embedding": {
        "query_text": "What is AI search?",
        "model_id": "mBGzipQB2gmRjlv_dOoB",
        "k": 2
      }
    }
  }
}
```

Usando um pipeline OpenSearch de ingestão com OpenTelemetry o Collector

Você pode usar o OpenTelemetry Collector para ingerir registros, rastreamentos e métricas nos pipelines de OpenSearch ingestão. Um único pipeline pode ser usado para ingerir todos os registros, rastreamentos e métricas de diferentes índices em um domínio ou coleção. Você também pode usar pipelines para ingerir somente registros, rastreamentos ou métricas individualmente.

Tópicos

- Pré-requisitos
- Etapa 1: configurar a função do pipeline

- Etapa 2: Criar o pipeline
- Conectividade entre contas
- Limitações
- CloudWatch Alarmes recomendados para fontes OpenTelemetry

Pré-requisitos

Ao configurar o arquivo OpenTelemetry de configuração, você deve configurar o seguinte para que a ingestão ocorra:

- A função de ingestão precisa da `osis:Ingest` permissão para interagir com o pipeline. Para obter mais informações, consulte Função de ingestão.
- O valor do endpoint deve incluir o endpoint do pipeline. Por exemplo, `https://pipeline-endpoint.us-east-1.osis.amazonaws.com.`.
- O valor do serviço deve ser`osis`.
- A opção de compactação do OTLP/HTTP exportador deve corresponder à opção de compactação na fonte selecionada do pipeline.

```
extensions:
    sigv4auth:
        region: "region"
        service: "osis"

exporters:
    otlphttp:
        logs_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/v1/logs"
        metrics_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/v1/
metrics"
        traces_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/v1/
traces"
        auth:
            authenticator: sigv4auth
        compression: none

service:
    extensions: [sigv4auth]
    pipelines:
        traces:
```

```
        receivers: [jaeger]
        exporters: [otlphttp]
```

Etapa 1: configurar a função do pipeline

Depois de definir a configuração do OpenTelemetry coletor, defina a função do pipeline que você deseja usar na configuração do pipeline. Não há permissões específicas que a função do pipeline precise para a fonte OTLP, apenas permissões para conceder aos pipelines acesso ao OpenSearch domínio ou à coleção.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline OpenSearch de ingestão como o seguinte, que especifica o OTLP como a origem. Você também pode configurar OpenTelemetry registros, métricas e rastreamentos como fontes individuais.

Configuração do pipeline de origem OTLP:

```
version: 2
otlp-pipeline:
    source:
        otlp:
            logs_path: /otlp-pipeline/v1/logs
            traces_path: /otlp-pipeline/v1/traces
            metrics_path: /otlp-pipeline/v1/metrics
    sink:
        - opensearch:
            hosts: ["https://search-mydomain.region.es.amazonaws.com"]
            index: "ss4o_metrics-otel-%{yyyy.MM.dd}"
            index_type: custom
            aws:
                region: "region"
```

OpenTelemetry Configuração do pipeline de registros:

```
version: 2
otel-logs-pipeline:
  source:
    otel_logs_source:
```

```
        path: /otel-logs-pipeline/v1/logs
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.region.es.amazonaws.com"]
        index: "ss4o_metrics-otel-%{yyyy.MM.dd}"
        index_type: custom
        aws:
            region: "region"
```

OpenTelemetry Configuração do pipeline de métricas:

```
version: 2
otel-metrics-pipeline:
  source:
    otel_metrics_source:
        path: /otel-metrics-pipeline/v1/metrics
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.region.es.amazonaws.com"]
        index: "ss4o_metrics-otel-%{yyyy.MM.dd}"
        index_type: custom
        aws:
            region: "region"
```

OpenTelemetry Configuração do pipeline de rastreamento:

```
version: 2
otel-trace-pipeline:
  source:
    otel_trace_source:
        path: /otel-traces-pipeline/v1/traces
  sink:
    - opensearch:
        hosts: ["https://search-mydomain.region.es.amazonaws.com"]
        index: "ss4o_metrics-otel-%{yyyy.MM.dd}"
        index_type: custom
        aws:
            region: "region"
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Conectividade entre contas

OpenSearch Os pipelines de ingestão com OpenTelemetry fontes têm capacidade de ingestão entre contas. O Amazon OpenSearch Ingestion permite que você compartilhe pipelines entre Contas da AWS uma nuvem privada virtual (VPC) e um endpoint de pipeline em uma VPC separada. Para obter mais informações, consulte the section called “Configurar pipelines para ingestão entre contas”.

Limitações

O pipeline OpenSearch de ingestão não pode receber solicitações maiores que 20 MB. Esse valor é configurado pelo usuário na `max_request_length` opção. Essa opção tem como padrão 10 MB.

CloudWatch Alarmes recomendados para fontes OpenTelemetry

As CloudWatch métricas a seguir são recomendadas para monitorar o desempenho do seu pipeline de ingestão. Essas métricas podem ajudar você a identificar a quantidade de dados processados nas exportações, a quantidade de eventos processados nos fluxos, os erros no processamento de exportações e eventos de fluxo e o número de documentos gravados no destino. Você pode configurar CloudWatch alarmes para realizar uma ação quando uma dessas métricas exceder um valor especificado por um determinado período de tempo.

As CloudWatch métricas da fonte OTLP são formatadas como. `{pipeline-name}.otlp.{logs | traces | metrics}.{metric-name}` Por exemplo, `.otel-pipeline.otlp.metrics.requestTimeouts.count`

No caso de usar uma OpenTelemetry fonte individual, as métricas serão formatadas como`{pipeline-name}.{source-name}.{metric-name}`. Por exemplo, `.trace-pipeline.otel_trace_source.requestTimeouts.count`

Todos os três tipos de OpenTelemetry dados terão as mesmas métricas, mas, para resumir, as métricas só serão listadas na tabela abaixo para dados do tipo de log de origem OTLP.

Métrica	Description
`otel-pipeline.BlockingBuffer.bufferUsage.value`	Indica quanto do buffer está sendo utilizado.

Métrica	Description
`otel-pipeline.otlp.logs.requestTimeouts.count`	O número de solicitações que atingiram o tempo limite.
`otel-pipeline.otlp.logs.requestsReceived.count`	O número de solicitações recebidas pelo OpenTelemetry Coletor.
`otel-pipeline.otlp.logs.badRequests.count`	O número de solicitações malformadas recebidas pelo OpenTelemetry Coletor.
`otel-pipeline.otlp.logs.requestsTooLarge.count`	O número de solicitações maior que o máximo de 20 MB recebidas pelo OpenTelemetry Coletor.
`otel-pipeline.otlp.logs.internalServerError.count`	O número de erros HTTP 500 recebidos do OpenTelemetry Collector.
`otel-pipeline.opensearch.bulkBadRequestErrors.count`	Contagem de erros durante solicitações em massa devido a uma solicitação malformada.
`otel-pipeline.opensearch.bulkRequestLatency.avg`	Latência média para solicitações de gravação em massa feitas para OpenSearch.
`otel-pipeline.opensearch.bulkRequestNotFoundErrors.count`	Número de solicitações em massa que falharam porque os dados de destino não puderam ser encontrados.
`otel-pipeline.opensearch.bulkRequestNumberOfRetries.count`	Número de novas tentativas por pipelines OpenSearch de ingestão para gravar o cluster. OpenSearch
`otel-pipeline.opensearch.bulkRequestSizeBytes.sum`	Tamanho total em bytes de todas as solicitações em massa feitas para OpenSearch.
`otel-pipeline.opensearch.documentErrors.count`	Número de erros ao enviar documentos para OpenSearch. Os documentos que causam os erros serão enviados para a DLQ.

Métrica	Description
`otel-pipeline.opensearch.documentsSuccess.count`	Número de documentos gravados com sucesso em um OpenSearch cluster ou coleção.
`otel-pipeline.opensearch.documentsSuccessFirstAttempt.count`	Número de documentos indexados com sucesso OpenSearch na primeira tentativa.
`otel-pipeline.opensearch.documentsVersionConflictErrors.count`	Contagem de erros devido a conflitos de versão em documentos durante o processamento.
`otel-pipeline.opensearch.PipelineLatency.avg`	Latência média do pipeline de OpenSearch ingestão para processar os dados lendo da origem até a gravação no destino.
`otel-pipeline.opensearch.PipelineLatency.max`	Latência máxima do pipeline de OpenSearch ingestão para processar os dados lendo da origem até a gravação no destino.
`otel-pipeline.opensearch.recordsIn.count`	Contagem de registros ingeridos com sucesso. OpenSearch Essa métrica é essencial para rastrear o volume de dados sendo processados e armazenados.
`otel-pipeline.opensearch.s3.dlqS3RecordsFailed.count`	Número de registros que falharam na gravação na DLQ.
`otel-pipeline.opensearch.s3.dlqS3RecordsSuccess.count`	Número de registros gravados no DLQ.
`otel-pipeline.opensearch.s3.dlqS3RequestLatency.count`	Contagem de medidas de latência para solicitações à fila de mensagens mortas do Amazon S3.
`otel-pipeline.opensearch.s3.dlqS3RequestLatency.sum`	Latência total para todas as solicitações para a fila de mensagens mortas do Amazon S3

Métrica	Description
`otel-pipeline.opensearch.s3.dlqS3RequestSizeBytes.sum`	Tamanho total em bytes de todas as solicitações feitas na fila de mensagens mortas do Amazon S3.
`otel-pipeline.recordsProcessed.count`	Número total de registros processados no pipeline, uma métrica fundamental para a produtividade geral.
`otel-pipeline.opensearch.bulkRequestInvalidInputErrors.count`	Contagem de erros de solicitação em massa OpenSearch devido à entrada inválida, crucial para monitorar a qualidade dos dados e problemas operacionais.

Usando um pipeline OpenSearch de ingestão com o Amazon Managed Service para Prometheus

Você pode usar o Amazon Managed Service for Prometheus como um destino para OpenSearch seu pipeline de ingestão para armazenar métricas em formato de série temporal. O coletor Prometheus permite que você OpenTelemetry envie métricas ou outros dados de séries temporais do seu pipeline para um espaço de trabalho do Amazon Managed Service for Prometheus para monitoramento, alertas e análises.

O plug-in `prometheus` sink permite que os pipelines de OpenSearch ingestão gravem dados de métricas nos espaços de trabalho do Amazon Managed Service for Prometheus usando o protocolo de gravação remota do Prometheus. Essa integração permite a você:

- Armazene dados de métricas de séries temporais no Amazon Managed Service for Prometheus
- Monitore e alerte sobre métricas usando o Amazon Managed Service para Prometheus e o Amazon Managed Grafana
- Encaminhe métricas para vários destinos simultaneamente (por exemplo, OpenSearch e o Amazon Managed Service for Prometheus)
- Processe OpenTelemetry métricas de agentes externos ou gere métricas dentro do pipeline

Tópicos

- Pré-requisitos
- Etapa 1: configurar a função do pipeline
- Etapa 2: Criar o pipeline
- Monitoramento e solução de problemas
- Limitações
- Práticas recomendadas

Pré-requisitos

Antes de configurar o coletor Prometheus, verifique se você tem o seguinte:

- Espaço de trabalho do Amazon Managed Service para Prometheus: crie um espaço de trabalho no mesmo e no Conta da AWS mesmo Região da AWS pipeline de ingestão. OpenSearch Para obter instruções, consulte Criação de um espaço de trabalho no Guia do usuário do Amazon Managed Service for Prometheus.
- Permissões do IAM: configure uma função do IAM com permissões para gravar no Amazon Managed Service for Prometheus. Para obter mais informações, consulte the section called “Etapa 1: configurar a função do pipeline”.

> **Note**
>
> Os espaços de trabalho do Amazon Managed Service para Prometheus devem usar chaves gerenciadas pelo serviço. AWS AWS KMS Atualmente, AWS KMS as chaves gerenciadas pelo cliente não são suportadas pelos coletores do Amazon Managed Service for Prometheus nos pipelines de ingestão. OpenSearch

Etapa 1: configurar a função do pipeline

O coletor do Prometheus herda automaticamente as permissões do IAM da função do pipeline para autenticação, portanto, nenhuma configuração de função adicional (`sts_role_arn`como) é necessária nas configurações do coletor.

O exemplo de política a seguir mostra as permissões necessárias para usar o Amazon Managed Service for Prometheus como coletor:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AMPRemoteWrite",
      "Effect": "Allow",
      "Action": [
        "aps:RemoteWrite"
      ],
      "Resource": "arn:aws:aps:region:account-id:workspace/workspace-id"
    }
  ]
}
```

Substitua os seguintes espaços reservados:

- *region*: Seu Região da AWS (por exemplo,us-east-1)
- *account-id*: Seu Conta da AWS ID
- *workspace-id*: Seu ID de espaço de trabalho do Amazon Managed Service para Prometheus

Você deve anexar essas permissões à sua função no pipeline.

Certifique-se de que sua função de funil tenha uma relação de confiança que permita à OpenSearch Ingestion assumi-la:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Etapa 2: Criar o pipeline

Depois de configurar suas permissões, você pode configurar um pipeline de OpenSearch ingestão para usar o Amazon Managed Service for Prometheus como coletor.

Configuração básica

O exemplo a seguir mostra uma configuração mínima do coletor Prometheus:

```
version: "2"
sink:
  - prometheus:
      url: "https://aps-workspaces.region.amazonaws.com/workspaces/workspace-id/api/v1/
remote_write"
      aws:
        region: "region"
```

Você deve especificar a `url` opção na configuração do `prometheus` coletor, que é o endpoint de gravação remota do Amazon Managed Service for Prometheus. Para formatar o URL, localize o ID do seu espaço de trabalho no console do Amazon Managed Service for Prometheus e construa o URL da seguinte forma:. `https://aps-workspaces.region.amazonaws.com/workspaces/workspace-id/api/v1/remote_write`

Opções de configuração

Use as seguintes opções para configurar o comportamento de agrupamento e descarga para o coletor Prometheus:

Opções de configuração do coletor Prometheus

Opção	Obrigatório	Tipo	Description
`max_events`	Não	Inteiro	O número máximo de eventos a serem acumulados antes de serem enviados para Prometheus. O padrão é `1000`.
`max_request_size`	Não	Contagem de bytes	O tamanho máximo da carga útil da solicitação antes da descarga. O padrão é `1mb`.

Opção	Obrigatório	Tipo	Description
`flush_interval`	Não	Duração	O tempo máximo de espera antes dos eventos de descarga. O padrão é `10s`. O valor máximo permitido é`60s`.

Exemplos de oleodutos

Exemplo 1: OpenTelemetry métricas para o Amazon Managed Service para Prometheus

Esse pipeline recebe OpenTelemetry métricas de um agente externo e as grava no Amazon Managed Service for Prometheus:

```
version: "2"
source:
  otel_metrics_source:
    path: "/v1/metrics"
    output_format: otel

sink:
  - prometheus:
      url: "https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/api/v1/remote_write"
      aws:
        region: "us-east-1"
```

Exemplo 2: coletor duplo — OpenSearch e Amazon Managed Service para Prometheus

Esse pipeline encaminha métricas para ambos OpenSearch e para o Amazon Managed Service for Prometheus:

```
version: "2"
source:
  otel_metrics_source:
    path: "/v1/metrics"
    output_format: otel

sink:
  - opensearch:
      hosts:
```

```
        - "https://search-domain-endpoint.us-east-1.es.amazonaws.com"
      index: "metrics-%{yyyy.MM.dd}"
      aws:
        region: "us-east-1"
        sts_role_arn: "arn:aws:iam::123456789012:role/OSI-Pipeline-Role"

  - prometheus:
      url: "https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/api/v1/remote_write"
      aws:
        region: "us-east-1"
```

Exemplo 3: Métricas com filtragem

Esse pipeline filtra as métricas antes de enviar para o Amazon Managed Service for Prometheus:

```
version: "2"
source:
  otel_metrics_source:
    path: "/v1/metrics"
    output_format: otel

processor:
  - drop_events:
      drop_when: '/name != "http.server.duration" and /name != "http.client.duration"'

sink:
  - prometheus:
      url: "https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/api/v1/remote_write"
      aws:
        region: "us-east-1"
```

Você pode usar um esquema pré-configurado do Amazon Managed Service for Prometheus para criar esses pipelines. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Criação de um pipeline com o Amazon Managed Service for Prometheus sink

Usando o AWS console

1. Navegue até o console OpenSearch de serviço.

2. Escolha Pipelines em Ingestão.
3. Selecione Criar pipeline.
4. Selecione Criar usando o blueprint e escolha as OpenTelemetry métricas para o blueprint do Amazon Prometheus.
5. Configure o pipeline:
   - Insira seu ID do espaço de trabalho do Amazon Managed Service for Prometheus
   - Especifique o ARN da função do pipeline
   - Defina as configurações de origem e processador conforme necessário
6. Revise e crie o pipeline.

Usando a AWS CLI

Crie um arquivo de configuração de pipeline (por exemplo,`amp-pipeline.yaml`) com a configuração desejada e execute:

```
aws osis create-pipeline \
  --pipeline-name my-amp-pipeline \
  --min-units 2 \
  --max-units 4 \
  --pipeline-configuration-body file://amp-pipeline.yaml
```

Usando AWS CloudFormation

```
Resources:
  MyAMPPipeline:
    Type: AWS::OSIS::Pipeline
    Properties:
      PipelineName: my-amp-pipeline
      MinUnits: 2
      MaxUnits: 4
      PipelineConfigurationBody: |
        version: "2"
        source:
          otel_metrics_source:
            path: "/v1/metrics"
            output_format: otel
        sink:
          - prometheus:
```

```
              url: "https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/api/v1/remote_write"
              aws:
                region: "us-east-1"
```

Monitoramento e solução de problemas

CloudWatch métricas

Monitore o desempenho do seu funil usando CloudWatch métricas:

- `DocumentsWritten`: Número de métricas gravadas com sucesso no Amazon Managed Service for Prometheus
- `DocumentsWriteFailed`: Número de métricas que falharam na gravação
- `RequestLatency`: Latência das solicitações de gravação remota

Problemas comuns

Problema: o Pipeline não consegue gravar no Amazon Managed Service para Prometheus

Soluções:

- Verifique se o ID do espaço de trabalho e a região no URL estão corretos
- Certifique-se de que a função do pipeline tenha `aps:RemoteWrite` permissão
- Verifique se o espaço de trabalho usa chaves gerenciadas pelo serviço AWS KMS
- Verifique se o pipeline e o espaço de trabalho estão no mesmo Conta da AWS

Problema: erros de autenticação

Soluções:

- Verifique se a relação de confiança `osis-pipelines.amazonaws.com` permite assumir a função do pipeline
- Certifique-se de que a função do pipeline tenha a `aps:RemoteWrite` permissão necessária

Problema: alta latência ou limitação

Soluções:

- Aumentar as unidades de capacidade da tubulação
- Implemente o agrupamento em lotes no processador
- Analise as cotas de serviços do Amazon Managed Service para Prometheus

Limitações

Considere as seguintes limitações ao configurar um pipeline de OpenSearch ingestão para o Amazon Managed Service for Prometheus:

- Os espaços de trabalho do Amazon Managed Service para Prometheus devem usar chaves gerenciadas pelo serviço. AWS AWS KMS Atualmente, não há suporte para AWS KMS chaves gerenciadas pelo cliente.
- O pipeline e o espaço de trabalho do Amazon Managed Service for Prometheus devem estar no mesmo local. Conta da AWS

Práticas recomendadas

- Use a mesma função do IAM: o coletor Prometheus usa automaticamente a função de pipeline. Se outros sumidouros forem usados, certifique-se de que `sts_role_arn` seja a mesma função da tubulação
- Monitore métricas: configure CloudWatch alarmes para falhas de gravação e alta latência
- Implemente a filtragem: use processadores para filtrar métricas desnecessárias antes de enviar para o Amazon Managed Service for Prometheus
- Capacidade do tamanho certo: comece com a capacidade mínima e escale com base no volume de métricas
- Use esquemas: utilize esquemas pré-configurados para casos de uso comuns

Usando um pipeline OpenSearch de ingestão com o Kafka

Você pode usar o plug-in Kafka para transmitir dados de clusters autogerenciados do Kafka para domínios do OpenSearch Amazon Service e coleções Serverless. OpenSearch OpenSearch A ingestão oferece suporte a conexões de clusters do Kafka configurados com redes públicas ou privadas (VPC). Este tópico descreve os pré-requisitos e as etapas para configurar um pipeline de ingestão, incluindo a definição de configurações de rede e métodos de autenticação, como mutual TLS (mTLS), SASL/SCRAM ou IAM.

Migrar dados de clusters do Kafka públicos

Você pode usar pipelines de OpenSearch ingestão para migrar dados de um cluster público autogerenciado do Kafka, o que significa que o nome DNS do domínio pode ser resolvido publicamente. Para fazer isso, configure um pipeline de OpenSearch ingestão com Kafka autogerenciado como origem e OpenSearch Service ou OpenSearch Serverless como destino. Isso processa seus dados de streaming de um cluster de origem autogerenciado para um domínio ou AWS coleção de destino gerenciado.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster Kafka autogerenciado com uma configuração de rede pública. O cluster deve conter os dados que você deseja ingerir no OpenSearch Service.
2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para obter mais informações, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
3. Configure a autenticação em seu cluster autogerenciado com AWS Secrets Manager. Ative a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

   O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
```

```
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline do Kafka, configure a função do pipeline que você deseja usar na configuração do pipeline e adicione permissão para gravar em um domínio de OpenSearch serviço ou coleção OpenSearch sem servidor, bem como permissão para ler segredos do Secrets Manager.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline OpenSearch de ingestão como o seguinte, que especifica o Kafka como a origem.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de um cluster de origem do Confluent Kafka para uma coleção de VPC sem servidor. OpenSearch Forneça uma política de acesso à rede na configuração do pipeline. Você pode usar um registro de esquema do Confluent para definir um esquema do Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
```

```
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para saber mais, consulte the section called “Trabalhar com esquemas”.

Migrar dados de clusters do Kafka em uma VPC

Você também pode usar pipelines OpenSearch de ingestão para migrar dados de um cluster Kafka autogerenciado executado em uma VPC. Para fazer isso, configure um pipeline de OpenSearch ingestão com Kafka autogerenciado como origem e OpenSearch Service ou OpenSearch Serverless como destino. Isso processa seus dados de streaming de um cluster de origem autogerenciado para um domínio ou AWS coleção de destino gerenciado.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster Kafka autogerenciado com uma configuração de rede VPC que contenha os dados que você deseja ingerir no Service. OpenSearch
2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.
3. Configure a autenticação em seu cluster autogerenciado com AWS Secrets Manager. Habilite a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Obtenha o ID da VPC que tem acesso ao Kafka autogerenciado. Escolha o CIDR da VPC a ser usado pela ingestão. OpenSearch

   > **Note**
   >
   > Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline de OpenSearch ingestão à sua VPC para usar o Kafka autogerenciado. Para fazer isso, encontre a seção Configuração de rede, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas ou selecione a sua própria. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.
   > Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre OpenSearch ingestão e autogerenciamento OpenSearch, certifique-se de que o CIDR autogerenciado da OpenSearch VPC seja diferente do CIDR para ingestão. OpenSearch

5. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

   O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

   JSON

   ```
   {
   ```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/domain-name"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline, configure o perfil de pipeline que você deseja usar na configuração do pipeline e adicione as seguintes permissões nesse perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Você deve fornecer as permissões do Amazon EC2 acima sobre a função do IAM que você usa para criar o pipeline de OpenSearch ingestão, pois o pipeline usa essas permissões para criar e excluir uma interface de rede em sua VPC. O pipeline só pode acessar o cluster do Kafka por meio dessa interface de rede.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline OpenSearch de ingestão como o seguinte, que especifica o Kafka como a origem.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de um cluster de origem do Confluent Kafka para uma coleção de VPC sem servidor. OpenSearch Forneça uma política de acesso à rede na configuração do pipeline. Você pode usar um registro de esquema do Confluent para definir um esquema do Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
```

```
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Migração de dados de OpenSearch clusters autogerenciados usando o Amazon Ingestion OpenSearch

Você pode usar um pipeline de OpenSearch ingestão da Amazon com autogerenciamento OpenSearch ou Elasticsearch para migrar dados para domínios do OpenSearch Amazon Service e coleções sem servidor. OpenSearch OpenSearch A ingestão oferece suporte a configurações de rede pública e privada para a migração de dados do autogerenciado OpenSearch e do Elasticsearch.

Migração de clusters públicos OpenSearch

Você pode usar pipelines de OpenSearch ingestão para migrar dados de um cluster autogerenciado OpenSearch ou do Elasticsearch com uma configuração pública, o que significa que o nome DNS do domínio pode ser resolvido publicamente. Para fazer isso, configure um pipeline de OpenSearch ingestão com autogerenciamento OpenSearch ou Elasticsearch como origem e OpenSearch serviço ou sem OpenSearch servidor como destino. Isso migra efetivamente seus dados de um cluster de origem autogerenciado para um domínio ou coleção AWS de destino gerenciado.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster autogerenciado OpenSearch ou Elastisearch que contenha os dados que você deseja migrar e configure um nome DNS público. Para obter instruções, consulte Criar um cluster na OpenSearch documentação.

2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para obter mais informações, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
3. Configure a autenticação em seu cluster autogerenciado com AWS Secrets Manager. Ative a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

   O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

   JSON

   ```
   {
     "Version":"2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:us-east-1:111122223333:domain/domain-name"
         ]
       }
     ]
   }
   ```

   Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do OpenSearch pipeline, configure a função do pipeline que você deseja usar na configuração do pipeline e adicione permissão para gravar em um domínio de OpenSearch serviço ou coleção OpenSearch sem servidor, bem como permissão para ler segredos do Secrets Manager.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que especifica OpenSearch como origem.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de uma fonte OpenSearch ou cluster do Elasticsearch para uma coleção de VPC sem servidor OpenSearch . Forneça uma política de acesso à rede na configuração do pipeline.

```
version: "2"
opensearch-migration-pipeline:
  source:
    opensearch:
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
```

```
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Migração de dados de OpenSearch clusters em uma VPC

Você também pode usar pipelines OpenSearch de ingestão para migrar dados de um cluster autogerenciado OpenSearch ou do Elasticsearch executado em uma VPC. Para fazer isso, configure um pipeline de OpenSearch ingestão com autogerenciamento OpenSearch ou Elasticsearch como origem e OpenSearch serviço ou sem OpenSearch servidor como destino. Isso migra efetivamente seus dados de um cluster de origem autogerenciado para um domínio ou coleção AWS de destino gerenciado.

Pré-requisitos

Antes de criar seu pipeline OpenSearch de ingestão, execute as seguintes etapas:

1. Crie um cluster autogerenciado OpenSearch ou Elastisearch com uma configuração de rede VPC que contenha os dados que você deseja migrar.
2. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor para onde você deseja migrar dados. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço e Criação de coleções.

3. Configure a autenticação em seu cluster autogerenciado com AWS Secrets Manager. Ative a alternância de segredos seguindo as etapas em Alternar segredos do AWS Secrets Manager.
4. Obtenha o ID da VPC que tem acesso ao OpenSearch autogerenciado ou ao Elasticsearch. Escolha o CIDR da VPC a ser usado pela ingestão. OpenSearch

> **Note**
>
> Se você estiver usando o Console de gerenciamento da AWS para criar seu pipeline, você também deve anexar seu pipeline de OpenSearch ingestão à sua VPC para usar o OpenSearch autogerenciado ou o Elasticsearch. Para fazer isso, encontre a seção Opções de rede de origem, marque a caixa de seleção Anexar à VPC e escolha seu CIDR em uma das opções padrão fornecidas. Você pode usar qualquer CIDR de um espaço de endereço privado, conforme definido em Melhor prática atual RFC 1918.
> Para fornecer um CIDR personalizado, selecione Outro no menu suspenso. Para evitar uma colisão de endereços IP entre OpenSearch ingestão e autogerenciamento OpenSearch, certifique-se de que o CIDR autogerenciado da OpenSearch VPC seja diferente do CIDR para ingestão. OpenSearch

5. Anexe uma política baseada em recursos ao seu domínio ou uma política de acesso a dados à sua coleção. Essas políticas de acesso permitem que o OpenSearch Inestion grave dados do seu cluster autogerenciado em seu domínio ou coleção.

   O exemplo de política de acesso ao domínio a seguir permite que a função de pipeline, que você cria na próxima etapa, grave dados em um domínio. Lembre-se de atualizar o `resource` com seu próprio ARN.

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
```

```
      "Resource": [
        "arn:aws:es:us-east-1:111122223333:domain/example.com"
      ]
    }
  ]
}
```

Para criar um perfil do IAM com as permissões corretas para acessar dados de gravação na coleção ou no domínio, consulte the section called “Configurar funções e usuários”.

Etapa 1: configurar a função do pipeline

Depois de configurar os pré-requisitos do pipeline, configure o perfil de pipeline que você deseja usar na configuração do pipeline e adicione as seguintes permissões nesse perfil:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:us-
east-1:111122223333:secret:secret-name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
```

```
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Você deve fornecer as permissões do Amazon EC2 acima sobre a função do IAM que você usa para criar o pipeline de OpenSearch ingestão, pois o pipeline usa essas permissões para criar e excluir uma interface de rede em sua VPC. O pipeline só pode acessar o OpenSearch cluster por meio dessa interface de rede.

Etapa 2: Criar o pipeline

Em seguida, você pode configurar um pipeline de OpenSearch ingestão como o seguinte, que especifica OpenSearch como origem.

Você pode especificar vários domínios OpenSearch de serviço como destinos para seus dados. Esse recurso permite o roteamento condicional ou a replicação de dados recebidos em vários domínios de serviço. OpenSearch

Você também pode migrar dados de uma fonte OpenSearch ou cluster do Elasticsearch para uma coleção de VPC sem servidor OpenSearch . Forneça uma política de acesso à rede na configuração do pipeline.

```
version: "2"
opensearch-migration-pipeline:
  source:
    opensearch:
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
```

```
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

É possível usar um esquema pré-configurado para criar esse pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

Use um pipeline OpenSearch de ingestão com o Amazon Kinesis Data Streams

Use um pipeline OpenSearch de ingestão com o Amazon Kinesis Data Streams para ingerir dados de registros de stream de vários streams em domínios e coleções do Amazon Service. OpenSearch O pipeline OpenSearch de ingestão incorpora a infraestrutura de ingestão de streaming para fornecer uma forma de alta escala e baixa latência de ingerir continuamente registros de streaming do Kinesis.

Tópicos

- Amazon Kinesis Data Streams como origem
- Várias contas do Amazon Kinesis Data Streams como origem

Amazon Kinesis Data Streams como origem

Com o procedimento a seguir, você aprenderá a configurar um pipeline de OpenSearch ingestão que usa o Amazon Kinesis Data Streams como fonte de dados. Esta seção aborda os pré-requisitos necessários, como criar um domínio de OpenSearch serviço ou uma coleção OpenSearch sem servidor e percorrer as etapas para configurar a função do pipeline e criar o pipeline.

Pré-requisitos

Para configurar o pipeline, você precisa de um ou mais Kinesis Data Streams. Esses fluxos devem estar recebendo registros ou estar prontos para receber registros de outras origens. Para obter mais informações, consulte Visão geral da OpenSearch ingestão.

Para configurar o pipeline

1. Crie um domínio OpenSearch de serviço ou uma coleção OpenSearch sem servidor

   Para criar um domínio ou uma coleção, consulte Introdução à OpenSearch ingestão.

   Para criar um perfil do IAM com as permissões certas para acessar dados de gravação na coleção ou no domínio, consulte Políticas baseadas no recurso.

2. Configurar o perfil do pipeline com permissões

   Configure o perfil de pipeline que você deseja usar na configuração do pipeline e adicione a ele as seguintes permissões. Substitua *placeholder values* por suas próprias informações.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:DescribeStreamConsumer",
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards",
                "kinesis:ListStreams",
                "kinesis:ListStreamConsumers",
                "kinesis:RegisterStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-1:111122223333:stream/stream-name"
            ]
        }
    ]
}
```

Se a criptografia do lado do servidor estiver habilitada nos fluxos, a política da AWS KMS a seguir permitirá descriptografar os registros. Substitua *placeholder values* por suas próprias informações.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowDecryptionOfCustomManagedKey",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id"
        }
    ]
}
```

Para que um pipeline grave dados em um domínio, o domínio deve ter uma política de acesso em nível de domínio que permita que a função de pipeline sts_role_arn o acesse.

O exemplo de política de acesso a domínio a seguir permite que o perfil de pipeline denominado `pipeline-role` criado na etapa anterior () grave dados no domínio `ingestion-domain`. Substitua *placeholder values* por suas próprias informações.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::your-account-id:role/pipeline-role"
      },
```

```
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:Região da AWS:account-id:domain/domain-name/*"
    }
  ]
}
```

3. Criar o pipeline

Configure um pipeline OpenSearch de ingestão especificando K inesis-data-streams como a origem. Você pode localizar um blueprint pronto disponível no Console OpenSearch de ingestão para criar esse pipeline. (Opcional) Para criar o pipeline usando o AWS CLI, você pode usar um blueprint chamado "**AWS-KinesisDataStreamsPipeline**". Substitua *placeholder values* por suas próprias informações.

```
version: "2"
kinesis-pipeline:
  source:
    kinesis_data_streams:
      acknowledgments: true
      codec:
        # Based on whether kinesis records are aggregated or not, you could choose
 json, newline or ndjson codec for processing the records.
        # JSON codec supports parsing nested CloudWatch Events into individual log
 entries that will be written as documents into OpenSearch.
        # json:
          # key_name: "logEvents"
          # These keys contain the metadata sent by CloudWatch Subscription Filters
          # in addition to the individual log events:
          # include_keys: [ 'owner', 'logGroup', 'logStream' ]
        newline:
      streams:
        - stream_name: "stream name"
          # Enable this if ingestion should start from the start of the stream.
          # initial_position: "EARLIEST"
          # checkpoint_interval: "PT5M"
          # Compression will always be gzip for CloudWatch, but will vary for other
 sources:
          # compression: "gzip"
        - stream_name: "stream name"
          # Enable this if ingestion should start from the start of the stream.
          # initial_position: "EARLIEST"
```

```
          # checkpoint_interval: "PT5M"
          # Compression will always be gzip for CloudWatch, but will vary for other
 sources:
          # compression: "gzip"

        # buffer_timeout: "1s"
        # records_to_accumulate: 100
        # Change the consumer strategy to "polling". Default consumer strategy will
 use enhanced "fan-out" supported by KDS.
        # consumer_strategy: "polling"
        # if consumer strategy is set to "polling", enable the polling config
 below.
        # polling:
          # max_polling_records: 100
          # idle_time_between_reads: "250ms"
      aws:
        # Provide the Role ARN with access to Amazon Kinesis Data Streams. This
 role should have a trust relationship with osis-pipelines.amazonaws.com
        sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
        # Provide the Região da AWS of the Data Stream.
        region: "us-east-1"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Serverless domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
        index: "index_${getMetadata(\"stream_name\")}"
        # Ensure adding unique document id as a combination of the metadata
 attributes available.
        document_id: "${getMetadata(\"partition_key\")}_
${getMetadata(\"sequence_number\")}_${getMetadata(\"sub_sequence_number\")}"
        aws:
          # Provide a Role ARN with access to the domain. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
          sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
          # Provide the Região da AWS of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch
 Serverless collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the
 serverless collection
```

```
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the OpenSearch Serverless
 domain is of version Elasticsearch 6.x
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
 compression setting is changed in the domain. See https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/gzip.html
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3
 bucket. Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the region of the bucket.
            region: "us-east-1"
            # Provide a Role ARN with access to the bucket. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
```

Opções de configuração

Para ver as opções de configuração do Kinesis, consulte Opções de configuração na OpenSearchdocumentação.

Atributos de metadados disponíveis

- stream_name: nome do Kinesis Data Streams de onde o registro foi ingerido
- partition_key: chave de partição do registro do Kinesis Data Streams que está sendo ingerido
- sequence_number: número da sequência do registro do Kinesis Data Streams que está sendo ingerido
- sequence_number: número da subsequência do registro do Kinesis Data Streams que está sendo ingerido

4. (Opcional) Configure as unidades de computação recomendadas (OCUs) para o pipeline do Kinesis Data Streams

   Um pipeline de origem do OpenSearch Kinesis Data Streams também pode ser configurado para ingerir registros de stream de mais de um stream. Para ter uma ingestão mais rápida,

recomendamos que você adicione uma unidade computacional adicional por cada novo fluxo adicionado.

Consistência de dados

OpenSearch A ingestão suporta o end-to-end reconhecimento para garantir a durabilidade dos dados. Quando o pipeline lê registros de fluxo do Kinesis, ele distribui dinamicamente o trabalho de leitura de registros de fluxo com base nos fragmentos associados aos fluxos. O Pipeline verificará automaticamente os fluxos quando receber uma confirmação após ingerir todos os registros no domínio ou na coleção. OpenSearch Isso evitará o processamento duplicado dos registros de fluxo.

Para criar o índice com base no nome do fluxo, defina-o na seção do coletor do opensearch como “index_${getMetadata(\“stream_name\”)}”.

Várias contas do Amazon Kinesis Data Streams como origem

Você pode conceder acesso a várias contas com o Amazon Kinesis Data Streams OpenSearch para que os pipelines de ingestão possam acessar o Kinesis Data Streams em outra conta como fonte. Conclua as seguintes etapas para habilitar o acesso entre contas:

Configure o acesso entre contas

1. Defina a política de recursos na conta que tem o fluxo do Kinesis

   Substitua *`placeholder values`* por suas próprias informações.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "StreamReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamSummary",
```

```
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:us-east-1:444455556666:stream/
stream-name"
        },
        {
            "Sid": "StreamEFOReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:us-
east-1:444455556666:stream/stream-name/consumer/consumer-name"
        }
    ]
}
```

2. (Opcional) Configurar o consumidor e a política de recursos do consumidor

   Essa é uma etapa opcional e apenas será necessária se você planejar usar a estratégia de consumidor de fanout avançado para ler registros de fluxo. Para ter mais informações, consulte Desenvolver consumidores de fanout avançado com throughput dedicado.

   a. Configurar consumidor

      Para reutilizar um consumidor existente, você pode ignorar esta etapa. Para obter mais informações, consulte a RegisterStreamConsumerReferência da API do Amazon Kinesis Data Streams.

      No exemplo de comando CLI a seguir, substitua o por suas *placeholder values* próprias informações.

Example Exemplo de comando da CLI:

```
aws kinesis register-stream-consumer \
--stream-arn "arn:aws:kinesis:Região da AWS:account-id:stream/stream-name" \
--consumer-name consumer-name
```

b. Configurar a política de recursos do consumidor

Na declaração a seguir, *placeholder values* substitua o por suas próprias informações.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "ConsumerEFOReadStatementID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/Pipeline-Role"
            },
            "Action": [
                "kinesis:DescribeStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": "arn:aws:kinesis:us-
east-1:444455556666:stream/stream-1/consumer/consumer-name"
        }
    ]
}
```

3. Configuração do pipeline

Para ingestão entre contas, adicione os seguintes atributos em `kinesis_data_streams` para cada fluxo:

- `stream_arn`: o arn do fluxo que pertence à conta na qual o fluxo existe

- `consumer_arn`: esse é um atributo opcional e deve ser especificado se a estratégia padrão de consumidor de fanout avançado for escolhida. Especifique o arn do consumidor real para esse campo. Substitua *`placeholder values`* por suas próprias informações.

```
version: "2"
     kinesis-pipeline:
       source:
         kinesis_data_streams:
           acknowledgments: true
           codec:
             newline:
           streams:
             - stream_arn: "arn:aws:kinesis:region:stream-account-id:stream/stream-
name"
               consumer_arn: "consumer arn"
               # Enable this if ingestion should start from the start of the
 stream.
               # initial_position: "EARLIEST"
               # checkpoint_interval: "PT5M"
             - stream_arn: "arn:aws:kinesis:region:stream-account-id:stream/stream-
name"
               consumer_arn: "consumer arn"
                # initial_position: "EARLIEST"

             # buffer_timeout: "1s"
             # records_to_accumulate: 100
             # Enable the consumer strategy to "polling". Default consumer strategy
 will use enhanced "fan-out" supported by KDS.
             # consumer_strategy: "polling"
             # if consumer strategy is set to "polling", enable the polling config
 below.
             # polling:
               # max_polling_records: 100
               # idle_time_between_reads: "250ms"
           aws:
             # Provide the Role ARN with access to Kinesis. This role should have a
 trust relationship with osis-pipelines.amazonaws.com
             sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
             # Provide the Região da AWS of the domain.
             region: "us-east-1"
```

```
      sink:
        - opensearch:
            # Provide an OpenSearch Serverless domain endpoint
            hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
            index: "index_${getMetadata(\"stream_name\")}"
            # Mapping for documentid based on partition key, shard sequence number
 and subsequence number metadata attributes
            document_id: "${getMetadata(\"partition_key\")}_
${getMetadata(\"sequence_number\")}_${getMetadata(\"sub_sequence_number\")}"
            aws:
              # Provide a Role ARN with access to the domain. This role should
 have a trust relationship with osis-pipelines.amazonaws.com
              sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
              # Provide the Região da AWS of the domain.
              region: "us-east-1"
              # Enable the 'serverless' flag if the sink is an OpenSearch
 Serverless collection
              serverless: false
                # serverless_options:
                # Specify a name here to create or update network policy for the
 serverless collection
              # network_policy_name: network-policy-name
            # Enable the 'distribution_version' setting if the OpenSearch
 Serverless domain is of version Elasticsearch 6.x
            # distribution_version: "es6"
            # Enable and switch the 'enable_request_compression' flag if
 the default compression setting is changed in the domain. See https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/gzip.html
            # enable_request_compression: true/false
            # Optional: Enable the S3 DLQ to capture any failed requests in an S3
 bucket. Delete this entire block if you don't want a DLQ.
            dlq:
              s3:
                # Provide an Amazon S3 bucket
                bucket: "your-dlq-bucket-name"
                # Provide a key path prefix for the failed requests
                # key_path_prefix: "alb-access-log-pipeline/logs/dlq"
                # Provide the Região da AWS of the bucket.
                region: "us-east-1"
                # Provide a Role ARN with access to the bucket. This role should
 have a trust relationship with osis-pipelines.amazonaws.com
                sts_role_arn: "arn:aws:iam::111122223333:role/Example-Role"
```

4. Perfil do pipeline OSI no Kinesis Data Streams

   a. Política do IAM

      Adicione a seguinte política ao perfil do pipeline. Substitua *placeholder values* por suas próprias informações.

      JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": [
            "arn:aws:kinesis:us-east-1:111122223333:stream/my-stream"
            ]
        },
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards",
                "kinesis:ListStreams",
                "kinesis:ListStreamConsumers",
                "kinesis:RegisterStreamConsumer"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-1:111122223333:stream/my-stream"
            ]
        }
    ]
```

```
}
```

b. Política de confiança

Para ingerir dados da conta do fluxo, você precisará estabelecer uma relação de confiança entre o perfil de ingestão do pipeline e a conta do fluxo. Adicione o que se seque ao perfil do pipeline. Substitua *placeholder values* por suas próprias informações.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::111122223333:root"
      },
     "Action": "sts:AssumeRole"
  }]
}
```

Próximas etapas

Depois de exportar seus dados para um pipeline, você pode consultá-los no domínio OpenSearch Service que está configurado como um coletor para o pipeline. Os seguintes recursos podem ajudá-lo a começar:

- Observabilidade
- the section called “Rastreamentos”
- Observabilidade

Usando um pipeline OpenSearch de ingestão com AWS Lambda

Use o AWS Lambda processador para enriquecer dados de qualquer fonte ou destino suportado pela OpenSearch ingestão usando código personalizado. Com o processador do Lambda, você pode

aplicar suas próprias transformações ou enriquecimentos de dados e depois retornar os eventos processados ao pipeline para processamento adicional. Esse processador permite o processamento de dados personalizado e oferece total controle sobre como os dados são manipulados antes de passar pelo pipeline.

> **Note**
>
> O limite de tamanho da carga útil para um único evento processado pelo processador do Lambda é de 5 MB. Além disso, o processador do Lambda só é compatível com respostas no formato de matriz JSON.

Pré-requisitos

Antes de criar um pipeline com um processador do Lambda, crie os seguintes recursos:

- Uma AWS Lambda função que enriquece e transforma seus dados de origem. Para obter instruções, consulte Criar sua primeira função do Lambda.
- Um domínio OpenSearch de serviço ou coleção OpenSearch sem servidor que será o coletor do pipeline. Para obter mais informações, consulte the section called “Criação OpenSearch de domínios de serviço” e the section called “Criação de coleções”.
- Um perfil de pipeline que inclui permissões para gravar no coletor do domínio ou da coleção. Para saber mais, consulte the section called “Perfis do pipeline”.

  O perfil do pipeline também precisa de uma política de permissões anexada que permita invocar a função do Lambda especificada na configuração do pipeline. Por exemplo:

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowinvokeFunction",
            "Effect": "Allow",
            "Action": [
                "lambda:invokeFunction",
                "lambda:InvokeAsync",
                "lambda:ListFunctions"
            ],
```

```
            "Resource": "arn:aws:lambda:us-
east-1:111122223333:function:function-name"

        }
    ]
}
```

Criar um pipeline

Para usar AWS Lambda como processador, configure um pipeline OpenSearch de ingestão e especifique `aws_lambda` como processador. Você também pode usar o esquema de enriquecimento personalizado do AWS Lambda para criar o pipeline. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

O exemplo de pipeline a seguir recebe dados de uma fonte HTTP, os enriquece usando um processador de data e o AWS Lambda processador e ingere os dados processados em um domínio. OpenSearch

```
version: "2"
lambda-processor-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
      - date:
        destination: "@timestamp"
        from_time_received: true
    - aws_lambda:
        function_name: "my-lambda-function"

        tags_on_failure: ["lambda_failure"]
        batch:
            key_name: "events"
        aws:
          region: region
  sink:
    - opensearch:
        hosts: [ "https://search-mydomain.us-east-1es.amazonaws.com" ]
        index: "table-index"
        aws:
          region: "region"
```

```
          serverless: false
```

A AWS Lambda função de exemplo a seguir transforma os dados recebidos adicionando um novo par de valores-chave (`"transformed": "true"`) a cada elemento na matriz de eventos fornecida e, em seguida, envia de volta a versão modificada.

```
import json

def lambda_handler(event, context):
    input_array = event.get('events', [])
    output = []
    for input in input_array:
        input["transformed"] = "true";
        output.append(input)

    return output
```

Agrupamento em lotes

Os pipelines enviam eventos em lote para o processador do Lambda e ajustam dinamicamente o tamanho do lote para garantir que fique abaixo do limite de 5 MB.

A seguinte imagem é um exemplo de lote do pipeline:

```
batch:
    key_name: "events"

input_arrary = event.get('events', [])
```

> **Note**
>
> Ao criar um pipeline, certifique-se de que a opção `key_name` na configuração do processador do Lambda corresponda à chave de evento no manipulador do Lambda.

Filtragem condicional

A filtragem condicional permite controlar quando seu AWS Lambda processador invoca a função Lambda com base em condições específicas nos dados do evento. Isso será especialmente útil quando você deseja processar seletivamente certos tipos de evento enquanto ignora outros.

O seguinte exemplo de configuração usa filtragem condicional:

```
processors:
  - aws_lambda:
      function_name: "my-lambda-function"
      aws:
        region: "region"
      lambda_when: "/sourceIp == 10.10.10.10"
```

Migração de dados entre domínios e coleções usando o Amazon Ingestion OpenSearch

Você pode usar pipelines OpenSearch de ingestão para migrar dados entre domínios do Amazon OpenSearch Service ou coleções de VPC sem servidor OpenSearch . Para fazer isso, você define um pipeline no qual configura um domínio ou coleção como origem e outro domínio ou coleção como coletor. Isso migra efetivamente seus dados de um domínio ou coleção para outro.

Para migrar dados, você deve ter os seguintes recursos:

- Um domínio de OpenSearch serviço de origem ou uma coleção de OpenSearch VPC sem servidor. Esse domínio ou coleção contém os dados que você deseja migrar. Se você estiver usando um domínio, ele deverá estar executando a OpenSearch versão 1.0 ou posterior ou a versão 7.4 ou posterior do Elasticsearch. O domínio também deve ter uma política de acesso que conceda as permissões apropriadas à sua perfil de pipeline.
- Um domínio separado ou coleção da VPC para o qual você deseja migrar seus dados. Esse domínio ou coleção funcionará como o coletor do pipeline.
- Uma função de pipeline que o OpenSearch Ingestion usará para ler e gravar em sua coleção ou domínio. Inclua o nome do recurso da Amazon (ARN) deste perfil na configuração do pipeline. Para saber mais, consulte os seguintes recursos:
  - the section called “Concedendo acesso aos pipelines aos domínios”
  - the section called “Conceder aos pipelines acesso às coleções”

Tópicos

- Limitações
- OpenSearch Serviço como fonte

- Especificação de vários coletores OpenSearch de domínio de serviço
- Migração de dados para uma coleção de OpenSearch VPC sem servidor

Limitações

As seguintes limitações se aplicam quando você designa domínios OpenSearch de serviço ou coleções OpenSearch sem servidor como coletores:

- Um pipeline não pode gravar em mais de um domínio da VPC.
- Você só pode migrar dados de ou para coleções OpenSearch sem servidor que usam acesso VPC. As coleções públicas não são compatíveis.
- Você não pode especificar uma combinação de VPC e domínios públicos em uma única configuração de pipeline.
- Você pode ter no máximo 20 coletores sem ser pipeline em uma única configuração de pipeline.
- Você pode especificar coletores de no máximo três diferentes Regiões da AWS em uma única configuração de pipeline.
- Um pipeline com vários coletores poderá sofrer uma redução na velocidade de processamento ao longo do tempo se algum dos coletores ficar inativo por muito tempo ou não for provisionado com capacidade suficiente para receber dados de entrada.

OpenSearch Serviço como fonte

O domínio ou coleção que você especifica como origem é de onde os dados são migrados.

Criar um perfil de pipeline no IAM

Para criar seu pipeline de OpenSearch ingestão, primeiro você deve criar uma função de pipeline para conceder acesso de leitura e gravação entre domínios ou coleções. Para fazer isso, execute as seguintes etapas:

1. Crie uma nova política de permissões no IAM para anexar ao perfil do pipeline. Conceda permissões de leitura da fonte e de gravação no coletor. Para obter mais informações sobre como definir permissões de pipeline do IAM para domínios de OpenSearch serviço, consulte the section called “Concedendo acesso aos pipelines aos domínios” e. the section called “Conceder aos pipelines acesso às coleções”
2. Especifique as permissões a seguir no perfil de pipeline do IAM para leitura a partir da origem:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"es:ESHttpGet",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_cat/
indices",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
scroll",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpPost",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search/
point_in_time",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/*/_search/
scroll"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpDelete",
         "Resource":[
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
point_in_time",
            "arn:aws:es:us-east-1:111122223333:domain/domain-name/_search/
scroll"
         ]
      }
   ]
}
```

Criar um pipeline

Depois de anexar a política à função do pipeline, use o blueprint de AWSOpenSearchDataMigrationPipelinemigração para criar o pipeline. Esse esquema inclui uma configuração padrão para migrar dados entre domínios ou coleções OpenSearch de serviços. Para obter mais informações, consulte the section called “Trabalhar com esquemas”.

> **Note**
>
> OpenSearch A ingestão usa a versão e a distribuição do domínio de origem para determinar qual mecanismo usar para a migração. Algumas versões oferecem suporte à `point_in_time` opção. OpenSearch O Serverless usa a `search_after` opção porque ela não suporta `point_in_time` ou. `scroll`

Novos índices podem estar sendo criados durante o processo de migração, ou documentos podem estar sendo atualizados enquanto a migração está em andamento. Por isso, talvez seja necessário fazer uma única ou várias verificações dos dados de índice do domínio para obter dados novos ou atualizados.

Especifique o número de verificações a serem executadas, definindo `index_read_count` e `interval` na configuração do pipeline. O exemplo a seguir mostra como fazer várias verificações:

```
scheduling:
    interval: "PT2H"
    index_read_count: 3
    start_time: "2023-06-02T22:01:30.00Z"
```

OpenSearch A ingestão usa a seguinte configuração para garantir que seus dados sejam gravados no mesmo índice e mantenham a mesma ID do documento:

```
index: "${getMetadata(\"opensearch-index\")}"
document_id: "${getMetadata(\"opensearch-document_id\")}"
```

Especificação de vários coletores OpenSearch de domínio de serviço

Você pode especificar vários domínios OpenSearch de serviço público como destinos para seus dados. Você pode usar esse recurso para realizar roteamento condicional ou replicar dados de

entrada em vários domínios de serviço. OpenSearch Você pode especificar até 10 domínios de OpenSearch serviço público diferentes como coletores.

No exemplo a seguir, os dados recebidos são roteados condicionalmente para diferentes OpenSearch domínios de serviço:

```
...
  route:
    - 2xx_status: "/response >= 200 and /response < 300"
    - 5xx_status: "/response >= 500 and /response < 600"
  sink:
    - opensearch:
        hosts: [ "https://search-response-2xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-2xx"
        routes:
          - 2xx_status
    - opensearch:
        hosts: [ "https://search-response-5xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-5xx"
        routes:
          - 5xx_status
```

Migração de dados para uma coleção de OpenSearch VPC sem servidor

Você pode usar o OpenSearch Inestion para migrar dados de um domínio de OpenSearch serviço de origem ou de uma coleção OpenSearch sem servidor para um coletor de coleta de VPC. Você deve fornecer uma política de acesso à rede na configuração do pipeline. Para obter mais informações sobre a ingestão de dados em coleções de VPC OpenSearch sem servidor, consulte. the section called “Tutorial: Ingestão de dados em uma coleção”

Para migrar dados para uma coleção da VPC

1. Crie uma coleção OpenSearch sem servidor. Para instruções, consulte the section called “Tutorial: Ingestão de dados em uma coleção”.
2. Crie uma política de rede para a coleção que especifique o acesso via VPC ao endpoint da coleção e ao endpoint do Dashboards. Para instruções, consulte the section called “Acesso à rede”.

3. Crie o perfil de pipeline se ainda não tiver um. Para instruções, consulte the section called “Perfis do pipeline”.
4. Criar o pipeline. Para instruções, consulte the section called “Trabalhar com esquemas”.

Usando o AWS SDKs para interagir com a Amazon OpenSearch Ingestion

Esta seção inclui um exemplo de como usar o AWS SDKs para interagir com o Amazon OpenSearch Ingestion. O exemplo de código demonstra como criar um domínio e um pipeline e, em seguida, ingerir dados no pipeline.

Tópicos

- Python

Python

O script de exemplo a seguir usa o AWS SDK para Python (Boto3) para criar uma função de pipeline do IAM, um domínio para gravar dados e um pipeline para ingerir dados. Em seguida, ele ingere um arquivo de log de amostra no pipeline usando a biblioteca HTTP de `requests`.

Execute os comandos a seguir para instalar as dependências necessárias:

```
pip install boto3
pip install botocore
pip install requests
pip install requests-auth-aws-sigv4
```

No script, substitua todas as instâncias de *`account-id`* por seu ID da Conta da AWS .

```
import boto3
import botocore
from botocore.config import Config
import requests
from requests_auth_aws_sigv4 import AWSSigV4
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
```

```
# key, and default region.

opensearch = boto3.client('opensearch', config=my_config)
iam = boto3.client('iam', config=my_config)
osis = boto3.client('osis', config=my_config)

domainName = 'test-domain'  # The name of the domain
pipelineName = 'test-pipeline' # The name of the pipeline

def createPipelineRole(iam, domainName):
    """Creates the pipeline role"""
    response = iam.create_policy(
        PolicyName='pipeline-policy',
        PolicyDocument=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect\":
\"Allow\",\"Action\":\"es:DescribeDomain\",\"Resource\":\"arn:aws:es:us-east-1:account-
id:domain\/{domainName}\"}},{{\"Effect\":\"Allow\",\"Action\":\"es:ESHttp*\",\"Resource
\":\"arn:aws:es:us-east-1:account-id:domain\/{domainName}\/*\"}}]}}'
    )
    policyarn = response['Policy']['Arn']

    response = iam.create_role(
        RoleName='PipelineRole',
        AssumeRolePolicyDocument='{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"Service\":\"osis-pipelines.amazonaws.com\"},\"Action\":
\"sts:AssumeRole\"}]}'
    )
    rolename=response['Role']['RoleName']

    response = iam.attach_role_policy(
        RoleName=rolename,
        PolicyArn=policyarn
    )

    print('Creating pipeline role...')
    time.sleep(10)
    print('Role created: ' + rolename)

def createDomain(opensearch, domainName):
    """Creates a domain to ingest data into"""
    response = opensearch.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_2.3',
        ClusterConfig={
            'InstanceType': 't2.small.search',
```

```
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect
\":\"Allow\",\"Principal\":{{\"AWS\":\"arn:aws:iam::account-id:role\/PipelineRole
\"}},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:account-id:domain\/
{domainName}\/*\"}}]}}',
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    return(response)

def waitForDomainProcessing(opensearch, domainName):
    """Waits for the domain to be active"""
    try:
        response = opensearch.describe_domain(
            DomainName=domainName
        )
        # Every 30 seconds, check whether the domain is processing.
        while 'Endpoint' not in response['DomainStatus']:
            print('Creating domain...')
            time.sleep(60)
            response = opensearch.describe_domain(
                DomainName=domainName)

        # Once we exit the loop, the domain is ready for ingestion.
        endpoint = response['DomainStatus']['Endpoint']
        print('Domain endpoint ready to receive data: ' + endpoint)
        createPipeline(osis, endpoint)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found.')
        else:
            raise error
```

```
def createPipeline(osis, endpoint):
    """Creates a pipeline using the domain and pipeline role"""
    try:
        definition = f'version: \"2\"\nlog-pipeline:\n  source:\n    http:\n      path:
 \"/${{pipelineName}}/logs\"\n  processor:\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://{endpoint}\" ]\n        index: \"application_logs\"\n        aws:\n
   region: \"us-east-1\"'
        response = osis.create_pipeline(
            PipelineName=pipelineName,
            MinUnits=4,
            MaxUnits=9,
            PipelineConfigurationBody=definition,
            PipelineRoleArn="arn:aws:iam::account-id:role/PipelineRole"
        )

        response = osis.get_pipeline(
                PipelineName=pipelineName
        )

        # Every 30 seconds, check whether the pipeline is active.
        while response['Pipeline']['Status'] == 'CREATING':
            print('Creating pipeline...')
            time.sleep(30)
            response = osis.get_pipeline(
                PipelineName=pipelineName)

        # Once we exit the loop, the pipeline is ready for ingestion.
        ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
        print('Pipeline ready to ingest data at endpoint: ' + ingestionEndpoint)
        ingestData(ingestionEndpoint)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceAlreadyExistsException':
            print('Pipeline already exists.')
            response = osis.get_pipeline(
                PipelineName=pipelineName
            )
            ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
            ingestData(ingestionEndpoint)
        else:
            raise error
```

```
def ingestData(ingestionEndpoint):
    """Ingests a sample log file into the pipeline"""
    endpoint = 'https://' + ingestionEndpoint
    r = requests.request('POST', f'{endpoint}/log-pipeline/logs',

 data='[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","requ
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]',
    auth=AWSSigV4('osis'))
    print('Ingesting sample log file into pipeline')
    print('Response: ' + r.text)

def main():
    createPipelineRole(iam, domainName)
    createDomain(opensearch, domainName)
    waitForDomainProcessing(opensearch, domainName)

if __name__ == "__main__":
    main()
```

Segurança na OpenSearch ingestão da Amazon

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isto como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos programas de conformidade da AWS.
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Essa documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o OpenSearch Inestion. Os tópicos a seguir mostram como configurar

a OpenSearch ingestão para atender aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos OpenSearch de ingestão.

Tópicos

- Configurando o acesso à VPC para pipelines de ingestão da Amazon OpenSearch
- Configurando pipelines OpenSearch de ingestão para ingestão entre contas
- Identity and Access Management para OpenSearch ingestão na Amazon
- Registro de chamadas da Amazon OpenSearch Ingestion API usando AWS CloudTrail
- API OpenSearch de ingestão e endpoints de interface da Amazon ()AWS PrivateLink

Configurando o acesso à VPC para pipelines de ingestão da Amazon OpenSearch

Você pode acessar seus pipelines de OpenSearch ingestão da Amazon usando uma interface VPC endpoint. Uma VPC é uma rede virtual dedicada à sua. Conta da AWSÉ logicamente isolado de outras redes virtuais na AWS nuvem. O acesso a um pipeline por meio de um VPC endpoint permite a comunicação segura entre o OpenSearch Inestion e outros serviços dentro da VPC sem a necessidade de um gateway de internet, dispositivo NAT ou conexão VPN. Todo o tráfego permanece seguro na nuvem. AWS

OpenSearch A ingestão estabelece essa conexão privada criando um endpoint de interface, alimentado por. AWS PrivateLink Criamos uma interface de rede do endpoint em cada sub-rede que você especificar durante a criação do pipeline. Essas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado ao pipeline de ingestão. OpenSearch Você também pode optar por criar e gerenciar os endpoints da interface por conta própria.

O uso de uma VPC permite que você imponha o fluxo de dados por meio de seus pipelines de OpenSearch ingestão dentro dos limites da VPC, e não pela Internet pública. Pipelines que não estão em uma VPC enviam e recebem dados por endpoints públicos e pela Internet.

Um pipeline com acesso à VPC pode gravar em domínios públicos ou de OpenSearch serviços de VPC e em coleções públicas ou sem servidor de VPC. OpenSearch

Tópicos

- Considerações
- Limitações
- Pré-requisitos
- Como configurar o acesso à VPC para um pipeline
- Endpoints da VPC autogerenciados
- Função vinculada ao serviço para acesso à VPC

Considerações

Considere o seguinte ao configurar o acesso à VPC para um pipeline.

- Um pipeline não precisa estar na mesma VPC que seu coletor. Você também não precisa estabelecer uma conexão entre os dois VPCs. OpenSearch A ingestão se encarrega de conectá-los para você.
- Você só pode especificar uma VPC para o pipeline.
- Ao contrário dos pipelines públicos, um pipeline de VPC deve estar na mesma Região da AWS do domínio ou coletor de coleções em que está gravando. Você pode configurar uma fonte do S3 para o pipeline a fim de gravar entre regiões.
- Você pode optar por implantar um pipeline em uma, duas ou três sub-redes da VPC. As sub-redes são distribuídas nas mesmas zonas de disponibilidade nas quais suas unidades de OpenSearch computação de ingestão (OCUs) estão implantadas.
- Se você implantar apenas um pipeline em uma sub-rede e a Zona de disponibilidade ficar inativa, você não conseguirá ingerir dados. Para garantir a alta disponibilidade, recomendamos que você configure pipelines com duas ou três sub-redes.
- A especificação de um grupo de segurança é opcional. Se você não fornecer um grupo de segurança, o OpenSearch Inestion usará o grupo de segurança padrão especificado na VPC.

Limitações

Pipelines com acesso a uma VPC têm as seguintes limitações.

- Não é possível alterar a configuração de rede de um pipeline depois de criá-la. Se você iniciar um pipeline em uma VPC, não poderá alterá-lo posteriormente para um endpoint público e vice-versa.
- Você pode iniciar o pipeline com endpoint da VPC de interface ou um endpoint público, mas não pode fazer ambos. Você deve escolher uma opção ou outra ao criar um pipeline.

- Após provisionar um pipeline com acesso a uma VPC, não será possível movê-lo para uma VPC diferente e você não pode mudar as sub-redes e as configurações do grupo de segurança.
- Se seu pipeline grava em um domínio ou em um coletor de coleções que utiliza acesso à VPC, você não pode voltar mais tarde e alterar o coletor (VPC ou público) após a criação do pipeline. Você deve excluir e recriar o pipeline com um novo coletor. Você ainda pode mudar de um coletor público para um coletor com acesso à VPC.
- Você não pode fornecer acesso de ingestão entre contas aos pipelines de VPC.

Pré-requisitos

Antes de poder provisionar um pipeline com acesso à VPC, você deve fazer o seguinte:

- Criar uma VPC

  Para criar sua VPC, você pode usar o console Amazon VPC, a AWS CLI ou uma das. AWS SDKs Para obter mais informações, consulte Trabalhando com VPCs no Guia do usuário da Amazon VPC. Se você já tiver uma VPC, ignore esta etapa.
- Reservar endereços IP

  OpenSearch A ingestão coloca uma interface de rede elástica em cada sub-rede que você especifica durante a criação do pipeline. Cada interface de rede está associada a um endereço IP. Você deve reservar um endereço IP por sub-rede para as interfaces de rede.

Como configurar o acesso à VPC para um pipeline

Você pode habilitar o acesso à VPC para um pipeline dentro do console OpenSearch de serviço ou usando o. AWS CLI

Console

Você configura o acesso à VPC durante a criação do pipeline. Em Opções de rede de origem, escolha Acesso à VPC e defina as seguintes configurações:

Configuração	Description
Gerenciamento de endpoints	Escolha se você mesmo quer criar seus VPC endpoints ou deixar que o OpenSearch Ingestion os crie para você.

Configuração	Description
VPC	Escolha o ID da nuvem privada virtual (VPC) que deseja usar. A VPC e o pipeline devem estar na mesma Região da AWS.
Sub-redes	Escolha uma ou mais sub-redes. OpenSearch O serviço coloca um endpoint VPC e interfaces de rede elástica nas sub-redes.
Grupos de segurança	Escolha um ou mais grupos de segurança de VPC que permitam que o aplicativo necessário alcance o pipeline de OpenSearch ingestão nas portas (80 ou 443) e protocolos (HTTP ou HTTPs) expostos pelo pipeline.
Opções de anexo de VPC	Se sua fonte exigir comunicação entre VPCs, como Amazon DocumentDB, OpenSearch autogerenciado ou Confluent Kafka OpenSearch , o Ingestion cria interfaces de rede elásticas ENIs () nas sub-redes que você especifica para se conectar a essas fontes. OpenSearch A ingestão é usada ENIs em cada zona de disponibilidade para alcançar as fontes especificadas. A opção Anexar à VPC conecta a VPC do plano de dados de OpenSearch ingestão à sua VPC especificada. Selecione uma reserva CIDR para a VPC gerenciada para implantar a interface de rede.

CLI

Para configurar o acesso à VPC usando o AWS CLI, especifique o `--vpc-options` parâmetro:

```
aws osis create-pipeline \
  --pipeline-name vpc-pipeline \
  --min-units 4 \
  --max-units 10 \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml"
```

Endpoints da VPC autogerenciados

Ao criar um pipeline, você pode usar o gerenciamento de endpoints para criar um pipeline com endpoints autogerenciados ou endpoints gerenciados por serviços. O gerenciamento de endpoints é opcional e o padrão é endpoints gerenciados pelo Ingestion. OpenSearch

Para criar um pipeline com um VPC endpoint autogerenciado no, consulte Criação de pipelines com Console de gerenciamento da AWS o console de serviço. OpenSearch Para criar um pipeline com um VPC endpoint autogerenciado no, você pode usar AWS CLI o parâmetro `--vpc-options` no comando create-pipeline:

```
--vpc-options SubnetIds=subnet-abcdef01234567890,VpcEndpointManagement=CUSTOMER
```

Você mesmo pode criar um endpoint em seu pipeline ao especificar o serviço de endpoint. Para encontrar seu serviço de endpoint, use o comando get-pipeline, que retorna uma resposta semelhante à seguinte:

```
"vpcEndpointService" : "com.amazonaws.osis.us-east-1.pipeline-
id-1234567890abcdef1234567890",
"vpcEndpoints" : [
  {
    "vpcId" : "vpc-1234567890abcdef0",
    "vpcOptions" : {
      "subnetIds" : [ "subnet-abcdef01234567890", "subnet-021345abcdef6789" ],
      "vpcEndpointManagement" : "CUSTOMER"
    }
  }
```

Use o `vpcEndpointService` from the response para criar um VPC endpoint com o ou. Console de gerenciamento da AWS AWS CLI

Se você usa endpoints da VPC autogerenciados, habilite os atributos de DNS `enableDnsSupport` e `enableDnsHostnames` em sua VPC. Observe que, se você tiver um pipeline com um endpoint autogerenciado que você interrompe e reinicia, deverá recriar o endpoint da VPC em sua conta.

Função vinculada ao serviço para acesso à VPC

Uma função vinculada ao serviço é um tipo exclusivo de função do IAM que delega permissões para um serviço de forma que ele possa criar e gerenciar recursos em seu nome. Se você escolher um

VPC endpoint gerenciado por serviços, o OpenSearch Ingestion exigirá uma função vinculada ao serviço chamada AWSServiceRoleForAmazonOpenSearchIngestionServicepara acessar sua VPC, criar o endpoint de pipeline e colocar interfaces de rede em uma sub-rede da sua VPC.

Se você escolher um VPC endpoint autogerenciado OpenSearch , a ingestão exigirá uma função vinculada ao serviço chamada. AWSServiceRoleForOpensearchIngestionSelfManagedVpce Para saber mais sobre essas funções, suas permissões e como excluí-las, consulte the section called "Perfil de criação de pipeline".

OpenSearch A ingestão cria automaticamente a função quando você cria um pipeline de ingestão. Para que essa criação automática seja bem-sucedida, o usuário que cria o primeiro pipeline em uma conta precisa ter permissões para a ação `iam:CreateServiceLinkedRole`. Para saber mais, consulte Permissões de funções vinculadas ao serviço no Manual do usuário do IAM. Você pode ver a função no console AWS Identity and Access Management (IAM) depois que ela for criada.

Configurando pipelines OpenSearch de ingestão para ingestão entre contas

Para fontes baseadas em push, como HTTP e OTel, o Amazon OpenSearch Ingestion permite que você compartilhe pipelines entre Contas da AWS uma nuvem privada virtual (VPC) e um endpoint de pipeline em uma VPC separada. As equipes que compartilham analytics com outras equipes da organização usam esse atributo como um meio mais otimizado de, por exemplo, compartilhar analytics de logs.

Esta seção usa a seguinte terminologia:

- Proprietário do pipeline — A conta que possui e gerencia o pipeline OpenSearch de ingestão. Apenas uma conta pode ser a proprietária de um pipeline.
- Conta de conexão: uma conta que se conecta e usa um pipeline compartilhado. Várias contas podem se conectar ao mesmo pipeline.

Para configurar VPCs para compartilhar pipelines de OpenSearch ingestão Contas da AWS, conclua as seguintes tarefas, conforme descrito aqui:

- (Proprietária do pipeline) Conceder às contas conectadas acesso a um pipeline
- (Conta de conexão) Criar uma conexão de endpoint de pipeline para cada VPC conectada

Antes de começar

Antes de configurar VPCs para compartilhar pipelines de OpenSearch ingestão Contas da AWS, conclua as seguintes tarefas:

Tarefa	Detalhes
Crie um ou mais pipelines OpenSearch de ingestão	Defina as unidades OpenSearch computaci onais mínimas (OSUs) para 2 ou mais. Para obter mais informações, consulte Criação de pipelines OpenSearch de ingestão da Amazon. Para obter informações sobre a atualização de um pipeline, consulte Atualização dos pipelines OpenSearch de ingestão da Amazon.
Crie um ou mais VPCs para OpenSearch ingestão	Para permitir o compartilhamento de pipelines entre contas, qualquer VPC envolvida para o pipeline e os endpoints do pipeline deve ser configurada com os seguintes valores de DNS: • `enableDnsSupport=true` • `enableDnsHostnames=true` Para acessar mais informações, consulte Atributos de DNS para sua VPC no Guia do usuário da Amazon VPC.

Conceder às contas conectadas acesso a um pipeline

Os procedimentos nesta seção descrevem como usar o console OpenSearch de serviços e configurar o acesso AWS CLI ao pipeline entre contas criando uma política de recursos. Uma política de recurso permite que o proprietário do pipeline especifique outras contas que podem acessar um pipeline. Uma vez criadas, as políticas do pipeline existirão enquanto ele existir ou até que a política seja excluída.

> **Note**
>
> As políticas de recursos não substituem a autorização OpenSearch de ingestão padrão usando permissões do IAM. As políticas de recursos são um mecanismo de autorização adicional para permitir o acesso ao pipeline entre contas.

Tópicos

- Conceder às contas de conexão acesso a um pipeline (console)
- Conceder às contas de conexão acesso a um pipeline (CLI)

Conceder às contas de conexão acesso a um pipeline (console)

Use o procedimento a seguir para conceder às contas conectadas acesso a um pipeline usando o console do Amazon OpenSearch Service.

Para criar uma conexão de endpoint do pipeline

1. No console do Amazon OpenSearch Service, no painel de navegação, expanda Ingestão e selecione Pipelines.
2. Na seção Pipelines, escolha o nome de um pipeline ao qual você deseja conceder acesso para uma conta de conexão.
3. Escolha a guia Endpoints da VPC.
4. Na seção Entidade principais autorizadas, escolha Autorizar conta.
5. No campo ID da Conta da AWS , insira o número de 12 dígitos do ID da conta e selecione Autorizar.

Conceder às contas de conexão acesso a um pipeline (CLI)

Use o procedimento a seguir para conceder às contas de conexão acesso a um pipeline usando a AWS CLI.

Para conceder às contas de conexão acesso a um pipeline

1. Atualize para a versão mais recente do AWS CLI (versão 2.0). Para saber mais, consulte Instalar ou atualizar para a versão mais recente da AWS CLI.

2. Abra a CLI na conta e Região da AWS com o pipeline que você deseja compartilhar.
3. Execute o comando a seguir para criar uma política de recurso para o pipeline. Essa política dá a permissão de `osis:CreatePipelineEndpoint` no pipeline. A política inclui um parâmetro no qual você pode listar Conta da AWS IDs para permitir.

> **Note**
>
> No comando a seguir, você deve usar a forma abreviada do ID da conta fornecendo apenas o ID da conta de doze dígitos. Usar um ARN não funcionará. Você também deve fornecer o nome do recurso da Amazon (ARN) do pipeline no parâmetro `resource-arn` da CLI e na política JSON em `Resource`, como mostrado.

```
aws --region region osis put-resource-policy \
  --resource-arn arn:aws:osis:region:pipeline-owner-account-ID:pipeline/pipeline-name
  --policy 'IAM-policy'
```

Use uma política como a seguinte para *IAM-policy*

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
  {
  "Sid": "AllowAccess",
  "Effect": "Allow",
  "Principal": {
  "AWS": [
  "111122223333",
  "444455556666"
  ]
  },
  "Action":
  "osis:CreatePipelineEndpoint",
  "Resource": "arn:aws:osis:us-east-1:123456789012:pipeline/pipeline-name"
  }
  ]
```

```
}
```

Criar uma conexão de endpoint de pipeline para cada VPC conectada

Depois que o proprietário do pipeline concede acesso a um pipeline da VPC usando o procedimento anterior, um usuário na conta de conexão cria um endpoint do pipeline na VPC. Esta seção inclui procedimentos para criar endpoints usando o console OpenSearch de serviço e o. AWS CLI Quando você cria um endpoint, o OpenSearch Inestion executa as seguintes ações:

- Cria a função AWSServiceRoleForAmazonOpenSearchIngestionServicevinculada ao serviço em sua conta, se ela ainda não existir. Essa função dá ao usuário na conta conectada permissão para chamar a ação CreatePipelineEndpointda API.
- Cria o endpoint do pipeline.
- Configura o endpoint do pipeline para ingerir dados do pipeline compartilhado na VPC do proprietário do pipeline.

Tópicos

- Criar uma conexão de endpoint de pipeline (console)
- Criar uma conexão de endpoint do pipeline (CLI)

Criar uma conexão de endpoint de pipeline (console)

Use o procedimento a seguir para criar uma conexão de endpoint de pipeline usando o console OpenSearch de serviço.

Para criar uma conexão de endpoint do pipeline

1. No console do Amazon OpenSearch Service, no painel de navegação, expanda Ingestão e selecione VPC endpoints.
2. Na página Endpoints da VPC, escolha Criar.
3. Em Localização do pipeline, escolha uma opção. Se escolher Conta corrente, selecione o pipeline na lista. Se escolher Entre contas, especifique o ARN do pipeline no campo. O proprietário do pipeline deve ter concedido acesso ao pipeline, como descrito em Conceder às contas conectadas acesso a um pipeline.
4. Na seção Configurações da VPC, em VPC, escolha uma VPC na lista.

5. Para Subnets, escolha uma sub-rede.
6. Em Grupos de segurança, escolha um grupo.
7. Escolha Criar endpoint.

Aguarde o status do endpoint que você criou passar a ser `ACTIVE`. Quando o pipeline estiver `ACTIVE`, você verá um novo campo denominado `ingestEndpointUrl`. Use esse endpoint para acessar o pipeline e ingerir dados usando um cliente como. FluentBit Para obter mais informações sobre como usar FluentBit para ingerir dados, consulteUsando um pipeline de OpenSearch ingestão com o FluentBit.

> **Note**
>
> A `ingestEndpointUrl` é a mesma URL de todas as contas de conexão.

Criar uma conexão de endpoint do pipeline (CLI)

Use o procedimento a seguir para criar uma conexão de endpoint do pipeline usando a AWS CLI.

Para criar uma conexão de endpoint do pipeline

1. Se ainda não o fez, atualize para a versão mais recente do AWS CLI (versão 2.0). Para saber mais, consulte Instalar ou atualizar para a versão mais recente da AWS CLI.
2. Abra a CLI na conta de conexão Região da AWS com o pipeline compartilhado.
3. Execute o comando a seguir para criar um endpoint do pipeline.

> **Note**
>
> Você deve fornecer pelo menos uma sub-rede e um grupo de segurança para a VPC da conta de conexão. O grupo de segurança deve incluir a porta 443 e ser compatível com os clientes na VPC da conta de conexão.

```
aws osis --region region create-pipeline-endpoint \
  --pipeline-arn arn:aws:osis:region:connecting-account-ID:pipeline/shared-
pipeline-name
```

```
  --vpc-options SecurityGroupIds={sg-security-group-ID-1,sg-security-group-
ID-2},SubnetIds=subnet-subnet-ID
```

4. Execute o comando a seguir para listar os endpoints na região especificada no comando anterior:

```
aws osis --region region list-pipeline-endpoints
```

Aguarde o status do endpoint que você criou passar a ser `ACTIVE`. Quando o pipeline estiver `ACTIVE`, você verá um novo campo denominado `ingestEndpointUrl`. Use esse endpoint para acessar o pipeline e ingerir dados usando um cliente como. FluentBit Para obter mais informações sobre como usar FluentBit para ingerir dados, consulteUsando um pipeline de OpenSearch ingestão com o FluentBit.

> **Note**
>
> A `ingestEndpointUrl` é a mesma URL de todas as contas de conexão.

Remover endpoints do pipeline

Se você não quiser mais fornecer acesso a um pipeline compartilhado, poderá remover um endpoint do pipeline usando um dos seguintes métodos:

- Exclua o endpoint do pipeline (conta de conexão).
- Revogue o endpoint do pipeline (proprietário do pipeline).

Use o procedimento a seguir para excluir um endpoint do pipeline de uma conta de conexão.

Para excluir um endpoint do pipeline (conta de conexão)

1. Abra a CLI na conta de conexão Região da AWS com o pipeline compartilhado.
2. Execute o seguinte comando para listar os endpoints da região:

```
aws osis --region region list-pipeline-endpoints
```

   Anote o ID do pipeline que você deseja excluir.

3. Execute o seguinte comando para excluir o endpoint do pipeline:

```
aws osis --region region delete-pipeline-endpoint \
  --endpoint-id 'ID'
```

Como proprietário do pipeline compartilhado, use o procedimento a seguir para revogar um endpoint do pipeline.

Para revogue um endpoint do pipeline (proprietário do pipeline)

1. Abra a CLI na conta de conexão Região da AWS com o pipeline compartilhado.
2. Execute o comando a seguir para listar as conexões de endpoint do pipeline:

```
aws osis --region region list-pipeline-endpoint-connections
```

   Anote o ID do pipeline que você deseja excluir.
3. Execute o seguinte comando para excluir o endpoint do pipeline:

```
aws osis --region region revoke-pipeline-endpoint-connections \
  --pipeline-arn pipeline-arn --endpoint-ids ID
```

   O comando é compatível com a especificação de apenas um ID de endpoint.

Identity and Access Management para OpenSearch ingestão na Amazon

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar os recursos de OpenSearch ingestão. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- Políticas baseadas em identidade para ingestão OpenSearch
- Ações políticas para OpenSearch ingestão
- Recursos de políticas para OpenSearch ingestão
- Chaves de condição da política para Amazon OpenSearch Inestion

- ABAC com ingestão OpenSearch
- Usando credenciais temporárias com OpenSearch o Ingestion
- Funções vinculadas ao serviço para ingestão OpenSearch
- Exemplos de políticas baseadas em identidade para ingestão OpenSearch

Políticas baseadas em identidade para ingestão OpenSearch

Compatível com políticas baseadas em identidade: sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que podem ser anexados a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte Referência de elemento de política JSON do IAM no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade para ingestão OpenSearch

Para ver exemplos de políticas baseadas em identidade de OpenSearch ingestão, consulte. the section called “Exemplos de políticas baseadas em identidade”

Ações políticas para OpenSearch ingestão

Compatível com ações de políticas: sim

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

As ações de política no OpenSearch Ingestion usam o seguinte prefixo antes da ação:

```
osis
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [
      "osis:action1",
      "osis:action2"
         ]
```

É possível especificar várias ações usando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

```
"Action": "osis:List*"
```

Para ver exemplos de políticas baseadas em identidade de OpenSearch ingestão, consulte. Exemplos de políticas baseadas em identidade para Serverless OpenSearch

Recursos de políticas para OpenSearch ingestão

Compatível com recursos de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Para ações que não oferecem compatibilidade com permissões em nível de recurso, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Chaves de condição da política para Amazon OpenSearch Inestion

Compatível com chaves de condição de política específicas de serviço: não

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` especifica quando as instruções são executadas com base em critérios definidos. É possível criar expressões condicionais que usem agentes de condição, como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Para ver uma lista das chaves de condição de OpenSearch ingestão, consulte Chaves de condição para OpenSearch ingestão da Amazon na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte Ações definidas pela Amazon OpenSearch Ingestion.

ABAC com ingestão OpenSearch

Compatível com ABAC (tags em políticas): sim

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos chamados de tags. Você pode anexar tags a entidades e AWS recursos do IAM e, em seguida, criar políticas ABAC para permitir operações quando a tag do diretor corresponder à tag no recurso.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no elemento de condição de uma política usando as `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou chaves de condição `aws:TagKeys`.

Se um serviço for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para saber mais sobre o ABAC, consulte Definir permissões com autorização do ABAC no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar controle de acesso por atributo (ABAC) no Guia do usuário do IAM.

Para obter mais informações sobre a marcação de recursos OpenSearch de ingestão, consulte. the section called “Uso de tags com pipelines”

Usando credenciais temporárias com OpenSearch o Ingestion

Compatível com credenciais temporárias: sim

As credenciais temporárias fornecem acesso de curto prazo aos AWS recursos e são criadas automaticamente quando você usa a federação ou troca de funções. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para ter mais informações, consulte Credenciais de segurança temporárias no IAM e Serviços da Serviços da AWS que funcionam com o IAM no Guia do usuário do IAM.

Funções vinculadas ao serviço para ingestão OpenSearch

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir o perfil de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.

OpenSearch A ingestão usa uma função vinculada ao serviço chamada. `AWSServiceRoleForAmazonOpenSearchIngestionService` A função vinculada ao serviço chamada também `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` está disponível para pipelines com endpoints da VPC autogerenciados. Para obter detalhes sobre como criar e gerenciar funções vinculadas ao serviço de OpenSearch ingestão, consulte. the section called "Perfil de criação de pipeline"

Exemplos de políticas baseadas em identidade para ingestão OpenSearch

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos OpenSearch de ingestão. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos pela Amazon OpenSearch Ingestion, incluindo o formato de cada um dos tipos de recursos, consulte Ações, recursos e chaves de condição para Amazon OpenSearch Ingestion na Referência de Autorização de Serviço. ARNs

Tópicos

- Melhores práticas de políticas
- Usando o OpenSearch Inestion no console
- Administrando pipelines de OpenSearch ingestão
- Ingestão de dados em um pipeline de OpenSearch ingestão

Melhores práticas de políticas

As políticas baseadas em identidade são muito poderosas. Eles determinam se alguém pode criar, acessar ou excluir recursos OpenSearch de ingestão em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos de OpenSearch ingestão em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para saber mais, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para funções de trabalho no Guia do usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para saber mais sobre como usar o IAM para aplicar permissões, consulte Políticas e permissões no IAM no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: é possível adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para saber mais, consulte Elementos da política JSON do IAM: condição no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para saber mais, consulte Validação de políticas do IAM Access Analyzer no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para saber mais, consulte Configuração de acesso à API protegido por MFA no Guia do Usuário do IAM.

Para saber mais sobre as práticas recomendadas do IAM, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.

Usando o OpenSearch Inestion no console

Para acessar o OpenSearch Inestion no console OpenSearch de serviço, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os recursos de OpenSearch ingestão em sua AWS conta. Se você criar uma política baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (como perfis do IAM) com essa política.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que corresponderem a operação da API que você estiver tentando executar.

A política a seguir permite que um usuário acesse a OpenSearch Ingestão no console OpenSearch de serviço:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ]
        }
    ]
}
```

Como alternativa, você pode usar a política the section called “AmazonOpenSearchIngestionReadOnlyAccess” AWS gerenciada, que concede acesso somente leitura a todos os recursos de OpenSearch ingestão para um. Conta da AWS

Administrando pipelines de OpenSearch ingestão

Essa política é um exemplo de política de “administrador de pipeline” que permite ao usuário gerenciar e administrar pipelines de OpenSearch ingestão da Amazon. O usuário pode criar, exibir e excluir pipelines.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:osis:us-east-1:111122223333:pipeline/*",
            "Action": [
                "osis:CreatePipeline",
                "osis:DeletePipeline",
                "osis:UpdatePipeline",
                "osis:ValidatePipeline",
                "osis:StartPipeline",
                "osis:StopPipeline"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Ingestão de dados em um pipeline de OpenSearch ingestão

Esse exemplo de política permite que um usuário ou outra entidade consuma dados em um pipeline de OpenSearch ingestão da Amazon em sua conta. O usuário não pode modificar os pipelines.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:osis:us-east-1:123456789012:pipeline/*",
            "Action": [
                "osis:Ingest"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Registro de chamadas da Amazon OpenSearch Ingestion API usando AWS CloudTrail

A Amazon OpenSearch Ingestion é integrada com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço na OpenSearch Ingestão.

CloudTrail captura todas as chamadas de API para OpenSearch ingestão como eventos.
As chamadas capturadas incluem chamadas da seção OpenSearch Ingestão do console de OpenSearch serviços e chamadas de código para as operações da API OpenSearch de ingestão.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para OpenSearch ingestão. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita para o OpenSearch Inestion, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o Guia AWS CloudTrail do usuário.

OpenSearch Informações de ingestão em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre na OpenSearch ingestão, essa atividade é registrada em um CloudTrail evento junto com outros

eventos de AWS serviço no histórico de eventos. Você pode exibir, pesquisar e baixar eventos recentes em sua Conta da AWS. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em seu Conta da AWS, incluindo eventos para OpenSearch ingestão, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as Regiões da AWS.

A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para saber mais, consulte:

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações de OpenSearch ingestão são registradas CloudTrail e documentadas na referência da API OpenSearch de ingestão. Por exemplo, as chamadas para as ações `CreateCollection`, `ListCollections` e `DeleteCollection` geram entradas nos arquivos de log do CloudTrail .

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário root ou AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte Elemento userIdentity do CloudTrail .

OpenSearch Entendendo as entradas do arquivo de registro de ingestão

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log.

Um evento representa uma solicitação única de qualquer fonte. Ele inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `DeletePipeline` ação.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/test-user",
        "accountId": "123456789012",
        "accessKeyId": "access-key",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-21T16:48:33Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-04-21T16:49:22Z",
    "eventSource": "osis.amazonaws.com",
    "eventName": "UpdatePipeline",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "123.456.789.012",
```

```
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36",
    "requestParameters": {
        "pipelineName": "my-pipeline",
        "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
 http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:\n
        log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received: true
\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n"
    },
    "responseElements": {
        "pipeline": {
            "pipelineName": "my-pipeline",sourceIPAddress
            "pipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/my-pipeline",
            "minUnits": 1,
            "maxUnits": 1,
            "status": "UPDATING",
            "statusReason": {
                "description": "An update was triggered for the pipeline. It is still
 available to ingest data."
            },
            "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
    http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:
\n          log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n",
            "createdAt": "Mar 29, 2023 1:03:44 PM",
            "lastUpdatedAt": "Apr 21, 2023 9:49:21 AM",
            "ingestEndpointUrls": [
                "my-pipeline-tu33ldsgdltgv7x7tjqiudvf7m.us-west-2.osis.amazonaws.com"
            ]
        }
    },
    "requestID": "12345678-1234-1234-1234-987654321098",
    "eventID": "12345678-1234-1234-1234-987654321098",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
```

```
    "recipientAccountId": "709387180454",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
        "clientProvidedHostHeader": "osis.us-west-2.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

API OpenSearch de ingestão e endpoints de interface da Amazon ()AWS PrivateLink

Você pode estabelecer uma conexão privada entre sua VPC e os endpoints da API OpenSearch Ingestion criando uma interface VPC endpoint. Os endpoints de interface são desenvolvidos pelo AWS PrivateLink.

AWS PrivateLink permite que você acesse de forma privada as operações da API de OpenSearch ingestão sem um gateway de internet, dispositivo NAT, conexão VPN ou conexão. Direct Connect Os recursos em sua VPC não precisam de endereços IP públicos para se comunicar com os endpoints da API de OpenSearch ingestão para criar, modificar ou excluir pipelines. O tráfego entre sua VPC e o OpenSearch Inestion não sai da rede Amazon.

> **Note**
>
> Este tópico aborda os endpoints da VPC para acessar a API de OpenSearch ingestão, que permite gerenciar pipelines (criar, atualizar, excluir) de dentro da sua VPC. Isso é diferente de configurar o acesso à VPC para os próprios pipelines, que controla como os dados são ingeridos nos pipelines a partir de fontes dentro da sua VPC. Para obter informações sobre como configurar o acesso à VPC para pipelines, consulte. the section called “Como configurar o acesso à VPC para pipelines”

Cada endpoint de interface é representado por uma ou mais interfaces de rede elástica nas sub-redes. Para obter mais informações sobre interfaces de rede elástica, consulte Interfaces de rede elástica no Guia do usuário do Amazon EC2.

Para obter mais informações sobre endpoints da VPC, consulte Endpoints da VPC de interface (AWS PrivateLink) no Guia do usuário da Amazon VPC. Para obter mais informações sobre as operações da API de OpenSearch ingestão, consulte a referência da API OpenSearch de ingestão.

Considerações sobre VPC endpoints

Antes de configurar uma interface VPC endpoint para endpoints da API de OpenSearch ingestão, certifique-se de revisar as propriedades e limitações do endpoint da interface no Guia do usuário da Amazon VPC.

Todas as operações da API de OpenSearch ingestão relevantes para o gerenciamento OpenSearch de recursos de ingestão estão disponíveis em sua VPC usando. AWS PrivateLink

As políticas de endpoint de VPC são compatíveis com endpoints da API de OpenSearch ingestão. Por padrão, o acesso total às operações da API de OpenSearch ingestão é permitido por meio do endpoint. Para obter mais informações, consulte Controlar o acesso a serviços com VPC endpoints no Guia do usuário da Amazon VPC.

Disponibilidade

OpenSearch Atualmente, a API de ingestão é compatível com VPC endpoints em OpenSearch todas as regiões de ingestão.

No momento, os endpoints FIPS não são suportados.

Criação de uma interface VPC endpoint para a API de ingestão OpenSearch

Você pode criar um VPC endpoint para a API de OpenSearch ingestão usando o console Amazon VPC ou o (). AWS Command Line Interface AWS CLI Para obter mais informações, consulte Criar um endpoint de interface no Guia do usuário da Amazon VPC.

Crie um VPC endpoint para a API de OpenSearch ingestão usando o nome do serviço. `com.amazonaws.region.osis`

Se você habilitar o DNS privado para o endpoint, poderá fazer solicitações de API para OpenSearch ingestão com o VPC endpoint usando seu nome DNS padrão para a região, por exemplo. AWS `osis.us-east-1.amazonaws.com`

Para mais informações, consulte Acessar um serviço por um endpoint de interface no Guia do usuário da Amazon VPC.

Criação de uma política de VPC endpoint para a API de ingestão OpenSearch

Você pode anexar uma política de endpoint ao seu VPC endpoint que controla o acesso à OpenSearch API de ingestão. Essa política especifica as seguintes informações:

- A entidade principal que pode realizar ações.
- As ações que podem ser realizadas.
- Os recursos aos quais as ações podem ser aplicadas.

Para mais informações, consulte Controlar o acesso a serviços com VPC endpoints no Guia do usuário da Amazon VPC.

Exemplo: política de VPC endpoint para OpenSearch ações da API de ingestão

Veja a seguir um exemplo de uma política de endpoint para a API OpenSearch de ingestão. Quando anexada a um endpoint, essa política concede acesso às ações listadas da API de OpenSearch ingestão para todos os diretores em todos os recursos.

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "osis:CreatePipeline",
            "osis:UpdatePipeline",
            "osis:DeletePipeline"
         ],
         "Resource":"*"
      }
   ]
}
```

Exemplo: política de VPC endpoint que nega todo o acesso de uma conta especificada AWS

A política de VPC endpoint a seguir nega à AWS conta `123456789012` todo o acesso aos recursos que usam o endpoint. A política permite todas as ações de outras contas.

```
{
  "Statement": [
    {
```

```
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    },
    {
      "Action": "*",
      "Effect": "Deny",
      "Resource": "*",
      "Principal": { "AWS": [ "123456789012" ] }
     }
   ]
}
```

Marcação de pipelines de OpenSearch ingestão da Amazon

As tags permitem que você atribua informações arbitrárias a um pipeline de OpenSearch ingestão da Amazon para que você possa categorizar e filtrar essas informações. Uma tag é um rótulo de metadados que você atribui ou AWS atribui a um AWS recurso. Cada tag consiste em uma chave e um valor. Em tags atribuídas por você, você mesmo define a chave e o valor. Por exemplo, você pode definir a chave como `stage` e o valor de um atributo como `test`.

As tags ajudam a:

- Identifique e organize seus AWS recursos. Muitos AWS serviços oferecem suporte à marcação, então você pode atribuir a mesma tag a recursos de serviços diferentes para indicar que os recursos estão relacionados. Por exemplo, você pode atribuir a mesma tag a um pipeline OpenSearch de ingestão que você atribui a um domínio do Amazon OpenSearch Service.
- Acompanhe seus AWS custos. Você ativa essas tags no Gerenciamento de Faturamento e Custos da AWS painel. AWS usa as tags para categorizar seus custos e entregar um relatório mensal de alocação de custos para você. Para saber mais, consulte Usar tags de alocação de custos no Guia do usuário do AWS Billing.
- Restrinja o acesso aos pipelines usando controle de acesso baseado em atributos. Para saber mais, consulte Controlar o acesso baseado em chaves de tag no Guia do Usuário do IAM.

No OpenSearch Ingestion, o recurso principal é um pipeline. Você pode usar o console OpenSearch de serviço, a AWS CLI, o OpenSearch Ingestion APIs ou o AWS SDKs para adicionar, gerenciar e remover tags de um pipeline.

Tópicos

- Permissões obrigatórias
- Uso de tags (console)
- Uso de tags (AWS CLI)

Permissões obrigatórias

OpenSearch A ingestão usa as seguintes permissões AWS Identity and Access Management Access Analyzer (IAM) para marcar pipelines:

- `osis:TagResource`
- `osis:ListTagsForResource`
- `osis:UntagResource`

Para obter mais informações sobre cada permissão, consulte Ações, recursos e chaves de condição para OpenSearch ingestão na Referência de autorização de serviço.

Uso de tags (console)

O console é a maneira mais simples marcar um pipeline com tags.

Como criar uma tag

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Selecione o pipeline ao qual você deseja adicionar tags e vá para guia Tags.
3. Escolha Gerenciar e Adicionar nova tag.
4. Insira uma chave de tag e um valor opcional.
5. Escolha Salvar.

Para excluir uma tag, siga as mesmas etapas e escolha Remover na página Gerenciar tags.

Para saber mais sobre como usar o console para trabalhar com tags, consulte Editor de tags no Guia de conceitos básicos do Console de Gerenciamento da AWS .

Uso de tags (AWS CLI)

Para marcar um pipeline usando o AWS CLI, envie uma `TagResource` solicitação:

```
aws osis tag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tags Key=service,Value=osis Key=source,Value=otel
```

Remova as tags de um pipeline usando o comando `UntagResource`:

```
aws osis untag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tag-keys service
```

É possível exibir as tags existentes para um pipeline com o comando `ListTagsForResource`:

```
aws osis list-tags-for-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
```

Registrando e monitorando a OpenSearch ingestão da Amazon com a Amazon CloudWatch

O Amazon OpenSearch Ingestion publica métricas e registros na Amazon. CloudWatch

Tópicos

- Monitoramento dos logs de pipeline
- Métricas do pipeline de monitoramento

Monitoramento dos logs de pipeline

Você pode ativar o registro nos pipelines de OpenSearch ingestão da Amazon para expor mensagens de erro e aviso geradas durante as operações do pipeline e a atividade de ingestão. OpenSearch A ingestão publica todos os registros no Amazon CloudWatch Logs. CloudWatch Os registros podem monitorar as informações nos arquivos de log e notificá-lo quando determinados limites forem atingidos. É possível também arquivar seus dados de log em armazenamento resiliente. Para obter mais informações, consulte o Guia do usuário do Amazon CloudWatch Logs.

Os registros da OpenSearch ingestão podem indicar falhas no processamento de solicitações, erros de autenticação da origem até o coletor e outros avisos que podem ser úteis para a solução de problemas. Para seus registros, a OpenSearch ingestão usa os níveis de registro de `INFO`,`WARN`,`ERROR`, e. `FATAL` Recomendamos habilitar a publicação de logs para todos os pipelines.

Permissões obrigatórias

Para permitir que o OpenSearch Inestion envie registros para o CloudWatch Logs, você precisa estar conectado como um usuário com determinadas permissões do IAM.

Você precisa das seguintes permissões de CloudWatch registros para criar e atualizar os recursos de entrega de registros:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:PutResourcePolicy",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:DescribeResourcePolicies",
                "logs:GetLogDelivery",
                "logs:ListLogDeliveries"
            ]
        }
    ]
}
```

Habilitar publicação de logs

Você pode ativar a publicação de logs em pipelines existentes ou ao criar um pipeline. Para ver as etapas para habilitar a publicação de logs durante a criação do pipeline, consulte the section called “Como criar pipelines”.

Console

Para habilitar a publicação de logs em um pipeline existente

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/osis/home. Você estará na página Pipelines.
2. Abra o pipeline no qual você deseja habilitar logs e escolha Ações, Editar opções de publicação de log.
3. Ative a opção Publicar em CloudWatch registros.
4. Crie um novo grupo de logs ou selecione um existente. Recomendamos que você formate o nome como um caminho, como `/aws/vendedlogs/OpenSearchIngestion/pipeline-name/audit-logs`. Esse formato facilita a aplicação de uma política de CloudWatch acesso que concede permissões a todos os grupos de registros em um caminho específico, como`/aws/vendedlogs/OpenSearchIngestion`.

   > **Important**
   >
   > Você deve incluir o prefixo `vendedlogs` no nome do grupo de logs, caso contrário, a criação falhará.

5. Escolha Salvar.

CLI

Para habilitar a publicação de registros usando o AWS CLI, envie a seguinte solicitação:

```
aws osis update-pipeline \
  --pipeline-name my-pipeline \
  --log-publishing-options  IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="/aws/vendedlogs/OpenSearchIngestion/pipeline-name"}
```

Métricas do pipeline de monitoramento

Você pode monitorar os pipelines OpenSearch de ingestão da Amazon usando a Amazon CloudWatch, que coleta dados brutos e os processa em métricas legíveis, quase em tempo real. Essas estatísticas são mantidas por 15 meses, de maneira que você possa acessar informações históricas e ter uma perspectiva melhor de como o aplicativo web ou o serviço está se saindo. Você também pode definir alarmes que observam determinados limites e enviam notificações ou realizam

ações quando esses limites são atingidos. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.

O console de OpenSearch ingestão exibe uma série de gráficos com base nos dados brutos da CloudWatch guia Desempenho de cada pipeline.

OpenSearch Métricas de relatórios de ingestão da maioria dos plug-ins compatíveis. Se certos plug-ins não tiverem sua própria tabela abaixo, isso significa que eles não tiveram nenhuma métrica específica do plug-in reportada. As métricas de pipeline estão publicadas no namespace `AWS/OSIS`.

Tópicos

- Métricas comuns
- Métricas do buffer
- Métricas do Signature V4
- Métricas de buffer de bloqueio limitado
- Métricas da fonte de rastreamento OTel
- Métricas do OTel: métricas de origem
- Métricas HTTP
- Métricas do S3
- Indicadores agregados
- Métricas de data
- métricas do Lambda
- Métricas do Grok
- Métricas brutas do OTel trace
- Métricas de grupo de monitoramento do OTel
- Métricas do mapa de serviço
- OpenSearch métricas
- Métricas do sistema e de medição

Métricas comuns

As métricas a seguir são comuns a todos os processadores e coletores.

Cada métrica é prefixada pelo nome do subpipeline e pelo nome do plug-in, no formato <*sub_pipeline_name*><><>*plugin*.*metric_name* Por exemplo, o nome completo da métrica `recordsIn.count` de um subpipeline chamado `my-pipeline` e o processador date seriam `my-pipeline.date.recordsIn.count`.

Sufixo métrico	Description
`recordsIn.count`	A entrada de registros em um componente do pipeline. Essa métrica se aplica a processadores e coletores. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsOut.count`	A saída de registros em um componente do pipeline. Essa métrica se aplica a processadores e origens. Estatísticas relevantes: soma Dimensão:`PipelineName`
`timeElapsed.count`	Uma contagem de pontos de dados registrados durante a execução de um componente do pipeline. Essa métrica se aplica a processadores e coletores. Estatísticas relevantes: soma Dimensão:`PipelineName`
`timeElapsed.sum`	O tempo total decorrido durante a execução de um componente do pipeline. Essa métrica se aplica a processadores e coletores, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`timeElapsed.max`	O tempo máximo decorrido durante a execução de um componente do pipeline. Essa métrica se aplica a processadores e coletores, em milissegundos.

Sufixo métrico	Description
	Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas do buffer

As métricas a seguir se aplicam ao buffer de bloqueio limitado padrão que o OpenSearch Ingestion configura automaticamente para todos os pipelines.

Cada métrica é prefixada pelo nome do subpipeline e pelo nome do buffer, no formato **<** *sub_pipeline_name* **><** **><** **>***buffer_name*. *metric_name* Por exemplo, o nome completo da métrica `recordsWritten.count` de um subpipeline chamado `my-pipeline` seria `my-pipeline.BlockingBuffer.recordsWritten.count`.

Sufixo métrico	Description
`recordsWritten.count`	O número de registros gravados em um buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsRead.count`	O número de registros lidos de um buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsInFlight.value`	O número de registros não verificados lidos de um buffer. Estatística relevante: média Dimensão:`PipelineName`
`recordsInBuffer.value`	O número de registros atualmente em um buffer. Estatística relevante: média Dimensão:`PipelineName`

Sufixo métrico	Description
`recordsProcessed.count`	O número de registros lidos de um buffer e processados por um pipeline. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsWriteFailed.count`	O número de registros que o pipeline não conseguiu gravar no coletor. Estatísticas relevantes: soma Dimensão:`PipelineName`
`writeTimeElapsed.count`	Uma contagem de pontos de dados registrados durante a gravação em um buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`writeTimeElapsed.sum`	O tempo total decorrido durante a gravação em um buffer, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`writeTimeElapsed.max`	O tempo máximo decorrido durante a gravação em um buffer, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`writeTimeouts.count`	A contagem dos tempos limite de gravação em um buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`readTimeElapsed.count`	Uma contagem de pontos de dados registrados durante a leitura de um buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`readTimeElapsed.sum`	O tempo total decorrido durante a leitura de um buffer, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`readTimeElapsed.max`	O tempo máximo decorrido durante a leitura de um buffer, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`checkpointTimeElapsed.count`	Uma contagem de pontos de dados registrados durante o checkpoint. Estatísticas relevantes: soma Dimensão:`PipelineName`
`checkpointTimeElapsed.sum`	O tempo total decorrido durante o checkpoint, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`checkpointTimeElapsed.max`	O tempo máximo decorrido durante o checkpoint, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas do Signature V4

As métricas a seguir se aplicam ao endpoint de ingestão de um pipeline e estão associadas aos plug-ins de origem (`http`, `otel_trace` e `otel_metrics`). Todas as solicitações para o endpoint de ingestão devem ser assinadas usando o Signature Version 4. Essas métricas podem ajudar você a identificar problemas de autorização ao se conectar ao seu pipeline ou confirmar que você está autenticando com sucesso.

Cada métrica é prefixada pelo nome do subpipeline e `osis_sigv4_auth`. Por exemplo, *`.sub_pipeline_name`*`.osis_sigv4_auth.httpAuthSuccess.count`

Sufixo métrico	Description
`httpAuthSuccess.count`	O número de solicitações bem-sucedidas do Signature V4 para o pipeline. Estatísticas relevantes: soma Dimensão:`PipelineName`
`httpAuthFailure.count`	O número de solicitações do Signature V4 que falharam no pipeline. Estatísticas relevantes: soma Dimensão:`PipelineName`
`httpAuthServerError.count`	O número de solicitações do Signature V4 ao pipeline que retornaram erros do servidor. Estatísticas relevantes: soma

Sufixo métrico	Description
	Dimensão:`PipelineName`

Métricas de buffer de bloqueio limitado

As métricas a seguir se aplicam ao buffer de bloqueio limitado. Cada métrica é prefixada pelo nome do subpipeline e `BlockingBuffer`. Por exemplo, *`.sub_pipeline_name`*`.BlockingBuffer.bufferUsage.value`

Sufixo métrico	Description
`bufferUsage.value`	Porcentagem de uso do `buffer_size` com base no número de registros no buffer. `buffer_size` represent a o número máximo de registros gravados no buffer, bem como registros em ação que não foram verificados. Estatística relevante: média Dimensão:`PipelineName`

Métricas da fonte de rastreamento OTel

As métricas a seguir se aplicam à fonte de OTel rastreamento. Cada métrica é prefixada pelo nome do subpipeline e `otel_trace_source`. Por exemplo, *`.sub_pipeline_name`*`.otel_trace_source.requestTimeouts.count`

Sufixo métrico	Description
`requestTimeouts.count`	O número de solicitações que atingiram o tempo limite. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestsReceived.count`	O número de solicitações recebidas pelo plug-in. Estatísticas relevantes: soma

Sufixo métrico	Description
	Dimensão:`PipelineName`
`successRequests.count`	O número de solicitações que foram processadas com êxito pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`badRequests.count`	O número de solicitações com um formato inválido que foram processadas pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestsTooLarge.count`	O número de solicitações cujas extensões no conteúdo são maiores do que a capacidade do buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`internalServerError.count`	O número de solicitações processadas pelo plug-in com um tipo de exceção personalizado. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.count`	Uma contagem de pontos de dados registrados durante o processamento de solicitações pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`requestProcessDuration.sum`	A latência total das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.max`	A latência máxima das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`payloadSize.count`	Uma contagem da distribuição dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.sum`	A distribuição total dos tamanhos da carga útil das solicitações recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.max`	A distribuição máxima dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas do OTel: métricas de origem

As métricas a seguir se aplicam à fonte de OTel métricas. Cada métrica é prefixada pelo nome do subpipeline e `otel_metrics_source`. Por exemplo, *`.sub_pipeline_name`*`.otel_metrics_source.requestTimeouts.count`

Sufixo métrico	Description
`requestTimeouts.count`	O número total de solicitações do plug-in que expiraram. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestsReceived.count`	O número total de solicitações recebidas pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`successRequests.count`	O número de solicitações processadas com sucesso (código de status de 200 respostas) pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.count`	Uma contagem da latência das solicitações processadas pelo plug-in, em segundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.sum`	A latência total das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`requestProcessDuration.max`	A latência máxima das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`payloadSize.count`	Uma contagem da distribuição dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.sum`	A distribuição total dos tamanhos da carga útil das solicitações recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.max`	A distribuição máxima dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas HTTP

As métricas a seguir se aplicam à fonte HTTP. Cada métrica é prefixada pelo nome do subpipeline e `http`. Por exemplo, *`.sub_pipeline_name`*`.http.requestsReceived.count`

Sufixo métrico	Description
`requestsReceived.count`	O número de solicitações recebido pelo endpoint do `/log/ingest` .

Sufixo métrico	Description
	Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestsRejected.count`	O número de solicitações rejeitadas pelo plug-in (código de status de resposta 429). Estatísticas relevantes: soma Dimensão:`PipelineName`
`successRequests.count`	O número de solicitações processadas com sucesso (código de status de 200 respostas) pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`badRequests.count`	O número de solicitações com tipo ou formato de conteúdo inválido processadas pelo plug-in (código de status de 400 respostas). Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestTimeouts.count`	O número de solicitações que atingem o tempo limite no servidor de origem HTTP (código de status de resposta 415). Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`requestsTooLarge.count`	O número de solicitações cujo tamanho dos eventos no conteúdo é maior que a capacidade do buffer (código de status de resposta 413). Estatísticas relevantes: soma Dimensão:`PipelineName`
`internalServerError.count`	O número de solicitações processadas pelo plug-in com um tipo de exceção personalizado (código de status de 500 respostas). Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.count`	Uma contagem da latência das solicitações processadas pelo plug-in, em segundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.sum`	A latência total das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestProcessDuration.max`	A latência máxima das solicitações processadas pelo plug-in, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Sufixo métrico	Description
`payloadSize.count`	Uma contagem da distribuição dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.sum`	A distribuição total dos tamanhos da carga útil das solicitaç ões recebidas, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`payloadSize.max`	A distribuição máxima dos tamanhos de carga das solicitações recebidas, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas do S3

As métricas a seguir se aplicam à fonte do S3. Cada métrica é prefixada pelo nome do subpipeline e `s3`. Por exemplo, *`.sub_pipeline_name`*`.s3.s3ObjectsFailed.count`

Sufixo métrico	Description
`s3ObjectsFailed.count`	O número total de objetos do S3 que o plug-in não conseguiu ler. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectsNotFound.count`	O número de objetos do S3 que o plug-in não conseguiu ler devido a um erro de `Not Found` do S3. Essas métricas também contam para a métrica `s3ObjectsFailed` .

Sufixo métrico	Description
	Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectsAccessDenied.count`	O número de objetos do S3 que o plug-in não conseguiu ler devido a um erro de `Access Denied` ou `Forbidden` do S3. Essas métricas também contam para a métrica `s3ObjectsFailed` . Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectReadTimeElapsed.count`	A quantidade de tempo que o plug-in leva para realizar uma solicitação GET para um objeto do S3, analisá-lo e gravar eventos no buffer. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectReadTimeElapsed.sum`	O tempo total que o plug-in leva para realizar uma solicitação GET para um objeto do S3, analisá-lo e gravar eventos no buffer, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectReadTimeElapsed.max`	O tempo máximo que o plug-in leva para realizar uma solicitação GET para um objeto do S3, analisá-lo e gravar eventos no buffer, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Sufixo métrico	Description
`s3ObjectSizeBytes.count`	A contagem da distribuição dos tamanhos dos objetos do S3, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectSizeBytes.sum`	A distribuição total dos tamanhos dos objetos do S3, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectSizeBytes.max`	A distribuição máxima dos tamanhos de objetos do S3, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`s3ObjectProcessedBytes.count`	A contagem da distribuição dos objetos do S3 processados pelo plug-in, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectProcessedBytes.sum`	A distribuição total dos objetos do S3 processados pelo plug-in, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`s3ObjectProcessedBytes.max`	A distribuição máxima dos objetos do S3 processados pelo plug-in, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`s3ObjectsEvents.count`	A contagem da distribuição dos eventos do S3 recebidos pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectsEvents.sum`	A distribuição total dos eventos do S3 recebidos pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3ObjectsEvents.max`	A distribuição máxima dos eventos do S3 recebidos pelo plug-in. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`sqsMessageDelay.count`	Uma contagem de pontos de dados registrados desde quando o S3 registra um horário de evento para a criação de um objeto até quando ele é totalmente analisado. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`sqsMessageDelay.sum`	O tempo total entre o momento em que o S3 registra o horário de um evento para a criação de um objeto e o momento em que ele é totalmente analisado, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`sqsMessageDelay.max`	O tempo máximo entre o momento em que o S3 grava um evento para a criação de um objeto e o momento em que ele é totalmente analisado, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`s3ObjectsSucceeded.count`	O número de objetos do S3 que o plug-in leu com sucesso. Estatísticas relevantes: soma Dimensão:`PipelineName`
`sqsMessagesReceived.count`	O número de mensagens do Amazon SQS recebidas da fila pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`
`sqsMessagesDeleted.count`	O número de mensagens do Amazon SQS excluídas da fila pelo plug-in. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`sqsMessagesFailed.count`	O número de mensagens do Amazon SQS que o plug-in não conseguiu analisar. Estatísticas relevantes: soma Dimensão:`PipelineName`

Indicadores agregados

As métricas a seguir se aplicam ao processador Aggregate (Agregar).
Cada métrica é prefixada pelo nome do subpipeline e `aggregate`. Por exemplo, *`.sub_pipeline_name`*`.aggregate.actionHandleEventsOut.count`

Sufixo métrico	Description
`actionHandleEvents Out.count`	O número de eventos que foram retornados da chamada `handleEvent` para a ação configurada. Estatísticas relevantes: soma Dimensão:`PipelineName`
`actionHandleEvents Dropped.count`	O número de eventos que foram retornados da chamada `handleEvent` para a ação configurada. Estatísticas relevantes: soma Dimensão:`PipelineName`
`actionHandleEvents ProcessingErrors.count`	O número de chamadas feitas para `handleEvent` para a ação configurada que resultaram em erro. Estatísticas relevantes: soma Dimensão:`PipelineName`
`actionConcludeGrou pEventsOut.count`	O número de eventos que foram retornados da chamada `concludeGroup` para a ação configurada.

Sufixo métrico	Description
	Estatísticas relevantes: soma Dimensão:`PipelineName`
`actionConcludeGrou pEventsDropped.count`	O número de eventos que não foram retornados da chamada `condludeGroup` para a ação configurada. Estatísticas relevantes: soma Dimensão:`PipelineName`
`actionConcludeGrou pEventsProcessingE rrors.count`	O número de chamadas feitas para `concludeGroup` para a ação configurada que resultaram em erro. Estatísticas relevantes: soma Dimensão:`PipelineName`
`currentAggregateGr oups.value`	O número atual de grupos. Esse indicador diminui quando os grupos são concluídos e aumenta quando um evento inicia a criação de um novo grupo. Estatística relevante: média Dimensão:`PipelineName`

Métricas de data

As métricas a seguir se aplicam ao processador de Date (Data).
Cada métrica é prefixada pelo nome do subpipeline e `date`. Por exemplo, *.sub_pipeline_name*`.date.dateProcessingMatchSuccess.count`

Sufixo métrico	Description
`dateProcessingMatc hSuccess.count`	O número de registros que correspondem a pelo menos um dos padrões especificados na opção de configuração `match`.

Sufixo métrico	Description
	Estatísticas relevantes: soma Dimensão:`PipelineName`
`dateProcessingMatchFailure.count`	O número de registros que não corresponderam a nenhum dos padrões especificados na opção de configuração `match`. Estatísticas relevantes: soma Dimensão:`PipelineName`

métricas do Lambda

As métricas a seguir se aplicam ao processador do AWS Lambda.
Cada métrica é prefixada pelo nome do subpipeline e `lambda`. Por exemplo, *`.sub_pipeline_name`*`.lambda.recordsSuccessfullySentToLambda.count`

Sufixo métrico	Description
`recordsSuccessfullySentToLambda.count`	O número de registros processados com sucesso pela função do Lambda. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsFailedToSendToLambda.count`	O número de registros que não puderam ser enviados para a função do Lambda. Estatísticas relevantes: soma Dimensão:`PipelineName`

`lambdaFunctionLatency.avg`
 	
`lambdaFunctionLatency.max`
	A latência das invocações da função do Lambda.
	
	Estatísticas relevantes: média, máxima

Sufixo métrico	Description
	Dimensão:`PipelineName`
`numberOfRequestsSucceeded.count`	O número total de solicitações de invocação do Lambda pelo bem-sucedidas. Estatísticas relevantes: soma Dimensão:`PipelineName`
`numberOfRequestsFailed.count`	O número total de solicitações de invocação do Lambda pelo malsucedidas. Estatísticas relevantes: soma Dimensão:`PipelineName`
`requestPayloadSize.avg`	O tamanho das cargas úteis das solicitações enviadas ao Lambda. Estatística relevante: média Dimensão:`PipelineName`
`responsePayloadSize.avg`	O tamanho das cargas úteis das respostas recebidas do Lambda. Estatística relevante: média Dimensão:`PipelineName`

Métricas do Grok

As métricas a seguir se aplicam ao processador Grok. Cada métrica é prefixada pelo nome do subpipeline e `grok`. Por exemplo, *`.sub_pipeline_name`*`.grok.grokProcessingMatch.count`

Sufixo métrico	Description
`grokProcessingMatch.count`	O número de registros que encontraram pelo menos uma correspondência de padrão na opção de configuração `match`. Estatísticas relevantes: soma Dimensão:`PipelineName`
`grokProcessingMismatch.count`	O número de registros que não corresponderam a nenhum dos padrões especificados na opção de configuração `match`. Estatísticas relevantes: soma Dimensão:`PipelineName`
`grokProcessingErrors.count`	O número de erros de processamento de registros. Estatísticas relevantes: soma Dimensão:`PipelineName`
`grokProcessingTimeouts.count`	O número de registros que atingiram o tempo limite durante a correspondência. Estatísticas relevantes: soma Dimensão:`PipelineName`
`grokProcessingTime.count`	Uma contagem de pontos de dados registrados enquanto um registro individual correspondia aos padrões da opção de configuração `match`. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`grokProcessingTime.sum`	O tempo total que cada registro individual leva para corresponder aos padrões da opção de configuração `match`, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`grokProcessingTime.max`	O tempo máximo que cada registro individual leva para corresponder aos padrões da opção de configuração `match`, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas brutas do OTel trace

As métricas a seguir se aplicam ao processador de OTel rastreamento bruto.
Cada métrica é prefixada pelo nome do subpipeline e `otel_trace_raw`. Por exemplo, *`.sub_pipeline_name`*`.otel_trace_raw.traceGroupCacheCount.value`

Sufixo métrico	Description
`traceGroupCacheCount.value`	O número de grupos de rastreamento no cache do grupo de rastreamento. Estatísticas relevantes: soma Dimensão:`PipelineName`
`spanSetCount.value`	O número de conjuntos de períodos na coleção de conjuntos de períodos. Estatísticas relevantes: soma Dimensão:`PipelineName`

Métricas de grupo de monitoramento do OTel

As métricas a seguir se aplicam ao processador do grupo de OTel rastreamento.
Cada métrica é prefixada pelo nome do subpipeline e `otel_trace_group`. Por exemplo, *`.sub_pipeline_name`*`.otel_trace_group.recordsInMissingTraceGroup.count`

Sufixo métrico	Description
`recordsInMissingTraceGroup.count`	O número de registros de entrada sem os campos do grupo de rastreamento. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsOutFixedTraceGroup.count`	O número de registros de saída com os campos do grupo de rastreamento preenchidos com sucesso. Estatísticas relevantes: soma Dimensão:`PipelineName`
`recordsOutMissingTraceGroup.count`	O número de registros de saída sem os campos do grupo de rastreamento. Estatísticas relevantes: soma Dimensão:`PipelineName`

Métricas do mapa de serviço

As métricas a seguir se aplicam ao processador Service-map stateful (Mapa de serviço com estado). Cada métrica é prefixada pelo nome do subpipeline e `service-map-stateful`. Por exemplo, *`.sub_pipeline_name`*`.service-map-stateful.spansDbSize.count`

Sufixo métrico	Description
`spansDbSize.value`	Os tamanhos de bytes na memória das extensões no MapDB nas durações da janela atual e anterior.

Sufixo métrico	Description
	Estatística relevante: média Dimensão:`PipelineName`
`traceGroupDbSize.value`	Os tamanhos de bytes na memória dos grupos de rastreamento no MapDB nas durações da janela atual e anterior. Estatística relevante: média Dimensão:`PipelineName`
`spansDbCount.value`	A contagem de intervalos das extensões no MapDB nas durações da janela atual e anterior. Estatísticas relevantes: soma Dimensão:`PipelineName`
`traceGroupDbCount.value`	A contagem de grupos de rastreamento das extensões no MapDB nas durações da janela atual e anterior. Estatísticas relevantes: soma Dimensão:`PipelineName`
`relationshipCount.value`	A contagem de relacionamentos armazenados nas durações da janela atual e anterior. Estatísticas relevantes: soma Dimensão:`PipelineName`

OpenSearch métricas

As métricas a seguir se aplicam ao OpenSearchcoletor. Cada métrica é prefixada pelo nome do subpipeline e `opensearch`. Por exemplo, `.`*`sub_pipeline_name`*`.opensearch.bulkRequestErrors.count`

Sufixo métrico	Description
`bulkRequestErrors.count`	O número total de erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`documentsSuccess.count`	O número de documentos enviados com sucesso ao OpenSearch Serviço por solicitação em massa, incluindo novas tentativas. Estatísticas relevantes: soma Dimensão:`PipelineName`
`documentsSuccessFirstAttempt.count`	O número de documentos enviados com sucesso ao OpenSearch Serviço por solicitação em massa na primeira tentativa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`documentErrors.count`	O número de documentos que não foram enviados por solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestFailed.count`	O número de solicitações em massa que falharam. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestNumber0fRetries.count`	O número de novas tentativas de solicitações em massa com falha

Sufixo métrico	Description
	Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkBadRequestErrors.count`	O número de `Bad Request` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestNotAllowedErrors.count`	O número de `Request Not Allowed` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestInvalidInputErrors.count`	O número de `Invalid Input` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestNotFoundErrors.count`	O número de `Request Not Found` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestTimeoutErrors.count`	O número de `Request Timeout` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`bulkRequestServerErrors.count`	O número de `Server Error` erros encontrados ao enviar solicitações em massa. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestSizeBytes.count`	Uma contagem da distribuição dos tamanhos de pacote das solicitações em massa, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestSizeBytes.sum`	A distribuição total dos tamanhos de pacote das solicitações em massa, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestSizeBytes.max`	A distribuição máxima dos tamanhos de pacote das solicitações em massa, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`bulkRequestLatency.count`	Uma contagem de pontos de dados registrados enquanto as solicitações são enviadas ao plug-in, incluindo novas tentativas. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`bulkRequestLatency.sum`	A latência total das solicitações enviadas ao plug-in, incluindo novas tentativas, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`bulkRequestLatency.max`	A latência máxima das solicitações enviadas ao plug-in, incluindo novas tentativas, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`s3.dlqS3RecordsSuccess.count`	O número de registros enviados com sucesso para a fila de mensagens não entregues do S3. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RecordsFailed.count`	O número de registros que não foram enviados para a fila de mensagens não entregues do S3. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RequestSuccess.count`	O número de solicitações bem-sucedidas para a fila de mensagens não entregues do S3. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`s3.dlqS3RequestFailed.count`	O número de solicitações com falha na fila de mensagens não entregues do S3. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RequestLatency.count`	Uma contagem de pontos de dados registrados enquanto as solicitações são enviadas para a fila de mensagens não entregues do S3, incluindo novas tentativas. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RequestLatency.sum`	A latência total das solicitações enviadas para a fila de mensagens não entregues do S3, incluindo novas tentativas, em milissegundos. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RequestLatency.max`	A latência máxima das solicitações enviadas para a fila de mensagens não entregues do S3, incluindo novas tentativas, em milissegundos. Estatísticas relevantes: máx. Dimensão:`PipelineName`
`s3.dlqS3RequestSizeBytes.count`	Uma contagem da distribuição dos tamanhos de carga das solicitações para a fila de mensagens não entregues do S3, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`

Sufixo métrico	Description
`s3.dlqS3RequestSizeBytes.sum`	A distribuição total dos tamanhos de carga das solicitações para a fila de mensagens não entregues do S3, em bytes. Estatísticas relevantes: soma Dimensão:`PipelineName`
`s3.dlqS3RequestSizeBytes.max`	A distribuição máxima dos tamanhos de carga das solicitações para a fila de mensagens não entregues do S3, em bytes. Estatísticas relevantes: máx. Dimensão:`PipelineName`

Métricas do sistema e de medição

As métricas a seguir se aplicam ao sistema geral OpenSearch de ingestão. Essas métricas não são prefixadas.

Métrica	Description
`system.cpu.usage.value`	A porcentagem de uso da CPU disponível para todos os nós de dados. Estatística relevante: média Dimensão: `PipelineName` , `area`, `id`
`system.cpu.count.value`	A quantidade total de uso da CPU para todos os nós de dados. Estatística relevante: média Dimensão: `PipelineName` , `area`, `id`

Métrica	Description
`jvm.memory.max.value`	A quantidade máxima de memória que pode ser usada para gerenciamento de memória, em bytes. Estatística relevante: média Dimensão: `PipelineName` , `area`, `id`
`jvm.memory.used.value`	A quantidade total de memória usada em bytes. Estatística relevante: média Dimensão: `PipelineName` , `area`, `id`sinal
`jvm.memory.committ ed.value`	A quantidade de memória comprometida para uso pela máquina virtual Java (JVM), em bytes. Estatística relevante: média Dimensão: `PipelineName` , `area`, `id`
`computeUnits`	O número de unidades OpenSearch computacionais de ingestão (ingestão OCUs) em uso por um pipeline. Estatísticas relevantes: máximo, soma, média Dimensão:`PipelineName`

Melhores práticas para OpenSearch ingestão na Amazon

Este tópico fornece as melhores práticas para criar e gerenciar pipelines de OpenSearch ingestão da Amazon e inclui diretrizes gerais que se aplicam a muitos casos de uso. Cada workload é única e tem características particulares, portanto, nenhuma recomendação genérica é exatamente certa para cada caso de uso.

Tópicos

- Práticas recomendadas gerais
- CloudWatch Alarmes recomendados

Práticas recomendadas gerais

As práticas recomendadas gerais a seguir se aplicam à criação e gerenciamento de pipelines.

- Para garantir a alta disponibilidade, configure pipelines de VPC com duas ou três sub-redes. Se você implantar apenas um pipeline em uma sub-rede e a Zona de disponibilidade ficar inativa, você não conseguirá ingerir dados.
- Em cada pipeline, recomendamos limitar o número de subpipelines a 5 ou menos.
- Se você estiver usando o plug-in de origem do S3, use arquivos do S3 de tamanho uniforme para obter um desempenho ideal.
- Se estiver usando o plug-in de origem do S3, adicione 30 segundos de tempo limite de visibilidade adicional para cada 0,25 GB de tamanho de arquivo no bucket do S3 para obter um desempenho ideal.
- Inclua uma fila de mensagens não entregues (DLQ – fila de mensagens não entregues) na configuração do pipeline para que você possa descarregar eventos com falha e torná-los acessíveis para análise. Se seus coletores rejeitarem dados devido a mapeamentos incorretos ou outros problemas, você poderá rotear os dados para o DLQ para avaliar e corrigir o problema.
- Se você estiver usando um processador agregado em um pipeline, recomendamos o uso de `“local_mode: true”` flag para otimizar o desempenho do pipeline.

CloudWatch Alarmes recomendados

CloudWatch os alarmes realizam uma ação quando uma CloudWatch métrica excede um valor especificado por um determinado período de tempo. Por exemplo, talvez você queira AWS enviar um e-mail se o status de integridade do seu cluster `red` for superior a um minuto. Esta seção inclui alguns alarmes recomendados para o Amazon OpenSearch Ingestion e como responder a eles.

Para obter mais informações sobre a configuração de alarmes, consulte Criação de CloudWatchalarmes da Amazon no Guia do usuário da Amazon CloudWatch .

Alarme	Problema
O `computeUnits` máximo é = o `maxUnits` configura	O pipeline atingiu a capacidade máxima e pode precisar de uma atualização de `maxUnits`. Aumente a capacidade máxima do seu pipeline

Alarme	Problema
do para 15 minutos, 3 vezes consecutivas	
`opensearch.documentErrors.count` soma é = soma de *`{sub_pipeline_name}`* `.opensearch.recordsIn.count` por 1 minuto, 1 vez consecutiva	O pipeline não consegue gravar no OpenSearch coletor. Verifique as permissões do pipeline e confirme se o domínio ou a coleção estão íntegros. Você também pode verificar se há eventos com falha na fila de mensagens não entregues (DLQ), se ela estiver configurada.
`bulkRequestLatency.max` máximo é >= x por 1 minuto, 1 vez consecutiva	O pipeline está passando por uma alta latência enviando dados para o OpenSearch coletor. Provavelmente, isso se deve ao fato de a pia estar subdimensionada ou a uma estratégia de fragmentação deficiente, que está fazendo com que o coletor deixe a desejar. A alta latência sustentada pode afetar o desempenho do pipeline e provavelmente causará uma contrapressão nos clientes.
`httpAuthFailure.count` soma >= 1 por 1 minuto, 1 vez consecutiva	As solicitações de ingestão não estão sendo autenticadas. Confirme se todos os clientes têm a autenticação Signature versão 4 ativada corretamente.
Média de `system.cpu.usage.value` >= 80% por 15 minutos, 3 vezes consecutivas	A utilização elevada e sustentada da CPU pode ser problemática. Considere aumentar a capacidade máxima do pipeline.

Alarme	Problema
Média de `bufferUsage.value` >= 80% por 15 minutos, 3 vezes consecutivas	O uso sustentado de alta bufferização pode ser problemático. Considere aumentar a capacidade máxima do pipeline.

Outros alarmes que você pode considerar

Considere configurar os seguintes alarmes, dependendo dos recursos do Amazon OpenSearch Ingestion que você usa regularmente.

Alarme	Problema
`dynamodb.exportJobFailure.count` soma 1	A tentativa de acionar uma exportação para o Amazon S3 falhou.
Média de `opensearch.EndtoEndLatency.avg` > X por 15 minutos, 4 vezes consecutivas	`EndtoEndLatency` é maior do que o desejado para leitura de fluxos do DynamoDB. Isso pode ser causado por um OpenSearch cluster subdimensionado ou por uma capacidade máxima de OCU de pipeline muito baixa para a taxa de transferência da WCU na tabela do DynamoDB. `EndtoEndLatency` será maior após uma exportação, mas deve diminuir com o tempo à medida que alcança os streams mais recentes do DynamoDB.
`dynamodb.changeEventsProcessed.count` soma == 0 por X minutos	Nenhum registro está sendo coletado dos fluxos do DynamoDB. Isso pode ser causado por falta de atividade na tabela ou por um problema no acesso aos fluxos do DynamoDB.
soma `opensearch.s3.dlqS3RecordsSuccess.count`	Um número maior de registros está sendo enviado para o DLQ do que para o OpenSearch coletor. Analise as métricas do plug-in OpenSearch sink para investigar e determinar a causa raiz.

Alarme	Problema
>= soma `opensearch.documentSuccess.count` por 1 minuto, 1 vez consecutiva	
`grok.grokProcessingTimeouts.count` é = soma de recordsIn.count por 1 minuto, 5 vezes consecutivas	Todos os dados atingem o tempo limite enquanto o processador Grok está tentando combinar padrões. Isso provavelmente está afetando o desempenho e diminuindo a velocidade do seu pipeline. Considere ajustar seus padrões para reduzir os tempos limite.
`grok.grokProcessingErrors.count` soma é >= 1 por 1 minuto, 1 vez consecutiva	O processador Grok não está conseguindo combinar os padrões com os dados no pipeline, resultando em erros. Revise seus dados e as configurações do plug-in do Grok para garantir que a correspondência de padrões seja a esperada.
`grok.grokProcessingMismatch.count` é = soma de recordsIn.count por 1 minuto, 5 vezes consecutivas	O processador Grok não consegue combinar padrões com os dados no pipeline. Revise seus dados e as configurações do plug-in do Grok para garantir que a correspondência de padrões seja a esperada.
`date.dateProcessingMatchFailure.count` soma = soma de recordsIn.count por 1 minuto, 5 vezes consecutivas	O processador de data não consegue combinar nenhum padrão com os dados no pipeline. Revise seus dados e as configurações do plug-in de data para garantir que o padrão seja o esperado.

Alarme	Problema
`s3.s3ObjectsFailed.count` soma >= 1 por 1 minuto, 1 vez consecutiva	Esse problema está ocorrendo porque o objeto S3 não existe ou porque o pipeline não tem privilégios suficientes. Analise as métricas de `s3ObjectsNotFound.count` e `s3ObjectsAccessDenied.count` para determinar a causa raiz. Confirme se o objeto S3 existe e and/or atualize as permissões.
`s3.sqsMessagesFailed.count` soma >= 1 por 1 minuto, 1 vez consecutiva	O plug-in do S3 falhou ao processar uma mensagem do Amazon SQS. Se você tiver um DLQ habilitado em sua fila do SQS, revise a mensagem de falha. A fila pode estar recebendo dados inválidos que o pipeline está tentando processar.
`http.badRequests.count` soma >= 1 por 1 minuto, 1 vez consecutiva	O cliente está enviando uma solicitação incorreta. Confirme se todos os clientes estão enviando a carga útil adequada.
`http.requestsTooLarge.count` soma >= 1 por 1 minuto, 1 vez consecutiva	As solicitações do plug-in HTTP de origem contêm muitos dados, excedendo a capacidade do buffer. Ajuste o tamanho do lote para seus clientes.
`http.internalServerError.count` soma >=0 por 1 minuto, 1 vez consecutiva	O plug-in HTTP de origem está tendo problemas para receber eventos.

Alarme	Problema
`http.requestTimeouts.count` soma >=0 por 1 minuto, 1 vez consecutiva	Os tempos limite de origem provavelmente são o resultado do subprovisionamento do pipeline. Considere aumentar o pipeline `maxUnits` para lidar com o workload (carga de trabalho) adicional.
`otel_trace.badRequests.count` soma >= 1 por 1 minuto, 1 vez consecutiva	O cliente está enviando uma solicitação incorreta. Confirme se todos os clientes estão enviando a carga útil adequada.
`otel_trace.requestsTooLarge.count` soma >= 1 por 1 minuto, 1 vez consecutiva	As solicitações do plug-in de origem do OTel Trace contêm muitos dados, excedendo a capacidade do buffer. Ajuste o tamanho do lote para seus clientes.
`otel_trace.internalServerError.count` soma >=0 por 1 minuto, 1 vez consecutiva	O plug-in de origem do OTel Trace está tendo problemas para receber eventos.
`otel_trace.requestTimeouts.count` soma >=0 por 1 minuto, 1 vez consecutiva	Os tempos limite de origem provavelmente são o resultado do subprovisionamento do pipeline. Considere aumentar o pipeline `maxUnits` para lidar com o workload (carga de trabalho) adicional.

Alarme	Problema
`otel_metrics.requestTimeouts.count` soma >=0 por 1 minuto, 1 vez consecutiva	Os tempos limite de origem provavelmente são o resultado do subprovisionamento do pipeline. Considere aumentar o pipeline `maxUnits` para lidar com o workload (carga de trabalho) adicional.

Amazon sem OpenSearch servidor

O Amazon OpenSearch Serverless é uma configuração sob demanda e com escalabilidade automática para o Amazon Service. OpenSearch Ao contrário dos OpenSearch domínios provisionados, que exigem gerenciamento manual da capacidade, uma coleção OpenSearch sem servidor escala automaticamente os recursos de computação com base nas necessidades do seu aplicativo.

OpenSearch O Serverless oferece uma solução econômica para cargas de trabalho pouco frequentes, intermitentes ou imprevisíveis. Ele otimiza os custos ajustando automaticamente a capacidade computacional de acordo com o uso da aplicação. As coleções sem servidor usam o mesmo volume de armazenamento de alta capacidade, distribuído e altamente disponível dos domínios de serviços OpenSearch provisionados.

OpenSearch As coleções sem servidor são sempre criptografadas. É possível escolher a chave de criptografia, mas não é possível desabilitar a criptografia. Para saber mais, consulte . the section called “Criptografia”

Benefícios

OpenSearch O Serverless tem os seguintes benefícios:

- Mais simples do que provisionado — o OpenSearch Serverless remove grande parte da complexidade do gerenciamento de clusters e da capacidade. OpenSearch Ele dimensiona e ajusta automaticamente seus clusters e cuida do gerenciamento do ciclo de vida de fragmentos e índices. Ele também gerencia atualizações de software de serviço e upgrades de OpenSearch versão. Todas as atualizações e upgrades não causam interrupções.
- Econômico — Ao usar o OpenSearch Serverless, você paga apenas pelos recursos que consome. Isso elimina a necessidade de provisionamento inicial e superprovisionamento para workloads de pico.
- Altamente disponível — o OpenSearch Serverless suporta cargas de trabalho de produção com redundância para proteger contra interrupções na Zona de Disponibilidade e falhas na infraestrutura.
- Escalável — O OpenSearch Serverless dimensiona automaticamente os recursos para manter taxas de ingestão de dados e tempos de resposta de consultas consistentemente rápidos.

O que é Amazon OpenSearch Serverless?

O Amazon OpenSearch Serverless é uma opção sob demanda e sem servidor para o OpenSearch Amazon Service que elimina a complexidade operacional do provisionamento, configuração e ajuste de clusters. OpenSearch É ideal para organizações que preferem não autogerenciar seus clusters ou que não têm recursos e expertise dedicados para operar implantações em grande escala. Com o OpenSearch Serverless, você pode pesquisar e analisar grandes volumes de dados sem gerenciar a infraestrutura subjacente.

Uma coleção OpenSearch sem servidor é um grupo de OpenSearch índices que trabalham juntos para dar suporte a uma carga de trabalho ou caso de uso específico. As coleções simplificam as operações em comparação com OpenSearch clusters autogerenciados, que exigem provisionamento manual.

As coleções usam o mesmo armazenamento de alta capacidade, distribuído e altamente disponível dos domínios de OpenSearch serviços provisionados, mas reduzem ainda mais a complexidade ao eliminar a configuração e o ajuste manuais. Os dados em uma coleção são criptografados em trânsito. OpenSearch O Serverless também oferece suporte a OpenSearch painéis, fornecendo uma interface para análise de dados.

Atualmente, as coleções sem servidor executam a OpenSearch versão 2.17.x. À medida que novas versões são lançadas, o OpenSearch Serverless atualiza automaticamente as coleções para incorporar novos recursos, correções de erros e melhorias de desempenho.

OpenSearch O Serverless suporta as mesmas operações de API de ingestão e consulta do pacote de código OpenSearch aberto, para que você possa continuar usando seus clientes e aplicativos existentes. Seus clientes devem ser compatíveis com OpenSearch 2.x para trabalhar com o OpenSearch Serverless. Para obter mais informações, consulte the section called “Ingestão de dados em coleções”.

Tópicos

- Casos de uso do OpenSearch Serverless
- Como funciona
- Escolha de um tipo de coleção
- Preços
- Suportado Regiões da AWS
- Limitações

- Comparando OpenSearch serviços e sem OpenSearch servidor

Casos de uso do OpenSearch Serverless

OpenSearch O Serverless oferece suporte a dois casos de uso principais:

- Análise de logs: o segmento de analytics de logs se concentra na análise de grandes volumes de dados de séries temporais, semiestruturados e gerados por máquina para obter informações operacionais e de comportamento do usuário.
- Pesquisa de texto completo: o segmento de pesquisa de texto completo alimenta aplicações em suas redes internas (sistemas de gerenciamento de conteúdo, documentos legais) e aplicações voltadas para a Internet, como a pesquisa de conteúdo de sites de comércio eletrônico.

Ao criar uma coleção, escolha um desses casos de uso. Para saber mais, consulte the section called "Escolha de um tipo de coleção".

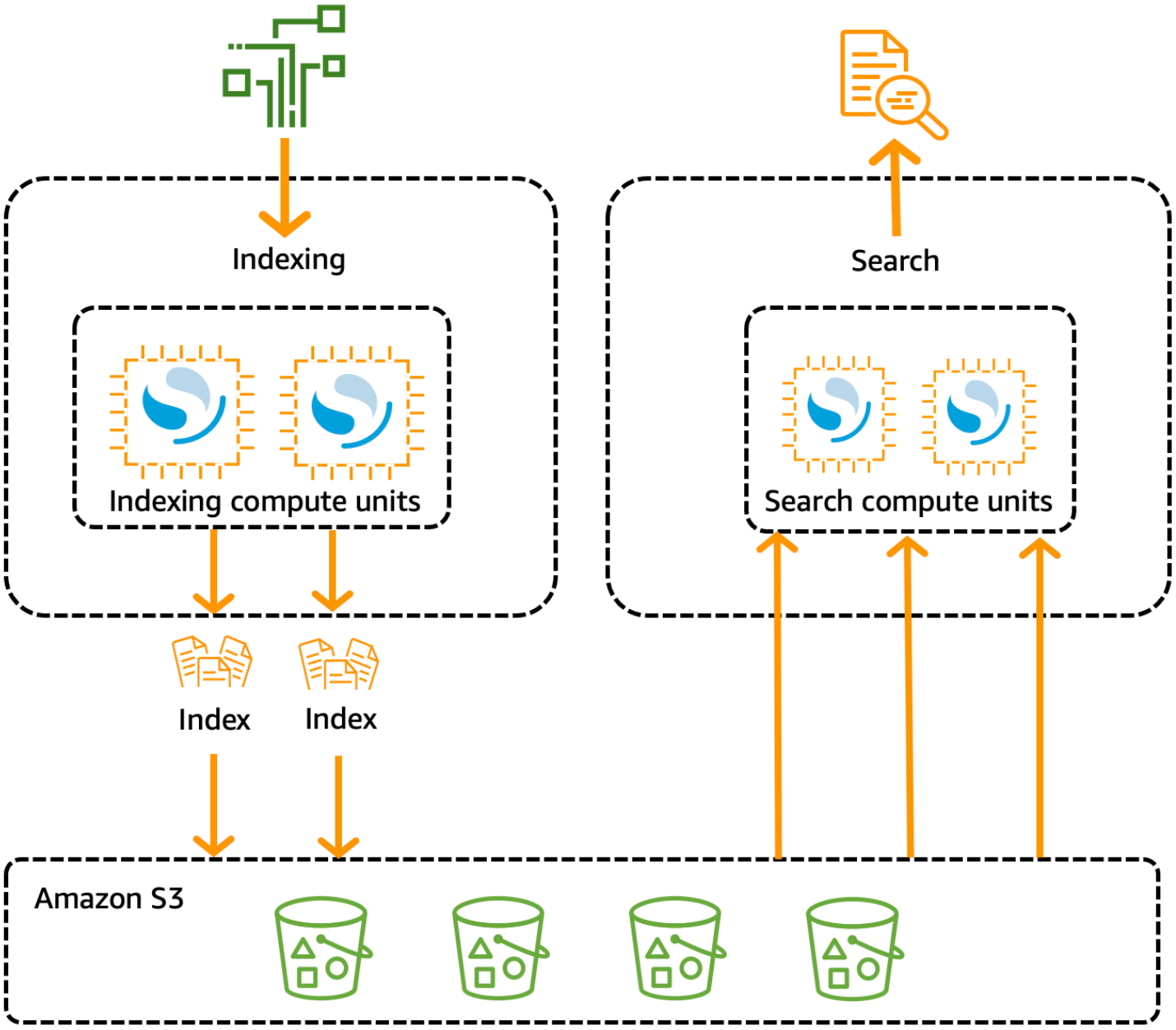
Como funciona

OpenSearch Os clusters tradicionais têm um único conjunto de instâncias que realizam operações de indexação e pesquisa, e o armazenamento de índices está estreitamente associado à capacidade computacional. Por outro lado, o OpenSearch Serverless usa uma arquitetura nativa da nuvem que separa os componentes de indexação (ingestão) dos componentes de pesquisa (consulta), com o Amazon S3 como principal armazenamento de dados para índices.

Essa arquitetura desacoplada permite escalar as funções de pesquisa e indexação de forma independente uma da outra e independentemente dos dados indexados no S3. A arquitetura também fornece isolamento para operações de ingestão e consulta para que elas possam ser executadas simultaneamente, sem contenção de recursos.

Quando você grava dados em uma coleção, o OpenSearch Serverless os distribui para as unidades computacionais de indexação. As unidades computacionais de indexação ingerem os dados recebidos e movem os índices para S3. Quando você realiza uma pesquisa nos dados da coleta, o OpenSearch Serverless encaminha as solicitações para as unidades computacionais de pesquisa que contêm os dados que estão sendo consultados. As unidades computacionais de pesquisa baixam os dados indexados diretamente do S3 (se ainda não estiverem armazenados em cache localmente), executam operações de pesquisa e realizam agregações.

A imagem a seguir ilustra essa arquitetura desacoplada:



OpenSearch A capacidade computacional sem servidor para ingestão, pesquisa e consulta de dados é medida em OpenSearch Unidades de Computação (). OCUs Cada OCU é uma combinação de 6 GiB de memória e CPU virtual (vCPU) correspondente e cria um pipeline de dados para o Amazon S3. Cada OCU inclui armazenamento efêmero de atividade muito alta que é suficiente para 120 GiB de dados de indexação.

Quando você cria sua primeira coleção, o OpenSearch Serverless instancia duas OCUs — uma para indexação e outra para pesquisa. Para garantir alta disponibilidade, ele também lança um conjunto de nós em espera em outra zona de disponibilidade. Para fins de desenvolvimento e teste, você pode desativar a configuração Ativar redundância para uma coleção, que elimina as duas

réplicas em espera e instancia apenas duas. OCUs Por padrão, as réplicas ativas redundantes estão habilitadas, o que significa que um total de quatro OCUs são instanciadas para a primeira coleção em uma conta.

Eles OCUs existem mesmo quando não há atividade em nenhum endpoint de coleta. Todas as coleções subsequentes as compartilham OCUs. Quando você cria coleções adicionais na mesma conta, o OpenSearch Serverless só adiciona mais OCUs para pesquisa e ingestão conforme necessário para dar suporte às coleções, de acordo com os limites de capacidade que você especificar. A capacidade é reduzida à medida que o uso da computação diminui.

Para obter informações sobre como você é cobrado por eles OCUs, consultethe section called "Preços".

Escolha de um tipo de coleção

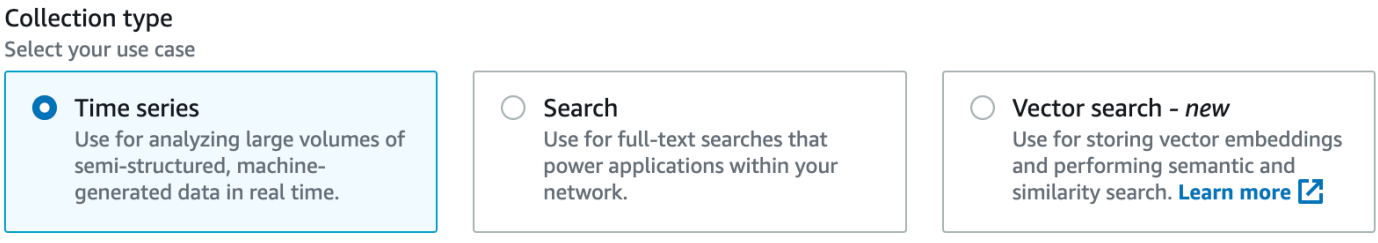
OpenSearch O Serverless oferece suporte a três tipos principais de coleção:

Séries temporais: o segmento de analytics de log que analisa em tempo real grandes volumes de dados semiestruturados gerados por máquinas, fornecendo insights sobre operações, segurança, comportamento do usuário e desempenho dos negócios.

Pesquisa: pesquisa em texto completo que habilita aplicações em redes internas, como sistemas de gerenciamento de conteúdo e repositórios de documentos jurídicos, assim como aplicações voltadas para a internet, como pesquisa em sites de comércio eletrônico e descoberta de conteúdo.

Pesquisa vetorial: a pesquisa semântica em incorporações vetoriais simplifica o gerenciamento de dados vetoriais e possibilita experiências de pesquisa aprimorada por machine learning (ML). É compatível com aplicações de IA generativa, como chatbots, assistentes pessoais e detecção de fraudes.

Você escolhe um tipo de coleção ao criar uma coleção pela primeira vez:



O tipo de coleção que você escolhe depende do tipo dos dados que planeja ingerir na coleção e de como você planeja consultá-los. Não é possível alterar o tipo da coleção depois de criá-la.

Os tipos de coleção têm as seguintes diferenças notáveis:

- Para coleções de pesquisa e pesquisa vetorial, todos os dados são armazenados no armazenamento a quente para garantir tempos de resposta rápidos às consultas. As coleções de séries temporais usam uma combinação de armazenamento de atividade alta e muito alta, em que os dados mais recentes são mantidos em armazenamento de atividade muito alta para otimizar os tempos de resposta da consulta para dados acessados com mais frequência.
- Para coleções de séries temporais e pesquisa vetorial, não é possível indexar por ID de documento personalizado nem atualizar por solicitações de upsert. Essa operação é reservada para casos de uso de pesquisa. Em vez disso, você pode atualizar por ID do documento. Para saber mais, consulte the section called “Operações e permissões de OpenSearch API suportadas”.
- Para pesquisas e coleções de séries temporais, você não pode usar índices do tipo k-NN.

Preços

AWS cobra pelos seguintes componentes OpenSearch sem servidor:

- Computação de ingestão de dados
- Computação de pesquisa e consulta
- Armazenamento retido no Amazon S3

Uma OCU compreende 6 GB de RAM, vCPU GP3 , armazenamento e transferência de dados correspondentes para o Amazon S3. A menor unidade pela qual você pode ser cobrado é 0,5 OCU. AWS fatura a OCU por hora, com granularidade de segundo nível. No extrato da sua conta, você vê uma entrada para computação em horas de OCU com um rótulo para ingestão de dados e um rótulo para pesquisa. AWS também cobra mensalmente pelos dados armazenados no Amazon S3. Ele não cobra pelo uso de OpenSearch painéis.

Ao criar uma coleção com réplicas ativas redundantes, você é cobrado pelo menos 1 OCU (0,5 OCU × 2) para ingestão, incluindo primária e em espera, e 1 OCU (0,5 OCU × 2) para pesquisa:

- 1 OCU (0,5 OCU × 2) por ingestão, incluindo primária e em espera
- 1 OCU (0,5 OCU × 2) por pesquisa

Se você desabilitar réplicas ativas redundantes, será cobrado um mínimo de 1 OCU (0,5 OCU x 2) pela primeira coleção em sua conta. Todas as coleções subsequentes podem compartilhá-las OCUs.

OpenSearch O Serverless adiciona mais OCUs em incrementos de 1 OCU com base na potência computacional e no armazenamento necessários para dar suporte às suas coleções. Você pode configurar um número máximo de OCUs para sua conta para controlar os custos.

> **Note**
>
> Coleções com itens exclusivos não AWS KMS keys podem ser compartilhadas OCUs com outras coleções.

OpenSearch O servidor tenta usar os recursos mínimos necessários para contabilizar as mudanças nas cargas de trabalho. O número de OCUs provisionados a qualquer momento pode variar e não é exato. Com o tempo, o algoritmo usado pelo OpenSearch Serverless continuará melhorando para minimizar melhor o uso do sistema.

Para obter detalhes completos sobre preços, consulte os preços OpenSearch do Amazon Service.

Suportado Regiões da AWS

OpenSearch O Serverless está disponível em um subconjunto Regiões da AWS desse OpenSearch Serviço disponível em. Para obter uma lista das regiões suportadas, consulte os endpoints e cotas do Amazon OpenSearch Service no. Referência geral da AWS

Limitações

OpenSearch O Serverless tem as seguintes limitações:

- Algumas operações de OpenSearch API não são suportadas. Consulte the section called “Operações e permissões de OpenSearch API suportadas”.
- Alguns OpenSearch plug-ins não são compatíveis. Consulte the section called “OpenSearch Plugins compatíveis”.
- Atualmente, não há como migrar automaticamente seus dados de um domínio de OpenSearch serviço gerenciado para uma coleção sem servidor. É necessário reindexar seus dados de um domínio para uma coleção.
- Não há suporte para acesso entre contas a coleções. Não é possível incluir coleções de outras contas em suas políticas de criptografia ou acesso a dados.
- Não há suporte para OpenSearch plug-ins personalizados.

- Instantâneos automatizados são compatíveis com coleções OpenSearch sem servidor. Não há suporte para instantâneos manuais. Para obter mais informações, consulte the section called "Fazer backup de coleções usando snapshots".
- Não há suporte para pesquisa e replicação entre regiões.
- Há limites no número de recursos de tecnologia sem servidor possíveis em uma única conta e região. Consulte Cotas OpenSearch sem servidor.
- O intervalo de atualização dos índices nas coleções de pesquisa vetorial é de aproximadamente 60 segundos. O intervalo de atualização dos índices nas coleções de pesquisa e série temporal é de aproximadamente 10 segundos.
- O número de fragmentos, o número de intervalos e o intervalo de atualização não são modificáveis e são gerenciados pelo Serverless. OpenSearch A estratégia de fragmentação é baseada no tipo de coleta e no tráfego. Por exemplo, uma coleção de séries temporais dimensiona os fragmentos primários com base nos gargalos do tráfego de gravação.
- Os recursos geoespaciais disponíveis nas OpenSearch versões até 2.1 são suportados.

Comparando OpenSearch serviços e sem OpenSearch servidor

No OpenSearch Serverless, alguns conceitos e recursos são diferentes dos recursos correspondentes para um domínio de serviço provisionado OpenSearch . Por exemplo, uma diferença importante é que o OpenSearch Serverless não tem o conceito de cluster ou nó.

A tabela a seguir descreve como os recursos e conceitos importantes do OpenSearch Serverless diferem do recurso equivalente em um domínio de serviço provisionado OpenSearch .

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
Domínios versus coleções	Os índices são mantidos em domínios, que são clusters OpenSearch pré-provi sionados. Para obter mais informações, consulte Criação e gerenciam ento de domínios.	Os índices são mantidos em coleções, que são agrupamentos lógicos de índices que representam uma workload ou um caso de uso específico. Para saber mais, consulte the section called "Gerenciar coleções".
Tipos de nós e gerenciam	Você cria um cluster com tipos de nós que atendem às	OpenSearch O Serverless dimensiona e provisiona automaticamente unidades de

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
ento de capacidade	suas especificações de custo e performance. É necessári o calcular seus próprios requisitos de armazenamento e escolher um tipo de instância para seu domínio. Para obter mais informaçõ es, consulte the section called “Dimensionamento de domínios”.	computação adicionais para sua conta com base no uso da capacidade. Para obter mais informações, consulte the section called “Gerenciamento de limites de capacidade”.
Faturamento	Você paga por hora de uso de uma instância do EC2 e pelo tamanho cumulativo de todos os volumes de armazenam ento do EBS anexados às suas instâncias. Para saber mais, consulte the section called “Preços”.	Você é cobrado em horas de OCU pela computação para ingestão de dados, computação para pesquisa e consulta e armazenamento retido no S3. Para saber mais, consulte the section called “Preços”.
Criptografia	A criptografia em repouso é opcional para domínios. Para saber mais, consulte the section called “Criptografia em repouso”.	A criptografia em repouso é obrigatória para coleções. Para saber mais, consulte the section called “Criptografia”.
Controle de acesso a dados	O acesso aos dados nos domínios é determinado pelas políticas do IAM e pelo controle de acesso minucioso.	O acesso aos dados nas coleções é determina do pelas políticas de acesso a dados.

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
OpenSearchOperações suportadas	OpenSearch O serviço oferece suporte a um subconjunto de todas as operações da OpenSearch API. Para obter mais informações, consulte the section called “Operações compatíveis”.	OpenSearch O Serverless oferece suporte a um subconjunto diferente de operações de OpenSearch API. Para obter mais informações, consulte the section called “Operações e plug-ins com suporte”.
Login no Dashboards	Faça login com um nome de usuário e senha. Para obter mais informações, consulte the section called “Acessando OpenSearch painéis como usuário principal”.	Se você estiver conectado ao AWS console e navegar até a URL do seu painel, você fará login automaticamente. Para obter mais informações, consulte the section called “Acessando OpenSearch painéis”.
APIs	Interaja programaticamente com o OpenSearch Serviço usando as operações da API do OpenSearch Serviço.	Interaja programaticamente com o OpenSearch Serverless usando as operações da API Serverless. OpenSearch
Acesso à rede	As configurações de rede de um domínio se aplicam ao endpoint do domínio, bem como ao endpoint do OpenSearch Dashboards. O acesso à rede para ambos está fortemente acoplado.	As configurações de rede do endpoint do domínio e do endpoint do OpenSearch Dashboards são dissociadas. Você pode optar por não configurar o acesso à rede para OpenSearch painéis. Para obter mais informações, consulte the section called “Acesso à rede”.

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
Assinatura de solicitações	Use os clientes REST de OpenSearch alto e baixo nível para assinar solicitações. Especifique o nome do serviço como `es`.	No momento, o OpenSearch Serverles s oferece suporte a um subconjunto de clientes aos quais o Service oferece suporte. OpenSearch Ao assinar solicitações, especifique o nome do serviço como `aoss`. O cabeçalho `x-amz-content-sha256` é obrigatório. Para obter mais informações, consulte the section called "Outros clientes".
OpenSearch atualizações de versão	Você atualiza manualmen te seus domínios à medida que novas versões do são OpenSearch disponibilizadas. Você é responsável por garantir que seu domínio atenda aos requisitos de atualização e que tenha resolvido quaisquer alterações importantes.	OpenSearch O Serverless atualiza automatic amente suas coleções para novas versões. OpenSearch As atualizações não acontecem necessariamente assim que uma nova versão é disponibilizada.
Atualizações de software de serviço	Você aplica manualmente as atualizações do software de serviço ao seu domínio assim que elas se tornam disponíve is.	OpenSearch O Serverless atualiza automatic amente suas coleções para consumir as últimas correções de bugs, recursos e melhorias de desempenho.
Acesso por VPC	É possível provisionar seu domínio em uma VPC. Você também pode criar endpoints OpenSearch VPC gerenciados por serviços adicionais para acessar o domínio.	Você cria um ou mais VPC endpoints OpenSearch gerenciados sem servidor para sua conta. Em seguida, você inclui esses endpoints nas políticas de rede.

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
Autenticação SAML	Você habilita a autenticação SAML por domínio. Para obter mais informações, consulte the section called “Autenticação SAML para painéis OpenSearch”.	Você configura um ou mais provedores de SAML no nível da conta e, em seguida, inclui o usuário e o grupo associados IDs nas políticas de acesso aos dados. Para obter mais informações, consulte the section called “Autenticação SAML”.
Transport Layer Security (TLS)	OpenSearch O serviço oferece suporte ao TLS 1.2, mas é recomendável usar o TLS 1.3.	OpenSearch O Serverless oferece suporte ao TLS 1.2, mas é recomendável usar o TLS 1.3.

Tutorial: Introdução ao Amazon OpenSearch Serverless

Este tutorial mostra as etapas básicas para colocar uma coleção de pesquisa Amazon OpenSearch Serverless em funcionamento rapidamente. Uma coleção de pesquisas permite que você alimente aplicativos em suas redes internas e aplicativos voltados para a Internet, como a pesquisa de sites de comércio eletrônico e de conteúdo.

Para saber como usar uma coleção de pesquisa vetorial, consulte the section called “Trabalho com coleções de pesquisa vetorial”. Para obter informações detalhadas sobre o uso das coleções, consulte the section called “Gerenciar coleções” e outros tópicos nesta aba.

Você concluirá as seguintes etapas neste tutorial:

1. Configurar permissões
2. Criar uma coleção
3. Transferir e pesquisar dados
4. Excluir a coleção

> **Note**
>
> Recomendamos que você use somente caracteres ASCII no `IndexName`. Se você não usar caracteres ASCII para o seu`IndexName`, as CloudWatch métricas `IndexName` in serão convertidas em um formato codificado de URL para caracteres não ASCII.

Etapa 1: configurar permissões

Para concluir este tutorial e usar o OpenSearch Serverless em geral, você deve ter as permissões corretas do IAM. Neste tutorial, você criará uma coleção, transferirá e pesquisará dados e, em seguida, excluirá a coleção.

Seu usuário ou função deve ter uma política baseada em identidade anexada com as seguintes permissões mínimas:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:UpdateSecurityPolicy",
        "iam:ListUsers",
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
```

```
}
```

Para obter mais informações sobre as permissões do IAM OpenSearch sem servidor, consulte. the section called “Gerenciamento de Identidade e Acesso”

Etapa 2: criar uma coleção

Uma coleção é um grupo de OpenSearch índices que trabalham juntos para dar suporte a uma carga de trabalho ou caso de uso específico.

Para criar uma coleção OpenSearch sem servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Escolha Coleções no painel de navegação à esquerda e escolha Criar coleção.
3. Dê à coleção o nome de movies (filmes).
4. Para o tipo de coleção, escolha Pesquisar. Para saber mais, consulte Choosing a network type (Escolher um tipo de rede).
5. Em Segurança, escolha Criação padrão.
6. Em Criptografia, selecione Usar Chave pertencente à AWS. Isso é o AWS KMS key que o OpenSearch Serverless usará para criptografar seus dados.
7. Em Rede, configure o acesso à rede para a coleção.
   - Para o tipo de acesso, selecione Público.
   - Para o tipo de recurso, escolha Habilitar acesso a OpenSearch endpoints e Habilitar acesso a OpenSearch painéis. Como você fará o upload e pesquisará dados usando OpenSearch painéis, é necessário habilitar os dois.
8. Escolha Próximo.
9. Em Configurar acesso aos dados, defina as configurações de acesso para a coleção. As políticas de acesso a dados permitem que usuários e funções acessem os dados em uma coleção. Neste tutorial, forneceremos a um único usuário as permissões necessárias para indexar e pesquisar dados na coleção movies (filmes).

   Crie uma única regra que forneça acesso à coleção de filmes. Nomeie a regra de Acesso à coleção de filmes.
10. Escolha Adicionar diretores, usuários e funções do IAM e selecione o usuário ou a função que você usará para fazer login nos OpenSearch painéis e indexar dados. Escolha Salvar.

11. Em Permissões de índices, selecione todas as permissões.
12. Escolha Próximo.
13. Para as configurações da política de acesso, escolha Criar uma nova política de acesso a dados e nomeie os filmes da política.
14. Escolha Próximo.
15. Reveja suas configurações da coleção e escolha Enviar. Aguarde alguns minutos até que o status da coleção mude para `Active`.

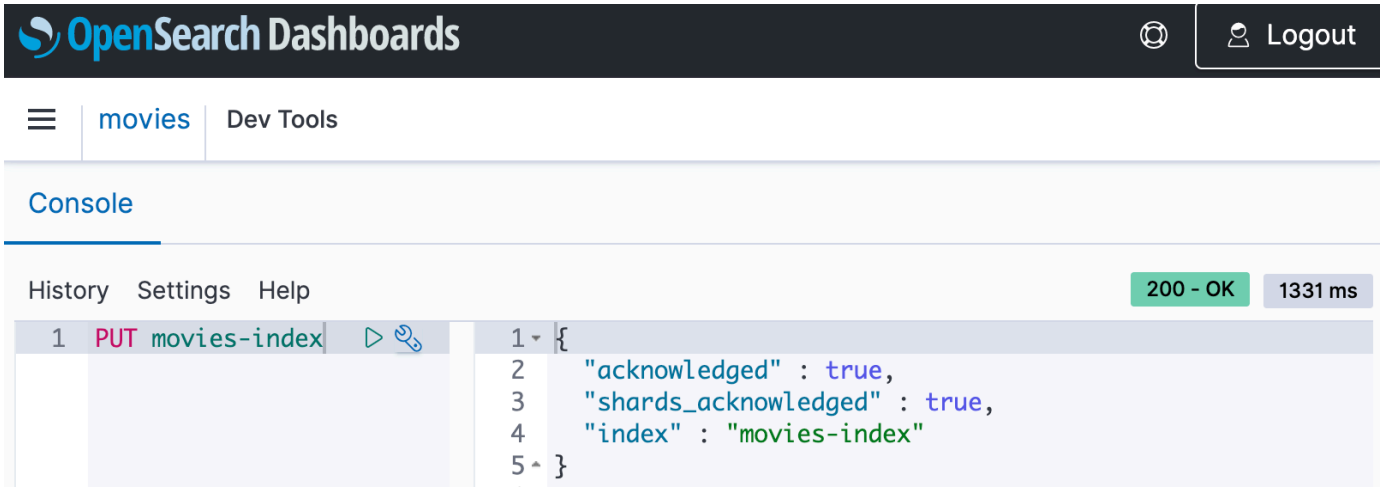
Etapa 3: Transferir e pesquisar dados

Você pode carregar dados para uma coleção OpenSearch sem servidor usando Postman ou cURL. Para resumir, esses exemplos usam Dev Tools no console OpenSearch Dashboards.

Para indexar e pesquisar dados na coleção de filmes

1. Escolha Coleções no painel de navegação à esquerda e escolha a coleção movies (filmes) para abrir sua página de detalhes.
2. Escolha o URL dos OpenSearch painéis para a coleção. O URL assume o formato `https://dashboards.{region}.aoss.amazonaws.com/_login/?collectionId={collection-id}`.
3. Em OpenSearch Painéis, abra o painel de navegação esquerdo e escolha Ferramentas de desenvolvimento.
4. Para criar um único índice chamado movies-index, envie a seguinte solicitação:

```
PUT movies-index
```



5. Para indexar um único documento em movies-index, envie a seguinte solicitação:

```
PUT movies-index/_doc/1
{
  "title": "Shawshank Redemption",
  "genre": "Drama",
  "year": 1994
}
```

6. Para pesquisar dados em OpenSearch painéis, você precisa configurar pelo menos um padrão de índice. OpenSearch usa esses padrões para identificar quais índices você deseja analisar. Abra o painel de navegação à esquerda, escolha Gerenciamento de pilhas, Padrões de índice e, em seguida, escolha Criar padrão de índice. Para este tutorial, insira movies.
7. Escolha Próxima etapa e, em seguida, Criar padrão de índice. Depois que o padrão é criado, você pode visualizar os vários campos do documento, como `title` e `genre`.
8. Para começar a pesquisar seus dados, abra novamente o painel de navegação à esquerda e escolha Descobrir, ou use a API de pesquisa nas Ferramentas de desenvolvimento.

Tratamento de erros

Ao executar operações de indexação e pesquisa, você pode obter as seguintes respostas de erro:

- `HTTP 507`: indica que ocorreu um erro interno do servidor. Esse erro geralmente indica que suas unidades de OpenSearch computação (OCUs) estão sobrecarregadas pelo volume ou pela complexidade de suas solicitações. Embora o OpenSearch Serverless seja escalado

automaticamente para gerenciar a carga, pode haver um atraso na implantação de recursos adicionais.

Para mitigar esse erro, implemente uma política de repetição de tentativa com recuo exponencial. Essa abordagem reduz temporariamente a taxa de solicitações para administrar a carga com eficiência. Para obter mais detalhes, consulte o comportamento de repetição no Guia de referência de ferramentas AWS SDKs e ferramentas.

- `HTTP 402`— Indica que você atingiu o limite máximo de capacidade da unidade OpenSearch computacional (OCU). Otimize a workload para reduzir o uso de OCUs ou solicite um aumento de cota.

Etapa 4: Excluir a coleção

Como a coleção movies (filmes) é usada apenas para fins de teste, você deverá excluí-la quando terminar os testes.

Para excluir uma coleção OpenSearch sem servidor

1. Volte para o console do Amazon OpenSearch Service.
2. Escolha Coleções no painel de navegação à esquerda e selecione a coleção movies (filmes).
3. Escolha Excluir e confirme a exclusão.

Próximas etapas

Agora que você sabe como criar uma coleção e indexar dados, talvez você queira tentar alguns dos seguintes exercícios:

- Veja opções mais avançadas para a criação de uma coleção. Para saber mais, consulte the section called “Gerenciar coleções”.
- Saiba como configurar políticas de segurança para gerenciar a segurança da coleção em escala. Para saber mais, consulte the section called “Segurança sem OpenSearch servidor”.
- Descubra outras formas de indexar dados em coleções. Para obter mais informações, consulte the section called “Ingestão de dados em coleções”.

Coleções Amazon OpenSearch Serverless

Uma coleção no Amazon OpenSearch Serverless é um agrupamento lógico de um ou mais índices que representam uma carga de trabalho de análise. OpenSearch O Serverless gerencia e ajusta automaticamente a coleção, exigindo o mínimo de entrada manual.

Tópicos

- Gerenciando coleções Amazon OpenSearch Serverless
- Trabalho com coleções de pesquisa vetorial
- Usando políticas de ciclo de vida de dados com o Amazon Serverless OpenSearch
- Usando o AWS SDKs para interagir com o Amazon OpenSearch Serverless
- Usando CloudFormation para criar coleções Amazon OpenSearch Serverless
- Fazer backup de coleções usando snapshots
- Suporte ao codec Zstandard no Amazon Serverless OpenSearch
- Economize armazenamento usando a fonte derivada

Gerenciando coleções Amazon OpenSearch Serverless

Uma coleção no Amazon OpenSearch Serverless é um agrupamento lógico de um ou mais índices que representam uma carga de trabalho de análise. OpenSearch O Serverless gerencia e ajusta automaticamente a coleção, exigindo o mínimo de entrada manual.

Tópicos

- Configurar permissões para coleções
- Enriquecimento semântico automático para Serverless
- Criação de coleções
- Acessando OpenSearch painéis
- Exibição das coleções
- Exclusão de coleções

Configurar permissões para coleções

OpenSearch O Serverless usa as seguintes permissões AWS Identity and Access Management (IAM) para criar e gerenciar coleções. É possível especificar as condições do IAM para restringir os usuários a coleções específicas.

- `aoss:CreateCollection`: cria uma coleção.
- `aoss:ListCollections`: lista coleções na conta atual.
- `aoss:BatchGetCollection`: obtém detalhes sobre uma ou mais coleções.
- `aoss:UpdateCollection`: modifica uma coleção.
- `aoss:DeleteCollection`: exclui uma coleção.

O exemplo de política de acesso baseada em identidade a seguir fornece as permissões mínimas necessárias para que um usuário gerencie uma única coleção de nome `Logs`:

```
[
   {
      "Sid":"Allows managing logs collections",
      "Effect":"Allow",
      "Action":[
         "aoss:CreateCollection",
         "aoss:ListCollections",
         "aoss:BatchGetCollection",
         "aoss:UpdateCollection",
         "aoss:DeleteCollection",
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringEquals":{
            "aoss:collection":"Logs"
         }
      }
   }
]
```

`aoss:CreateAccessPolicy` e `aoss:CreateSecurityPolicy` estão incluídos porque as políticas de criptografia, de rede e de acesso a dados são necessárias para que uma coleção

funcione adequadamente. Para saber mais, consulte the section called “Gerenciamento de Identidade e Acesso”.

> **Note**
>
> Se você estiver criando a primeira coleção em sua conta, também precisará da permissão `iam:CreateServiceLinkedRole`. Para obter mais informações, consulte the section called “Função de criação de coleção”.

Enriquecimento semântico automático para Serverless

Introdução

O recurso de enriquecimento semântico automático pode ajudar a melhorar a relevância da pesquisa em até 20% em relação à pesquisa léxica. O enriquecimento semântico automático elimina o trabalho pesado indiferenciado de gerenciar sua própria infraestrutura de modelo de ML (aprendizado de máquina) e a integração com o mecanismo de pesquisa. O recurso está disponível para todos os três tipos de coleção sem servidor: Pesquisa, Série Temporal e Vetor.

O que é pesquisa semântica

Os mecanismos de pesquisa tradicionais dependem da word-to-word correspondência (conhecida como pesquisa léxica) para encontrar resultados para consultas. Embora isso funcione bem para consultas específicas, como números de modelos de televisão, ele tem dificuldades com pesquisas mais abstratas. Por exemplo, ao pesquisar por “sapatos para a praia”, uma pesquisa léxica apenas corresponde às palavras individuais “sapatos”, “praia”, “para” e “o” nos itens do catálogo, potencialmente sem produtos relevantes, como “sandálias resistentes à água” ou “calçados de surfe”, que não contêm os termos de pesquisa exatos.

A pesquisa semântica retorna resultados de consulta que incorporam não apenas a correspondência às palavras-chave, mas a intenção e o significado contextual da pesquisa do usuário. Por exemplo, se um usuário pesquisar “como tratar uma dor de cabeça”, um sistema de busca semântica pode retornar os seguintes resultados:

- Remédios para enxaqueca
- Técnicas de controle da dor
- Over-the-counter analgésicos

Detalhes do modelo e benchmark de desempenho

Embora esse recurso lide com as complexidades técnicas nos bastidores sem expor o modelo subjacente, fornecemos transparência por meio de uma breve descrição do modelo e resultados de benchmark para ajudá-lo a tomar decisões informadas sobre a adoção de recursos em suas cargas de trabalho críticas.

O enriquecimento semântico automático usa um modelo esparso pré-treinado e gerenciado por serviços que funciona de forma eficaz sem exigir ajustes personalizados. O modelo analisa os campos que você especifica, expandindo-os em vetores esparsos com base em associações aprendidas de diversos dados de treinamento. Os termos expandidos e seus pesos de significância são armazenados no formato nativo do índice Lucene para uma recuperação eficiente. Otimizamos esse processo usando o modo somente documentos, em que a codificação ocorre somente durante a ingestão de dados. As consultas de pesquisa são meramente tokenizadas em vez de processadas por meio de um modelo esparso, tornando a solução econômica e eficiente.

Nossa validação de desempenho durante o desenvolvimento do recurso usou o conjunto de dados de recuperação de passagens MS MARCO, com passagens com média de 334 caracteres. Para pontuação de relevância, medimos a média de ganho cumulativo com desconto normalizado (NDCG) para os primeiros 10 resultados de pesquisa (ndcg @10) no benchmark BEIR para conteúdo em inglês e a média de ndcg @10 no MIRACL para conteúdo multilíngue. Avaliamos a latência por meio de medições do percentil 90 (p90) do lado do cliente e a resposta de pesquisa p90 obteve valores. Esses benchmarks fornecem indicadores básicos de desempenho para a relevância da pesquisa e os tempos de resposta. Aqui estão os principais números de referência -

- Idioma inglês - Melhoria de relevância de 20% em relação à pesquisa léxica. Também reduziu a latência de pesquisa do P90 em 7,7% em relação à pesquisa lexical (BM25 é 26 ms e o enriquecimento semântico automático é 24 ms).
- Multilíngue - Melhoria de relevância de 105% em relação à pesquisa lexical, enquanto a latência da pesquisa P90 aumentou 38,4% em relação à pesquisa lexical (BM25 é 26 ms e o enriquecimento semântico automático é 36 ms).

Dada a natureza exclusiva de cada carga de trabalho, recomendamos que você avalie esse recurso em seu ambiente de desenvolvimento usando seus próprios critérios de benchmarking antes de tomar decisões de implementação.

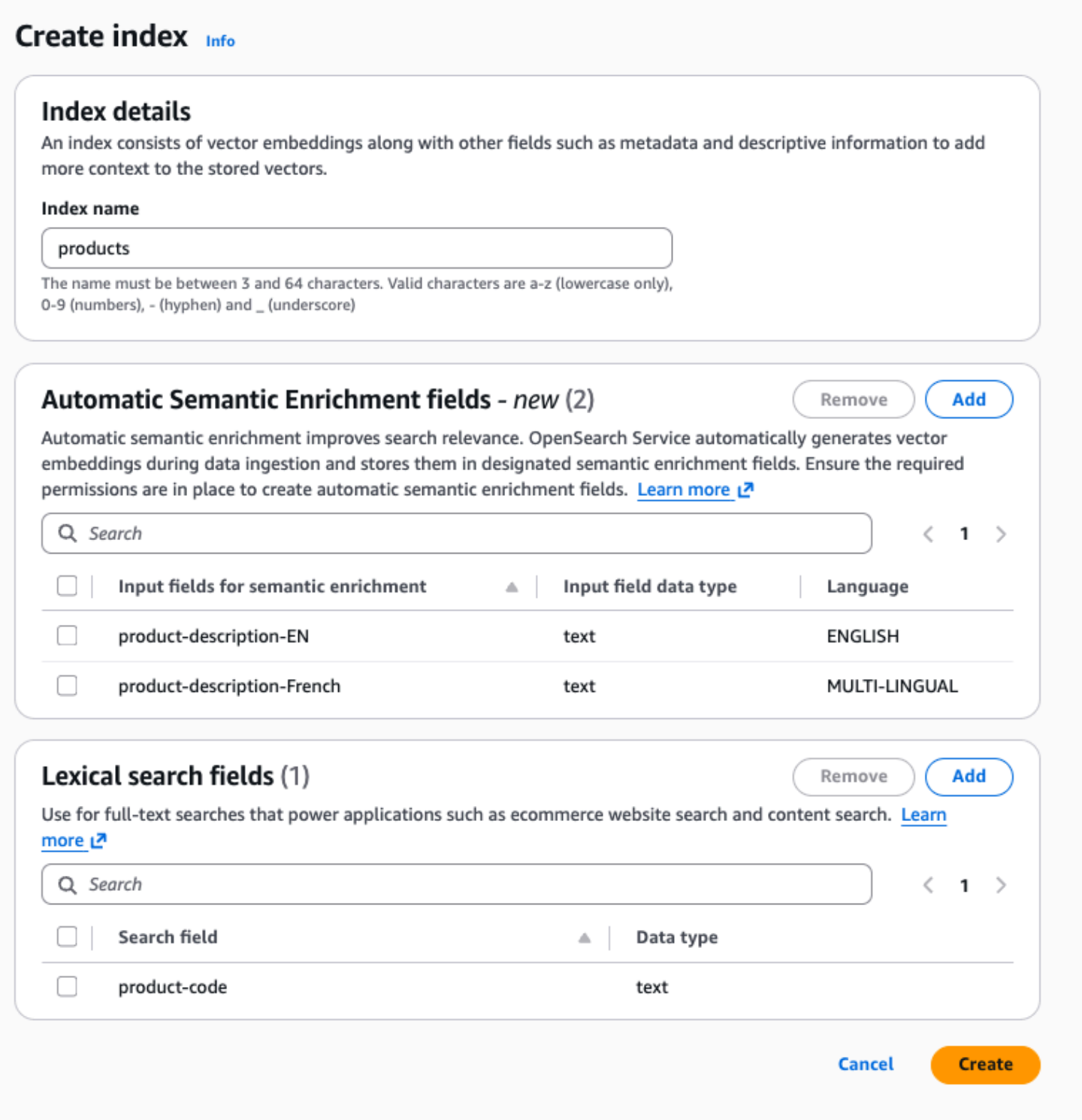
Idiomas suportados

O recurso oferece suporte ao inglês. Além disso, o modelo também oferece suporte para árabe, bengali, chinês, finlandês, francês, hindi, indonésio, japonês, coreano, persa, russo, espanhol, suaíli e telugu.

Configure um índice automático de enriquecimento semântico para coleções sem servidor

É fácil configurar um índice com o enriquecimento semântico automático ativado para seus campos de texto e você pode gerenciá-lo por meio do console e de CloudFormation modelos durante a criação do novo índice. APIs Para habilitá-lo para um índice existente, você precisa recriar o índice com o enriquecimento semântico automático ativado para campos de texto.

Experiência de console - O AWS console permite criar facilmente um índice com campos de enriquecimento semântico automático. Depois de selecionar uma coleção, você encontrará o botão criar índice na parte superior do console. Depois de clicar no botão criar índice, você encontrará opções para definir campos de enriquecimento semântico automático. Em um índice, você pode ter combinações de enriquecimento semântico automático para inglês e multilíngue, bem como campos lexicais.



Experiência de API - Para criar um índice automático de enriquecimento semântico usando a interface de linha de AWS comando (AWS CLI), use o comando create-index:

```
aws opensearchserverless create-index \
--id [collection_id] \
```

```
--index-name [index_name] \
--index-schema [index_body] \
```

No exemplo de esquema de índice a seguir, o campo title_semantic tem um tipo de campo definido como texto e o parâmetro semantic_enrichment definido como status ENABLED. Definir o parâmetro semantic_enrichment ativa o enriquecimento semântico automático no campo title_semantic. Você pode usar o campo language_options para especificar inglês ou multilíngue.

```
    aws opensearchserverless create-index \
    --id XXXXXXXXX \
    --index-name 'product-catalog' \
    --index-schema '{
    "mappings": {
        "properties": {
            "product_id": {
                "type": "keyword"
            },
            "title_semantic": {
                "type": "text",
                "semantic_enrichment": {
                    "status": "ENABLED",
                    "language_options": "english"
                }
            },
            "title_non_semantic": {
                "type": "text"
            }
        }
    }
}'
```

Para descrever o índice criado, use o seguinte comando:

```
aws opensearchserverless get-index \
--id [collection_id] \
--index-name [index_name] \
```

Você também pode usar CloudFormation modelos (Tipo:AWS::OpenSearchServerless::CollectionIndex) para criar uma pesquisa semântica durante o provisionamento da coleção, bem como após a criação da coleção.

Ingestão e pesquisa de dados

Depois de criar um índice com o enriquecimento semântico automático ativado, o recurso funciona automaticamente durante o processo de ingestão de dados, sem necessidade de configuração adicional.

Ingestão de dados: quando você adiciona documentos ao seu índice, o sistema automaticamente:

- Analisa os campos de texto que você designou para enriquecimento semântico
- Gera codificações semânticas usando o modelo esparso gerenciado por OpenSearch serviços
- Armazena essas representações enriquecidas junto com seus dados originais

Esse processo usa conectores OpenSearch de ML e pipelines de ingestão integrados, que são criados e gerenciados automaticamente nos bastidores.

Pesquisa: os dados de enriquecimento semântico já estão indexados, então as consultas são executadas com eficiência sem invocar o modelo de ML novamente. Isso significa que você obtém maior relevância de pesquisa sem sobrecarga adicional de latência de pesquisa.

Configurar permissões para enriquecimento semântico automático

Antes de criar um índice de enriquecimento semântico automático, é necessário configurar as permissões necessárias. Esta seção explica as permissões necessárias e como configurá-las.

Permissões de política do IAM

Use a seguinte política AWS Identity and Access Management (IAM) para conceder as permissões necessárias para trabalhar com o enriquecimento semântico automático:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AutomaticSemanticEnrichmentPermissions",
            "Effect": "Allow",
```

```
            "Action": [
                "aoss:CreateIndex",
                "aoss:GetIndex",
                "aoss:UpdateIndex",
                "aoss:DeleteIndex",
                "aoss:APIAccessAll"
            ],
            "Resource": "*"
        }
    ]
}
```

Permissões de chave

- As permissões `aoss:*Index` habilitam o gerenciamento de índices
- A `aoss:APIAccessAll` permissão permite operações OpenSearch de API
- Para restringir as permissões a uma coleção específica, substitua `"Resource": "*"` pelo ARN da coleção

Configurar permissões de acesso a dados

Para configurar um índice para enriquecimento semântico automático, você deve ter as políticas de acesso a dados apropriadas que concedam permissão para acessar recursos de índice, pipeline e coleção de modelos. Para saber mais sobre políticas de acesso a dados, consulte Controle de acesso a dados para Amazon OpenSearch Serverless. Para obter o procedimento de configuração de uma política de acesso a dados, consulte Criação de políticas de acesso a dados (console).

Permissões de acesso a dados

```
[
    {
        "Description": "Create index permission",
        "Rules": [
            {
                "ResourceType": "index",
                "Resource": ["index/collection_name/*"],
                "Permission": [
                  "aoss:CreateIndex",
                  "aoss:DescribeIndex",
                  "aoss:UpdateIndex",
                  "aoss:DeleteIndex"
```

```
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create pipeline permission",
        "Rules": [
            {
                "ResourceType": "collection",
                "Resource": ["collection/collection_name"],
                "Permission": [
                  "aoss:CreateCollectionItems",
                  "aoss:DescribeCollectionItems"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create model permission",
        "Rules": [
            {
                "ResourceType": "model",
                "Resource": ["model/collection_name/*"],
                "Permission": ["aoss:CreateMLResource"]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
]
```

Permissões de acesso à rede

Para permitir que APIs o serviço acesse coleções particulares, você deve configurar políticas de rede que permitam o acesso necessário entre a API do serviço e a coleção. Para obter mais informações sobre políticas de rede, consulte Acesso à rede para Amazon OpenSearch Serverless.

```
[
   {
      "Description":"Enable automatic semantic enrichment in a private collection",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/collection_name"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceServices":[
         "aoss.amazonaws.com"
      ],
   }
]
```

Para configurar as permissões de acesso à rede para uma coleção privada

1. Faça login no console OpenSearch de serviço em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Políticas de rede. Depois, siga um destes procedimentos:
   - Escolha o nome de uma política existente e escolha Editar
   - Escolha Criar política de rede e configure os detalhes da política
3. Na área Tipo de acesso, escolha Privado (recomendado) e selecione Acesso privado ao serviço da AWS .
4. No campo de pesquisa, escolha Serviço e depois aoss.amazonaws.com.
5. Na área Tipo de recurso, selecione a caixa Habilitar acesso ao OpenSearch endpoint.
6. Em Pesquisar coleções ou inserir termos de prefixo específicos, no campo de pesquisa, selecione Nome da coleção. Em seguida, insira ou selecione o nome das coleções a serem associadas à política de rede.
7. Escolha Criar para uma nova política de rede ou Atualizar para uma política de rede existente.

Regravações de consultas

O enriquecimento semântico automático converte automaticamente suas consultas de "correspondência" existentes em consultas de pesquisa semântica sem exigir modificações na consulta. Se uma consulta de correspondência fizer parte de uma consulta composta, o sistema percorrerá sua estrutura de consulta, localizará consultas de correspondência e as substituirá por consultas neurais esparsas. Atualmente, o recurso suporta apenas a substituição de consultas "correspondentes", seja uma consulta autônoma ou parte de uma consulta composta. "multi_match" não é suportado. Além disso, o recurso oferece suporte a todas as consultas compostas para substituir suas consultas de correspondência aninhadas. As consultas compostas incluem: bool, boosting, constant_score, dis_max, function_score e hybrid.

Limitações do enriquecimento semântico automático

A pesquisa semântica automática é mais eficaz quando aplicada a campos small-to-medium de tamanho contendo conteúdo em linguagem natural, como títulos de filmes, descrições de produtos, avaliações e resumos. Embora a pesquisa semântica aumente a relevância para a maioria dos casos de uso, ela pode não ser ideal para determinados cenários. Considere as seguintes limitações ao decidir se deve implementar o enriquecimento semântico automático para seu caso de uso específico.

- Documentos muito longos — O modelo esparso atual processa apenas os primeiros 8.192 tokens de cada documento em inglês. Para documentos multilíngues, são 512 tokens. Para artigos longos, considere implementar a fragmentação de documentos para garantir o processamento completo do conteúdo.
- Cargas de trabalho de análise de registros — O enriquecimento semântico aumenta significativamente o tamanho do índice, o que pode ser desnecessário para a análise de registros, onde a correspondência exata normalmente é suficiente. O contexto semântico adicional raramente melhora a eficácia da pesquisa de registros o suficiente para justificar o aumento dos requisitos de armazenamento.
- O enriquecimento semântico automático não é compatível com o recurso Fonte Derivada.
- Limitação — As solicitações de inferência de indexação estão atualmente limitadas a 100 TPS para o Serverless. OpenSearch Esse é um limite flexível; entre em contato com o AWS Support para obter limites mais altos.

Preços

OpenSearch O Serverless cobra o enriquecimento semântico automático com base nas unidades de OpenSearch computação (OCUs) consumidas durante a geração de vetores esparsos no momento da indexação. Você é cobrado somente pelo uso real durante a indexação. Você pode monitorar esse consumo usando a SemanticSearch OCU CloudWatch métrica da Amazon. Para obter detalhes específicos sobre os limites do token do modelo, a taxa de transferência de volume por OCU e um exemplo de cálculo de amostra, visite Preços de OpenSearch serviços.

Criação de coleções

Você pode usar o console ou o AWS CLI para criar uma coleção sem servidor. Essas etapas abordam como criar uma pesquisa ou uma coleção de séries temporais. Para criar uma coleção de pesquisa vetorial, consultethe section called “Trabalho com coleções de pesquisa vetorial”.

Tópicos

- Criação de uma coleção (console)
- Criação de uma coleção (CLI)

Criação de uma coleção (console)

Use os procedimentos desta seção para criar uma coleção usando o Console de gerenciamento da AWS. Essas etapas abordam como criar uma pesquisa ou uma coleção de séries temporais. Para criar uma coleção de pesquisa vetorial, consultethe section called “Trabalho com coleções de pesquisa vetorial”.

Tópicos

- Definir configurações de coleção
- Configurar campos de pesquisa adicionais

Definir configurações de coleção

Use o procedimento a seguir para configurar as informações sobre a coleção.

Para definir as configurações de coleção usando o console

1. Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/home/.

2. Expanda Sem Servidor no painel de navegação à esquerda e escolha Coleções.
3. Escolha Criar coleção.
4. Forneça um nome e uma descrição para a coleção. O nome deve atender aos seguintes critérios:

   - É exclusivo para sua conta e Região da AWS
   - Contém apenas letras minúsculas a-z, números de 0 a 9 e hífens (-).
   - Contém de 3 a 32 caracteres
5. Escolha um tipo de coleção:

   - Séries temporais: segmento de analytics de logs que se concentra na análise de grandes volumes de dados semiestruturados gerados por máquina. Pelo menos 24 horas de dados são armazenadas em índices ativos, e o restante permanece no armazenamento ativo.
   - Pesquisa: pesquisa de texto completo que alimenta aplicações em suas redes internas e aplicações voltadas para a Internet. Todos os dados de pesquisa são armazenados em um armazenamento de atividade muito alta para garantir tempos de resposta rápidos às consultas.

     > **Note**
     >
     > Escolha essa opção se você estiver habilitando a pesquisa semântica automática, como descrito em Definir configurações de coleção.
   - Pesquisa vetorial: pesquisa semântica em incorporações vetoriais que simplifica o gerenciamento de dados vetoriais. Potencializa experiências de pesquisa aumentada de machine learning (ML) e aplicações de IA generativa, como chatbots, assistentes pessoais e detecção de fraudes.

   Para saber mais, consulte the section called “Escolha de um tipo de coleção”.
6. Em Tipo de implantação, escolha a configuração de redundância para a coleção. Por padrão, cada coleção tem redundância, o que significa que cada uma das unidades OpenSearch computacionais de indexação e pesquisa (OCUs) tem suas próprias réplicas em espera em uma zona de disponibilidade diferente. Para fins de desenvolvimento e teste, você pode optar por desativar a redundância, o que reduz o número de OCUs em sua coleção para dois. Para obter mais informações, consulte the section called “Como funciona”.
7. Em Segurança, escolha Criação padrão.

8. Em Criptografia, escolha uma AWS KMS chave para criptografar seus dados. OpenSearch O Serverless notifica você se o nome da coleção que você inseriu corresponder a um padrão definido em uma política de criptografia. É possível optar por manter essa correspondência ou substituí-la por configurações de criptografia exclusivas. Para saber mais, consulte the section called “Criptografia”.
9. Em Configurações de acesso à rede, configure o acesso à rede para a coleção.

   - Em Tipo de acesso, selecione público ou privado.

     Se você escolher privado, especifique quais endpoints da VPC e Serviços da AWS podem acessar a coleção.

     - Endpoints da VPC para acesso: especifique um ou mais endpoints da VPC para permitir o acesso por eles. Para criar um VPC da endpoint, consulte the section called “Endpoint VPC do plano de dados”.
     - AWS service (Serviço da AWS) acesso privado — Selecione um ou mais serviços compatíveis aos quais permitir o acesso.
   - Em Tipo de recurso, selecione se os usuários podem acessar a coleção por meio de seu OpenSearchendpoint (para fazer chamadas de API por meio de cURL, Postman e assim por diante), por meio OpenSearch do endpoint Dashboards (para trabalhar com visualizações e fazer chamadas de API por meio do console) ou ambos.

     > **Note**
     >
     > AWS service (Serviço da AWS) o acesso privado se aplica somente ao OpenSearch endpoint, não ao endpoint do OpenSearch Dashboards.

   OpenSearch O Serverless notifica você se o nome da coleção inserido corresponder a um padrão definido em uma política de rede. É possível optar por manter essa correspondência ou substituí-la por configurações de rede personalizadas. Para saber mais, consulte the section called “Acesso à rede”.
10. (Opcional) Adicione uma ou mais tags à coleção. Para saber mais, consulte the section called “Aplicação de tags nas coleções”.
11. Escolha Próximo.

Configurar campos de pesquisa adicionais

As opções que você vê na segunda página do fluxo de trabalho de criação de coleção dependem do tipo de coleção sendo criada. Esta seção descreve como configurar campos de pesquisa adicionais para cada tipo de coleção. Esta seção também descreve como configurar o enriquecimento semântico automático. Ignore qualquer seção que não se aplique a seu tipo de coleção.

Tópicos

- Configurar enriquecimento semântico automático
- Configurar campos de pesquisa de séries temporais
- Configurar campos de pesquisa lexical
- Configurar campos de pesquisa vetorial

Configurar enriquecimento semântico automático

Ao criar ou editar uma coleção, você pode configurar o enriquecimento semântico automático, o que simplifica a implementação e os recursos da pesquisa semântica no Amazon Service. OpenSearch A pesquisa semântica retorna resultados de consulta que incorporam não apenas a correspondência às palavras-chave, mas a intenção e o significado contextual da pesquisa do usuário. Para saber mais, consulte Enriquecimento semântico automático para Serverless.

Para configurar o enriquecimento semântico automático

1. Na seção Detalhes do índice, em Nome do índice, especifique um nome.
2. Na seção Campos de enriquecimento semântico automático, escolha Adicionar campo de pesquisa semântica.
3. No campo Nome do campo de entrada para enriquecimento semântico, insira o nome do campo que você deseja enriquecer.
4. O Tipo de dado é Texto. Não é possível alterar esse valor.
5. Em Idioma, escolha português ou multilíngue.
6. Escolha Adicionar campo.
7. Depois de concluir a configuração dos campos opcionais para a coleção, escolha Avançar. Revise as alterações e escolha Enviar para criar a coleção.

Configurar campos de pesquisa de séries temporais

As opções na seção Campos de pesquisa de séries temporais referem-se a dados de séries temporais e fluxos de dados. Para saber mais sobre esses assuntos, consulte Gerenciando dados de séries temporais no Amazon OpenSearch Service com fluxos de dados.

Para configurar campos de pesquisa de séries temporais

1. Na seção Campos de pesquisa de séries temporais, escolha Adicionar campo de série temporal.
2. Em Nome do filtro, insira um nome.
3. Em Tipo de dados, escolha um tipo na lista.
4. Escolha Adicionar campo
5. Depois de concluir a configuração dos campos opcionais para a coleção, escolha Avançar. Revise as alterações e escolha Enviar para criar a coleção.

Configurar campos de pesquisa lexical

A pesquisa lexical busca uma correspondência exata entre uma consulta de pesquisa e os termos ou palavras-chave indexados.

Para configurar campos de pesquisa lexical

1. Na seção Campos de pesquisa lexical, escolha Adicionar campo de pesquisa.
2. Em Nome do filtro, insira um nome.
3. Em Tipo de dados, escolha um tipo na lista.
4. Escolha Adicionar campo
5. Depois de concluir a configuração dos campos opcionais para a coleção, escolha Avançar. Revise as alterações e escolha Enviar para criar a coleção.

Configurar campos de pesquisa vetorial

Para configurar campos de pesquisa vetorial

1. Na seção Campos vetoriais, escolha Adicionar campo vetorial.
2. Em Nome do filtro, insira um nome.
3. Em Mecanismo, escolha um tipo na lista.

4. Insira o número de dimensões.
5. Em Métrica de distância, escolha um tipo na lista.
6. Depois de concluir a configuração dos campos opcionais para a coleção, escolha Avançar.
7. Revise as alterações e escolha Enviar para criar a coleção.

Criação de uma coleção (CLI)

Use os procedimentos desta seção para criar uma coleção OpenSearch sem servidor usando o. AWS CLI

Tópicos

- Antes de começar
- Criar uma coleção
- Criando uma coleção com um índice de enriquecimento semântico automático

Antes de começar

Antes de criar uma coleção usando o AWS CLI, use o procedimento a seguir para criar as políticas necessárias para a coleção.

> **Note**
>
> Em todos os procedimentos a seguir, quando você especifica um nome para uma coleção, ele deve atender aos seguintes critérios:
>
> - É exclusivo para sua conta e Região da AWS
> - Contém apenas letras minúsculas a-z, números de 0 a 9 e hífens (-).
> - Contém de 3 a 32 caracteres

Para criar as políticas necessárias para uma coleção

1. Abra AWS CLI e execute o comando a seguir para criar uma política de criptografia com um padrão de recurso que corresponda ao nome pretendido da coleção.

```
&aws opensearchserverless create-security-policy \
```

```
  --name policy name \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection\/collection name\"]}],\"AWSOwnedKey\":true}"
```

Por exemplo, se você planeja nomear sua coleção como logs-application, é possível criar uma política de criptografia como esta:

```
&aws opensearchserverless create-security-policy \
  --name logs-policy \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection\/logs-application\"]}],\"AWSOwnedKey\":true}"
```

Se você planeja usar a política para cobranças adicionais, é possível tornar a regra mais ampla, como `collection/logs*` ou `collection/*`.

2. Execute o comando a seguir para definir as configurações de rede para a coleção usando uma política de rede. É possível criar políticas de rede depois de criar uma coleção, mas recomendamos fazer isso previamente.

```
&aws opensearchserverless create-security-policy \
  --name policy name \
  --type network --policy "[{\"Description\":\"description\",\"Rules\":
[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/collection name\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/collection name\"]}],
\"AllowFromPublic\":true}]"
```

Usando o exemplo anterior de logs-application, é possível criar a seguinte política de rede:

```
&aws opensearchserverless create-security-policy \
  --name logs-policy \
  --type network --policy "[{\"Description\":\"Public access for logs collection
\",\"Rules\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/logs-
application\"]},{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/logs-
application\"]}],\"AllowFromPublic\":true}]"
```

Criar uma coleção

O procedimento a seguir usa a ação CreateCollectionda API para criar uma coleção do tipo `SEARCH` ou`TIMESERIES`. Se você não especificar um tipo de coleção na solicitação, ela assumirá o padrão `TIMESERIES`. Para saber mais sobre tipos, consulte Escolha de um tipo de coleção. Para criar

uma coleção de pesquisa vetorial, consultethe section called “Trabalho com coleções de pesquisa vetorial”.

Se sua coleção for criptografada com um Chave pertencente à AWS, o `kmsKeyArn` é `auto` em vez de um ARN.

> **Important**
>
> Depois de criar uma coleção, você não poderá acessá-la, a menos que ela corresponda a uma política de acesso a dados. Para saber mais, consulte Controle de acesso a dados para Amazon OpenSearch Serverless.

Como criar uma coleção

1. Verifique se você criou as políticas necessárias descritas em Antes de começar.
2. Execute o comando a seguir. Em `type`, especifique `SEARCH` ou `TIMESERIES`.

```
&aws opensearchserverless create-collection --name "collection name" --
type collection type --description "description"
```

Criando uma coleção com um índice de enriquecimento semântico automático

Use o procedimento a seguir para criar uma nova coleção OpenSearch sem servidor com um índice configurado para enriquecimento semântico automático. O procedimento usa a ação da CreateIndexAPI OpenSearch Serverless.

Para criar uma nova coleção com um índice configurado para enriquecimento semântico automático

Execute o comando a seguir para criar uma coleção e um índice.

```
&aws opensearchserverless create-index \
--region Region ID \
--id collection name --index-name index name \
--index-schema \
'mapping in json'
```

Aqui está um exemplo.

```
&aws opensearchserverless create-index \
```

```
--region us-east-1 \
--id conversation_history --index-name conversation_history_index \
--index-schema \
'{
    "mappings": {
        "properties": {
            "age": {
                "type": "integer"
            },
            "name": {
                "type": "keyword"
            },
            "user_description": {
                "type": "text"
            },
            "conversation_history": {
                "type": "text",
                "semantic_enrichment": {
                    "status": "ENABLED",
                    // Specifies the sparse tokenizer for processing multi-lingual text
                    "language_option": "MULTI-LINGUAL",
                    // If embedding_field is provided, the semantic embedding field
 will be set to the given name rather than original field name + "_embedding"
                    "embedding_field": "conversation_history_user_defined"
                }
            },
            "book_title": {
                "type": "text",
                "semantic_enrichment": {
                    // No embedding_field is provided, so the semantic embedding field
 is set to "book_title_embedding"
                    "status": "ENABLED",
                    "language_option": "ENGLISH"
                }
            },
            "abstract": {
                "type": "text",
                "semantic_enrichment": {
                    // If no language_option is provided, it will be set to English.
                    // No embedding_field is provided, so the semantic embedding field
 is set to "abstract_embedding"
                    "status": "ENABLED"
                }
            }
```

```
        }
    }
}'
```

Acessando OpenSearch painéis

Depois de criar uma coleção com o Console de gerenciamento da AWS, você pode navegar até a URL dos OpenSearch painéis da coleção. Você pode encontrar o URL dos Painéis escolhendo Coleções no painel de navegação esquerdo e selecionando a coleção para abrir a página de detalhes. O URL assume o formato `https://dashboards.us-east-1.aoss.amazonaws.com/_login/?collectionId=07tjusf2h91cunochc`. Depois de navegar até o URL, você fará login no Dashboards automaticamente.

Se você já tiver o URL dos OpenSearch painéis disponível, mas não estiver no Console de gerenciamento da AWS, chamar o URL dos painéis pelo navegador será redirecionado para o console. Depois de inserir suas AWS credenciais, você fará login automaticamente nos painéis. Para obter informações sobre como acessar coleções para SAML, consulte Acessando OpenSearch painéis com SAML.

O tempo limite do console do OpenSearch Dashboards é de uma hora e não é configurável.

> **Note**
>
> Em 10 de maio de 2023, OpenSearch introduziu um endpoint global comum para OpenSearch painéis. Agora você pode navegar até OpenSearch Painéis no navegador com uma URL que assume o formato`https://dashboards.us-east-1.aoss.amazonaws.com/_login/?collectionId=07tjusf2h91cunochc`. Para garantir a compatibilidade com versões anteriores, continuaremos oferecendo suporte aos endpoints de OpenSearch painéis específicos da coleção existente com o formato. `https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/_dashboards`

Exibição das coleções

Você pode visualizar as coleções existentes Conta da AWS na sua guia Coleções do console do Amazon OpenSearch Service.

Para listar as coleções junto com as suas IDs, envie uma ListCollectionssolicitação.

```
&aws opensearchserverless list-collections
```

Exemplo de resposta

```
{
   "collectionSummaries":[
      {
         "arn":"arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "id":"07tjusf2h91cunochc",
         "name":"my-collection",
         "status":"CREATING"
      }
   ]
}
```

Para limitar os resultados da pesquisa, use filtros de coleções. Esta solicitação filtra a resposta para coleções no estado `ACTIVE`:

```
&aws opensearchserverless list-collections --collection-filters '{ "status":
 "ACTIVE" }'
```

Para obter informações mais detalhadas sobre uma ou mais coleções, incluindo o OpenSearch endpoint e o endpoint do OpenSearch Dashboards, envie uma solicitação: BatchGetCollection

```
&aws opensearchserverless batch-get-collection --ids "07tjusf2h91cunochc"
 "1iu5usc4rame"
```

> **Note**
>
> É possível incluir `--names` ou `--ids` na solicitação, mas não os dois.

Exemplo de resposta

```
{
   "collectionDetails":[
      {
         "id": "07tjusf2h91cunochc",
         "name": "my-collection",
```

```
         "status": "ACTIVE",
         "type": "SEARCH",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://07tjusf2h91cunochc.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/
_dashboards"
      },
      {
         "id": "178ukvtg3i82dvopdid",
         "name": "another-collection",
         "status": "ACTIVE",
         "type": "TIMESERIES",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/178ukvtg3i82dvopdid",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com/_dashboards"
      }
   ],
   "collectionErrorDetails":[]
}
```


Exclusão de coleções

A exclusão de uma coleção exclui todos os dados e índices da coleção. Você não poderá recuperar coleções depois de excluí-las.

Para excluir uma coleção usando o console

1. No painel Coleções do console do Amazon OpenSearch Service, selecione a coleção que você deseja excluir.
2. Escolha Excluir e confirme a exclusão.

Para excluir uma coleção usando o AWS CLI, envie uma DeleteCollectionsolicitação:

```
&aws opensearchserverless delete-collection --id 07tjusf2h91cunochc
```

Exemplo de resposta

```
{
   "deleteCollectionDetail":{
      "id":"07tjusf2h91cunochc",
      "name":"my-collection",
      "status":"DELETING"
   }
}
```

Trabalho com coleções de pesquisa vetorial

O tipo de coleta de pesquisa vetorial no OpenSearch Serverless fornece um recurso de pesquisa por similaridade que é escalável e de alto desempenho. Isso facilita a criação de experiências modernas de pesquisa aumentada de machine learning (ML) e aplicativos de inteligência artificial generativa (IA) sem precisar gerenciar a infraestrutura subjacente do banco de dados de vetores.

Os casos de uso de coleções de pesquisa vetorial incluem pesquisas de imagens, pesquisas de documentos, recuperação de músicas, recomendações de produtos, pesquisas de vídeo, pesquisas baseadas em localização, detecção de fraudes e detecção de anomalias.

Como o mecanismo vetorial do OpenSearch Serverless é alimentado pelo recurso de pesquisa k-Nearest Neighbor (k-NN) OpenSearch, você obtém a mesma funcionalidade com a simplicidade de um ambiente sem servidor. O mecanismo é compatível com a API do plug-in k-NN. Com essas operações, você pode aproveitar pesquisas em texto completo, filtragem avançada, agregações, consultas geoespaciais, consultas aninhadas para uma recuperação mais rápida dos dados e resultados de pesquisa aprimorados.

O mecanismo vetorial fornece métricas de distância, como distância euclidiana, similaridade de cosseno, similaridade de produtos escalares, e também pode acomodar 16.000 dimensões. Você pode armazenar campos com vários tipos de dados para metadados, como números, booleanos, datas, palavras-chave e pontos geográficos. Também é possível armazenar campos com texto para obter informações descritivas e adicionar mais contexto aos vetores armazenados. A colocalização conjunta dos tipos de dados reduz a complexidade, aumenta a capacidade de manutenção e evita a duplicação de dados, desafios de compatibilidade de versões e problemas de licenciamento.

> **Note**
>
> Observe as seguintes informações:
>
> - O Amazon OpenSearch Serverless oferece suporte à quantização escalar Faiss de 16 bits, que pode ser usada para realizar conversões entre vetores flutuantes de 32 bits e vetores de 16 bits. Para saber mais, consulte Quantização escalar de Faiss de 16-bits. Você também pode usar vetores binários para reduzir os custos de memória. Para saber mais, consulte Vetores binários.
> - O Amazon OpenSearch Serverless oferece suporte à pesquisa vetorial baseada em disco. A pesquisa vetorial baseada em disco reduz significativamente os custos operacionais de workloads vetoriais em ambientes com pouca memória. Para saber mais, consulte Pesquisa vetorial baseada em disco.

Conceitos básicos de coleções de pesquisa de vetores

Neste tutorial, você conclui as etapas a seguir para armazenar, pesquisar e recuperar incorporações vetoriais em tempo real:

1. Configurar permissões
2. Criar uma coleção
3. Transferir e pesquisar dados
4. Excluir a coleção

Etapa 1: configurar permissões

Para concluir este tutorial (e usar o OpenSearch Serverless em geral), você deve ter as permissões corretas AWS Identity and Access Management (IAM). Neste tutorial, você criará uma coleção, carregará e pesquisará dados e, em seguida, excluirá a coleção.

Seu usuário ou função deve ter uma política baseada em identidade anexada com as seguintes permissões mínimas:

JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "iam:ListUsers",
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Para obter mais informações sobre as permissões do IAM OpenSearch sem servidor, consulte. the section called “Gerenciamento de Identidade e Acesso”

Etapa 2: criar uma coleção

Uma coleção é um grupo de OpenSearch índices que trabalham juntos para dar suporte a uma carga de trabalho ou caso de uso específico.

Para criar uma coleção OpenSearch sem servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Escolha Coleções no painel de navegação à esquerda e escolha Criar coleção.
3. Nomeie o armazenamento da coleção.
4. Para o tipo de coleção, escolha Pesquisa vetorial. Para saber mais, consulte the section called “Escolha de um tipo de coleção”.
5. Em Tipo de implantação, desmarque Habilitar redundância (réplicas ativas). Isso cria uma coleção no modo de desenvolvimento ou teste e reduz o número de unidades de OpenSearch computação (OCUs) em sua coleção para duas. Se quiser criar um ambiente de produção neste tutorial, deixe a caixa de seleção marcada.

6. Em Segurança, selecione Criação fácil para simplificar sua configuração de segurança. Por padrão, todos os dados no mecanismo vetorial são criptografados em trânsito e em repouso. O mecanismo vetorial oferece suporte a permissões refinadas do IAM para que você possa definir quem pode criar, atualizar e excluir criptografias, redes, coleções e índices.
7. Escolha Próximo.
8. Reveja suas configurações da coleção e escolha Enviar. Aguarde alguns minutos até que o status da coleção mude para `Active`.

Etapa 3: Transferir e pesquisar dados

Um índice é uma coleção de documentos com um esquema de dados comum que fornece uma maneira de armazenar, pesquisar e recuperar suas incorporações vetoriais e outros campos. Você pode criar e carregar dados para índices em uma coleção OpenSearch sem servidor usando o console Dev Tools em OpenSearch painéis ou uma ferramenta HTTP, como Postman ou awscurl. Este tutorial usa o Dev Tools.

Para indexar e pesquisar dados na coleção habitacional

1. Para criar um único índice para a nova coleção, envie a solicitação a seguir no console do Dev Tools. Por padrão, isso cria um índice com um `nmslib` mecanismo e uma distância euclidiana.

```
PUT housing-index
{
   "settings": {
      "index.knn": true
   },
   "mappings": {
      "properties": {
         "housing-vector": {
            "type": "knn_vector",
            "dimension": 3
         },
         "title": {
            "type": "text"
         },
         "price": {
            "type": "long"
         },
         "location": {
            "type": "geo_point"
```

```
          }
       }
    }
}
```

2. Para indexar um único documento em housing-index, envie a seguinte solicitação:

```
POST housing-index/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
  "price": "2800",
  "location": "47.71, 122.00"
}
```

3. Para pesquisar propriedades semelhantes às do seu índice, envie a seguinte consulta:

```
GET housing-index/_search
{
    "size": 5,
    "query": {
        "knn": {
            "housing-vector": {
                "vector": [
                    10,
                    20,
                    30
                ],
                "k": 5
            }
        }
    }
}
```

Etapa 4: Excluir a coleção

Como a coleção habitação é usada apenas para fins de teste, você deverá excluí-la quando terminar os testes.

Para excluir uma coleção OpenSearch sem servidor

1. Volte para o console do Amazon OpenSearch Service.
2. Escolha Coleções no painel de navegação à esquerda e selecione a coleção propriedades.
3. Selecione Excluir para confirmar a exclusão.

Pesquisa com filtro

Você pode usar filtros para refinar os resultados da pesquisa semântica. Para criar um índice e realizar uma pesquisa com filtro nos seus documentos, substitua Carregar e pesquisar dados no tutorial anterior pelas instruções a seguir. As outras etapas permanecem as mesmas. Para saber mais sobre os filtros, consulte pesquisa k-NN com filtros.

Para indexar e pesquisar dados na coleção habitacional

1. Para criar um único índice para a coleção, envie a seguinte solicitação no console do Dev Tools:

```
PUT housing-index-filtered
{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "housing-vector": {
        "type": "knn_vector",
        "dimension": 3,
        "method": {
          "engine": "faiss",
          "name": "hnsw"
        }
      },
      "title": {
        "type": "text"
      },
      "price": {
        "type": "long"
      },
      "location": {
        "type": "geo_point"
      }
```

```
    }
  }
}
```

2. Para indexar um único documento em housing-index-filtered, envie a seguinte solicitação:

```
POST housing-index-filtered/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
  "price": "2800",
  "location": "47.71, 122.00"
}
```

3. Para pesquisar seus dados em busca de um apartamento em Seattle por um preço específico e dentro de uma determinada distância de um ponto geográfico, envie a seguinte solicitação:

```
GET housing-index-filtered/_search
{
  "size": 5,
  "query": {
    "knn": {
      "housing-vector": {
        "vector": [
          0.1,
          0.2,
          0.3
        ],
        "k": 5,
        "filter": {
          "bool": {
            "must": [
              {
                "query_string": {
                  "query": "Find me 2 bedroom apartment in Seattle under $3000 ",
                  "fields": [
                    "title"
                  ]
                }
```

```
              },
              {
                "range": {
                  "price": {
                    "lte": 3000
                  }
                }
              },
              {
                "geo_distance": {
                  "distance": "100miles",
                  "location": {
                    "lat": 48,
                    "lon": 121
                  }
                }
              }
            ]
          }
        }
      }
    }
  }
}
```

Workloads em escala de bilhões

Coleções de pesquisa vetorial oferecem suporte a workloads com bilhões de vetores. Você não precisa reindexar para fins de ajuste de escala, pois o ajuste de escala automático faz isso por você. Se você tiver milhões de vetores (ou mais) com um grande número de dimensões e precisar de mais de 200 OCUs, entre em contato com o AWS Support para aumentar o número máximo de unidades OpenSearch computacionais (OCUs) para sua conta.

Limitações

As coleções de pesquisa vetorial têm as seguintes limitações:

- As coleções de pesquisa vetorial não são compatíveis com o mecanismo Apache Lucene ANN.
- As coleções de pesquisa vetorial são compatíveis apenas com o algoritmo HNSW com Faiss e não são compatíveis com FIV e IVFQ.

- As coleções de pesquisa vetorial não são compatíveis com as operações de API de aquecimento, estatísticas e treinamento de modelo.
- As coleções de pesquisa vetorial não oferecem suporte a scripts embutidos ou armazenados.
- As informações de contagem de índices não estão disponíveis nas coleções Console de gerenciamento da AWS de pesquisa vetorial.
- O intervalo de atualização dos índices nas coleções de pesquisa vetorial é de 60 segundos.

Próximas etapas

Agora que você sabe como criar uma coleção de pesquisa vetorial e indexar os dados, talvez você queira testar alguns dos seguintes exercícios:

- Use o cliente OpenSearch Python para trabalhar com coleções de pesquisa vetorial. Veja este tutorial em GitHub.
- Use o cliente OpenSearch Java para trabalhar com coleções de pesquisa vetorial. Veja este tutorial em GitHub.
- Configure LangChain para usar OpenSearch como um repositório de vetores. LangChain é uma estrutura de código aberto para o desenvolvimento de aplicativos alimentados por modelos de linguagem. Para obter mais informações, consulte a documentação do LangChain .

Usando políticas de ciclo de vida de dados com o Amazon Serverless OpenSearch

Uma política de ciclo de vida de dados no Amazon OpenSearch Serverless define por quanto tempo o OpenSearch Serverless retém os dados em uma coleção de séries temporais. Por exemplo, você pode definir uma política para reter dados de log por 30 dias antes que o OpenSearch Serverless os exclua.

Você pode configurar uma política separada para cada índice em cada coleção de séries temporais em seu Conta da AWS. OpenSearch O Serverless retém os documentos pelo menos pela duração especificada na política. Depois os exclui automaticamente assim que possível, normalmente dentro de 48 horas ou 10% do período de retenção, o que for maior.

Somente coleções de séries temporais oferecem suporte às políticas de ciclo de vida dos dados. As coleções de pesquisa e pesquisa vetorial não são.

Tópicos

- Políticas de ciclo de vida de dados
- Permissões obrigatórias
- Precedência das políticas
- Sintaxe da política
- Criar políticas de ciclo de vida de dados
- Atualização de políticas de ciclo de vida de dados
- Como excluir políticas de ciclo de vida dos dados

Políticas de ciclo de vida de dados

Em uma política de ciclo de vida dos dados, você especifica uma série de regras. A política de ciclo de vida de dados permite gerenciar o período de retenção de dados associados a índices ou coleções que correspondam a essas regras. Essas regras definem o período de retenção dos dados em um índice ou grupo de índices. Cada regra consiste em um tipo de recurso (`index`), um período de retenção e uma lista de recursos (índices) aos quais o período de retenção se aplica.

Você define o período de retenção com um dos seguintes formatos:

- `"MinIndexRetention": "24h"`— O OpenSearch Serverless retém os dados do índice do período especificado em horas ou dias. Você pode definir esse período para `24h` a `3650d`.
- `"NoMinIndexRetention": true`— O OpenSearch Serverless retém os dados do índice indefinidamente.

No exemplo de política a seguir, a primeira regra especifica um período de retenção de 15 dias para todos os índices da coleção `marketing`. A segunda regra especifica que todos os nomes de índice que começam com `log` na coleção `finance` não têm período de retenção definido e serão mantidos indefinidamente.

```
{
   "lifeCyclePolicyDetail": {
      "type": "retention",
      "name": "my-policy",
      "policyVersion": "MTY4ODI0NTM2OTk1N18x",
      "policy": {
         "Rules": [
            {
```

```
            "ResourceType":"index",
            "Resource":[
               "index/marketing/*"
            ],
            "MinIndexRetention": "15d"
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/finance/log*"
            ],
            "NoMinIndexRetention": true
         }
         ]
      },
      "createdDate": 1688245369957,
      "lastModifiedDate": 1688245369957
   }
}
```

No exemplo de regra de política a seguir, o OpenSearch Serverless retém indefinidamente os dados em todos os índices de todas as coleções da conta.

```
{
   "Rules": [
      {
         "ResourceType": "index",
         "Resource": [
            "index/*/*"
         ]
      }
   ],
   "NoMinIndexRetention": true
}
```

Permissões obrigatórias

As políticas de ciclo de vida do OpenSearch Serverless usam as seguintes permissões AWS Identity and Access Management (IAM). Você pode especificar as condições do IAM para restringir os usuários a políticas de ciclo de vida dos dados associadas a coleções e índices específicos.

- `aoss:CreateLifecyclePolicy` – criar uma política de ciclo de vida dos dados.

- `aoss:ListLifecyclePolicies` – listar todas as políticas de ciclo de vida dos dados na conta atual.
- `aoss:BatchGetLifecyclePolicy`: visualize uma política de ciclo de vida de dados associada a um nome de conta ou política.
- `aoss:BatchGetEffectiveLifecyclePolicy`: visualize uma política de ciclo de vida de dados para um determinado recurso (`index` é o único recurso compatível).
- `aoss:UpdateLifecyclePolicy`: modifique uma determinada política de ciclo de vida de dados e altere sua configuração ou recurso de retenção.
- `aoss:DeleteLifecyclePolicy` – excluir uma política de ciclo de vida dos dados.

A política de acesso baseada em identidade a seguir permite que um usuário exiba todas as políticas de ciclo de vida dos dados e atualize as políticas com o padrão de recursos `index/application-logs`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateLifecyclePolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListLifecyclePolicies",
                "aoss:BatchGetLifecyclePolicy"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

Precedência das políticas

Pode haver situações em que as regras das políticas de ciclo de vida se sobreponham, dentro ou entre as políticas. Quando isso acontece, uma regra com um nome de recurso ou padrão mais específico para um índice substitui uma regra com um nome de recurso ou padrão mais geral para qualquer índice que seja comum às duas regras.

Por exemplo, na política a seguir, duas regras se aplicam a um índice `index/sales/logstash`. Nessa situação, a segunda regra tem precedência porque `index/sales/log*` é a correspondência mais longa para `index/sales/logstash`. Portanto, o OpenSearch Serverless não define um período de retenção para o índice.

```
{
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/*",
            ],
            "MinIndexRetention": "15d"
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/log*",
            ],
            "NoMinIndexRetention": true
         }
      ]
   }
```

Sintaxe da política

Forneça uma ou mais regras. Essas regras definem as configurações do ciclo de vida dos dados para seus índices sem OpenSearch servidor.

Cada regra contém os seguintes elementos: Você pode fornecer `MinIndexRetention` ou `NoMinIndexRetention` em cada regra, mas não em ambas.

Elemento	Description
Tipo de atributo	O tipo de recurso ao qual a regra se aplica. A única opção compatível com políticas de ciclo de vida de dados é `index`
Recurso	Uma lista de and/or padrões de nomes de recursos. Padrões consistem em um prefixo e um curinga (*), que permitem que as permissões associada s sejam aplicadas a vários recursos. Por exemplo, `.index/`*`<collection-name|p attern>`* `/`*`<index-name|pattern>`*
MinIndexRetention	O período limitado, em dias (d) ou horas (h), para reter o documento no índice. O limite mínimo é `24h` e o máximo é`3650d`.
NoMinIndexRetention	Se`true`, o OpenSearch Serverless retém documentos indefinidamente.

No exemplo a seguir, a primeira regra se aplica a todos os índices do padrão `autoparts-inventory` (`index/autoparts-inventory/*`) e exige que os dados sejam retidos por pelo menos 20 dias antes que qualquer ação, como exclusão ou arquivamento, possa ocorrer.

A segunda regra visa os índices que correspondem ao padrão `auto*/gear` (`index/auto*/gear`), definindo um período mínimo de retenção de 24 horas.

A terceira regra se aplica especificamente ao índice `tires` e não tem período mínimo de retenção, o que significa que os dados desse índice podem ser excluídos ou arquivados imediatamente ou com base em outros critérios. Essas regras ajudam a gerenciar a retenção de dados de índices com tempos de retenção variáveis ou sem restrições de retenção.

```
{
  "Rules": [
    {
      "ResourceType": "index",
      "Resource": [
```

```
        "index/autoparts-inventory/*"
      ],
      "MinIndexRetention": "20d"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/auto*/gear"
      ],
      "MinIndexRetention": "24h"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/autoparts-inventory/tires"
      ],
      "NoMinIndexRetention": true
    }
  ]
}
```

Criar políticas de ciclo de vida de dados

Para criar uma política de ciclo de vida de dados, você define regras que gerenciam a retenção e a exclusão dos dados com base nos critérios especificados.

Console

Para criar uma política de ciclo de vida de dados

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Políticas de ciclo de vida de dados.
3. Escolha Criar uma política de ciclo de vida de dados.
4. Insira um nome descritivo para a política.
5. Em Ciclo de vida dos dados, escolha Adicionar e selecione as coleções e os índices para a política.

   Comece escolhendo as coleções às quais os índices pertencem. Em seguida, escolha o índice na lista ou insira um padrão de índice. Para selecionar todas as coleções como origens, insira um asterisco (*).

6. Para retenção de dados, você pode optar por reter os dados indefinidamente ou desmarcar Ilimitado (nunca excluir) e especificar um período após o qual o OpenSearch Serverless excluirá automaticamente os dados do Amazon S3.
7. Escolha Salvar e Criar.

AWS CLI

Para criar uma política de ciclo de vida de dados usando o AWS CLI, use o create-lifecycle-policycomando com as seguintes opções:

- `--name`: o nome da política.
- `--type`: o tipo de política. Atualmente, o único valor disponível é `retention`.
- `--policy`: a política de ciclo de vida de dados. Esse parâmetro aceita tanto políticas em linha quanto arquivos .json. As políticas em linha devem ser codificadas como uma string JSON com escape. Para fornecer a política em um arquivo, use o formato `--policy file://my-policy.json`.

Example

```
aws opensearchserverless create-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy "{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":[\"index/autoparts-inventory/*\"],\"MinIndexRetention\": \"81d\"},{\"ResourceType\":\"index\",\"Resource\":[\"index/sales/orders*\"],\"NoMinIndexRetention\":true}]}"
```

Atualização de políticas de ciclo de vida de dados

Para atualizar uma política de ciclo de vida de dados, você pode modificar as regras existentes para refletir as alterações nos requisitos de retenção ou exclusão de dados. Isso permite que você adapte suas políticas à medida que suas necessidades de gerenciamento de dados evoluem.

Pode haver alguns minutos de intervalo entre o momento em que você atualiza a política e o momento em que o OpenSearch Serverless começa a aplicar os novos períodos de retenção.

Console

Para atualizar uma política de ciclo de vida de dados

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Políticas de ciclo de vida de dados.
3. Selecione a política de ciclo de vida de dados que você deseja atualizar e escolha Editar.
4. Modifique a política usando o editor visual ou o editor JSON.
5. Escolha Salvar.

AWS CLI

Para atualizar uma política de ciclo de vida de dados usando o AWS CLI, use o update-lifecycle-policycomando.

Você deve incluir o parâmetro `--policy-version` na solicitação. É possível recuperar a versão da política usando os comandos list-lifecycle-policies ou batch-get-lifecycle-policy. Recomendamos incluir a versão mais recente da política para evitar a substituição acidental das alterações feitas por outras pessoas.

A solicitação a seguir atualiza uma política de ciclo de vida de dados com um novo documento de política JSON.

Example

```
aws opensearchserverless update-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy-version MTY2MzY5MTY1MDA3Ml8x \
  --policy file://my-new-policy.json
```

Como excluir políticas de ciclo de vida dos dados

Quando você exclui uma política de ciclo de vida de dados, o OpenSearch Serverless não a aplica mais em nenhum índice correspondente.

Console

Para excluir uma política de ciclo de vida de dados

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Políticas de ciclo de vida de dados.
3. Selecione a política que você deseja excluir e depois Excluir.

AWS CLI

Para excluir uma política de ciclo de vida de dados usando o AWS CLI, use o delete-lifecycle-policycomando.

Example

```
aws opensearchserverless delete-lifecycle-policy \
  --name my-policy \
  --type retention
```

Usando o AWS SDKs para interagir com o Amazon OpenSearch Serverless

Esta seção inclui exemplos de como usar o AWS SDKs para interagir com o Amazon OpenSearch Serverless. Esses exemplos de códigos mostram como criar políticas e coleções de segurança, e como consultar coleções.

> **Note**
>
> No momento, estamos criando esses exemplos de código. Se você quiser contribuir com uma amostra de código (Java, Go etc.), abra uma pull request diretamente no GitHub repositório.

Tópicos

- Python
- JavaScript

Python

O script de exemplo a seguir usa o AWS SDK para Python (Boto3), assim como o cliente opensearch-py para Python, para criar políticas de criptografia, rede e acesso a dados, criar uma coleção correspondente e indexar alguns dados de exemplo.

Execute os comandos a seguir para instalar as dependências necessárias:

```
pip install opensearch-py
pip install boto3
pip install botocore
pip install requests-aws4auth
```

No script, é necessário substituir o elemento `Principal` pelo o nome do recurso da Amazon (ARN) do usuário ou da função do usuário que está assinando a solicitação. Você também pode, opcionalmente, modificar a `region`.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3
import botocore
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

client = boto3.client('opensearchserverless')
service = 'aoss'
region = 'us-east-1'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def createEncryptionPolicy(client):
    """Creates an encryption policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Encryption policy for TV collections',
            name='tv-policy',
            policy="""
```

```
                {
                    \"Rules\":[
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ]
                        }
                    ],
                    \"AWSOwnedKey\":true
                }
                """,
            type='encryption'
        )
        print('\nEncryption policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] The policy name or rules conflict with an existing
 policy.')
        else:
            raise error


def createNetworkPolicy(client):
    """Creates a network policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Network policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Description\":\"Public access for TV collection\",
                    \"Rules\":[
                        {
                            \"ResourceType\":\"dashboard\",
                            \"Resource\":[\"collection\/tv-*\"]
                        },
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[\"collection\/tv-*\"]
                        }
                    ],
```

```
                    \"AllowFromPublic\":true
                }]
                """,
            type='network'
        )
        print('\nNetwork policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A network policy with this name already exists.')
        else:
            raise error


def createAccessPolicy(client):
    """Creates a data access policy that matches all collections beginning with tv-"""
    try:
        response = client.create_access_policy(
            description='Data access policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Rules\":[
                        {
                            \"Resource\":[
                                \"index\/tv-*\/*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateIndex\",
                                \"aoss:DeleteIndex\",
                                \"aoss:UpdateIndex\",
                                \"aoss:DescribeIndex\",
                                \"aoss:ReadDocument\",
                                \"aoss:WriteDocument\"
                            ],
                            \"ResourceType\": \"index\"
                        },
                        {
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateCollectionItems\"
```

```
                            ],
                            \"ResourceType\": \"collection\"
                        }
                    ],
                    \"Principal\":[
                        \"arn:aws:iam::123456789012:role\/Admin\"
                    ]
                }]
                """,
            type='data'
        )
        print('\nAccess policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] An access policy with this name already exists.')
        else:
            raise error


def createCollection(client):
    """Creates a collection"""
    try:
        response = client.create_collection(
            name='tv-sitcoms',
            type='SEARCH'
        )
        return(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A collection with this name already exists. Try
 another name.')
        else:
            raise error


def waitForCollectionCreation(client):
    """Waits for the collection to become active"""
    response = client.batch_get_collection(
        names=['tv-sitcoms'])
    # Periodically check collection status
    while (response['collectionDetails'][0]['status']) == 'CREATING':
```

```
        print('Creating collection...')
        time.sleep(30)
        response = client.batch_get_collection(
            names=['tv-sitcoms'])
    print('\nCollection successfully created:')
    print(response["collectionDetails"])
    # Extract the collection endpoint from the response
    host = (response['collectionDetails'][0]['collectionEndpoint'])
    final_host = host.replace("https://", "")
    indexData(final_host)


def indexData(host):
    """Create an index and add some sample data"""
    # Build the OpenSearch client
    client = OpenSearch(
        hosts=[{'host': host, 'port': 443}],
        http_auth=awsauth,
        use_ssl=True,
        verify_certs=True,
        connection_class=RequestsHttpConnection,
        timeout=300
    )
    # It can take up to a minute for data access rules to be enforced
    time.sleep(45)

    # Create index
    response = client.indices.create('sitcoms-eighties')
    print('\nCreating index:')
    print(response)

    # Add a document to the index.
    response = client.index(
        index='sitcoms-eighties',
        body={
            'title': 'Seinfeld',
            'creator': 'Larry David',
            'year': 1989
        },
        id='1',
    )
    print('\nDocument added:')
    print(response)
```

```
def main():
    createEncryptionPolicy(client)
    createNetworkPolicy(client)
    createAccessPolicy(client)
    createCollection(client)
    waitForCollectionCreation(client)


if __name__ == "__main__":
    main()
```

JavaScript

O script de exemplo a seguir usa o SDK para JavaScript em Node.js, bem como o cliente opensearch-js para JavaScript, criar políticas de criptografia, rede e acesso a dados, criar uma coleção correspondente, criar um índice e indexar alguns dados de amostra.

Execute os comandos a seguir para instalar as dependências necessárias:

```
npm i aws-sdk
npm i aws4
npm i @opensearch-project/opensearch
```

No script, é necessário substituir o elemento `Principal` pelo o nome do recurso da Amazon (ARN) do usuário ou da função do usuário que está assinando a solicitação. Você também pode, opcionalmente, modificar a `region`.

```
var AWS = require('aws-sdk');
var aws4 = require('aws4');
var {
    Client,
    Connection
} = require("@opensearch-project/opensearch");
var {
    OpenSearchServerlessClient,
    CreateSecurityPolicyCommand,
    CreateAccessPolicyCommand,
    CreateCollectionCommand,
    BatchGetCollectionCommand
} = require("@aws-sdk/client-opensearchserverless");
var client = new OpenSearchServerlessClient();
```

```
async function execute() {
    await createEncryptionPolicy(client)
    await createNetworkPolicy(client)
    await createAccessPolicy(client)
    await createCollection(client)
    await waitForCollectionCreation(client)
}

async function createEncryptionPolicy(client) {
    // Creates an encryption policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateSecurityPolicyCommand({
            description: 'Encryption policy for TV collections',
            name: 'tv-policy',
            type: 'encryption',
            policy: " \
        { \
            \"Rules\":[ \
                { \
                    \"ResourceType\":\"collection\", \
                    \"Resource\":[ \
                        \"collection\/tv-*\" \
                    ] \
                } \
            ], \
            \"AWSOwnedKey\":true \
        }"
        });
        const response = await client.send(command);
        console.log("Encryption policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] The policy name or rules conflict with an
 existing policy.');
        } else
            console.error(error);
    };
}

async function createNetworkPolicy(client) {
    // Creates a network policy that matches all collections beginning with 'tv-'
    try {
```

```
        var command = new CreateSecurityPolicyCommand({
            description: 'Network policy for TV collections',
            name: 'tv-policy',
            type: 'network',
            policy: " \
            [{ \
                \"Description\":\"Public access for television collection\", \
                \"Rules\":[ \
                    { \
                        \"ResourceType\":\"dashboard\", \
                        \"Resource\":[\"collection\/tv-*\"] \
                    }, \
                    { \
                        \"ResourceType\":\"collection\", \
                        \"Resource\":[\"collection\/tv-*\"] \
                    } \
                ], \
                \"AllowFromPublic\":true \
            }]"
        });
        const response = await client.send(command);
        console.log("Network policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A network policy with that name already
 exists.');
        } else
            console.error(error);
    };
}

async function createAccessPolicy(client) {
    // Creates a data access policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateAccessPolicyCommand({
            description: 'Data access policy for TV collections',
            name: 'tv-policy',
            type: 'data',
            policy: " \
            [{ \
                \"Rules\":[ \
                    { \
                        \"Resource\":[ \
```

```
                        \"index\/tv-*\/*\" \
                    ], \
                    \"Permission\":[ \
                        \"aoss:CreateIndex\", \
                        \"aoss:DeleteIndex\", \
                        \"aoss:UpdateIndex\", \
                        \"aoss:DescribeIndex\", \
                        \"aoss:ReadDocument\", \
                        \"aoss:WriteDocument\" \
                    ], \
                    \"ResourceType\": \"index\" \
                }, \
                { \
                    \"Resource\":[ \
                        \"collection\/tv-*\" \
                    ], \
                    \"Permission\":[ \
                        \"aoss:CreateCollectionItems\" \
                    ], \
                    \"ResourceType\": \"collection\" \
                } \
            ], \
            \"Principal\":[ \
                \"arn:aws:iam::123456789012:role\/Admin\" \
            ] \
        }]"
    });
    const response = await client.send(command);
    console.log("Access policy created:");
    console.log(response['accessPolicyDetail']);
} catch (error) {
    if (error.name === 'ConflictException') {
        console.log('[ConflictException] An access policy with that name already
 exists.');
    } else
        console.error(error);
};
}

async function createCollection(client) {
    // Creates a collection to hold TV sitcoms indexes
    try {
        var command = new CreateCollectionCommand({
            name: 'tv-sitcoms',
```

```
            type: 'SEARCH'
        });
        const response = await client.send(command);
        return (response)
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A collection with this name already
 exists. Try another name.');
        } else
            console.error(error);
    };
}

async function waitForCollectionCreation(client) {
    // Waits for the collection to become active
    try {
        var command = new BatchGetCollectionCommand({
            names: ['tv-sitcoms']
        });
        var response = await client.send(command);
        while (response.collectionDetails[0]['status'] == 'CREATING') {
            console.log('Creating collection...')
            await sleep(30000) // Wait for 30 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        console.log('Collection successfully created:');
        console.log(response['collectionDetails']);
        // Extract the collection endpoint from the response
        var host = (response.collectionDetails[0]['collectionEndpoint'])
        // Pass collection endpoint to index document request
        indexDocument(host)
    } catch (error) {
        console.error(error);
    };
}

async function indexDocument(host) {

    var client = new Client({
```

```
        node: host,
        Connection: class extends Connection {
            buildRequestObject(params) {
                var request = super.buildRequestObject(params)
                request.service = 'aoss';
                request.region = 'us-east-1'; // e.g. us-east-1
                var body = request.body;
                request.body = undefined;
                delete request.headers['content-length'];
                request.headers['x-amz-content-sha256'] = 'UNSIGNED-PAYLOAD';
                request = aws4.sign(request, AWS.config.credentials);
                request.body = body;

                return request
            }
        }
    });

    // Create an index
    try {
        var index_name = "sitcoms-eighties";

        var response = await client.indices.create({
            index: index_name
        });

        console.log("Creating index:");
        console.log(response.body);

        // Add a document to the index
        var document = "{ \"title\": \"Seinfeld\", \"creator\": \"Larry David\", \"year
\": \"1989\" }\n";

        var response = await client.index({
            index: index_name,
            body: document
        });

        console.log("Adding document:");
        console.log(response.body);
    } catch (error) {
        console.error(error);
    };
}
```

```
execute()
```

Usando CloudFormation para criar coleções Amazon OpenSearch Serverless

Você pode usar CloudFormation para criar recursos Amazon OpenSearch Serverless, como coleções, políticas de segurança e VPC endpoints. Para obter uma CloudFormation referência abrangente sobre OpenSearch Serverless, consulte Amazon OpenSearch Serverless no Guia do usuário.CloudFormation

O CloudFormation modelo de exemplo a seguir cria uma política simples de acesso a dados, uma política de rede e uma política de segurança, bem como uma coleção correspondente. É uma boa maneira de começar a trabalhar rapidamente com o Amazon OpenSearch Serverless e provisionar os elementos necessários para criar e usar uma coleção.

> **Important**
>
> Este exemplo usa o acesso à rede pública, o que não é recomendado para workloads de produção. Recomendamos usar o acesso à VPC para proteger as coleções. Para obter mais informações, consulte AWS::OpenSearchServerless::VpcEndpoint e the section called "Endpoint VPC do plano de dados".

```
AWSTemplateFormatVersion: 2010-09-09
Description: 'Amazon OpenSearch Serverless template to create an IAM user, encryption
 policy, data access policy and collection'
Resources:
  IAMUSer:
    Type: 'AWS::IAM::User'
    Properties:
      UserName:  aossadmin
  DataAccessPolicy:
    Type: 'AWS::OpenSearchServerless::AccessPolicy'
    Properties:
      Name: quickstart-access-policy
      Type: data
      Description: Access policy for quickstart collection
      Policy: !Sub >-
```

```
        [{"Description":"Access for cfn user","Rules":
[{"ResourceType":"index","Resource":["index/*/*"],"Permission":["aoss:*"]},
        {"ResourceType":"collection","Resource":["collection/quickstart"],"Permission":
["aoss:*"]}],
        "Principal":["arn:aws:iam::${AWS::AccountId}:user/aossadmin"]}]
  NetworkPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-network-policy
      Type: network
      Description: Network policy for quickstart collection
      Policy: >-
        [{"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}, {"ResourceType":"dashboard","Resource":["collection/
quickstart"]}],"AllowFromPublic":true}]
  EncryptionPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-security-policy
      Type: encryption
      Description: Encryption policy for quickstart collection
      Policy: >-
        {"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}],"AWSOwnedKey":true}
  Collection:
    Type: 'AWS::OpenSearchServerless::Collection'
    Properties:
      Name: quickstart
      Type: TIMESERIES
      Description: Collection to holds timeseries data
    DependsOn: EncryptionPolicy
Outputs:
  IAMUser:
    Value: !Ref IAMUSer
  DashboardURL:
    Value: !GetAtt Collection.DashboardEndpoint
  CollectionARN:
    Value: !GetAtt Collection.Arn
```

Fazer backup de coleções usando snapshots

Snapshots são point-in-time backups de suas coleções Amazon OpenSearch Serverless que fornecem recursos de recuperação de desastres. OpenSearch O Serverless cria e gerencia

automaticamente instantâneos de suas coleções, garantindo a continuidade dos negócios e a proteção dos dados. Cada snapshot contém metadados de índice (configurações e mapeamentos para seus índices), metadados de cluster (modelos de índice e aliases) e dados de índice (todos os documentos e dados armazenados em seus índices).

OpenSearch O Serverless fornece backups automáticos de hora em hora, sem configuração manual, sem sobrecarga de manutenção, sem custos adicionais de armazenamento, recuperação rápida da perda acidental de dados e a capacidade de restaurar índices específicos a partir de um instantâneo.

Antes de trabalhar com instantâneos, entenda essas considerações importantes. A criação de um instantâneo leva tempo para ser concluída e não é instantânea. Novos documentos ou atualizações durante a criação do instantâneo não serão incluídos no instantâneo. Você só pode restaurar os snapshots para a sua coleção original, não para uma nova coleção. Quando restaurados, os índices recebem novos UUIDs que diferem de suas versões originais. A restauração de um índice aberto existente no OpenSearch Serverless substituirá os dados desse índice, desde que um novo nome de índice ou um padrão de prefixo não seja fornecido. Isso difere do comportamento principal. OpenSearch Você pode executar somente uma operação de restauração por vez e não pode iniciar várias operações de restauração na mesma coleção simultaneamente. A tentativa de restaurar índices durante uma operação de restauração ativa faz com que a operação falhe. Durante uma operação de restauração, suas solicitações aos índices falham.

Permissões obrigatórias

Para trabalhar com snapshots, configure as seguintes permissões em sua política de acesso a dados. Para saber mais sobre políticas de acesso a dados, consulte Políticas de acesso a dados versus políticas do IAM.

Política de acesso a dados	APIs
perda: DescribeSnapshot	GET /_ cat/snapshots/aoss -automatizado OBTENHA _snapshot/aoss-automated/snapshot/
perda: RestoreSnapshot	POST /_ /_restaurar snapshot/aoss-automated/snapshot
perda: DescribeCollectionItems	GET /_cat/recovery

Você pode configurar políticas usando os seguintes AWS CLI comandos:

1. create-access-policy
2. delete-access-policy
3. get-access-policy
4. update-access-policy

Aqui está um exemplo de comando da CLI para criar uma política de acesso. No comando, substitua o *example* conteúdo por suas informações específicas.

```
aws opensearchserverless create-access-policy \
--type data \
--name Example-data-access-policy \
--region aws-region \
--policy '[
  {
    "Rules": [
      {
        "Resource": [
          "collection/Example-collection"
        ],
        "Permission": [
          "aoss:DescribeSnapshot",
          "aoss:RestoreSnapshot",
          "aoss:DescribeCollectionItems"
        ],
        "ResourceType": "collection"
      }
    ],
    "Principal": [
      "arn:aws:iam::111122223333:user/UserName"
    ],
    "Description": "Data policy to support snapshot operations."
  }
]'
```

Trabalhar com snapshots

Por padrão, quando você cria uma nova coleção, o OpenSearch Serverless cria automaticamente instantâneos a cada hora. Medida a ser tomada Cada instantâneo inclui todos os índices da coleção.

Depois que o OpenSearch Serverless criar instantâneos, você poderá listá-los e revisar os detalhes do instantâneo usando os procedimentos a seguir.

Listar instantâneos

Use os procedimentos a seguir para listar todos os instantâneos em uma coleção e revisar seus detalhes.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, selecione Coleções.
3. Escolha o nome da sua coleção para abrir a página de detalhes.
4. Escolha a guia Instantâneos para exibir todos os instantâneos gerados.
5. Analise as informações do snapshot, incluindo:
   - ID do instantâneo - identificador exclusivo do instantâneo
   - Status - Estado atual (disponível, em andamento)
   - Hora de criação - Quando o instantâneo foi tirado

AWS CLI

- Use o comando a seguir para listar todos os instantâneos em uma coleção:

```
GET /_cat/snapshots/aoss-automated
```

OpenSearch O Serverless retorna uma resposta como a seguinte:

```
id                                status  start_epoch start_time end_epoch
 end_time    duration    indexes successful_shards failed_shards total_shards
snapshot-ExampleSnapshotID1     SUCCESS 1737964331  07:52:11   1737964382
 07:53:02    50.4s       1
snapshot-ExampleSnapshotID2     SUCCESS 1737967931  08:52:11   1737967979
 08:52:59    47.7s       2
snapshot-ExampleSnapshotID3     SUCCESS 1737971531  09:52:11   1737971581
 09:53:01    49.1s       3
snapshot-ExampleSnapshotID4 IN_PROGRESS 1737975131  10:52:11   -          -
       4.8d       3
```

Obtenha detalhes do instantâneo

Use os procedimentos a seguir para recuperar informações detalhadas sobre um instantâneo específico.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, selecione Coleções.
3. Escolha o nome da sua coleção para abrir a página de detalhes.
4. Escolha a guia Snapshots.
5. Escolha o ID do trabalho de snapshot para exibir informações detalhadas sobre o snapshot, incluindo metadados, índices incluídos e informações de tempo.

AWS CLI

- Use o comando a seguir para recuperar informações sobre um instantâneo. No comando, substitua o *example* conteúdo por suas informações específicas.

```
GET _snapshot/aoss-automated/snapshot/
```

Exemplo de solicitação:

```
GET _snapshot/aoss-automated/snapshot-ExampleSnapshotID1/
```

Exemplo de resposta:

```
{
    "snapshots": [
        {
            "snapshot": "snapshot-ExampleSnapshotID1-5e01-4423-9833Example",
            "uuid": "Example-5e01-4423-9833-9e9eb757Example",
            "version_id": 136327827,
            "version": "2.11.0",
            "remote_store_index_shallow_copy": true,
            "indexes": [
                "Example-index-0117"
            ],
            "data_streams": [],
```

```
            "include_global_state": true,
            "metadata": {},
            "state": "SUCCESS",
            "start_time": "2025-01-27T09:52:11.953Z",
            "start_time_in_millis": 1737971531953,
            "end_time": "2025-01-27T09:53:01.062Z",
            "end_time_in_millis": 1737971581062,
            "duration_in_millis": 49109,
            "failures": [],
            "shards": {
                "total": 0,
                "failed": 0,
                "successful": 0
            }
        }
    ]
}
```

A resposta do instantâneo inclui vários campos principais: `id` fornece um identificador exclusivo para a operação do instantâneo, `status` retorna o estado atual `SUCCESS` ou `IN_PROGRESS` `duration` indica o tempo gasto para concluir a operação do instantâneo e `indexes` retorna o número de índices incluídos no instantâneo.

Restauração a partir de um snapshot

A restauração a partir de um instantâneo recupera dados de um backup feito anteriormente. Esse processo é crucial para recuperação de desastres e gerenciamento de dados no OpenSearch Serverless. Antes de restaurar, entenda que os índices restaurados serão diferentes UUIDs de suas versões originais. A restauração em um índice aberto existente no OpenSearch Serverless substituirá os dados desse índice, desde que um novo nome de índice ou um padrão de prefixo não seja fornecido, os instantâneos só poderão ser restaurados em sua coleção original (a restauração entre coleções não é suportada) e as operações de restauração afetarão o desempenho do cluster, portanto, planeje adequadamente.

Use os procedimentos a seguir para restaurar índices de backup a partir de um snapshot.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, selecione Coleções.

3. Escolha o nome da sua coleção para abrir a página de detalhes.
4. Escolha a guia Instantâneos para exibir os instantâneos disponíveis.
5. Escolha o instantâneo do qual você deseja restaurar e, em seguida, escolha Restaurar do instantâneo.
6. Na caixa de diálogo Restaurar a partir do instantâneo:
   - Em Nome do instantâneo, verifique o ID do instantâneo selecionado.
   - Para Escopo do Snapshot, escolha uma das seguintes opções:
     - Todos os índices na coleção - Restaure todos os índices do instantâneo
     - Índices específicos - Selecione índices individuais para restaurar
   - Em Destino, escolha a coleção para a qual restaurar.
   - (Opcional) Defina as configurações de renomeação para renomear os índices restaurados:
     - Não renomeie - mantenha os nomes dos índices originais
     - Adicionar prefixo aos nomes de índices restaurados - Adicione um prefixo para evitar conflitos
     - Renomear usando expressão regular - Use padrões avançados de renomeação
   - (Opcional) Defina as configurações de notificação para ser notificado quando a restauração for concluída ou encontrar erros.
7. Escolha Salvar para iniciar a operação de restauração.

OpenSearch API

1. Execute o comando a seguir para identificar o snapshot apropriado.

   ```
   GET /_snapshot/aoss-automated/_all
   ```

   Para obter uma lista menor de snapshots, execute o comando a seguir.

   ```
   GET /_cat/snapshots/aoss-automated
   ```

2. Execute o comando a seguir para verificar os detalhes do snapshot antes de restaurar. No comando, substitua o *example* conteúdo por suas informações específicas.

   ```
   GET _snapshot/aoss-automated/snapshot-ExampleSnapshotID1/
   ```

3. Execute o comando a seguir para restaurar a partir de um snapshot específico.

```
POST /_snapshot/aoss-automated/snapshot-ID/_restore
```

Você pode personalizar a operação de restauração incluindo um corpo de solicitação. Aqui está um exemplo.

```
POST /_snapshot/aoss-automated/snapshot-
ExampleSnapshotID1-5e01-4423-9833Example/_restore
{
  "indices": "opensearch-dashboards*,my-index*",
  "ignore_unavailable": true,
  "include_global_state": false,
  "include_aliases": false,
  "rename_pattern": "opensearch-dashboards(.+)",
  "rename_replacement": "restored-opensearch-dashboards$1"
}
```

4. Execute o seguinte comando para visualizar o progresso da restauração.

```
GET /_cat/recovery
```

> **Note**
>
> Ao restaurar um instantâneo com um comando que inclui um corpo de solicitação, você pode usar vários parâmetros para controlar o comportamento da restauração. O `indices` parâmetro especifica quais índices devem ser restaurados e oferece suporte a padrões curinga. `ignore_unavailable`Defina para continuar a operação de restauração mesmo que um índice no snapshot esteja ausente. Use `include_global_state` para determinar se o estado do cluster deve ser restaurado e `include_aliases` para controlar se os aliases associados devem ser restaurados. Os `rename_replacement` parâmetros `rename_pattern` e renomeiam os índices durante a operação de restauração.

Suporte ao codec Zstandard no Amazon Serverless OpenSearch

Os codecs de índice determinam como os campos armazenados de um índice são compactados e armazenados no disco e no S3. O codec de índice é controlado pela `index.codec` configuração

estática que especifica o algoritmo de compactação. Essa configuração afeta o tamanho do fragmento de índice e o desempenho da operação do índice.

Por padrão, os índices no OpenSearch Serverless usam o codec padrão com o algoritmo de compactação. LZ4 OpenSearch O Serverless também oferece suporte `zstd` e `zstd_no_dict` codecs com níveis de compressão configuráveis de 1 a 6.

> **Important**
>
> Como `index.codec` é uma configuração estática, ela não pode ser alterada após a criação do índice.

Para obter mais detalhes, consulte a documentação do OpenSearch Index Codecs.

Criando um índice com o codec ZSTD

Você pode especificar o codec ZSTD durante a criação do índice usando a configuração: `index.codec`

```
PUT /your_index
{
  "settings": {
    "index.codec": "zstd"
  }
}
```

Níveis de compressão

Os codecs ZSTD suportam níveis de compressão opcionais por meio da `index.codec.compression_level` configuração, aceitando números inteiros no intervalo [1, 6]. Níveis de compressão mais altos resultam em melhores taxas de compressão (menor armazenamento), mas em velocidades de compressão e descompressão mais lentas. O nível de compactação padrão é 3.

```
PUT /your_index
{
  "settings": {
    "index.codec": "zstd",
    "index.codec.compression_level": 2
  }
```

```
}
```

Avaliação comparativa de desempenho

Com base em testes de benchmark com o conjunto de dados nyc_taxi, a compressão ZSTD alcançou uma compressão 26-32% melhor em comparação com a linha de base em diferentes combinações de, e níveis de compressão. `zstd` `zstd_no_dict`

Métrica	STD L1	STD L6	STD_NO_DICT L1	STD_NO_DICT L6
Redução do tamanho do índice	28,10%	32%	26,90%	28,70%
Alteração da taxa de transferência de indexação	-0,50%	-23,80%	-0,50%	-5,30%
Melhoria da latência do Match-all Query p90	-16,40%	29,50%	-16,40%	23,40%
Melhoria da latência do Range Query p90	90,90%	92,40%	-282,90%	92,50%
Quantidade de distância p90 Agg (melhoria da latência)	2%	24,70%	2%	13,80%

Para obter mais detalhes, consulte o AWS OpenSearch blog.

Economize armazenamento usando a fonte derivada

Por padrão, o OpenSearch Serverless armazena cada documento ingerido no `_source` campo, que contém o corpo do documento JSON original e indexa campos individuais para pesquisa. Embora o `_source` campo não seja pesquisável, ele é mantido para que o documento completo possa ser retornado ao executar solicitações de busca, como get e search. Quando a fonte derivada é ativada, o OpenSearch Serverless ignora o armazenamento do `_source` campo e, em vez disso, o reconstrói dinamicamente sob demanda — por exemplo, durante operações de pesquisa, obtenção, mget, reindexação ou atualização. Usar a configuração de origem derivada pode reduzir o uso do armazenamento em até 50%.

Configuração

Para configurar a fonte derivada para seu índice, crie o índice usando a `index.derived_source.enabled` configuração:

```
PUT my-index1
{
  "settings": {
    "index": {
      "derived_source": {
        "enabled": true
      }
    }
  }
}
```

Considerações importantes

- Somente determinados tipos de campo são suportados. Para obter uma lista dos campos e limitações compatíveis, consulte a OpenSearch documentação. Se você criar um índice com fonte derivada e um campo sem suporte, a criação do índice falhará. Se você tentar ingerir um documento com um campo não suportado em um índice derivado ativado pela fonte, a ingestão falhará. Use esse recurso somente quando estiver ciente dos tipos de campo que serão adicionados ao seu índice.
- A configuração `index.derived_source.enabled` é estática. Isso não pode ser alterado após a criação do índice.

Limitações nas respostas de consulta

Quando a fonte derivada está ativada, ela impõe certas limitações sobre como as respostas da consulta são geradas e retornadas.

- Os campos de data com vários formatos especificados sempre usam o primeiro formato na lista para todos os documentos solicitados, independentemente do formato original ingerido.
- Os valores dos pontos geográficos são retornados em um `{"lat": lat_val, "lon": lon_val}` formato fixo e podem perder alguma precisão.
- As matrizes de vários valores podem ser classificadas e os campos de palavras-chave podem ser desduplicados.

Para obter mais detalhes, consulte o OpenSearch blog.

Avaliação comparativa de desempenho

Com base em testes de benchmark com o conjunto de dados nyc_taxi, a fonte derivada alcançou uma redução de 58% no tamanho do índice em comparação com a linha de base.

Métrica	Fonte derivada
Redução do tamanho do índice	58,3%
Alteração da taxa de transferência de indexação	3,7%
Indexando a alteração da latência p90	6,9%
Melhoria da latência do Match-all Query p90	19%
Melhoria da latência do Range Query p90	-18,8%
Quantidade de distância p90 Agg (melhoria da latência)	-7,3%

Para obter mais detalhes, consulte o OpenSearch blog.

Grupos de OpenSearch coleção Amazon Serverless

Os grupos de coleção no Amazon OpenSearch Serverless organizam várias coleções e permitem o compartilhamento de recursos computacionais entre coleções com chaves KMS diferentes. Esse modelo de computação compartilhada reduz os custos ao eliminar a necessidade de unidades de OpenSearch processamento separadas (OCUs) para cada chave KMS.

Cada coleção OpenSearch Serverless que você cria é protegida com criptografia de dados em repouso, usada AWS KMS para armazenar e gerenciar suas chaves de criptografia. Coleções dentro do mesmo grupo de coleções compartilham recursos computacionais e espaço de memória da OCU, mesmo quando usam chaves KMS diferentes para criptografia.

Os grupos de coleta fornecem isolamento para os requisitos de segurança e desempenho. Você pode agrupar coleções com a mesma chave KMS em um único grupo de coleta para isolamento de segurança ou combinar coleções com chaves KMS diferentes no mesmo grupo para otimização

de custos. Essa flexibilidade permite equilibrar os requisitos de segurança com a eficiência dos recursos.

Quando você adiciona uma coleção a um grupo de coleções, o OpenSearch Serverless a atribui aos recursos computacionais compartilhados do grupo. O sistema gerencia automaticamente a distribuição das cargas de trabalho entre esses recursos, mantendo a segurança criptografando os dados de cada coleção com a chave KMS designada. Os controles de acesso continuam sendo aplicados no nível da coleção, e os recursos computacionais compartilhados acessam várias chaves do KMS conforme necessário para servir às coleções no grupo.

Você controla a alocação de recursos definindo limites mínimos e máximos de OCU no nível do grupo de coleta. Esses limites se aplicam a todas as coleções do grupo e ajudam você a gerenciar custos e, ao mesmo tempo, garantir um desempenho consistente.

Por padrão, há uma cota de serviço (limite) para o número de coleções em um grupo de coleções, o número de índices em uma coleção e o número de OCUs em um grupo de coleções. Para obter mais informações, consulte Cotas OpenSearch sem servidor.

Principais conceitos

Grupo de coleção

: Um AWS recurso que atua como um contêiner para uma ou mais coleções. Cada grupo de coleta tem um identificador exclusivo e pode ter seus próprios limites de capacidade e configurações.

Computação compartilhada

: Coleções dentro do mesmo grupo de coleções compartilham o mesmo conjunto de OCUs, independentemente das chaves KMS que elas usam. Esse compartilhamento reduz os custos ao eliminar a necessidade de recursos computacionais separados para cada chave KMS.

Limites de capacidade

: Você pode definir limites mínimos e máximos de OCU para operações de indexação e pesquisa no nível do grupo de coleta. Esses limites ajudam a controlar os custos e garantir um desempenho consistente.

Limites de capacidade do grupo de coleta

Os grupos de coleta fornecem controle granular sobre a alocação de recursos por meio dos limites mínimo e máximo da OCU. Esses limites se aplicam a todas as coleções dentro do grupo e operam independentemente das configurações de capacidade no nível da conta.

Por padrão, há uma cota de serviço (limite) para o número de coleções em um grupo de coleções, o número de índices em uma coleção e o número de OCUs em um grupo de coleções. Para obter mais informações, consulte Cotas OpenSearch sem servidor.

Entendendo os limites de capacidade do grupo de coleta

Você pode configurar limites mínimos e máximos de OCU para operações de indexação e pesquisa no nível do grupo de coleta. Esses limites controlam como o OpenSearch Serverless dimensiona os recursos para coleções no grupo:

- OCU mínima — O número mínimo OCUs que o OpenSearch Serverless mantém para o grupo de coleta, garantindo um desempenho de linha de base consistente.
  - Se a carga de trabalho exigir menos OCUs do que o valor mínimo especificado, o OpenSearch Serverless ainda manteria o valor mínimo especificado de OCUs e o faturamento refletiria o mesmo.
  - Se a carga de trabalho exigir um número maior de OCUs do que o valor mínimo especificado, o OpenSearch Serverless manteria o nível de OCUs necessário para a carga de trabalho e o faturamento refletiria a maior utilização da OCU.
- OCU máxima — O número máximo OCUs desse OpenSearch Serverless pode ser expandido para o grupo de coleta, ajudando você a controlar os custos.

Os limites de capacidade do grupo de coleta são dissociados dos limites no nível da conta. As configurações máximas de OCU no nível da conta se aplicam somente às coleções não associadas a nenhum grupo de coleta, enquanto as configurações máximas de OCU do grupo de coleta se aplicam às coleções dentro desse grupo específico.

Valores-limite de capacidade válidos

Ao definir limites mínimos e máximos de OCU para um grupo de coleta, você só pode usar valores do seguinte conjunto: 1, 2, 4, 8, 16 e múltiplos de 16 (como 32, 48, 64, 80, 96) até um máximo de 1.696. OCUs

Os limites mínimo e máximo de OCU são opcionais quando você cria um grupo de coleta. Se você não especificar um limite máximo de OCU, o OpenSearch Serverless usará um valor padrão de 96. OCUs

O limite mínimo da OCU deve ser menor ou igual ao limite máximo da OCU.

Entendendo a relação entre os limites de OCU no nível da conta e do grupo de coleção

Ao planejar sua capacidade OpenSearch sem servidor, é importante entender como os limites de OCU no nível da conta e os limites de OCU do grupo de coleção interagem. A soma das configurações máximas de OCU em todos os grupos de coleta mais a configuração máxima de OCU no nível da conta deve ser menor ou igual ao limite de cota de serviço por conta. Para ver os valores-limite atuais, consulte Cotas OpenSearch sem servidor.

> **Note**
>
> As configurações máximas de OCU no nível da conta se aplicam somente às coleções não associadas a nenhum grupo de coleções. As coleções dentro dos grupos de coleta são regidas pelos respectivos limites do grupo de coleta, não pelos limites no nível da conta.

Essa restrição se aplica tanto à indexação quanto à pesquisa OCUs de forma independente. Por exemplo, se você definir configurações no nível da conta e grupos de coleção, deverá garantir que o total não exceda o limite da cota de serviço para indexação OCUs e, separadamente, não exceda o limite da cota de serviço para pesquisa. OCUs Além disso, você pode criar no máximo 300 grupos de coleção por conta.

Exemplo: capacidade de planejamento com limites no nível da conta e do grupo de coleções

Se você definir a OCU máxima de pesquisa no nível da conta como 500 e o limite da cota de serviço for 1.700:

- E crie 2 grupos de coleta, a soma da OCU máxima para os 2 grupos de coleta não deve ser superior a 1.200 (1.700 - 500)
- Você pode deixar cada grupo de coleta na OCU máxima padrão de 96 (96 + 96 + 500 = 692), deixando espaço para crescimento futuro
- Ou você pode aumentar o máximo de cada grupo de coleta para 600 (600 + 600 + 500 = 1.700), usando a capacidade total permitida pela cota de serviço

Esse relacionamento é fundamental para o planejamento da capacidade. Antes de criar novos grupos de coleta ou aumentar os limites máximos de OCU, verifique se sua alocação total não excede o limite da cota de serviço. Se você atingir esse limite, deverá reduzir as configurações máximas de OCU nos grupos de coleta existentes ou diminuir as configurações máximas de OCU em nível de conta para abrir espaço para novas alocações.

Configurando limites de capacidade

Você pode definir limites de capacidade ao criar um grupo de coleta ou atualizá-los posteriormente. Para configurar limites de capacidade usando o AWS CLI, use os UpdateCollectionGroupcomandos CreateCollectionGroupou:

```
aws opensearchserverless create-collection-group \
    --name my-collection-group \
    --capacity-limits
 maxIndexingCapacityInOCU=32,maxSearchCapacityInOCU=32,minIndexingCapacityInOCU=4,minSearchCapa
```

Para atualizar os limites de capacidade de um grupo de coleta existente:

```
aws opensearchserverless update-collection-group \
    --id abcdef123456 \
    --capacity-limits
 maxIndexingCapacityInOCU=48,maxSearchCapacityInOCU=48,minIndexingCapacityInOCU=8,minSearchCapa
```

Monitorando a capacidade do grupo de coleta

OpenSearch O Serverless emite as seguintes métricas do Amazon CloudWatch Logs em intervalos de um minuto para ajudar você a monitorar a utilização da OCU e os limites de capacidade no nível do grupo de coleta:

- `IndexingOCU`— O número de indexação OCUs atualmente em uso pelo grupo de coleções.
- `SearchOCU`— O número de pesquisas OCUs atualmente em uso pelo grupo de coleções.

OpenSearch O Serverless também emite métricas de OCU no nível da conta para coleções não associadas a nenhum grupo de coleções. Você pode agregar essas métricas CloudWatch para visualizar a soma de todos os grupos de coleções e coleções OCUs em nível de conta.

Configure alarmes para notificá-lo quando seu grupo de coleta se aproximar dos limites de capacidade, para que você possa ajustar as configurações conforme necessário. Para obter mais

informações sobre métricas OpenSearch sem servidor, consulte. the section called “Monitoramento OpenSearch sem servidor”

Como os limites de capacidade são aplicados

OpenSearch O Serverless impõe limites de capacidade do grupo de coleta durante as operações de escalabilidade. Quando suas coleções precisam de recursos adicionais, o OpenSearch Serverless se expande até o limite máximo de OCU. Quando a demanda diminui, o OpenSearch Serverless diminui, mas mantém pelo menos o limite mínimo de OCU para garantir um desempenho consistente.

Os limites de capacidade são aplicados somente quando o grupo de coleta contém pelo menos uma coleção. Grupos de coleta vazios não consomem OCUs nem impõem limites de capacidade.

Se uma operação de escalabilidade exceder o limite máximo da OCU ou violar o requisito mínimo da OCU, a OpenSearch Serverless rejeitará a operação para manter a conformidade com os limites configurados.

Criptografia e chaves KMS em grupos de coleção

Cada coleção OpenSearch Serverless que você cria é protegida com criptografia de dados em repouso, usada AWS KMS para armazenar e gerenciar suas chaves de criptografia. Ao trabalhar com grupos de coleções, você tem flexibilidade na forma como especifica a chave KMS para suas coleções.

Você pode fornecer a chave KMS associada a uma coleção de duas maneiras:

- Na CreateCollection solicitação — especifique a chave KMS diretamente ao criar a coleção usando o `encryption-config` parâmetro.
- Nas políticas de segurança — defina a associação da chave KMS em uma política de segurança de criptografia.

Quando você especifica uma chave KMS em ambos os locais, a chave KMS fornecida na CreateCollection solicitação tem precedência sobre a configuração da política de segurança.

Essa flexibilidade simplifica o gerenciamento de coleções em grande escala, especialmente quando você precisa criar várias coleções com chaves KMS exclusivas. Em vez de criar e gerenciar milhares de políticas de criptografia, você pode especificar a chave KMS diretamente durante a criação da coleção.

Compartilhamento OCUs entre diferentes chaves KMS

Os grupos de coleções permitem o compartilhamento de recursos computacionais entre coleções com chaves KMS diferentes. Coleções no mesmo grupo de coleções compartilham espaço de memória da OCU, independentemente de suas chaves de criptografia. Esse modelo de computação compartilhada reduz os custos ao eliminar a necessidade de separar cada OCUs chave KMS.

Os grupos de coleta fornecem isolamento para os requisitos de segurança e desempenho. Você pode agrupar coleções com a mesma chave KMS em um único grupo de coleta para isolamento de segurança ou combinar coleções com chaves KMS diferentes no mesmo grupo para otimização de custos. Essa flexibilidade permite equilibrar os requisitos de segurança com a eficiência dos recursos.

O sistema mantém a segurança criptografando os dados de cada coleção com a chave KMS designada. Os controles de acesso continuam sendo aplicados no nível da coleção, e os recursos computacionais compartilhados acessam várias chaves do KMS conforme necessário para servir às coleções no grupo.

Permissões KMS necessárias

Ao especificar uma chave KMS na CreateCollection solicitação, você precisa das seguintes permissões adicionais:

- `kms:DescribeKey`— Permite que o OpenSearch Serverless recupere informações sobre a chave KMS.
- `kms:CreateGrant`— Permite que o OpenSearch Serverless crie uma concessão para a chave KMS para permitir operações de criptografia.

Essas permissões não são necessárias ao usar chaves AWS próprias.

Crie grupos de coleções

Este tópico descreve como criar, configurar e gerenciar grupos de coleta no Amazon OpenSearch Serverless. Use grupos de coleções para organizar coleções e compartilhar recursos computacionais para otimizar custos. Defina limites mínimos e máximos de OCU no nível do grupo de coleta para controlar o desempenho e os gastos.

Criar um grupo de coleções

Use os procedimentos a seguir para criar um novo grupo de coleta e definir suas configurações. Crie um grupo de coleção usando o console OpenSearch Serverless ou o. AWS CLI AWS SDKs Ao criar um grupo de coleta, você especifica limites de capacidade e outras opções de configuração.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, escolha Grupos de coleção
3. Escolha Criar grupo de coleção.
4. Em Nome do grupo de coleta, insira um nome para seu grupo de coleta. O nome deve ter de 3 a 32 caracteres, começar com uma letra minúscula e conter somente letras minúsculas, números e hífens.
5. (Opcional) Em Descrição, insira uma descrição para seu grupo de coleta.
6. Na seção Gerenciamento de capacidade, configure os limites da OCU:
   - Capacidade máxima de indexação — O número máximo de indexação para o OCUs qual as coleções desse grupo podem ser escaladas.
   - Capacidade máxima de pesquisa — O número máximo de pesquisas para as quais as coleções OCUs desse grupo podem ser escaladas.
   - Capacidade mínima de indexação — O número mínimo de indexação OCUs a ser mantido para um desempenho consistente.
   - Capacidade mínima de pesquisa — O número mínimo de pesquisas OCUs a serem mantidas para um desempenho consistente.
7. (Opcional) Na seção Tags, adicione tags para ajudar a organizar e identificar seu grupo de coleção.
8. Escolha Criar grupo de coleção.

AWS CLI

- Use o create-collection-groupcomando para criar um novo grupo de coleções. No comando, substitua o *example* conteúdo por suas próprias informações específicas.

```
aws opensearchserverless create-collection-group \
    --name my-collection-group \
    --description "Collection group for production workloads" \
    --capacity-limits
 maxIndexingCapacityInOCU=20,maxSearchCapacityInOCU=20,minIndexingCapacityInOCU=2,minSea
 \
    --tags key=Environment,value=Production key=Team,value=DataEngineering
```

O comando retorna detalhes sobre o grupo de coleção criado, incluindo seu ID exclusivo e ARN.

Adicionar uma nova coleção a um grupo de coleções

Ao criar uma nova coleção, especifique um nome de grupo de coleta existente ao qual associar a coleção. Use os procedimentos a seguir para adicionar uma nova coleção a um grupo de coleta.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, escolha Coleções
3. Escolha Criar coleção.
4. Em Nome da coleção, insira um nome para sua coleção. O nome deve ter de 3 a 28 caracteres, começar com uma letra minúscula e conter somente letras minúsculas, números e hífens.
5. (Opcional) Em Descrição, insira uma descrição para sua coleção.
6. Na seção Grupo de coleções, selecione o grupo de coleta ao qual você deseja que a coleção seja atribuída. Uma coleção só pode pertencer a um grupo de coleções por vez.

   (Opcional) Você também pode optar por Criar um novo grupo. Isso levará você ao fluxo de trabalho Criar grupo de coleção. Depois de concluir a criação do grupo de coleta, retorne à etapa 1 desse procedimento para começar a criar sua nova coleção.
7. Continue com o fluxo de trabalho para criar a coleção.

> **Important**
>
> Não saia do fluxo de trabalho Criar coleção. Isso interromperá a configuração da coleção. A página de detalhes da coleção aparece após a conclusão da configuração.

AWS CLI

- Use o comando create-collection para criar uma nova coleção e adicioná-la a um grupo de coleções existente. No comando, substitua o *example* conteúdo por suas próprias informações específicas.

```
aws opensearchserverless create-collection \
    --name my-collection \
    --type SEARCH \
    --collection-group-name my-collection-group \
    --description "Collection for search workloads"
```

Gerencie grupos de OpenSearch coleção Amazon Serverless

Depois de criar grupos de coleta Amazon OpenSearch Serverless, você pode modificar suas configurações conforme suas necessidades mudarem. Use essas operações de gerenciamento para atualizar os limites de capacidade e visualizar os detalhes do grupo de coleta. Essas mudanças ajudam você a otimizar a alocação de recursos e a manter a organização eficiente de suas coleções.

Exibir grupos de coleções

Exiba seus grupos de coleta OpenSearch sem servidor para revisar suas configurações, coleções associadas e status atual.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, escolha Coleções
3. Escolha a guia Grupos de coleções. Os grupos de coleções da sua conta são exibidos.

4. Escolha o Nome de um grupo de coleta para exibir seus detalhes.

AWS CLI

- Use o list-collection-groupscomando para listar todos os grupos de coleções em sua conta. Use o batch-get-collection-groupcomando para ver detalhes sobre grupos de coleta específicos. Nos comandos a seguir, substitua o *example* conteúdo por suas próprias informações específicas.

  Para listar todos os grupos de coleções:

  ```
  aws opensearchserverless list-collection-groups
  ```

  Para obter detalhes sobre grupos de coleções específicos:

  ```
  aws opensearchserverless batch-get-collection-group \
      --names my-collection-group another-group
  ```

Atualizar as configurações do grupo de coleções

Atualize as configurações do grupo de coleta OpenSearch Serverless para modificar configurações, como limites de capacidade e descrição.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, escolha Coleções
3. Escolha a guia Grupos de coleções. Os grupos de coleções da sua conta são exibidos.
4. Escolha o Nome de um grupo de coleta para exibir seus detalhes.
5. Em Detalhes do grupo de coleções, escolha Editar.
6. Faça as alterações e escolha Salvar.

AWS CLI

- Use o update-collection-groupcomando para atualizar a descrição e os limites de capacidade de um grupo de coleta existente. No comando a seguir, substitua o *example* conteúdo pelas suas próprias informações.

```
aws opensearchserverless update-collection-group \
    --id abcdef123456 \
    --description "Updated description for production workloads" \
    --capacity-limits
 maxIndexingCapacityInOCU=30,maxSearchCapacityInOCU=30,minIndexingCapacityInOCU=4,minSea
```

As alterações nos limites de capacidade entram em vigor imediatamente e podem afetar o comportamento de escalabilidade das coleções no grupo.

Excluir grupos de coleções

Antes de excluir um grupo de coleções, você deve primeiro remover todas as coleções do grupo. Você não pode excluir um grupo de coleções que contenha coleções.

Console

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/.
2. No painel de navegação esquerdo, escolha Sem servidor e, em seguida, escolha Coleções
3. Escolha a guia Grupos de coleções. Os grupos de coleções da sua conta são exibidos.
4. Escolha o nome do grupo de coleta que você deseja excluir.

> **Important**
>
> Remova todas as coleções do grupo de coleta atualizando cada coleção para remover a associação do grupo de coleta ou movendo-as para outros grupos de coleta.

5. Na parte superior da página, escolha Delete (Excluir).
6. Confirme a exclusão e escolha Excluir.

AWS CLI

- Use o delete-collection-groupcomando para excluir um grupo de coleções.

  > **Important**
  >
  > Remova todas as coleções do grupo de coleta atualizando cada coleção para remover a associação do grupo de coleta ou movendo-as para outros grupos de coleta.

  No comando a seguir, substitua o *example* conteúdo pelas suas próprias informações.

  Exclua o grupo de coleta vazio:

```
aws opensearchserverless delete-collection-group \
    --id abcdef123456
```

Gerenciando limites de capacidade para Amazon OpenSearch Serverless

Com o Amazon OpenSearch Serverless, você não precisa gerenciar a capacidade sozinho. OpenSearch O Serverless dimensiona automaticamente a capacidade computacional da sua conta com base na carga de trabalho atual. A capacidade computacional sem servidor é medida em Unidades de OpenSearch Computação (). OCUs Cada OCU é uma combinação de 6 GiB de memória e CPU virtual (vCPU) correspondente e cria um pipeline de dados para o Amazon S3. Para obter mais informações sobre a arquitetura desacoplada no OpenSearch Serverless, consulte. the section called “Como funciona”

Quando você cria sua primeira coleção, o OpenSearch Serverless instancia OCUs com base nas suas configurações de redundância. Por padrão, as réplicas ativas redundantes estão habilitadas, o que instancia quatro OCUs (duas para indexação e duas para pesquisa). Isso garante alta disponibilidade com nós em espera em outra zona de disponibilidade.

Para desenvolvimento e teste, você pode desativar a configuração Ativar redundância para uma coleção. Isso remove as réplicas em espera e usa apenas duas OCUs (uma para indexação e outra para pesquisa).

Eles OCUs sempre existem, mesmo quando não há atividade de indexação ou pesquisa. Todas as coleções subsequentes podem compartilhá-las OCUs, exceto as coleções com AWS KMS chaves exclusivas, que instanciam seu próprio conjunto de. OCUs Todas as coleções associadas a um grupo de coleções podem compartilhar o mesmo conjunto de OCUs. Somente um tipo de coleção (pesquisa, série temporal ou pesquisa vetorial) pode ser incluído em um único grupo de coleta. Para obter mais informações, consulte the section called “Grupos de coleções”.

OpenSearch O Serverless se expande e adiciona automaticamente à OCUs medida que seu uso de indexação e pesquisa aumenta. Quando o tráfego diminui, a capacidade volta ao número mínimo OCUs necessário para o tamanho dos dados.

Para pesquisas e coletas de séries temporais, o número OCUs necessário quando ocioso é proporcional ao tamanho dos dados e à contagem do índice. Para coleções de vetores, os requisitos da OCU dependem da memória (RAM) para armazenar gráficos vetoriais e do espaço em disco para armazenar índices. Quando não estão ociosos, os requisitos da OCU são responsáveis por ambos os fatores.

As coleções de vetores armazenam dados de índice no armazenamento local da OCU. Os limites de RAM da OCU são atingidos mais rapidamente do que os limites de disco, o que restringe as coleções de vetores por espaço de RAM.

Com a redundância ativada, a capacidade da OCU é reduzida para um mínimo de 1 OCU (0,5 OCU x 2) para indexação e 1 OCU (0,5 OCU x 2) para pesquisa. Quando você desativa a redundância, sua coleção pode ser reduzida para 0,5 OCU para indexação e 0,5 OCU para pesquisa.

A escalação também leva em consideração o número de fragmentos necessários para a coleção ou índice. Cada OCU suporta um número específico de fragmentos, e o número de índices deve ser proporcional à contagem de fragmentos. O número total de bases OCUs necessárias é o máximo de seus requisitos de dados, memória e fragmentos. Para obter mais informações, consulte os recursos de pesquisa econômicos do Amazon OpenSearch Serverless, em qualquer escala, no blog de AWS Big Data.

Para coleções de pesquisa e pesquisa vetorial, todos os dados são armazenados em índices de alta atividade para garantir tempos de resposta rápidos às consultas. Coleções de séries temporais usam uma combinação de armazenamento de atividade alta e muito alta, mantendo os dados

mais recentes em armazenamento de atividade muito alta para otimizar os tempos de resposta da consulta para dados acessados com mais frequência. Para obter mais informações, consulte the section called “Escolha de um tipo de coleção”.

> **Note**
>
> Uma coleção de pesquisa vetorial não pode ser compartilhada OCUs com coleções de pesquisa e séries temporais, mesmo que a coleta de pesquisa vetorial use a mesma chave KMS das coleções de pesquisa ou de séries temporais. Um novo conjunto de OCUs será criado para sua primeira coleção de vetores. As coleções OCUs de vetores são compartilhadas entre as mesmas coleções de chaves do KMS.

Para gerenciar a capacidade de suas coleções e controlar os custos, você pode especificar a capacidade máxima geral de indexação e pesquisa para a conta corrente e a região, e o OpenSearch Serverless dimensiona seus recursos de coleta automaticamente com base nessas especificações.

Como a capacidade de indexação e de pesquisa são escaladas separadamente, você especifica limites no nível de conta para cada uma:

- Capacidade máxima de indexação — O OpenSearch Serverless pode aumentar a capacidade de indexação até esse número de. OCUs
- Capacidade máxima de pesquisa — O OpenSearch Serverless pode aumentar a capacidade de pesquisa até esse número de. OCUs

> **Note**
>
> No momento, as configurações de capacidade só se aplicam ao nível da conta. Você não pode configurar limites de capacidade por coleção.

Sua meta deve ser garantir que a capacidade máxima seja alta o suficiente para lidar com picos de workload. Com base em suas configurações, o OpenSearch Serverless escala automaticamente o número de suas coleções OCUs para processar a carga de trabalho de indexação e pesquisa.

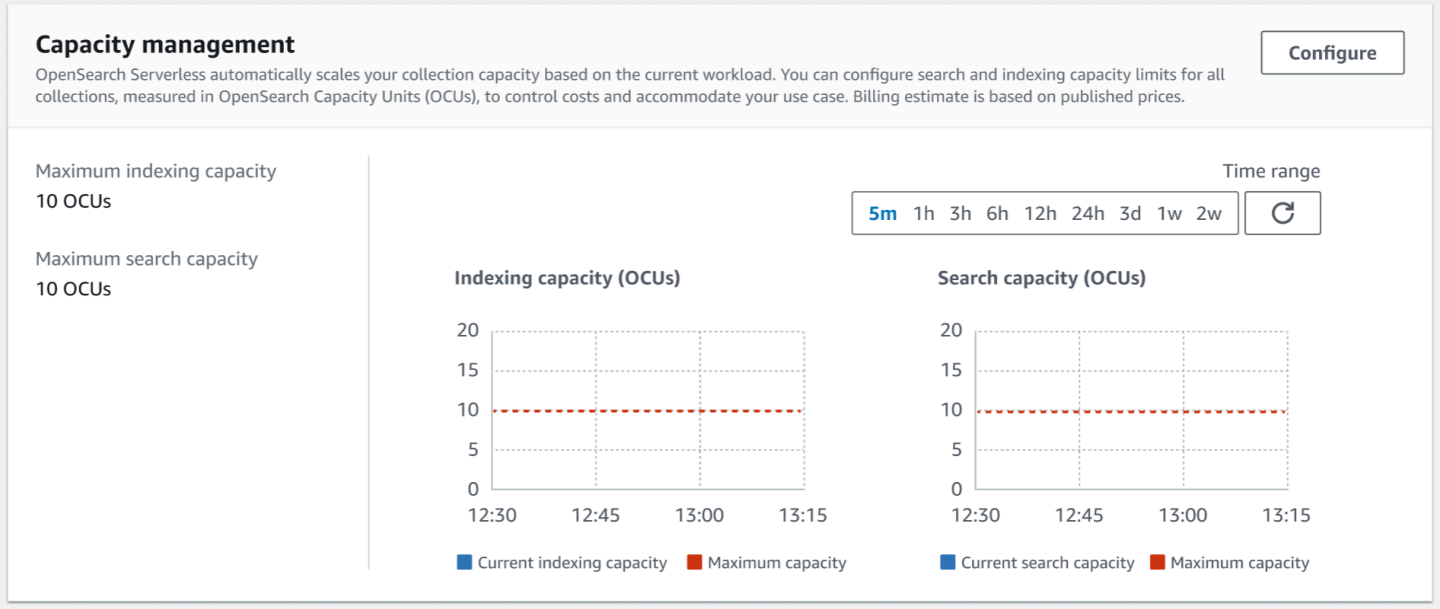
Tópicos

- Definição de configurações de capacidade

- Limites máximos de capacidade
- Monitoramento do uso da capacidade

Definição de configurações de capacidade

Para definir as configurações de capacidade no console OpenSearch Serverless, expanda Serverless no painel de navegação esquerdo e selecione Dashboard. Especifique a capacidade máxima de indexação e pesquisa em Gerenciamento de capacidade:



Para configurar a capacidade usando o AWS CLI, envie uma UpdateAccountSettingssolicitação:

```
aws opensearchserverless update-account-settings \
    --capacity-limits '{ "maxIndexingCapacityInOCU": 8,"maxSearchCapacityInOCU": 9 }'
```

Limites máximos de capacidade

O total máximo de índices que uma coleção pode conter é 1000. Para todos os três tipos de coleções, a capacidade máxima padrão da OCU é 10 OCUs para indexação e 10 OCUs para pesquisa. A capacidade mínima de OCUs permitidas para uma conta é 1 OCU [0,5 OCU x 2] para indexação e 1 OCU [0,5 OCU x 2] para pesquisa. Para todas as coleções, a capacidade máxima permitida é 1.700 OCUs para indexação e 1.700 OCUs para pesquisa. Você pode configurar a contagem de OCUs para ser qualquer número de 2 até a capacidade máxima permitida, em múltiplos de 2.

Cada OCU inclui armazenamento quente efêmero suficiente para 120 GiB de dados de índice. OpenSearch O Serverless suporta até 1 TiB de dados por índice em coleções de pesquisa e pesquisa vetorial e 100 TiB de dados ativos por índice em uma coleção de séries temporais. Para coletas de séries temporais, você pode ingerir mais dados, que podem ser armazenados como dados quentes no S3.

Para ver uma lista de todas as cotas, consulte Cotas OpenSearch sem servidor.

Monitoramento do uso da capacidade

Você pode monitorar as CloudWatch métricas `SearchOCU` e em `IndexingOCU` nível de conta para entender como suas coleções estão aumentando. É recomendável definir alarmes para notificação caso sua conta se aproxime de um limite das métricas relacionadas à capacidade, para que você possa ajustar as configurações de capacidade de acordo.

Você também pode usar essas métricas para determinar se as configurações de capacidade máxima são apropriadas ou se você precisa ajustá-las. Analise essas métricas para concentrar seus esforços para otimizar a eficiência de suas coleções. Para obter mais informações sobre as métricas para as quais o OpenSearch Serverless envia CloudWatch, consulte. the section called “Monitoramento OpenSearch sem servidor”

Ingestão de dados em coleções Amazon OpenSearch Serverless

Essas seções fornecem detalhes sobre os pipelines de ingestão compatíveis para ingestão de dados em coleções Amazon OpenSearch Serverless. Eles também abrangem alguns dos clientes que você pode usar para interagir com as operações da OpenSearch API. Seus clientes devem ser compatíveis com OpenSearch 2.x para se integrarem ao OpenSearch Serverless.

Tópicos

- Permissões mínimas necessárias
- OpenSearch Ingestão
- Fluent Bit
- Amazon Data Firehose
- Go
- Java
- JavaScript

- Logstash
- Python
- Ruby
- Assinar solicitações HTTP com outros clientes

Permissões mínimas necessárias

Para ingerir dados em uma coleção OpenSearch sem servidor, o diretor que está gravando os dados deve ter as seguintes permissões mínimas atribuídas em uma política de acesso a dados:

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/target-collection/logs"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:WriteDocument",
               "aoss:UpdateIndex"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/my-user"
      ]
   }
]
```

As permissões podem ser mais amplas se você planejar gravar em índices adicionais. Por exemplo, em vez de especificar um único índice de destino, você pode conceder permissão para todos os índices (índice/ *target-collection* /*) ou para um subconjunto de índices (índice//). *target-collection logs**

Para obter uma referência de todas as operações de OpenSearch API disponíveis e suas permissões associadas, consultethe section called “Operações e plug-ins com suporte”.

OpenSearch Ingestão

Em vez de usar um cliente terceirizado para enviar dados diretamente para uma coleção OpenSearch sem servidor, você pode usar o Amazon OpenSearch Ingestion. Você configura seus produtores de dados para enviar dados para OpenSearch Ingestão, e ele entrega automaticamente os dados para a coleção que você especificar. Você também pode configurar a OpenSearch ingestão para transformar seus dados antes de entregá-los. Para obter mais informações, consulte OpenSearch Ingestão da Amazon.

Um pipeline OpenSearch de ingestão precisa de permissão para gravar em uma coleção OpenSearch Serverless configurada como coletor. Essas permissões incluem a capacidade de descrever a coleção e enviar solicitações HTTP para ela. Para obter instruções sobre como usar a OpenSearch ingestão para adicionar dados a uma coleção, consultethe section called “Conceder aos pipelines acesso às coleções”.

Para começar a usar o OpenSearch Ingestion, consultethe section called “Tutorial: Ingestão de dados em uma coleção”.

Fluent Bit

Você pode usar AWS a imagem Fluent Bit e o plug-in OpenSearch de saída para ingerir dados em coleções sem OpenSearch servidor.

> **Note**
>
> Você deve ter a versão 2.30.0 ou posterior da imagem AWS for Fluent Bit para fazer a integração com o Serverless. OpenSearch

Exemplo de configuração:

Este exemplo de seção de saída do arquivo de configuração mostra como usar uma coleção OpenSearch Serverless como destino. A adição importante é o parâmetro `AWS_Service_Name`, que é `aoss`. `Host` é o endpoint da coleção.

```
[OUTPUT]
    Name  opensearch
    Match *
    Host  collection-endpoint.us-west-2.aoss.amazonaws.com
    Port  443
```

```
    Index  my_index
    Trace_Error On
    Trace_Output On
    AWS_Auth On
    AWS_Region <region>
    AWS_Service_Name aoss
    tls     On
    Suppress_Type_Name On
```

Amazon Data Firehose

O Firehose oferece suporte ao OpenSearch Serverless como destino de entrega. Para obter instruções sobre como enviar dados para o OpenSearch Serverless, consulte Criação de um stream de entrega do Kinesis Data Firehose e OpenSearch Escolha sem servidor para seu destino no Guia do desenvolvedor do Amazon Data Firehose.

O perfil do IAM que você fornece ao Firehose para entrega deve ser especificado em uma política de acesso a dados com a permissão mínima de `aoss:WriteDocument` para a coleção de destino, e você deve ter um índice preexistente para o qual enviar os dados. Para obter mais informações, consulte the section called “Permissões mínimas necessárias”.

Antes de enviar dados para o OpenSearch Serverless, talvez seja necessário realizar transformações nos dados. Para saber mais sobre como usar funções do Lambda para executar essa tarefa, consulte Transformação de dados do Amazon Kinesis Data Firehose no mesmo guia.

Go

O código de exemplo a seguir usa o cliente opensearch-go para Go para estabelecer uma conexão segura com a coleção OpenSearch Serverless especificada e criar um único índice. Você deve fornecer valores para `region` e `host`.

```
package main

import (
  "context"
  "log"
  "strings"
  "github.com/aws/aws-sdk-go-v2/aws"
  "github.com/aws/aws-sdk-go-v2/config"
  opensearch "github.com/opensearch-project/opensearch-go/v2"
  opensearchapi "github.com/opensearch-project/opensearch-go/v2/opensearchapi"
```

```
  requestsigner "github.com/opensearch-project/opensearch-go/v2/signer/awsv2"
)

const endpoint = "" // serverless collection endpoint

func main() {
 ctx := context.Background()

 awsCfg, err := config.LoadDefaultConfig(ctx,
  config.WithRegion("<AWS_REGION>"),
  config.WithCredentialsProvider(
   getCredentialProvider("<AWS_ACCESS_KEY>", "<AWS_SECRET_ACCESS_KEY>",
 "<AWS_SESSION_TOKEN>"),
  ),
 )
 if err != nil {
  log.Fatal(err) // don't log.fatal in a production-ready app
 }

 // create an AWS request Signer and load AWS configuration using default config folder
 or env vars.
 signer, err := requestsigner.NewSignerWithService(awsCfg, "aoss") // "aoss" for Amazon
 OpenSearch Serverless
 if err != nil {
  log.Fatal(err) // don't log.fatal in a production-ready app
 }

 // create an opensearch client and use the request-signer
 client, err := opensearch.NewClient(opensearch.Config{
  Addresses: []string{endpoint},
  Signer:    signer,
 })
 if err != nil {
  log.Fatal("client creation err", err)
 }

 indexName := "go-test-index"

  // define index mapping
 mapping := strings.NewReader(`{
  "settings": {
    "index": {
         "number_of_shards": 4
         }
```

```
        }
  }`)

 // create an index
 createIndex := opensearchapi.IndicesCreateRequest{
  Index: indexName,
    Body: mapping,
 }
 createIndexResponse, err := createIndex.Do(context.Background(), client)
 if err != nil {
  log.Println("Error ", err.Error())
  log.Println("failed to create index ", err)
  log.Fatal("create response body read err", err)
 }
 log.Println(createIndexResponse)

 // delete the index
 deleteIndex := opensearchapi.IndicesDeleteRequest{
  Index: []string{indexName},
 }

 deleteIndexResponse, err := deleteIndex.Do(context.Background(), client)
 if err != nil {
  log.Println("failed to delete index ", err)
  log.Fatal("delete index response body read err", err)
 }
 log.Println("deleting index", deleteIndexResponse)
}

func getCredentialProvider(accessKey, secretAccessKey, token string)
 aws.CredentialsProviderFunc {
 return func(ctx context.Context) (aws.Credentials, error) {
  c := &aws.Credentials{
   AccessKeyID:     accessKey,
   SecretAccessKey: secretAccessKey,
   SessionToken:    token,
  }
  return *c, nil
 }
}
```

Java

O código de exemplo a seguir usa o cliente opensearch-java para Java para estabelecer uma conexão segura com a coleção OpenSearch Serverless especificada e criar um único índice. Você deve fornecer valores para `region` e `host`.

A diferença importante em relação aos domínios OpenSearch de serviço é o nome do serviço (`aoss`em vez de`es`).

```
// import OpenSearchClient to establish connection to OpenSearch Serverless collection
import org.opensearch.client.opensearch.OpenSearchClient;

SdkHttpClient httpClient = ApacheHttpClient.builder().build();
// create an opensearch client and use the request-signer
OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

String index = "sample-index";

// create an index
CreateIndexRequest createIndexRequest = new
 CreateIndexRequest.Builder().index(index).build();
CreateIndexResponse createIndexResponse = client.indices().create(createIndexRequest);
System.out.println("Create index reponse: " + createIndexResponse);

// delete the index
DeleteIndexRequest deleteIndexRequest = new
 DeleteIndexRequest.Builder().index(index).build();
DeleteIndexResponse deleteIndexResponse = client.indices().delete(deleteIndexRequest);
System.out.println("Delete index reponse: " + deleteIndexResponse);

httpClient.close();
```

O exemplo de código a seguir estabelece novamente uma conexão segura e depois pesquisa um índice.

```
import org.opensearch.client.opensearch.OpenSearchClient;

SdkHttpClient httpClient = ApacheHttpClient.builder().build();

OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

Response response = client.generic()
    .execute(
        Requests.builder()
            .endpoint("/" + "users" + "/_search?typed_keys=true")
            .method("GET")
            .json("{"
                + "    \"query\": {"
                + "        \"match_all\": {}"
                + "    }"
                + "}")
            .build());

httpClient.close();
```

JavaScript

O código de exemplo a seguir usa o cliente opensearch-js JavaScript para estabelecer uma conexão segura com a coleção OpenSearch Serverless especificada, criar um único índice, adicionar um documento e excluir o índice. Você deve fornecer valores para `node` e `region`.

A diferença importante em relação aos domínios OpenSearch de serviço é o nome do serviço (`aoss`em vez de`es`).

Version 3

Este exemplo usa a versão 3 do SDK para JavaScript Node.js.

```
const { defaultProvider } = require('@aws-sdk/credential-provider-node');
```

```
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () => {
                const credentialsProvider = defaultProvider();
                return credentialsProvider();
            },
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';

    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index })).body) {
        console.log((await client.indices.create({ index })).body);
    }

    // add a document to the index
    const document = { foo: 'bar' };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);

    // delete the index
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Version 2

Este exemplo usa a versão 2 do SDK para JavaScript Node.js.

```
const AWS = require('aws-sdk');
```

```
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () =>
                new Promise((resolve, reject) => {
                    AWS.config.getCredentials((err, credentials) => {
                        if (err) {
                            reject(err);
                        } else {
                            resolve(credentials);
                        }
                    });
                }),
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';

    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index }),body) {
        console.log((await client.indices.create({
            index
        })).body);
    }

    // add a document to the index
    const document = {
        foo: 'bar'
    };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);

    // delete the index
```

```
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Logstash

Você pode usar o OpenSearch plug-in Logstash para publicar registros em coleções sem OpenSearch servidor.

Para usar o Logstash para enviar dados para o Serverless OpenSearch

1. Instale a versão 2.0.0 ou posterior do logstash-output-opensearchplug-in usando Docker ou Linux.

   Docker

   O Docker hospeda o software Logstash OSS com o plug-in de OpenSearch saída pré-instalado: opensearchproject/ -output-plugin. logstash-oss-with-opensearch É possível puxar a imagem como qualquer outra imagem:

   ```
   docker pull opensearchproject/logstash-oss-with-opensearch-output-plugin:latest
   ```

   Linux

   Primeiro, instale a versão mais recente do Logstash, caso ainda não a tenha. Em seguida, instale a versão 2.0.0 do plug-in de saída:

   ```
   cd logstash-8.5.0/
   bin/logstash-plugin install --version 2.0.0 logstash-output-opensearch
   ```

   Se o plug-in já estiver instalado, atualize-o para a versão mais recente:

   ```
   bin/logstash-plugin update logstash-output-opensearch
   ```

   A partir da versão 2.0.0 do plug-in, o AWS SDK usa a versão 3. Se você estiver usando uma versão do Logstash anterior à 8.4.0, deverá remover todos os plug-ins pré-instalados e instalar o AWS plug-in: `logstash-integration-aws`

```
/usr/share/logstash/bin/logstash-plugin remove logstash-input-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-input-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sns
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-cloudwatch

/usr/share/logstash/bin/logstash-plugin install --version 0.1.0.pre logstash-
integration-aws
```

2. Para que o plug-in OpenSearch de saída funcione com o OpenSearch Serverless, você deve fazer as seguintes modificações na seção de `opensearch` saída do logstash.conf:

   - Especifique `aoss` como o `service_name` em `auth_type`.
   - Especifique seu endpoint de coleção para `hosts`.
   - Adicione os parâmetros `default_server_major_version` e `legacy_template`. Esses parâmetros são necessários para que o plug-in funcione com o OpenSearch Serverless.

```
output {
  opensearch {
    hosts => "collection-endpoint:443"
    auth_type => {
      ...
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

   Esse exemplo de arquivo de configuração obtém sua entrada de arquivos em um bucket do S3 e os envia para uma coleção OpenSearch Serverless:

```
input {
  s3  {
    bucket => "my-s3-bucket"
    region => "us-east-1"
  }
}
```

```
output {
  opensearch {
    ecs_compatibility => disabled
    hosts => "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com:443"
    index => my-index
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

3. Em seguida, execute o Logstash com a nova configuração para testar o plug-in:

```
bin/logstash -f config/test-plugin.conf
```

Python

O código de exemplo a seguir usa o cliente opensearch-py para Python para estabelecer uma conexão segura com a coleção OpenSearch Serverless especificada, criar um único índice e pesquisar esse índice. Você deve fornecer valores para `region` e `host`.

A diferença importante em relação aos domínios OpenSearch de serviço é o nome do serviço (`aoss`em vez de`es`).

```
from opensearchpy import OpenSearch, RequestsHttpConnection, AWSV4SignerAuth
import boto3

host = ''  # serverless collection endpoint, without https://
region = ''  # e.g. us-east-1

service = 'aoss'
credentials = boto3.Session().get_credentials()
auth = AWSV4SignerAuth(credentials, region, service)

# create an opensearch client and use the request-signer
```

```
client = OpenSearch(
    hosts=[{'host': host, 'port': 443}],
    http_auth=auth,
    use_ssl=True,
    verify_certs=True,
    connection_class=RequestsHttpConnection,
    pool_maxsize=20,
)

# create an index
index_name = 'books-index'
create_response = client.indices.create(
    index_name
)

print('\nCreating index:')
print(create_response)

# index a document
document = {
  'title': 'The Green Mile',
  'director': 'Stephen King',
  'year': '1996'
}

response = client.index(
    index = 'books-index',
    body = document,
    id = '1'
)


# delete the index
delete_response = client.indices.delete(
    index_name
)

print('\nDeleting index:')
print(delete_response)
```

Ruby

A `opensearch-aws-sigv4` gema fornece acesso ao OpenSearch Serverless, junto com o OpenSearch Service, pronto para uso. Ele tem todos os recursos do cliente opensearch-ruby porque é uma dependência desse gem.

Ao instanciar o signatário do Sigv4, especifique `aoss` como nome do serviço:

```
require 'opensearch-aws-sigv4'
require 'aws-sigv4'

signer = Aws::Sigv4::Signer.new(service: 'aoss',
                                region: 'us-west-2',
                                access_key_id: 'key_id',
                                secret_access_key: 'secret')

# create an opensearch client and use the request-signer
client = OpenSearch::Aws::Sigv4Client.new(
  { host: 'https://your.amz-opensearch-serverless.endpoint',
    log: true },
  signer)

# create an index
index = 'prime'
client.indices.create(index: index)

# insert data
client.index(index: index, id: '1', body: { name: 'Amazon Echo',
                                            msrp: '5999',
                                            year: 2011 })

# query the index
client.search(body: { query: { match: { name: 'Echo' } } })

# delete index entry
client.delete(index: index, id: '1')

# delete the index
client.indices.delete(index: index)
```

Assinar solicitações HTTP com outros clientes

Os requisitos a seguir se aplicam ao assinar solicitações em coleções OpenSearch sem servidor quando você cria solicitações HTTP com outros clientes.

- O nome do serviço deve ser especificado como `aoss`.
- O cabeçalho `x-amz-content-sha256` é obrigatório para todas as solicitações do AWS Signature Version 4. Ele fornece um hash da carga da solicitação. Se houver uma carga útil de solicitação, defina o valor como seu hash criptográfico () do Secure Hash Algorithm (SHA). SHA256 Se não houver carga de solicitação, defina o valor como `e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855`, que é o hash de uma string vazia.

Tópicos

- Indexar com cURL
- Indexar com o Postman

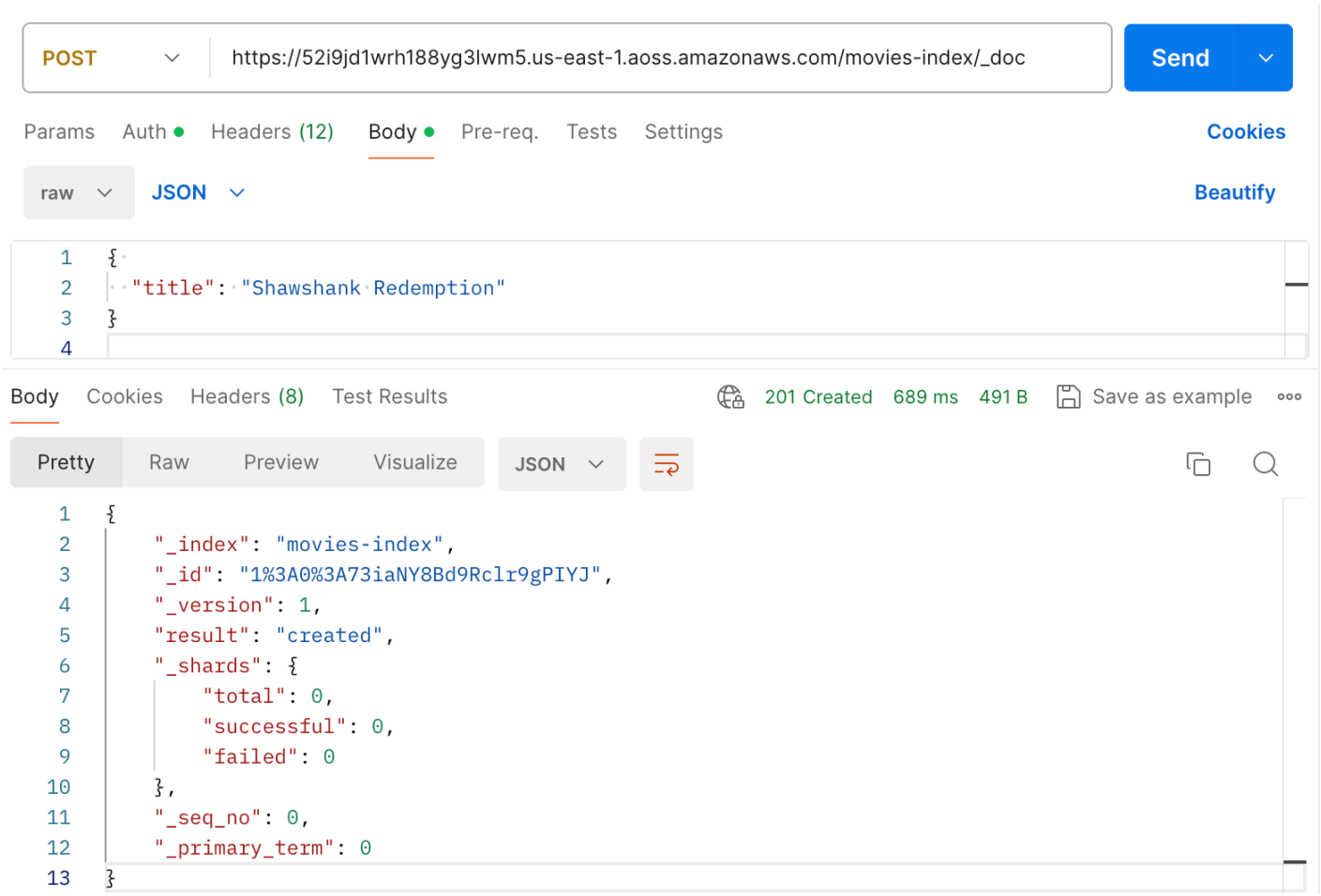
Indexar com cURL

O exemplo de solicitação a seguir usa a Client URL Request Library (cURL) para enviar um único documento para um índice denominado `movies-index` em uma coleção:

```
curl -XPOST \
    --user "$AWS_ACCESS_KEY_ID":"$AWS_SECRET_ACCESS_KEY" \
    --aws-sigv4 "aws:amz:us-east-1:aoss" \
    --header "x-amz-content-sha256: $REQUEST_PAYLOAD_SHA_HASH" \
    --header "x-amz-security-token: $AWS_SESSION_TOKEN" \
    "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com/movies-index/_doc" \
    -H "Content-Type: application/json" -d '{"title": "Shawshank Redemption"}'
```

Indexar com o Postman

A imagem a seguir mostra como enviar solicitações para uma coleção usando o Postman. Para obter instruções sobre como autenticar, consulte Fluxo de trabalho de autenticação com AWS assinatura no Postman.



Configurar o Machine Learning no Amazon OpenSearch Serverless

Machine Learning

O Machine Learning (ML) fornece recursos de ML na forma de algoritmos de ML e modelos remotos. Com acesso a esses modelos, você pode executar vários fluxos de trabalho de IA, como RAG ou pesquisa semântica. O ML é compatível com experimentação e implantação na produção de casos de uso de IA generativa usando os mais recentes modelos hospedados externamente que você pode configurar com conectores. Depois de configurar um conector, você deve configurá-lo em um modelo e depois implantá-lo para realizar previsões.

Conectores

Os conectores facilitam o acesso aos modelos hospedados em plataformas de ML de terceiros. Eles servem como gateway entre seu OpenSearch cluster e um modelo remoto. Para saber mais, consulte a documentação a seguir:

- Criação de conectores para plataformas de ML de terceiros no site de OpenSearch documentação
- Conectores para plataformas externas
- Conectores para Serviços da AWS

> **Important**
>
> - Ao criar uma política de confiança, adicione-a **`ml.opensearchservice.amazonaws.com`** como princípio OpenSearch de serviço.
> - Ignore as etapas na página Conectores que mostram como configurar um domínio na política.
> - Adicione a instrução `iam:PassRole` na etapa Configurar permissões.
> - Ignore a etapa Mapear a função de ML em OpenSearch Painéis. Não é necessário configurar perfil de backend. Isso se aplica aos conectores para Serviços da AWS e aos conectores para plataformas externas.
> - Em sua solicitação do SigV4 para o endpoint de coleção, defina o nome do serviço como **`aoss`** em vez de **`es`**.

Modelos da

Um modelo é a funcionalidade central usada em vários fluxos de trabalho de IA. Geralmente, você associa o conector a um modelo para realizar previsões usando o conector. Depois que o modelo está no estado implantado, você pode executar a previsão. Para obter mais informações, consulte Registrar um modelo hospedado em uma plataforma de terceiros no site da OpenSearch Documentação.

> **Note**
>
> Nem todos os recursos do modelo são compatíveis com o OpenSearch Serverless, como os modelos locais. Para obter mais informações, consulte Machine Learning APIs e recursos não suportados.

Configurar permissões para Machine Learning

A seção a seguir descreve as políticas de acesso a dados de coleções necessárias para Machine Learning (ML). Substitua *placeholder values* o por suas informações específicas. Para obter mais informações, consulte Permissões de políticas com suporte.

```
{
    "Rules": [
        {
            "Resource": [
                "model/collection_name/*"
            ],
            "Permission": [
                "aoss:DescribeMLResource",
                "aoss:CreateMLResource",
                "aoss:UpdateMLResource",
                "aoss:DeleteMLResource",
                "aoss:ExecuteMLResource"
            ],
            "ResourceType": "model"
        }
    ],
    "Principal": [
        "arn:aws:iam::account_id:role/role_name"
    ],
    "Description": "ML full access policy for collection_name"
}
```

- aoss:Describe MLResource — Concede permissão para pesquisar e consultar conectores, modelos e grupos de modelos.
- aoss:create MLResource — Concede permissão para criar conectores, modelos e grupos de modelos.
- aoss:Update MLResource — Concede permissão para atualizar conectores, modelos e grupos de modelos.
- aoss:Delete MLResource — Concede permissão para excluir conectores, modelos e grupos de modelos.
- aoss:Execute MLResource — Concede permissão para realizar previsões em modelos.

Machine Learning APIs e recursos não suportados

Não suportado APIs

Os seguintes Machine Learning (ML) não APIs são compatíveis com o Amazon OpenSearch Serverless:

- Funcionalidade de modelo local
- API de treinamento de modelo
- API de algoritmo de previsão de modelo
- API de previsão em lote de modelo
- API de agentes e ferramentas correspondentes
- Servidor MCP APIs
- Memória APIs
- Controlador APIs
- API de algoritmo de execução
- API de perfil de ML
- API de estatísticas de ML

Para obter mais informações sobre ML APIs, consulte ML APIs no site da OpenSearch documentação.

Atributos não compatíveis

Os seguintes recursos de ML não são compatíveis com o Amazon OpenSearch Serverless:

- Agentes e ferramentas
- Modelos locais
- O processador de inferência de ML nos pipelines de pesquisa e ingestão
  - Processador de ingestão de inferência de ML
  - Processador de respostas de pesquisa de inferência de ML
  - Processador de solicitações de pesquisa de inferência de ML

Para obter mais informações sobre esses recursos, consulte a seguinte documentação no site da OpenSearch Documentação:

- Machine learning
- Processador de inferência de ML
- Pipelines de pesquisa

Configurar a pesquisa neural e a pesquisa híbrida sem OpenSearch servidor

Pesquisa neural

O Amazon OpenSearch Serverless oferece suporte à funcionalidade de pesquisa neural para operações de pesquisa semântica em seus dados. A pesquisa neural usa modelos de machine learning para entender o significado semântico e o contexto das consultas, fornecendo resultados de pesquisa mais relevantes do que as pesquisas tradicionais baseadas em palavras-chave. Esta seção explica como configurar a Pesquisa Neural no OpenSearch Serverless, incluindo as permissões necessárias, os processadores compatíveis e as principais diferenças em relação à implementação padrão OpenSearch .

Com a Pesquisa Neural, você pode realizar nos dados uma pesquisa semântica que considera o significado semântico para entender a intenção das consultas de pesquisa. Esse recurso conta com os seguintes componentes:

- Processador de pipeline de ingestão de incorporação de texto
- Consulta neural
- Consulta neural esparsa

Pesquisa híbrida

Com a pesquisa híbrida, você pode melhorar a relevância da pesquisa combinando recursos de pesquisa semântica e pesquisa de palavras-chave. Para usar a pesquisa híbrida, crie um pipeline de pesquisa que processe os resultados da pesquisa e combine as pontuações dos documentos. Para obter mais informações, consulte Pipelines de pesquisa no site da OpenSearch documentação. Use os seguintes componentes para implementar a pesquisa híbrida:

- Processador de pipeline de pesquisa com normalização

Técnicas de normalização compatíveis

- `min_max`
- `l2`

Técnicas de combinação compatíveis

- `arithmetic_mean`
- `geometric_mean`
- `harmonic_mean`

Para obter mais informações sobre técnicas de normalização e combinação, consulte os campos do corpo da solicitação no site da OpenSearchdocumentação.

- Consulta híbrida

Consultas neurais e híbridas

Por padrão, OpenSearch calcula as pontuações dos documentos usando o BM25 algoritmo Okapi baseado em palavras-chave, que funciona bem para consultas de pesquisa que contêm palavras-chave. A pesquisa neural fornece novos tipos de consulta para consultas em linguagem natural e a capacidade de combinar pesquisa semântica e por palavra-chave.

Example: neural

```
"neural": {
  "vector_field": {
    "query_text": "query_text",
    "query_image": "image_binary",
    "model_id": "model_id",
    "k": 100
  }
}
```

Para obter mais informações, consulte Consulta neural no site da OpenSearch documentação.

Example: hybrid

```
"hybrid": {
      "queries": [
        array of lexical, neural, or combined queries
      ]
    }
```

Para obter mais informações, consulte Consulta híbrida no site da OpenSearch documentação.

Para configurar componentes de pesquisa semântica no Amazon OpenSearch Serverless, siga as etapas no tutorial de pesquisa neural no site da OpenSearch documentação. Lembre-se dessas diferenças importantes:

- OpenSearch O Serverless suporta somente modelos remotos. Você deve configurar conectores para modelos hospedados remotamente. Não é necessário implantar nem remover modelos remotos. Para obter mais informações, consulte Introdução à pesquisa semântica e híbrida no site da OpenSearch Documentação.
- Espere ter até 15 segundos de latência ao pesquisar no índice vetorial ou pesquisar canais de pesquisa e ingestão criados recentemente.

Configurar permissões do

A Pesquisa Neural no OpenSearch Serverless requer as seguintes permissões. Para obter mais informações, consulte Permissões de políticas com suporte.

Example: política de pesquisa neural

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "NeuralSearch",
            "Effect": "Allow",
            "Action": [
              "aoss:CreateIndex",
              "aoss:CreateCollection",
              "aoss:UpdateCollection",
              "aoss:DeleteIndex",
              "aoss:DeleteCollection"
            ],
            "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/your-
collection-name"
        }
    ]
}
```

- aoss: *Index: cria um índice vetorial no qual as incorporações de texto são armazenadas.
- aoss: * CollectionItems — Cria canais de ingestão e pesquisa.
- aoss: * MLResource — Cria e registra modelos de incorporação de texto.
- aoss: APIAccess All — Fornece acesso às operações de OpenSearch APIs busca e ingestão.

A seção a seguir descreve as políticas de acesso a dados de coleções necessárias para pesquisa neural. Substitua *placeholder values* o por suas informações específicas.

```
            {
                "ResourceType": "index",
                "Resource": ["index/collection_name/*"],
                "Permission": [
                  "aoss:CreateIndex",
                  "aoss:DescribeIndex",
                  "aoss:UpdateIndex",
                  "aoss:DeleteIndex"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create pipeline permission",
        "Rules": [
            {
                "ResourceType": "collection",
                "Resource": ["collection/collection_name"],
                "Permission": [
                  "aoss:CreateCollectionItems",
                  "aoss:DescribeCollectionItems",
                  "aoss:UpdateCollectionItems",
                  "aoss:DeleteCollectionItems"
                ]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    },
    {
        "Description": "Create model permission",
        "Rules": [
            {
                "ResourceType": "model",
                "Resource": ["model/collection_name/*"],
                "Permission": ["aoss:CreateMLResources"]
            }
        ],
        "Principal": [
            "arn:aws:iam::account_id:role/role_name"
        ]
    }
]
```

Configurar fluxos de trabalho no Amazon Serverless OpenSearch

Fluxos de trabalho

Os fluxos de trabalho apoiam os criadores na inovação de aplicativos de IA em. OpenSearch O processo atual de uso de ofertas de aprendizado de máquina (ML) OpenSearch, como a Pesquisa Semântica, requer tarefas complexas de configuração e pré-processamento, além de consultas detalhadas do usuário, que podem ser demoradas e propensas a erros. Os fluxos de trabalho são uma estrutura de simplificação para encadear várias chamadas de API. OpenSearch

Para configuração e uso, consulte Automatização de configurações no site. OpenSearch Ao usar fluxos de trabalho OpenSearch sem servidor, considere estas diferenças importantes:

- OpenSearch O Serverless usa somente modelos remotos nas etapas do fluxo de trabalho. Não é necessário implantar esses modelos.
- OpenSearch O Serverless não é compatível com a etapa de reindexação do fluxo de trabalho.
- Ao pesquisar fluxos de trabalho e estados de fluxo de trabalho após outras chamadas de API, espere até 15 segundos de latência para que as atualizações apareçam.

OpenSearch As coleções sem servidor oferecem suporte a fluxos de trabalho quando usadas como fonte de dados em seu OpenSearch aplicativo de interface do usuário. Para saber mais, consulte Gerenciar associações de fontes de dados.

Configurar permissões do

Antes de criar e provisionar um modelo, verifique se você tem as permissões necessárias. Se precisar de ajuda, entre em contato com o administrador da conta. OpenSearch Os fluxos de trabalho sem servidor exigem as seguintes permissões. Você pode definir o escopo das permissões para uma coleção específica definindo o ARN do recurso de coleção na política do IAM.

Example: política de fluxos de trabalho

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "NeuralSearch",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111122223333:role/Cognito_identitypoolname/Auth_Role"
        ]
      },
      "Action": [
        "aoss:CreateIndex",
        "aoss:CreateCollection",
        "aoss:UpdateCollection",
        "aoss:DeleteIndex",
        "aoss:DeleteCollection"
      ],
      "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/your-
collection-name"
    }
  ]
}
```

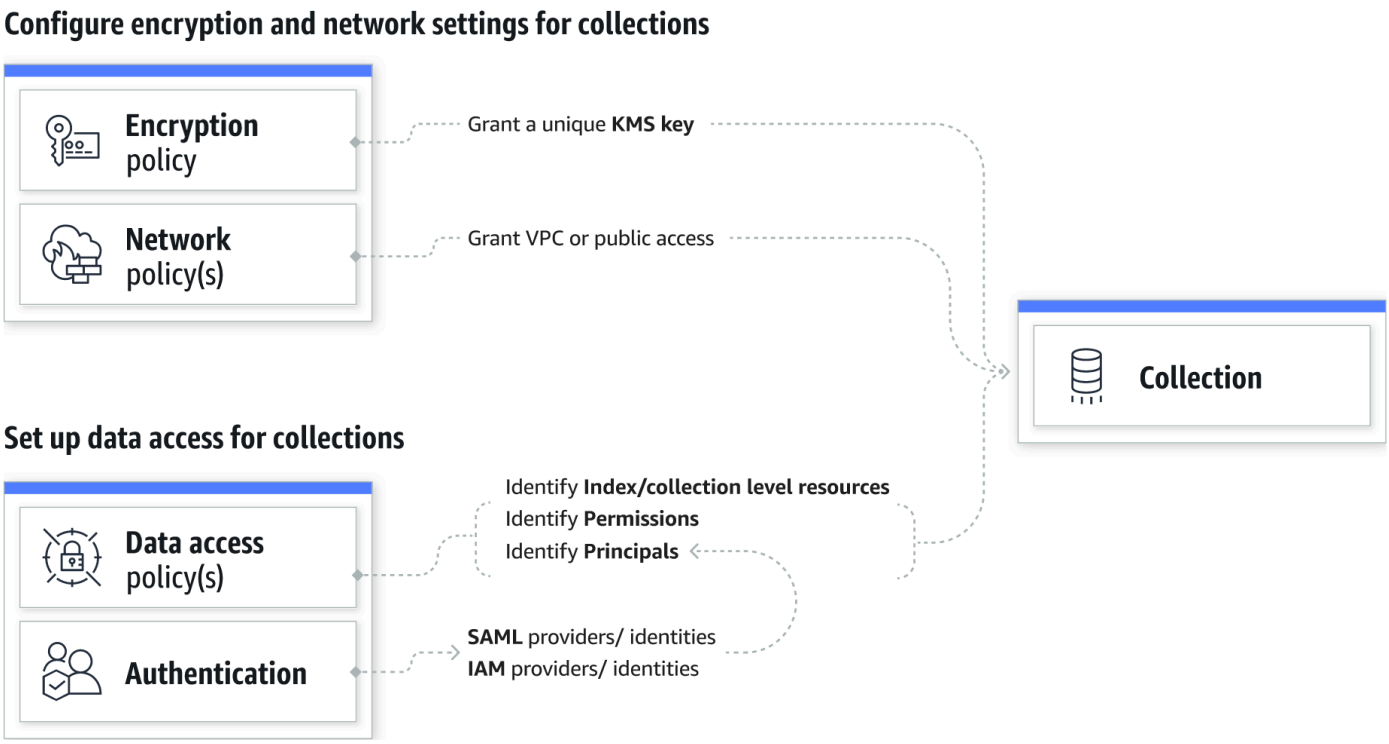
- aoss: * CollectionItems — Concede permissão para criar e gerenciar modelos e provisionar pipelines de pesquisa e ingestão.
- aoss: *Index — Concede permissão para criar e excluir índices usando operações de API. OpenSearch
- aoss: * MLResource — Concede permissão para provisionar etapas do fluxo de trabalho que usam o Configure Machine Learning.

Visão geral da segurança no Amazon OpenSearch Serverless

A segurança no Amazon OpenSearch Serverless difere fundamentalmente da segurança no Amazon OpenSearch Service das seguintes maneiras:

Recurso	OpenSearch Serviço	OpenSearch Sem servidor
Controle de acesso a dados	O acesso aos dados é determinado por políticas do IAM e por controle de acesso minucioso.	O acesso aos dados é determinado por políticas de acesso a dados.
Criptografia em repouso	A criptografia em repouso é opcional para domínios.	A criptografia em repouso é obrigatória para coleções.
Configuração e administração da segurança	Você deve configurar a rede, a criptografia e o acesso aos dados individualmente para cada domínio.	É possível usar políticas de segurança para gerenciar as configurações de segurança de várias coleções em escala.

O diagrama a seguir ilustra os componentes de segurança que compõem uma coleção funcional. Uma coleção deve ter uma chave de criptografia atribuída, configurações de acesso à rede e uma política de acesso a dados correspondente que conceda permissão aos seus recursos.



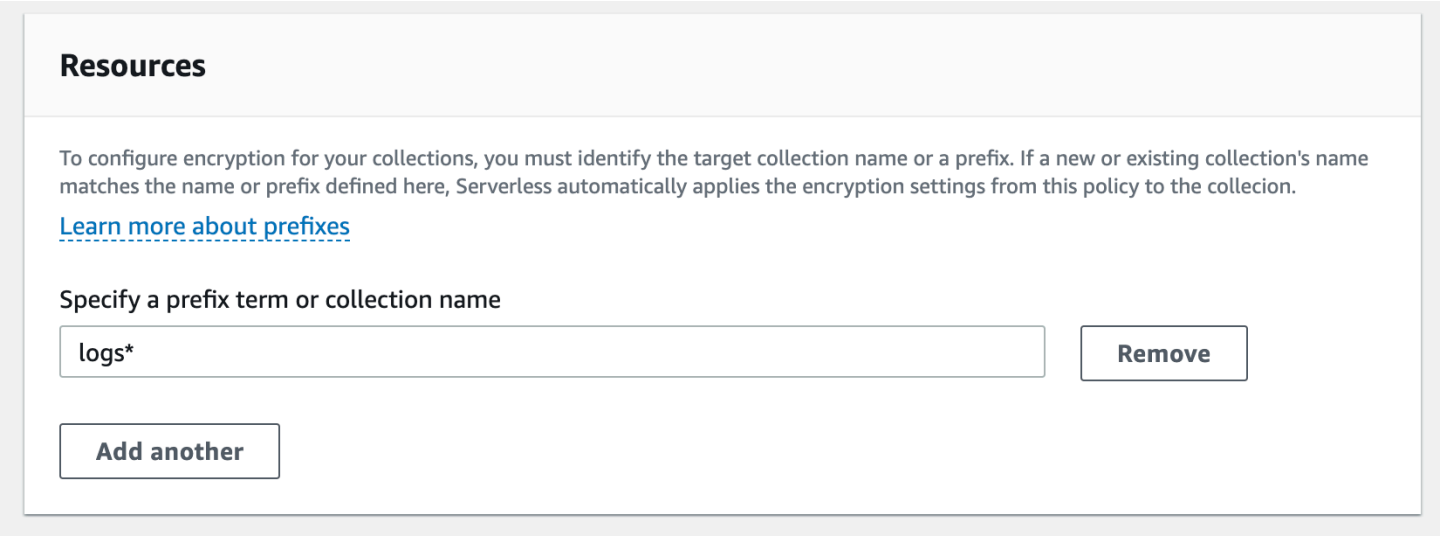
Tópicos

- Políticas de criptografia
- Políticas de rede
- Políticas de acesso a dados
- Autenticação SAML e IAM
- Segurança da infraestrutura
- Introdução à segurança no Amazon OpenSearch Serverless
- Identity and Access Management para Amazon OpenSearch Serverless
- Criptografia no Amazon OpenSearch Serverless
- Acesso à rede para Amazon OpenSearch Serverless
- Conformidade com FIPS no Amazon Serverless OpenSearch
- Controle de acesso a dados para Amazon OpenSearch Serverless
- Acesso ao plano de dados por meio de AWS PrivateLink
- Controle o acesso ao avião por meio de AWS PrivateLink
- Autenticação SAML para Amazon Serverless OpenSearch
- Validação de conformidade para Amazon OpenSearch Serverless

Políticas de criptografia

As políticas de criptografia definem se suas coleções são criptografadas com uma chave gerenciada pelo cliente Chave pertencente à AWS ou com uma chave gerenciada pelo cliente. As políticas de criptografia consistem em dois componentes: um padrão de recursos e uma chave de criptografia. O padrão de recursos define a qual coleção ou coleções a política se aplica. A chave de criptografia determina como as coleções associadas serão protegidas.

Para aplicar uma política a várias coleções, inclua um curinga (*) na regra da política. Por exemplo, a política a seguir se aplica a todas as coleções com nomes que começam com “logs”.

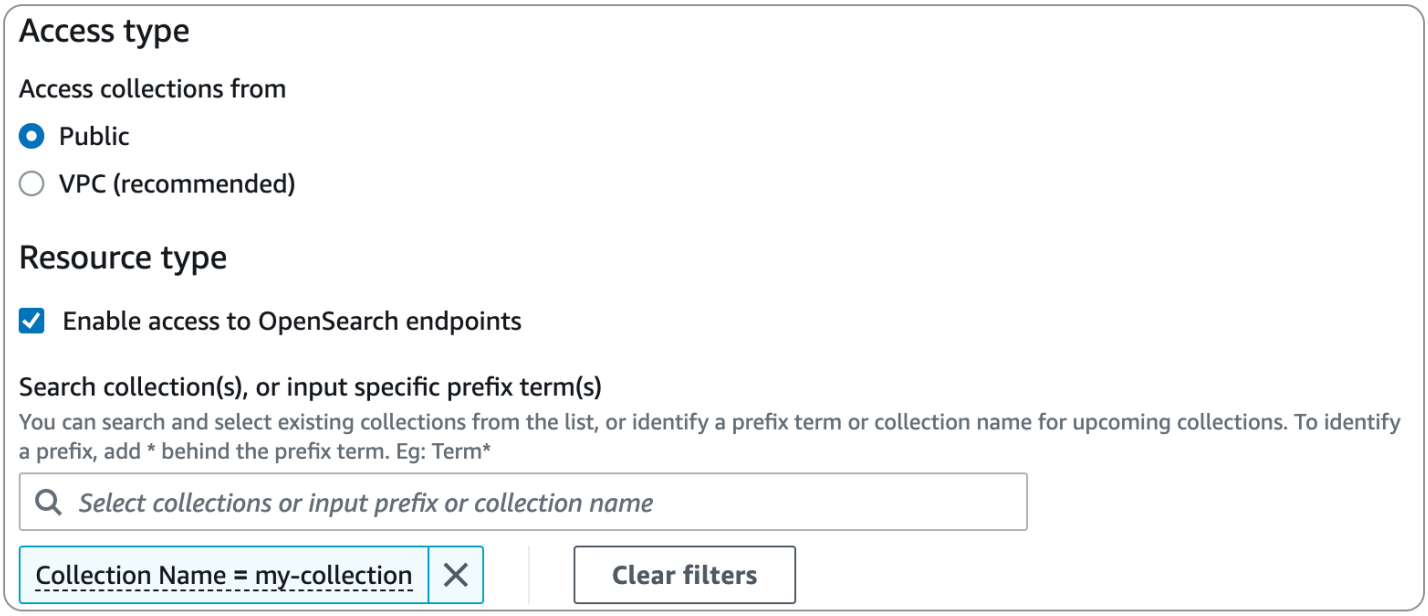


As políticas de criptografia simplificam o processo de criação e gerenciamento de coleções, especialmente quando isso é feito de forma programática. É possível criar uma coleção especificando um nome, e uma chave de criptografia será automaticamente atribuída a ela na criação.

Políticas de rede

As políticas de rede definem se as coleções são acessíveis de modo privado ou pela internet em redes públicas. As coleções particulares podem ser acessadas por meio de endpoints VPC OpenSearch gerenciados sem servidor ou por pontos específicos, Serviços da AWS como o Amazon Bedrock, usando acesso privado.AWS service (Serviço da AWS) Assim como as políticas de criptografia, as políticas de rede podem ser aplicadas a várias coleções, o que permite gerenciar o acesso à rede para muitas coleções em grande escala.

As políticas de rede consistem em dois componentes: um tipo de acesso e um tipo de recurso. O tipo de acesso pode ser público ou privado. O tipo de recurso determina se o acesso escolhido se aplica ao endpoint da coleção, ao endpoint do OpenSearch Dashboards ou a ambos.



Se você planeja configurar o acesso à VPC dentro de uma política de rede, primeiro deve criar um ou mais VPC endpoints gerenciados OpenSearch sem servidor. Esses endpoints permitem que você acesse o OpenSearch Serverless como se estivesse em sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão. Direct Connect

O acesso privado ao só Serviços da AWS pode ser aplicado ao endpoint da coleção, não ao OpenSearch endpoint do OpenSearch Dashboards. Serviços da AWS não pode ter acesso aos OpenSearch painéis.

Políticas de acesso a dados

As políticas de acesso a dados definem como seus usuários acessam os dados em suas coleções. As políticas de acesso a dados ajudam você a gerenciar coleções em grande escala atribuindo automaticamente permissões de acesso a coleções e índices que correspondam a um padrão específico. Várias políticas podem ser aplicadas a um único recurso.

As políticas de acesso a dados consistem em um conjunto de regras, cada uma com três componentes: um tipo de recurso, recursos concedidos e um conjunto de permissões. O tipo de recurso pode ser uma coleção ou um índice. Os recursos concedidos podem ser collection/index nomes ou padrões com um curinga (*). A lista de permissões especifica a quais operações de OpenSearch API a política concede acesso. Além disso, a política contém uma lista de entidades

principais, que especificam os perfis e usuários do IAM e as identidades SAML aos quais conceder acesso.



Para saber mais sobre o formato de uma política de acesso a dados, consulte a sintaxe da política.

Antes de criar uma política de acesso a dados, é necessário ter um ou mais usuários ou perfis do IAM, ou identidades SAML, aos quais fornecer acesso na política. Consulte a próxima seção para obter detalhes.

> **Note**
>
> Mudar de acesso público para privado para sua coleção removerá a guia Índices no console de coleção OpenSearch sem servidor.

Autenticação SAML e IAM

As entidades principais do IAM e as identidades do SAML são um dos alicerces de uma política de acesso a dados. Na declaração `principal` de uma política de acesso, é possível incluir usuários e perfis do IAM e identidades SAML. Em seguida, essas entidades principais recebem as permissões que você especifica nas regras de política associadas.

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/marketing/orders*"
```

```
            ],
            "Permission":[
               "aoss:*"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/Dale",
         "arn:aws:iam::123456789012:role/RegulatoryCompliance",
         "saml/123456789012/myprovider/user/Annie"
      ]
   }
]
```

Você configura a autenticação SAML diretamente no OpenSearch Serverless. Para obter mais informações, consulte the section called “Autenticação SAML”.

Segurança da infraestrutura

O Amazon OpenSearch Serverless é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte AWS Cloud Security. Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte Proteção de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar o Amazon OpenSearch Serverless pela rede. Os clientes devem ser compatíveis com o Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3. Para obter uma lista das cifras compatíveis com o TLS 1.3, consulte Protocolos e cifras do TLS na documentação do Elastic Load Balancing.

Além disso, as solicitações devem ser assinadas usando um ID de chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Introdução à segurança no Amazon OpenSearch Serverless

Os tutoriais a seguir ajudam você a começar a usar o Amazon OpenSearch Serverless. Ambos os tutoriais realizam as mesmas etapas básicas, mas um usa o console enquanto o outro usa a AWS CLI.

Observe que os casos de uso nestes tutoriais são simplificados. As políticas de rede e segurança são bastante abertas. Nas workloads de produção, recomendamos que você configure recursos de segurança mais robustos, como autenticação SAML, acesso por VPC e políticas de acesso a dados restritivas.

Tópicos

- Tutorial: Introdução à segurança no Amazon OpenSearch Serverless (console)
- Tutorial: Introdução à segurança no Amazon OpenSearch Serverless (CLI)

Tutorial: Introdução à segurança no Amazon OpenSearch Serverless (console)

Este tutorial mostra as etapas básicas para criar e gerenciar políticas de segurança usando o console Amazon OpenSearch Serverless.

Você concluirá as seguintes etapas neste tutorial:

1. Configurar permissões
2. Criar uma política de criptografia
3. Criar uma política de rede
4. Configurar uma política de acesso a dados
5. Criar uma coleção
6. Transferir e pesquisar dados

Este tutorial explica como configurar uma coleção usando Console de gerenciamento da AWS o. Para ver as mesmas etapas usando o AWS CLI, consultethe section called “Tutorial: Conceitos básicos de segurança (CLI)”.

Etapa 1: configurar permissões

> **Note**
>
> É possível pular esta etapa se já estiver usando uma política baseada em identidade mais ampla, como `Action":"aoss:*"` ou `Action":"*"`. Em ambientes de produção, no entanto, recomendamos que você siga a entidade principal do privilégio mínimo e atribua somente as permissões mínimas necessárias para concluir uma tarefa.

Para concluir este tutorial, você deve ter as permissões corretas do IAM. Seu usuário ou função deve ter uma política baseada em identidade anexada com as seguintes permissões mínimas:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:CreateCollection",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:ListSecurityPolicies",
        "aoss:CreateAccessPolicy",
        "aoss:GetAccessPolicy",
        "aoss:ListAccessPolicies"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Para obter uma lista completa das permissões OpenSearch sem servidor, consulte. the section called "Gerenciamento de Identidade e Acesso"

Etapa 2: criar uma política de criptografia

As políticas de criptografia especificam a AWS KMS chave que o OpenSearch Serverless usará para criptografar a coleção. Você pode criptografar coleções com uma chave Chave gerenciada pela AWS ou uma chave diferente. Por simplicidade, neste tutorial, criptografaremos nossa coleção com uma Chave gerenciada pela AWS.

Para criar uma política de criptografia

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, expanda Sem Servidor e escolha Políticas de criptografia.

3. Escolha Criar política de criptografia.
4. Nomeie a política como books-policy. Para a descrição, insira Política de criptografia para coleção de livros.
5. Em Recursos, insira livros, que é como você chamará sua coleção. Se você quiser ser mais amplo, inclua um asterisco (`books*`) para que a política se aplique a todas as coleções que comecem com a palavra “books” (livros).
6. Para Criptografia, mantenha a opção AWS Usar chave própria selecionada.
7. Escolha Criar.

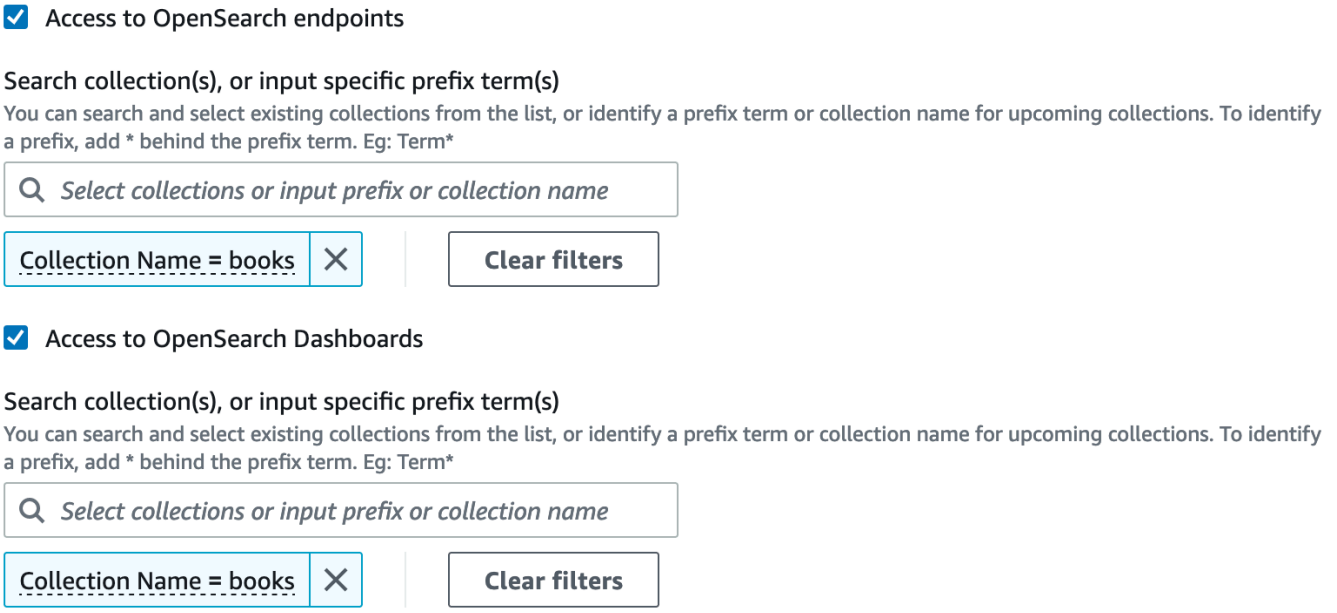
Etapa 3: criar uma política de rede

As políticas de rede determinam se sua coleção pode ser acessada pela Internet a partir de redes públicas ou se ela deve ser acessada por meio de VPC endpoints OpenSearch gerenciados sem servidor. Neste tutorial, configuraremos o acesso público.

Para criar uma política de rede

1. Escolha Políticas de rede no painel de navegação à esquerda, e escolha Criar política de rede.
2. Nomeie a política como books-policy. Para a descrição, insira Política de rede para coleção de livros.
3. Na Regra 1, nomeie a regra como Acesso público para coleção de livros .
4. Para simplificar, neste tutorial, configuraremos o acesso público para a coleção livros. Para o tipo de acesso, selecione Público.
5. Vamos acessar a coleção a partir dos OpenSearch painéis. Para fazer isso, você precisa configurar o acesso à rede para painéis e o OpenSearch endpoint, caso contrário, os painéis não funcionarão.

   Para o tipo de recurso, habilite o acesso aos OpenSearch endpoints e o acesso aos OpenSearch painéis.
6. Em ambas as caixas de entrada, insira Nome da coleção = livros. Essa configuração reduz o escopo da política para que ela se aplique somente a uma única coleção (`books`). Sua regra deve ser semelhante a esta:



7. Escolha Criar.

Etapa 4: Criar uma política de acesso a dados

Os dados da sua coleção não estarão acessíveis até que você configure o acesso aos dados. As políticas de acesso a dados são separadas da política baseada em identidade do IAM que você configurou na etapa 1. Elas permitem que os usuários acessem os dados reais de uma coleção.

Neste tutorial, forneceremos a um único usuário as permissões necessárias para indexar dados na coleção livros.

Para criar uma política de acesso a dados

1. No painel de navegação à esquerda, escolha Políticas de acesso a dados e, em seguida, Criar política de acesso.
2. Nomeie a política como books-policy. Para a descrição, insira Política de acesso a dados para coleção de livros.
3. Selecione JSON para o método de definição de política e cole a seguinte política no editor JSON.

   Substitua o ARN principal pelo ARN da conta que você usará para fazer login nos OpenSearch painéis e indexar dados.

```
[
```

```
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/books/*"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:DescribeIndex",
               "aoss:ReadDocument",
               "aoss:WriteDocument",
               "aoss:UpdateIndex",
               "aoss:DeleteIndex"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/my-user"
      ]
   }
]
```

Esta política fornece a um único usuário as permissões mínimas necessárias para criar um índice na coleção livros, indexar alguns dados e pesquisá-los.

4. Escolha Criar.

Etapa 5: Criar uma coleção

Agora que você configurou as políticas de criptografia e rede, será possível criar uma coleção correspondente e as configurações de segurança serão aplicadas automaticamente a ela.

Para criar uma coleção OpenSearch sem servidor

1. Escolha Coleções no painel de navegação à esquerda e escolha Criar coleção.
2. Dê o nome de livros à coleção.
3. Para o tipo de coleção, escolha Pesquisar.
4. Em Criptografia, OpenSearch Serverless informa que o nome da coleção corresponde à política de criptografia. `books-policy`

5. Em Configurações de acesso à rede, o OpenSearch Serverless informa que o nome da coleção corresponde à `books-policy` política de rede.
6. Escolha Próximo.
7. Em Opções de política de acesso a dados, o OpenSearch Serverless informa que o nome da coleção corresponde à política de acesso a `books-policy` dados.
8. Escolha Próximo.
9. Reveja a configuração da coleção e escolha Enviar. Normalmente, as coleções levam menos de um minuto para serem inicializadas.

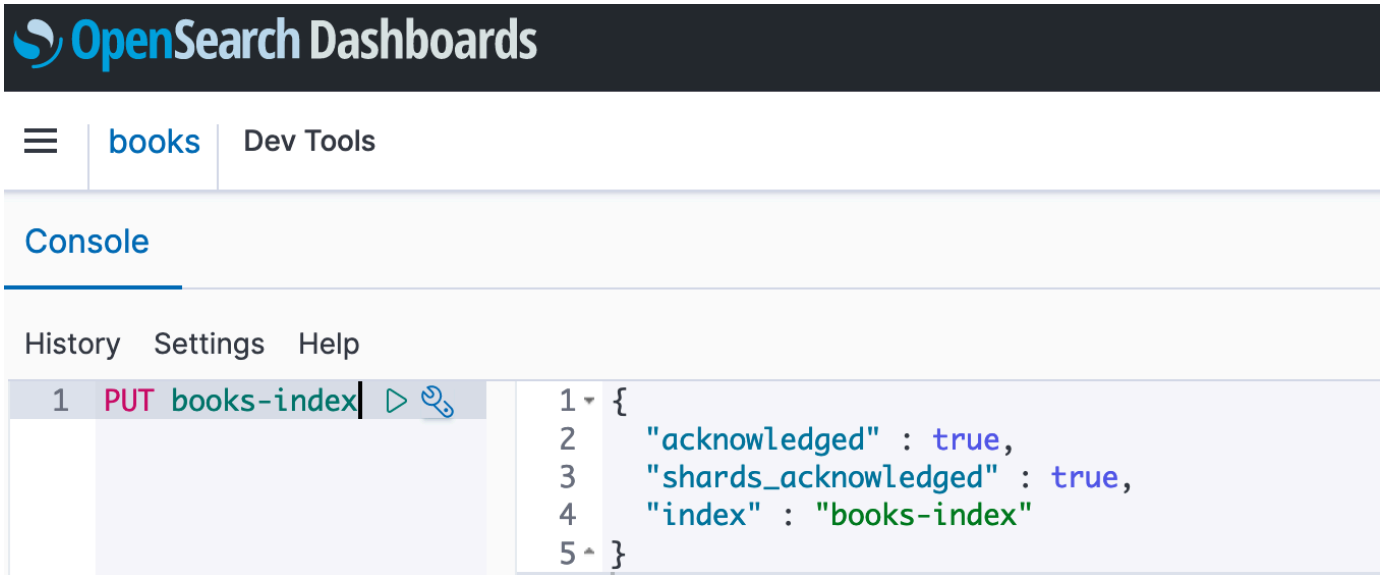
Etapa 6: transferir e pesquisar dados

Você pode carregar dados para uma coleção OpenSearch sem servidor usando Postman ou curl. Para resumir, esses exemplos usam Dev Tools no console OpenSearch Dashboards.

Para indexar e pesquisar dados em uma coleção

1. Escolha Coleções no painel de navegação à esquerda e escolha a coleção livros para abrir sua página de detalhes.
2. Escolha o URL dos OpenSearch painéis para a coleção. O URL assume o formato `https://collection-id.us-east-1.aoss.amazonaws.com/_dashboards`.
3. Faça login nos OpenSearch painéis usando as chaves de AWS acesso e secretas do principal que você especificou em sua política de acesso a dados.
4. Em OpenSearch Painéis, abra o menu de navegação à esquerda e escolha Ferramentas de desenvolvimento.
5. Para criar um único índice chamado books-index, execute o seguinte comando:

```
PUT books-index
```



6. Para indexar um único documento em books-index, execute o seguinte comando:

```
PUT books-index/_doc/1
{
  "title": "The Shining",
  "author": "Stephen King",
  "year": 1977
}
```

7. Para pesquisar dados em OpenSearch painéis, você precisa configurar pelo menos um padrão de índice. OpenSearch usa esses padrões para identificar quais índices você deseja analisar. Abra o menu principal do Dashboards, escolha Gerenciamento de pilhas, escolha Padrões de índice e, em seguida, escolha Criar padrão de índice. Para este tutorial, insira books-index.
8. Escolha Próxima etapa e, em seguida, Criar padrão de índice. Depois que o padrão é criado, você pode visualizar os vários campos do documento, como `author` e `title`.
9. Para começar a pesquisar seus dados, abra o menu principal novamente e escolha Descobrir, ou use a API de pesquisa.

Tutorial: Introdução à segurança no Amazon OpenSearch Serverless (CLI)

Este tutorial orienta você pelas etapas descritas no tutorial de introdução do console sobre segurança, mas usa o console AWS CLI em vez do OpenSearch Service console.

Você concluirá as seguintes etapas neste tutorial:

1. Criar uma política do IAM
2. Anexar a política do IAM ao perfil do IAM
3. Criar uma política de criptografia
4. Criar uma política de rede
5. Criar uma coleção
6. Configurar uma política de acesso a dados
7. Recuperar o endpoint da coleta
8. Carregar dados para sua conexão
9. Pesquisar dados em sua coleção

O objetivo deste tutorial é configurar uma única coleção OpenSearch Serverless com configurações bastante simples de criptografia, rede e acesso a dados. Por exemplo, configuraremos o acesso à rede pública, uma Chave gerenciada pela AWS para criptografia e uma política simplificada de acesso a dados que concede permissões mínimas a um único usuário.

Em um cenário de produção, considere implementar uma configuração mais robusta, incluindo autenticação SAML, uma chave de criptografia personalizada e acesso pela VPC.

Para começar a usar as políticas de segurança no OpenSearch Serverless

1.
> **Note**
>
> É possível pular esta etapa se já estiver usando uma política baseada em identidade mais ampla, como `Action":"aoss:*"` ou `Action":"*"`. Em ambientes de produção, no entanto, recomendamos que você siga a entidade principal do privilégio mínimo e atribua somente as permissões mínimas necessárias para concluir uma tarefa.

   Para começar, crie uma AWS Identity and Access Management política com as permissões mínimas necessárias para executar as etapas deste tutorial. Daremos o nome de `TutorialPolicy` à política:

```
aws iam create-policy \
  --policy-name TutorialPolicy \
  --policy-document "{\"Version\": \"2012-10-17\",\"Statement\":
 [{\"Action\": [\"aoss:ListCollections\",\"aoss:BatchGetCollection\",
```

```
\"aoss:CreateCollection\",\"aoss:CreateSecurityPolicy\",\"aoss:GetSecurityPolicy\",
\"aoss:ListSecurityPolicies\",\"aoss:CreateAccessPolicy\",\"aoss:GetAccessPolicy\",
\"aoss:ListAccessPolicies\"],\"Effect\": \"Allow\",\"Resource\": \"*\"}]}"
```

Exemplo de resposta

```
{
    "Policy": {
        "PolicyName": "TutorialPolicy",
        "PolicyId": "ANPAW6WRAECKG6QJWUV7U",
        "Arn": "arn:aws:iam::123456789012:policy/TutorialPolicy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2022-10-16T20:57:18+00:00",
        "UpdateDate": "2022-10-16T20:57:18+00:00"
    }
}
```

2. Anexe `TutorialPolicy` ao perfil do IAM que indexará e pesquisará dados na coleção. Daremos o nome de `TutorialRole` ao usuário:

```
aws iam attach-role-policy \
  --role-name TutorialRole \
  --policy-arn arn:aws:iam::123456789012:policy/TutorialPolicy
```

3. Antes de criar uma coleção, você precisa criar uma política de criptografia que atribua uma Chave pertencente à AWS à coleção livros que você criará em uma etapa posterior.

   Envie a seguinte solicitação para criar uma política de criptografia para a coleção livros:

```
aws opensearchserverless create-security-policy \
  --name books-policy \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection\/books\"]}],\"AWSOwnedKey\":true}"
```

   Exemplo de resposta

```
{
```

```
    "securityPolicyDetail": {
        "type": "encryption",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDAwNTk5MF8x",
        "policy": {
            "Rules": [
                {
                    "Resource": [
                        "collection/books"
                    ],
                    "ResourceType": "collection"
                }
            ],
            "AWSOwnedKey": true
        },
        "createdDate": 1669240005990,
        "lastModifiedDate": 1669240005990
    }
}
```

4. Crie uma política de rede que forneça acesso público à coleção livros:

```
aws opensearchserverless create-security-policy --name books-policy --type network
 \
  --policy "[{\"Description\":\"Public access for books collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/books\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/books\"]}],
\"AllowFromPublic\":true}]"
```

Exemplo de resposta

```
{
    "securityPolicyDetail": {
        "type": "network",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDI1Njk1NV8x",
        "policy": [
            {
                "Rules": [
                    {
                        "Resource": [
                            "collection/books"
                        ],
```

```
                        "ResourceType": "dashboard"
                    },
                    {
                        "Resource": [
                            "collection/books"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "AllowFromPublic": true,
                "Description": "Public access for books collection"
            }
        ],
        "createdDate": 1669240256955,
        "lastModifiedDate": 1669240256955
    }
}
```

5. Crie a coleção livros:

```
aws opensearchserverless create-collection --name books --type SEARCH
```

Exemplo de resposta

```
{
    "createCollectionDetail": {
        "id": "8kw362bpwg4gx9b2f6e0",
        "name": "books",
        "status": "CREATING",
        "type": "SEARCH",
        "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
        "kmsKeyArn": "auto",
        "createdDate": 1669240325037,
        "lastModifiedDate": 1669240325037
    }
}
```

6. Crie uma política de acesso a dados que forneça as permissões mínimas para indexar e pesquisar dados na coleção livros. Substitua o ARN da entidade principal pelo ARN do `TutorialRole` da etapa 1:

```
aws opensearchserverless create-access-policy \
  --name books-policy \
  --type data \
  --policy "[{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":
[\"index\/books\/books-index\"],\"Permission\":[\"aoss:CreateIndex
\",\"aoss:DescribeIndex\",\"aoss:ReadDocument\",\"aoss:WriteDocument
\",\"aoss:UpdateIndex\",\"aoss:DeleteIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:role\/TutorialRole\"]}]"
```

Exemplo de resposta

```
{
    "accessPolicyDetail": {
        "type": "data",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDM5NDY1M18x",
        "policy": [
            {
                "Rules": [
                    {
                        "Resource": [
                            "index/books/books-index"
                        ],
                        "Permission": [
                            "aoss:CreateIndex",
                            "aoss:DescribeIndex",
                            "aoss:ReadDocument",
                            "aoss:WriteDocument",
                            "aoss:UpdateDocument",
                            "aoss:DeleteDocument"
                        ],
                        "ResourceType": "index"
                    }
                ],
                "Principal": [
                    "arn:aws:iam::123456789012:role/TutorialRole"
                ]
            }
        ],
        "createdDate": 1669240394653,
        "lastModifiedDate": 1669240394653
    }
```

```
}
```

O `TutorialRole` agora deve ser capaz de indexar e pesquisar documentos na coleção livros.

7. Para fazer chamadas para a OpenSearch API, você precisa do endpoint da coleção. Envie a seguinte solicitação para recuperar o parâmetro `collectionEndpoint`:

```
aws opensearchserverless batch-get-collection --names books
```

Exemplo de resposta

```
{
    "collectionDetails": [
        {
            "id": "8kw362bpwg4gx9b2f6e0",
            "name": "books",
            "status": "ACTIVE",
            "type": "SEARCH",
            "description": "",
            "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
            "createdDate": 1665765327107,
            "collectionEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com",
            "dashboardEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com/_dashboards"
        }
    ],
    "collectionErrorDetails": []
}
```

> **Note**
>
> Não será possível ver o endpoint da coleção até que o status da coleção mude para `ACTIVE`. Talvez seja necessário fazer várias chamadas para verificar o status até que a coleção seja criada com êxito.

8. Use uma ferramenta HTTP, como o Postman ou curl, para indexar dados na coleção livros. Criaremos um índice chamado books-index e adicionaremos um único documento.

Envie a solicitação a seguir para o endpoint da coleção que você recuperou na etapa anterior, usando as credenciais do `TutorialRole`.

```
PUT https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_doc/1
{
  "title": "The Shining",
  "author": "Stephen King",
  "year": 1977
}
```

Exemplo de resposta

```
{
  "_index" : "books-index",
  "_id" : "1",
  "_version" : 1,
  "result" : "created",
  "_shards" : {
    "total" : 0,
    "successful" : 0,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 0
}
```

9. Para começar a pesquisar dados em sua coleção, use a API de pesquisa. A consulta a seguir executa uma pesquisa básica:

```
GET https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_search
```

Exemplo de resposta

```
{
    "took": 405,
    "timed_out": false,
    "_shards": {
        "total": 6,
        "successful": 6,
        "skipped": 0,
```

```
        "failed": 0
    },
    "hits": {
        "total": {
            "value": 2,
            "relation": "eq"
        },
        "max_score": 1.0,
        "hits": [
            {
                "_index": "books-index:0::3xJq14MBUaOS0wL26UU9:0",
                "_id": "F_bt4oMBLle5pYmm5q4T",
                "_score": 1.0,
                "_source": {
                    "title": "The Shining",
                    "author": "Stephen King",
                    "year": 1977
                }
            }
        ]
    }
}
```

Identity and Access Management para Amazon OpenSearch Serverless

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar recursos sem OpenSearch servidor. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- Políticas baseadas em identidade para servidores sem servidor OpenSearch
- Ações políticas para OpenSearch Serverless
- Recursos de políticas para OpenSearch Serverless
- Chaves de condição de política para Amazon OpenSearch Serverless
- ABAC com Serverless OpenSearch
- Usando credenciais temporárias com Serverless OpenSearch

- Funções vinculadas a serviços para Serverless OpenSearch
- Outros tipos de política
- Exemplos de políticas baseadas em identidade para Serverless OpenSearch
- Suporte do IAM Identity Center para Amazon OpenSearch Serverless

Políticas baseadas em identidade para servidores sem servidor OpenSearch

Compatível com políticas baseadas em identidade: sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que podem ser anexados a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte Referência de elemento de política JSON do IAM no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade para Serverless OpenSearch

Para ver exemplos de políticas baseadas em identidade OpenSearch sem servidor, consulte. the section called “Exemplos de políticas baseadas em identidade”

Ações políticas para OpenSearch Serverless

Compatível com ações de políticas: sim

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

As ações de política no OpenSearch Serverless usam o seguinte prefixo antes da ação:

```
aoss
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [
      "aoss:action1",
      "aoss:action2"
         ]
```

É possível especificar várias ações usando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `Describe`, inclua a seguinte ação:

```
"Action": "aoss:List*"
```

Para ver exemplos de políticas baseadas em identidade OpenSearch sem servidor, consulte. Exemplos de políticas baseadas em identidade para Serverless OpenSearch

Recursos de políticas para OpenSearch Serverless

Compatível com recursos de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Para ações que não oferecem compatibilidade com permissões em nível de recurso, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Chaves de condição de política para Amazon OpenSearch Serverless

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` especifica quando as instruções são executadas com base em critérios definidos. É possível criar expressões condicionais que usem agentes de condição, como “igual a”

ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Além do controle de acesso baseado em atributos (ABAC), o OpenSearch Serverless oferece suporte às seguintes chaves de condição:

- `aoss:collection`
- `aoss:CollectionId`
- `aoss:index`

É possível usar essas chaves de condição mesmo ao fornecer permissões para políticas de acesso e políticas de segurança. Por exemplo:

```
[
   {
      "Effect":"Allow",
      "Action":[
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringLike":{
            "aoss:collection":"log"
         }
      }
   }
]
```

Neste exemplo, a condição se aplica às políticas que contenham regras que correspondam a um nome ou padrão de coleção. As condições têm o seguinte comportamento:

- `StringEquals`: aplica-se a políticas com regras que contenham a string de recurso “log” exata (ou seja, `collection/log`).
- `StringLike`: aplica-se a políticas com regras que contenham uma string de recurso que inclua a string “log” (ou seja, `collection/log`, mas também `collection/logs-application` ou `collection/applogs123`).

> **Note**
>
> As chaves de condição coleção não se aplicam ao nível do índice. Por exemplo, na política acima, a condição não se aplicaria a uma política de acesso ou segurança contendo a string de recurso `index/logs-application/*`.

Para ver uma lista de chaves de condição OpenSearch sem servidor, consulte Chaves de condição para Amazon OpenSearch Serverless na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte Ações definidas pelo Amazon OpenSearch Serverless.

ABAC com Serverless OpenSearch

Compatível com ABAC (tags em políticas): sim

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos chamados de tags. Você pode anexar tags a entidades e AWS recursos do IAM e, em seguida, criar políticas ABAC para permitir operações quando a tag do diretor corresponder à tag no recurso.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no elemento de condição de uma política usando as `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou chaves de condição `aws:TagKeys`.

Se um serviço for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para saber mais sobre o ABAC, consulte Definir permissões com autorização do ABAC no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar controle de acesso por atributo (ABAC) no Guia do usuário do IAM.

Para obter mais informações sobre a marcação de recursos OpenSearch sem servidor, consulte. the section called “Aplicação de tags nas coleções”

Usando credenciais temporárias com Serverless OpenSearch

Compatível com credenciais temporárias: sim

As credenciais temporárias fornecem acesso de curto prazo aos AWS recursos e são criadas automaticamente quando você usa a federação ou troca de funções. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para ter mais informações, consulte Credenciais de segurança temporárias no IAM e Serviços da Serviços da AWS que funcionam com o IAM no Guia do usuário do IAM.

Funções vinculadas a serviços para Serverless OpenSearch

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir o perfil de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar e gerenciar funções vinculadas a serviços OpenSearch sem servidor, consulte. the section called “Função de criação de coleção”

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- Políticas de controle de serviço (SCPs) — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em AWS Organizations. AWS Organizations é um serviço para agrupar e gerenciar centralmente várias AWS contas que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as suas contas. O SCP limita as permissões para entidades nas contas dos membros, incluindo cada usuário raiz AWS da conta. Para obter mais informações sobre Organizations e SCPs, consulte Políticas de controle de serviços no Guia AWS Organizations do Usuário.
- Políticas de controle de recursos (RCPs) — RCPs são políticas JSON que você pode usar para definir o máximo de permissões disponíveis para recursos em suas contas sem atualizar as políticas do IAM anexadas a cada recurso que você possui. A RCP limita as permissões para recursos nas contas-membro e pode afetar as permissões efetivas para identidades, incluindo o usuário-raiz da conta da AWS , independentemente de pertencerem a sua organização. Para obter mais informações sobre Organizations e RCPs, incluindo uma lista de AWS serviços que oferecem

suporte RCPs, consulte Políticas de controle de recursos (RCPs) no Guia AWS Organizations do usuário.

Exemplos de políticas baseadas em identidade para Serverless OpenSearch

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos OpenSearch sem servidor. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos pelo Amazon OpenSearch Serverless, incluindo o formato de cada um dos ARNs tipos de recursos, consulte Ações, recursos e chaves de condição do Amazon OpenSearch Serverless na Referência de Autorização de Serviço.

Tópicos

- Melhores práticas de políticas
- Usando o OpenSearch Serverless no console
- Administrando coleções sem OpenSearch servidor
- Visualizando OpenSearch coleções sem servidor
- Usando operações OpenSearch de API
- ABAC para operações de OpenSearch API

Melhores práticas de políticas

As políticas baseadas em identidade são muito poderosas. Eles determinam se alguém pode criar, acessar ou excluir recursos OpenSearch sem servidor em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos OpenSearch sem servidor em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas

AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para saber mais, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para funções de trabalho no Guia do usuário do IAM.

- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para saber mais sobre como usar o IAM para aplicar permissões, consulte Políticas e permissões no IAM no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: é possível adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para saber mais, consulte Elementos da política JSON do IAM: condição no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para saber mais, consulte Validação de políticas do IAM Access Analyzer no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para saber mais, consulte Configuração de acesso à API protegido por MFA no Guia do Usuário do IAM.

Para saber mais sobre as práticas recomendadas do IAM, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.

Usando o OpenSearch Serverless no console

Para acessar o OpenSearch Serverless no console OpenSearch de serviço, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os recursos OpenSearch sem servidor em sua AWS conta. Se você criar uma política

baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (como perfis do IAM) com essa política.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que corresponderem a operação da API que você estiver tentando executar.

A política a seguir permite que um usuário acesse o OpenSearch Serverless no console de OpenSearch serviço:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "aoss:ListCollections",
                "aoss:BatchGetCollection",
                "aoss:ListAccessPolicies",
                "aoss:ListSecurityConfigs",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:ListVpcEndpoints",
                "aoss:GetAccessPolicy",
                "aoss:GetAccountSettings",
                "aoss:GetSecurityConfig",
                "aoss:GetSecurityPolicy"
            ]
        }
    ]
}
```

Administrando coleções sem OpenSearch servidor

Essa política é um exemplo de política de “administrador de coleções” que permite ao usuário gerenciar e administrar coleções Amazon OpenSearch Serverless. O usuário pode criar, exibir e excluir coleções.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:aoss:us-east-1:111122223333:collection/*",
            "Action": [
                "aoss:CreateCollection",
                "aoss:DeleteCollection",
                "aoss:UpdateCollection"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "aoss:CreateAccessPolicy",
                "aoss:CreateSecurityPolicy"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Visualizando OpenSearch coleções sem servidor

Este exemplo de política permite que um usuário visualize detalhes de todas as coleções Amazon OpenSearch Serverless em sua conta. O usuário não pode modificar as coleções nem as políticas de segurança associadas.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Resource": "*",
```

```
            "Action": [
                "aoss:ListAccessPolicies",
                "aoss:ListCollections",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:BatchGetCollection"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Usando operações OpenSearch de API

As operações da API do plano de dados consistem nas funções que você usa no OpenSearch Serverless para obter valor em tempo real do serviço. As operações da API do ambiente de gerenciamento consistem nas funções que você usa para configurar o ambiente.

Para acessar o plano de dados APIs e os OpenSearch painéis do Amazon OpenSearch Serverless a partir do navegador, você precisa adicionar duas permissões do IAM para recursos de coleta. Essas permissões são `aoss:APIAccessAll` e `aoss:DashboardsAccessAll`.

> **Note**
>
> A partir de 10 de maio de 2023, o OpenSearch Serverless exige essas duas novas permissões do IAM para recursos de coleta. A `aoss:APIAccessAll` permissão permite o acesso ao plano de dados e a `aoss:DashboardsAccessAll` permissão permite OpenSearch painéis a partir do navegador. A falha na adição das duas novas permissões do IAM resulta em um erro 403.

Este exemplo de política permite que um usuário acesse o plano APIs de dados de uma coleção específica em sua conta e acesse os OpenSearch painéis de todas as coleções em sua conta.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        },
        {
            "Effect": "Allow",
            "Action": "aoss:DashboardsAccessAll",
            "Resource": "arn:aws:aoss:us-east-1:111122223333:dashboards/default"
        }
    ]
}
```

Ambos `aoss:APIAccessAll` `aoss:DashboardsAccessAll` fornecem permissão total do IAM aos recursos da coleção, enquanto a permissão Dashboards também fornece acesso aos OpenSearch painéis. Cada permissão funciona de forma independente, portanto, uma negação explícita de `aoss:APIAccessAll` não bloqueia o acesso aos recursos de `aoss:DashboardsAccessAll`, incluindo as Ferramentas de desenvolvimento. O mesmo vale para uma negaçãoaoss:DashboardsAccessAll. OpenSearch O Serverless oferece suporte às seguintes chaves de condição globais:

- `aws:CalledVia`
- `aws:CalledViaAWSService`
- `aws:CalledViaFirst`
- `aws:CalledViaLast`
- `aws:CurrentTime`
- `aws:EpochTime`
- `aws:PrincipalAccount`
- `aws:PrincipalArn`
- `aws:PrincipallsAWSService`
- `aws:PrincipalOrgID`
- `aws:PrincipalOrgPaths`
- `aws:PrincipalType`
- `aws:PrincipalServiceName`

- aws:PrincipalServiceNamesList
- aws:ResourceAccount
- aws:ResourceOrgID
- aws:ResourceOrgPaths
- aws:RequestedRegion
- aws:ResourceTag
- aws:SourceIp
- aws:SourceVpce
- aws:SourceVpc
- aws:userid
- aws:username
- aws:VpcSourceIp

O seguinte exemplo mostra o uso de aws:SourceIp no bloco de condição da política do IAM da entidade principal para chamadas ao plano de dados:

```
"Condition": {
    "IpAddress": {
         "aws:SourceIp": "203.0.113.0"
    }
}
```

O seguinte exemplo mostra o uso de aws:SourceVpc no bloco de condição da política do IAM da entidade principal para chamadas ao plano de dados:

```
"Condition": {
    "StringEquals": {
        "aws:SourceVpc": "vpc-0fdd2445d8EXAMPLE"
    }
}
```

Além disso, é oferecido suporte para as seguintes chaves específicas do OpenSearch Serverless:

- aoss:CollectionId
- aoss:collection

O seguinte exemplo mostra o uso de `aoss:collection` no bloco de condição da política do IAM da entidade principal para chamadas ao plano de dados:

```
"Condition": {
    "StringLike": {
         "aoss:collection": "log-*"
    }
}
```

ABAC para operações de OpenSearch API

As políticas baseadas em identidade permitem que você use tags para controlar o acesso ao plano de dados Amazon OpenSearch Serverless. APIs A política a seguir é um exemplo para permitir que diretores anexados acessem o plano de dados APIs se a coleção tiver a `team:devops` tag:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/team": "devops"
                }
            }
        }
    ]
}
```

A política a seguir é um exemplo para negar que diretores anexados acessem o plano de dados APIs e o acesso aos painéis se a coleção tiver a `environment:production` tag:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collection-id"
        }
    ]
}
```

O Amazon OpenSearch Serverless não oferece suporte a chaves `RequestTag` de condição `TagKeys` globais para planos de dados. APIs

Suporte do IAM Identity Center para Amazon OpenSearch Serverless

Suporte do IAM Identity Center para Amazon OpenSearch Serverless

Você pode usar os diretores do IAM Identity Center (usuários e grupos) para acessar dados do Amazon OpenSearch Serverless por meio dos Amazon Applications. OpenSearch Para habilitar o suporte do IAM Identity Center para Amazon OpenSearch Serverless, você precisará habilitar o uso do IAM Identity Center. Para saber mais sobre como fazer isso, consulte O que é o IAM Identity?

> **Note**
>
> Para acessar as coleções do Amazon OpenSearch Serverless usando usuários ou grupos do IAM Identity Center, você deve usar o recurso OpenSearch UI (Applications). O acesso direto aos painéis OpenSearch sem servidor usando as credenciais do IAM Identity Center não é suportado. Para obter mais informações, consulte Introdução à interface do OpenSearch usuário.

Depois que a instância do IAM Identity Center é criada, o administrador da conta do cliente precisa criar um aplicativo do IAM Identity Center para o OpenSearch serviço Amazon Serverless. Isso pode ser feito chamando o CreateSecurityConfig:. O administrador da conta de cliente pode especificar quais atributos serão usados para autorizar a solicitação. Os atributos padrão usados são `UserId` e `GroupId.`

A integração do IAM Identity Center para o Amazon OpenSearch Serverless usa as seguintes permissões AWS do IAM Identity Center (IAM):

- `aoss:CreateSecurityConfig`: criar um provedor do IAM Identity Center
- `aoss:ListSecurityConfig`: listar todos os provedores do IAM Identity Center na conta atual.
- `aoss:GetSecurityConfig`: visualizar informações do provedor do IAM Identity Center.
- `aoss:UpdateSecurityConfig`: modificar uma determinada configuração do IAM Identity Center
- `aoss:DeleteSecurityConfig`: excluir um provedor do IAM Identity Center

A seguinte política de acesso baseada em identidade pode ser usada para gerenciar todas as configurações do IAM Identity Center:

JSON

```
{
"Version": "2012-10-17",
    "Statement": [
        {
"Action": [
                "aoss:CreateSecurityConfig",
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> O elemento `Resource` deve ser um caractere curinga.

Criar um provedor do IAM Identity Center (console)

Você pode criar um provedor do IAM Identity Center para habilitar a autenticação com o OpenSearch aplicativo. Para habilitar a autenticação do IAM Identity Center para OpenSearch painéis, execute as seguintes etapas:

1. Faça login no console do Amazon OpenSearch Service.
2. No painel de navegação à esquerda, expanda Tecnologia sem servidor e escolha Autenticação.
3. Escolha Autenticação do IAM Identity Center.
4. Selecione Editar
5. Marque a caixa ao lado de Autenticar com o IAM Identity Center.
6. Selecione a chave de atributo usuário e grupo no menu suspenso. Os atributos de usuário serão usados para autorizar usuários com base em `UserName`, `UserId` e `Email`. Os atributos de usuário serão usados para autorizar usuário com base em `GroupName` e `GroupId`.
7. Selecione a instância do IAM Identity Center.
8. Selecione Salvar.

Criar provedor do IAM Identity Center (AWS CLI)

Para criar um provedor do IAM Identity Center usando o AWS Command Line Interface (AWS CLI), use o seguinte comando:

```
aws opensearchserverless create-security-config \
--region us-east-2 \
--name "iamidentitycenter-config" \
--description "description" \
--type "iamidentitycenter" \
--iam-identity-center-options '{
    "instanceArn": "arn:aws:sso:::instance/ssoins-99199c99e99ee999",
    "userAttribute": "UserName",
    "groupAttribute": "GroupId"
}'
```

Depois que um IAM Identity Center é habilitado, os clientes somente podem modificar os atributos de usuário e grupo.

```
aws opensearchserverless update-security-config \
--region us-east-1 \
--id <id_from_list_security_configs> \
--config-version <config_version_from_get_security_config> \
--iam-identity-center-options-updates '{
    "userAttribute": "UserId",
    "groupAttribute": "GroupId"
}'
```

Para visualizar o provedor do IAM Identity Center usando o AWS Command Line Interface, use o seguinte comando:

```
aws opensearchserverless list-security-configs --type iamidentitycenter
```

Excluir provedor do IAM Identity Center

O IAM Identity Center oferece duas instâncias de provedores, uma para a conta da organização e outra para a sua conta de membro. Se precisar alterar a instância do IAM Identity Center, você terá que excluir a configuração de segurança por meio da API `DeleteSecurityConfig` e criar uma nova configuração de segurança usando a nova instância do IAM Identity Center. O seguinte comando pode ser usado para excluir um provedor do IAM Identity Center:

```
aws opensearchserverless delete-security-config \
--region us-east-1 \
--id <id_from_list_security_configs>
```

Conceder ao IAM Identity Center acesso a dados de coleção

Depois que o provedor do IAM Identity Center está habilitado, você pode atualizar a política de acesso a dados da coleção para incluir as entidades principais do IAM Identity Center. As entidades principais do IAM Identity Center precisam ser atualizadas no seguinte formato:

```
[
   {
"Rules":[
       ...
      ],
```

```
      "Principal":[
         "iamidentitycenter/<iamidentitycenter-instance-id>/user/<UserName>",
         "iamidentitycenter/<iamidentitycenter-instance-id>/group/<GroupId>"
      ]
   }
]
```

> **Note**
>
> O Amazon OpenSearch Serverless oferece suporte a apenas uma instância do IAM Identity Center para todas as coleções de clientes e pode suportar até 100 grupos para um único usuário. Se você tentar usar mais do que o número permitido de instâncias, terá inconsistência no processamento da autorização da política de acesso a dados e receberá uma mensagem de erro `403`.

É possível conceder acesso a coleções, índices ou ambos. Se você quiser que usuários diferentes tenham permissões diferentes, precisará criar várias regras. Para obter uma lista das permissões disponíveis, consulte Identity and Access Management no Amazon OpenSearch Service. Para obter informações sobre como formatar uma política de acesso, consulte Conceder a identidades SAML acesso a dados de coleções.

Criptografia no Amazon OpenSearch Serverless

Criptografia em repouso

Cada coleção Amazon OpenSearch Serverless que você cria é protegida com criptografia de dados em repouso, um recurso de segurança que ajuda a impedir o acesso não autorizado aos seus dados. A criptografia em repouso usa AWS Key Management Service (AWS KMS) para armazenar e gerenciar suas chaves de criptografia. Ela usa o algoritmo Advanced Encryption Standard com chaves de 256 bits (AES-256) para executar a criptografia.

Tópicos

- Políticas de criptografia
- Considerações
- Permissões obrigatórias
- Política de chaves para uma chave gerenciada pelo cliente
- Como o OpenSearch Serverless usa subsídios em AWS KMS

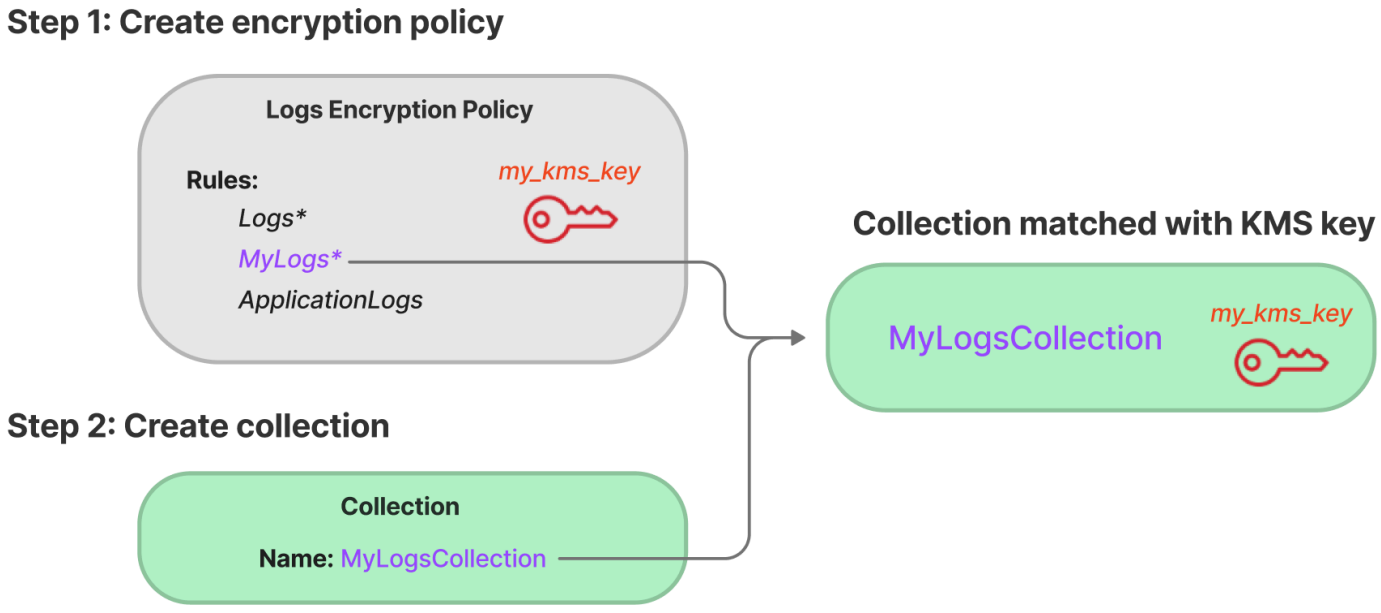
- Criação de políticas de criptografia (console)
- Criação de políticas de criptografia (AWS CLI)
- Exibição de políticas de criptografia
- Atualização de políticas de criptografia
- Exclusão de políticas de criptografia

Políticas de criptografia

Com as políticas de criptografia, é possível gerenciar várias coleções em grande escala atribuindo automaticamente uma chave de criptografia às coleções recém-criadas que correspondam a um nome ou padrão específico.

Ao criar uma política de criptografia, é possível especificar um prefixo, que é uma regra de correspondência baseada em curingas, como `MyCollection*`, ou inserir um único nome de coleção. Em seguida, quando você criar uma coleção que corresponda a esse padrão de nome ou prefixo, a política e a chave do KMS correspondente serão automaticamente atribuídas a ela.

Ao criar uma coleção, você pode especificar uma AWS KMS chave de duas maneiras: por meio de políticas de segurança ou diretamente na `CreateCollection` solicitação. Se você fornecer uma AWS KMS chave como parte da `CreateCollection` solicitação, ela terá precedência sobre qualquer política de segurança correspondente. Com essa abordagem, você tem a flexibilidade de substituir as configurações de criptografia baseadas em políticas para coleções específicas quando necessário.



As políticas de criptografia contêm os seguintes elementos:

- `Rules`: uma ou mais regras de correspondência de coleções, cada uma com os seguintes subelementos:
  - `ResourceType`: no momento, a única opção é “collection” (coleção). As políticas de criptografia se aplicam somente aos recursos de coleção.
  - `Resource`: um ou mais nomes ou padrões de coleção aos quais a política será aplicada, no formato `collection/<collection name|pattern>`.
- `AWSOwnedKey`: opção de uso de uma Chave pertencente à AWS.
- `KmsARN`: se você definir `AWSOwnedKey` como falso, especifique o nome do recurso da Amazon (ARN) da chave do KMS com a qual criptografar as coleções associadas. Se você incluir esse parâmetro, o OpenSearch Serverless ignorará o parâmetro. `AWSOwnedKey`

O exemplo de política a seguir atribuirá uma chave gerenciada pelo cliente a qualquer coleção futura denominada `autopartsinventory`, bem como às coleções que comecem com o termo “sales” (vendas):

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:kms:us-east-1:123456789012:key/93fd6da4-a317-4c17-
bfe9-382b5d988b36"
}
```

Mesmo que uma política corresponda a um nome de coleção, é possível optar por substituir essa atribuição automática durante a criação da coleção se o padrão do recurso contiver um caractere curinga (*). Se você optar por substituir a atribuição automática de chaves, o OpenSearch Serverless cria uma política de criptografia para você chamada auto-**< *collection-name* >** e a anexa à coleção. Inicialmente, a política só se aplica a uma única coleção, mas é possível modificá-la para incluir coleções adicionais.

Se você modificar as regras de política para que não correspondam mais a uma coleção, a chave do KMS associada não terá a atribuição a essa coleção cancelada. A coleção permanece sempre criptografada com sua chave de criptografia inicial. Se você desejar alterar a chave de criptografia de uma coleção, deverá recriar a coleção.

Se as regras de várias políticas corresponderem a uma coleção, a regra mais específica será usada. Por exemplo, se uma política contiver uma regra para `collection/log*` e outra para `collection/logSpecial`, a chave de criptografia da segunda política será usada porque é mais específica.

Você não pode usar um nome ou prefixo em uma política se ela já existir em outra política. OpenSearch O Serverless exibirá um erro se você tentar configurar padrões de recursos idênticos em políticas de criptografia diferentes.

Considerações

Considere o seguinte ao configurar a criptografia de suas coleções:

- A criptografia em repouso é obrigatória para todas as coleções do Sem Servidor.
- Você tem a opção de usar uma chave gerenciada pelo cliente ou uma Chave pertencente à AWS. Se você escolher uma chave gerenciada pelo cliente, recomendamos habilitar a rotação automática de chaves.
- Não é possível alterar a chave de criptografia de uma coleção depois que a coleção é criada. Escolha cuidadosamente qual AWS KMS usar na primeira vez que você configura uma coleção.
- Uma coleção só pode corresponder a uma única política de criptografia.
- Coleções com chaves KMS exclusivas não podem compartilhar Unidades de OpenSearch Computação (OCUs) com outras coleções. Cada coleção com uma chave exclusiva requer suas próprias 4 OCUs.
- Se você atualizar a chave do KMS em uma política de criptografia, a alteração não afetará as coleções correspondentes existentes com as chaves do KMS já atribuídas.
- OpenSearch O Serverless não verifica explicitamente as permissões do usuário nas chaves gerenciadas pelo cliente. Se um usuário tiver permissões para acessar uma coleção por meio de uma política de acesso a dados, ele poderá ingerir e consultar os dados criptografados com a chave associada.

Permissões obrigatórias

A criptografia em repouso para OpenSearch Serverless usa as seguintes permissões AWS Identity and Access Management (IAM). É possível especificar as condições do IAM para restringir os usuários a coleções específicas.

- `aoss:CreateSecurityPolicy`: cria uma política de criptografia.
- `aoss:ListSecurityPolicies`: lista todas as políticas e coleções de criptografia às quais elas estão vinculadas.
- `aoss:GetSecurityPolicy`: exibe os detalhes de uma política de criptografia específica.
- `aoss:UpdateSecurityPolicy`: modifica uma política de criptografia.
- `aoss:DeleteSecurityPolicy`: exclui uma política de criptografia.

O exemplo a seguir de política de acesso baseada em identidade fornece as permissões mínimas necessárias para que um usuário gerencie políticas de criptografia com o padrão de recursos `collection/application-logs`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "aoss:CreateSecurityPolicy",
            "aoss:UpdateSecurityPolicy",
            "aoss:DeleteSecurityPolicy",
            "aoss:GetSecurityPolicy"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "aoss:collection":"application-logs"
            }
         }
      },
      {
         "Effect":"Allow",
```

```
            "Action":[
               "aoss:ListSecurityPolicies"
            ],
            "Resource":"*"
         }
      ]
}
```

Política de chaves para uma chave gerenciada pelo cliente

Se você selecionar uma chave gerenciada pelo cliente para proteger uma coleção, a OpenSearch Serverless obterá permissão para usar a chave KMS em nome do diretor que faz a seleção. Esse diretor, um usuário ou uma função, deve ter as permissões na chave KMS que o OpenSearch Serverless exige. É possível fornecer essas permissões em uma política de chaves ou em uma política do IAM.

OpenSearch O Serverless faz `GenerateDataKey` e faz chamadas à API `Decrypt` KMS durante operações de manutenção, como escalonamento automático e atualizações de software. Você pode observar essas chamadas fora dos padrões de tráfego típicos. Essas chamadas fazem parte das operações normais do serviço e não indicam tráfego ativo de usuários.

OpenSearch O Serverless gera um `KMSKeyInaccessibleException` quando não consegue acessar a chave KMS que criptografa seus dados em repouso. Isso ocorre quando você desativa ou exclui a chave KMS ou revoga as concessões que permitem que o OpenSearch Serverless use a chave.

No mínimo, o OpenSearch Serverless exige as seguintes permissões em uma chave gerenciada pelo cliente:

- kms: DescribeKey
- kms: CreateGrant

Por exemplo:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
```

```
    {
        "Action": "kms:DescribeKey",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::123456789012:user/Dale"
        },
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "kms:ViaService": "aoss.us-east-1.amazonaws.com"
            }
        }
    },
    {
        "Action": "kms:CreateGrant",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::123456789012:user/Dale"
        },
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "kms:ViaService": "aoss.us-east-1.amazonaws.com"
            },
            "ForAllValues:StringEquals": {
                "kms:GrantOperations": [
                    "Decrypt",
                    "GenerateDataKey"
                ]
            },
            "Bool": {
                "kms:GrantIsForAWSResource": "true"
            }
        }
    }
  ]
}
```

OpenSearch Sem servidor, crie uma concessão com as permissões kms: GenerateDataKey e kms:decrypt.

Para saber mais, consulte Uso de políticas de chaves no AWS KMS no Guia do desenvolvedor do AWS Key Management Service .

Como o OpenSearch Serverless usa subsídios em AWS KMS

OpenSearch O Serverless exige uma concessão para usar uma chave gerenciada pelo cliente.

Quando você cria uma política de criptografia em sua conta com uma nova chave, o OpenSearch Serverless cria uma concessão em seu nome enviando uma CreateGrantsolicitação para. AWS KMS As concessões AWS KMS são usadas para dar acesso OpenSearch sem servidor a uma chave KMS em uma conta de cliente.

OpenSearch O Serverless exige que a concessão use sua chave gerenciada pelo cliente para as seguintes operações internas:

- Envie DescribeKeysolicitações AWS KMS para verificar se a ID simétrica da chave gerenciada pelo cliente fornecida é válida.
- Envie GenerateDataKeysolicitações para a chave KMS para criar chaves de dados com as quais criptografar objetos.
- Envie solicitações de descriptografia para AWS KMS descriptografar as chaves de dados criptografadas para que elas possam ser usadas para criptografar seus dados.

É possível revogar o acesso à concessão, ou remover o acesso do serviço à chave gerenciada pelo cliente a qualquer momento. Se você fizer isso, o OpenSearch Serverless não poderá acessar nenhum dos dados criptografados pela chave gerenciada pelo cliente, o que afeta todas as operações que dependem desses dados, levando a `AccessDeniedException` erros e falhas nos fluxos de trabalho assíncronos.

OpenSearch O Serverless retira as concessões em um fluxo de trabalho assíncrono quando uma determinada chave gerenciada pelo cliente não está associada a nenhuma política ou coleção de segurança.

Criação de políticas de criptografia (console)

Em uma política de criptografia, você especifica uma chave do KMS e uma série de padrões de coleção aos quais a política se aplicará. Qualquer nova coleção que corresponda a um dos padrões definidos na política receberá a chave do KMS correspondente quando você criar a coleção. Recomendamos que você crie políticas de criptografia antes de começar a criar coleções.

Para criar uma política de criptografia OpenSearch sem servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, expanda Sem Servidor e escolha Políticas de criptografia.
3. Escolha Criar política de criptografia.
4. Forneça um nome e uma descrição para a política.
5. Em Recursos, insira um ou mais padrões de recursos para essa política de criptografia. Todas as coleções recém-criadas na Conta da AWS e região atual que correspondam a um dos padrões serão automaticamente atribuídas a essa política. Por exemplo, se você inserir `ApplicationLogs` (sem nenhum curinga) e depois criar uma coleção com esse nome, a política e a chave do KMS correspondente serão atribuídas a essa coleção.

   Você também pode fornecer um prefixo como `Logs*`, que atribuirá a política a qualquer nova coleção com nomes começando com `Logs`. Usando curingas, é possível gerenciar as configurações de criptografia para várias coleções em grande escala.
6. Em Criptografia, escolha uma chave do KMS para usar.
7. Escolha Criar.

Próxima etapa: criar coleções

Depois de configurar uma ou mais políticas de criptografia, será possível começar a criar coleções que correspondam às regras definidas nessas políticas. Para instruções, consulte the section called “Criação de coleções”.

Na etapa Criptografias da criação da coleção, o OpenSearch Serverless informa que o nome inserido corresponde ao padrão definido em uma política de criptografia e atribui automaticamente a chave KMS correspondente à coleção. Se o padrão do recurso contiver um curinga (*), será possível optar por substituir a correspondência e selecionar sua própria chave.

Criação de políticas de criptografia (AWS CLI)

Para criar uma política de criptografia usando as operações da API OpenSearch Serverless, você especifica padrões de recursos e uma chave de criptografia no formato JSON. A CreateSecurityPolicysolicitação aceita políticas embutidas e arquivos.json.

As políticas de criptografia têm o formato a seguir. Esse arquivo `my-policy.json` de exemplo corresponde a qualquer coleção futura denominada `autopartsinventory`, bem como a qualquer coleção com nomes iniciando por `sales`.

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:kms:us-east-1:123456789012:key/93fd6da4-a317-4c17-
bfe9-382b5d988b36"
}
```

Para usar uma chave de propriedade do serviço, defina `AWSOwnedKey` como `true`:

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":true
}
```

A solicitação a seguir cria a política de criptografia:

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy file://my-policy.json
```

Em seguida, use a operação da CreateCollectionAPI para criar uma ou mais coleções que correspondam a um dos padrões de recursos.

Exibição de políticas de criptografia

Antes de criar uma coleção, talvez você queira pré-visualizar as políticas de criptografia existentes em sua conta para ver qual delas tem um padrão de recurso que corresponda ao nome da sua coleção. A ListSecurityPoliciessolicitação a seguir lista todas as políticas de criptografia em sua conta:

```
aws opensearchserverless list-security-policies --type encryption
```

A solicitação retorna informações sobre todas as políticas de criptografia configuradas. Use o conteúdo do elemento `policy` para visualizar as regras de padrões definidas na política:

```
{
   "securityPolicyDetails": [
      {
         "createdDate": 1663693217826,
         "description": "Sample encryption policy",
         "lastModifiedDate": 1663693217826,
         "name": "my-policy",
         "policy": "{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"]}],\"AWSOwnedKey\":true}",
         "policyVersion": "MTY2MzY5MzIxNzgyNl8x",
         "type": "encryption"
      }
   ]
}
```

Para ver informações detalhadas sobre uma política específica, incluindo a chave KMS, use o GetSecurityPolicycomando.

Atualização de políticas de criptografia

Se você atualizar a chave do KMS em uma política de criptografia, a alteração só se aplicará às coleções recém-criadas que correspondam ao nome ou padrão configurado. Isso não afeta as coleções existentes que já tenham chaves do KMS atribuídas.

O mesmo se aplica às regras de correspondência das políticas. Se você adicionar, modificar ou excluir uma regra, a alteração só se aplicará às coleções recém-criadas. As coleções existentes não perdem suas chaves do KMS atribuídas se você modificar as regras de uma política para que ela não corresponda mais ao nome de uma coleção.

Para atualizar uma política de criptografia no console OpenSearch sem servidor, escolha Políticas de criptografia, selecione a política a ser modificada e escolha Editar. Faça suas alterações e escolha Salvar.

Para atualizar uma política de criptografia usando a API OpenSearch Serverless, use a UpdateSecurityPolicyoperação. A solicitação a seguir atualiza uma política de criptografia com um novo documento JSON de política:

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy-version 2 \
    --policy file://my-new-policy.json
```

Exclusão de políticas de criptografia

Quando você exclui uma política de criptografia, todas as coleções que estiverem usando a chave do KMS definida na política não são afetadas. Para excluir uma política no console OpenSearch sem servidor, selecione a política e escolha Excluir.

Você também pode usar a DeleteSecurityPolicyoperação:

```
aws opensearchserverless delete-security-policy --name my-policy --type encryption
```

Criptografia em trânsito

No OpenSearch Serverless, todos os caminhos em uma coleção são criptografados em trânsito usando o Transport Layer Security 1.2 (TLS) com uma cifra AES-256 padrão do setor. O acesso a todos APIs os painéis do Opensearch também é feito por meio do TLS 1.2. O TLS é um conjunto de protocolos criptográficos padrão do setor usados para criptografar informações que são trocadas pela rede.

Acesso à rede para Amazon OpenSearch Serverless

As configurações de rede de uma coleção Amazon OpenSearch Serverless determinam se a coleção pode ser acessada pela Internet a partir de redes públicas ou se deve ser acessada de forma privada.

O acesso privado pode ser aplicado a um ou ambos os seguintes itens:

- OpenSearch VPC endpoints gerenciados sem servidor
- Compatível Serviços da AWS , como Amazon Bedrock

Você pode configurar o acesso à rede separadamente para o endpoint de uma coleção e o OpenSearchendpoint correspondente do OpenSearch Dashboards.

O acesso à rede é o mecanismo de isolamento para permitir o acesso de diferentes redes de origem. Por exemplo, se o endpoint de OpenSearch painéis de uma coleção estiver acessível publicamente, mas o endpoint da OpenSearch API não, um usuário poderá acessar os dados da coleção somente por meio de painéis ao se conectar a partir de uma rede pública. Se eles tentarem ligar OpenSearch APIs diretamente de uma rede pública, eles serão bloqueados. As configurações de rede podem ser usadas para essas permutações de origem para tipo de recurso. O Amazon OpenSearch Serverless oferece suporte a ambos IPv4 e IPv6 à conectividade.

Tópicos

- Políticas de rede
- Considerações
- Permissões necessárias para configurar políticas de rede
- Precedência das políticas
- Criação de políticas de rede (console)
- Criação de políticas de rede (AWS CLI)
- Exibição de políticas de rede
- Atualização de políticas de rede
- Exclusão de políticas de rede

Políticas de rede

As políticas de rede permitem que você gerencie várias coleções em escala, atribuindo automaticamente configurações de acesso à rede a coleções que correspondam às regras definidas na política.

Em uma política de rede, você especifica uma série de regras. Essas regras definem as permissões de acesso aos endpoints da coleção e aos endpoints do OpenSearch Dashboards. Cada regra consiste em um tipo de acesso (público ou privado) e um tipo de recurso (coleção e/ou endpoint de

OpenSearch painéis). Para cada tipo de recurso (`collection` e `dashboard`), você especifica uma série de regras que definem a quais coleções a política se aplicará.

Neste exemplo de política, a primeira regra especifica o acesso por endpoint da VPC ao endpoint da coleção e ao endpoint do Dashboards para todas as coleções que comecem com o termo `marketing*`. Também especifica o acesso ao Amazon Bedrock.

> **Note**
>
> O acesso privado Serviços da AWS , como o Amazon Bedrock, só se aplica ao endpoint da coleção, não ao OpenSearch endpoint do OpenSearch Dashboards. Mesmo que `ResourceType` sejadashboard, Serviços da AWS não é possível conceder acesso aos OpenSearch painéis.

A segunda regra especifica o acesso público à coleção `finance`, mas somente para o endpoint da coleção (sem acesso ao Dashboards).

```
[
   {
      "Description":"Marketing access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/marketing*"
            ]
         },
         {
            "ResourceType":"dashboard",
            "Resource":[
               "collection/marketing*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
```

```
      ],
   },
   {
      "Description":"Sales access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Essa política fornece acesso público somente aos OpenSearch painéis para coleções que começam com “finanças”. Qualquer tentativa de acessar diretamente a OpenSearch API falhará.

```
[
  {
    "Description": "Dashboards access",
    "Rules": [
      {
        "ResourceType": "dashboard",
        "Resource": [
          "collection/finance*"
        ]
      }
    ],
    "AllowFromPublic": true
  }
]
```

As políticas de rede podem ser aplicadas tanto às coleções existentes quanto às futuras. Por exemplo, é possível criar uma coleção e depois criar uma política de rede com uma regra que corresponda ao nome da coleção. Não é necessário criar políticas de rede para criar coleções.

Considerações

Considere o seguinte ao configurar o acesso de rede para suas coleções:

- Se você planeja configurar o acesso ao VPC endpoint para uma coleção, primeiro deve criar pelo menos um VPC endpoint gerenciado sem servidorOpenSearch .
- O acesso privado Serviços da AWS só se aplica ao endpoint da coleção, não ao OpenSearch endpoint do OpenSearch Dashboards. Mesmo que `ResourceType` sejadashboard, Serviços da AWS não é possível conceder acesso aos OpenSearch painéis.
- Se uma coleção for acessível a partir de redes públicas, ela também poderá ser acessada por todos os VPC endpoints OpenSearch gerenciados sem servidor e tudo mais. Serviços da AWS
- Várias políticas de rede podem ser aplicadas a uma única coleção. Para saber mais, consulte the section called “Precedência das políticas”.

Permissões necessárias para configurar políticas de rede

O acesso à rede para OpenSearch Serverless usa as seguintes permissões AWS Identity and Access Management (IAM). É possível especificar as condições do IAM para restringir os usuários a políticas de rede associadas a coleções específicas.

- `aoss:CreateSecurityPolicy`: crie uma política de acesso à rede.
- `aoss:ListSecurityPolicies`: lista todas as políticas de rede na conta atual.
- `aoss:GetSecurityPolicy`: exibe uma especificação de política de acesso à rede.
- `aoss:UpdateSecurityPolicy`: modifica uma determinada política de acesso à rede e altera o ID da VPC ou a designação de acesso público.
- `aoss:DeleteSecurityPolicy`: exclui uma política de acesso à rede (depois que ela for separada de todas as coleções).

A política de acesso baseada em identidade a seguir permite que um usuário exiba todas as políticas de rede e atualize as políticas com o padrão de recursos `collection/application-logs`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateSecurityPolicy"
```

```
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListSecurityPolicies",
                "aoss:GetSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Além disso, o OpenSearch Serverless exige `aoss:DashboardsAccessAll` permissões `aoss:APIAccessAll` e permissões para recursos de coleta. Para obter mais informações, consulte the section called “Usando operações OpenSearch de API”.

Precedência das políticas

Pode haver situações em que as regras das políticas de rede se sobreponham, dentro ou entre as políticas. Quando isso acontece, uma regra que especifique o acesso público substitui uma regra que especifique o acesso privado para quaisquer coleções que sejam comuns a ambas as regras.

Por exemplo, na política a seguir, ambas as regras atribuem acesso de rede à coleção `finance`, mas uma regra especifica o acesso por VPC enquanto a outra especifica o acesso público. Nessa situação, o acesso público substitui o acesso por VPC somente para a coleção finance (porque ele existe em ambas as regras), de modo que a coleção finance será acessível a partir de redes públicas. A coleção de vendas terá acesso por VPC a partir do endpoint especificado.

```
[
   {
```

```
      "Description":"Rule 1",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/sales",
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ]
   },
   {
      "Description":"Rule 2",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Se vários endpoints da VPC de regras diferentes se aplicarem a uma coleção, as regras serão aditivas e a coleção poderá ser acessada de todos os endpoints especificados. Se você definir `AllowFromPublictrue`, mas também fornecer um ou mais `SourceVPCEs` ou`SourceServices`, o OpenSearch Serverless ignorará os endpoints de VPC e os identificadores de serviço, e as coleções associadas terão acesso público.

Criação de políticas de rede (console)

As políticas de rede podem ser aplicadas tanto às coleções existentes quanto às futuras. Recomendamos que você crie políticas de rede antes de começar a criar coleções.

Para criar uma política de rede OpenSearch sem servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.

2. No painel de navegação à esquerda, expanda Sem Servidor e escolha Políticas de rede.
3. Escolha Criar política de rede.
4. Forneça um nome e uma descrição para a política.
5. Forneça uma ou mais regras. Essas regras definem permissões de acesso para suas coleções OpenSearch sem servidor e seus endpoints de OpenSearch painéis.

   Cada regra contém os seguintes elementos:

Elemento	Description
Nome da regra	Um nome que descreve o conteúdo da regra. Por exemplo, “Acesso por VPC para a equipe de marketing”.
Tipo de acesso	Escolha entre acesso público ou privado. Selecione uma das seguintes opções, ou ambas: • VPC endpoints para acesso — especifiq ue um ou mais VPC endpoints gerenciad os sem servidor — OpenSearch VPC endpoints gerenciados. • AWS service (Serviço da AWS) acesso privado — Selecione um ou mais compatíveis Serviços da AWS.
Tipo de atributo	Selecione se deseja fornecer acesso aos OpenSearch endpoints (o que permite fazer chamadas para a OpenSearch API), aos OpenSearch painéis (que permitem o acesso às visualizações e à interface do usuário para OpenSearch plug-ins) ou ambos. **Note** AWS service (Serviço da AWS) o acesso privado só se aplica

Elemento	Description
	ao endpoint da coleção, não ao OpenSearch endpoint do OpenSearch Dashboards. Mesmo se você selecionar OpenSearch Painéis, só Serviços da AWS poderá receber acesso ao endpoint.

Para cada tipo de recurso selecionado, você pode escolher coleções existentes às quais aplicar as configurações de política e and/or criar um ou mais padrões de recursos. Os padrões de recursos consistem em um prefixo e um caractere curinga (*), e definem a quais coleções as configurações de política se aplicarão.

Por exemplo, se você incluir um padrão chamado `Marketing*`, qualquer coleção nova ou existente cujos nomes comecem com “Marketing” terá as configurações de rede desta política aplicadas automaticamente a elas. Um único caractere curinga (`*`) aplica a política a todas as coleções atuais e futuras.

Além disso, você pode especificar o nome de uma coleção futura sem um caractere curinga, como`Finance`. OpenSearch O Serverless aplicará as configurações de política a qualquer coleção recém-criada com esse nome exato.

6. Quando estiver satisfeito com sua configuração de política, escolha Criar.

Criação de políticas de rede (AWS CLI)

Para criar uma política de rede usando as operações da API OpenSearch Serverless, você especifica regras no formato JSON. A CreateSecurityPolicysolicitação aceita políticas embutidas e arquivos.json. Todas as coleções e padrões devem assumir o formato `collection/<collection name|pattern>`.

> **Note**
>
> O tipo de recurso `dashboards` só permite a permissão para OpenSearch painéis, mas para que os OpenSearch painéis funcionem, você também deve permitir o acesso à coleção das mesmas fontes. Veja a segunda política a seguir como um exemplo.

Para especificar o acesso privado, inclua um dos seguintes elementos, ou ambos:

- `SourceVPCEs`— Especifique um ou mais VPC endpoints OpenSearch gerenciados sem servidor.
- `SourceServices`— Especifique o identificador de um ou mais compatíveis Serviços da AWS. No momento, os seguintes identificadores de serviço não são compatíveis:
  - `bedrock.amazonaws.com`: Amazon Bedrock

O exemplo de política de rede a seguir fornece acesso privado a um endpoint da VPC e ao Amazon Bedrock, a endpoints de coleção apenas para coleções que comecem com o prefixo `log*`. Usuários autenticados não podem entrar nos OpenSearch painéis; eles só podem acessar o endpoint de coleta de forma programática.

```
[
   {
      "Description":"Private access for log collections",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/log*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
      ],
   }
]
```

A política a seguir fornece acesso público ao OpenSearch endpoint e aos OpenSearch painéis para uma única coleção chamada. `finance` Se a coleção não existir, as configurações de rede serão aplicadas à coleção se e quando ela for criada.

```
[
   {
      "Description":"Public access for finance collection",
```

```
      "Rules":[
         {
            "ResourceType":"dashboard",
            "Resource":[
               "collection/finance"
            ]
         },
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

A solicitação a seguir cria a política de rede acima:

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type network \
    --policy "[{\"Description\":\"Public access for finance collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/finance\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/finance\"]}],
\"AllowFromPublic\":true}]"
```

Para fornecer a política em um arquivo JSON, use o formato `--policy file://my-policy.json`

Exibição de políticas de rede

Antes de criar uma coleção, talvez você queira pré-visualizar as políticas de rede existentes em sua conta para ver qual delas tem um padrão de recurso que corresponda ao nome da sua coleção. A ListSecurityPoliciessolicitação a seguir lista todas as políticas de rede em sua conta:

```
aws opensearchserverless list-security-policies --type network
```

A solicitação retorna informações sobre todas as políticas de rede configuradas. Para visualizar as regras de padrões definidas em uma política específica, encontre as informações sobre políticas no

conteúdo do elemento `securityPolicySummaries` na resposta. Observe o `name` final `type` desta política e use essas propriedades em uma GetSecurityPolicysolicitação para receber uma resposta com os seguintes detalhes da política:

```
{
    "securityPolicyDetail": [
        {
            "type": "network",
            "name": "my-policy",
            "policyVersion": "MTY2MzY5MTY1MDA3Ml8x",
            "policy": "[{\"Description\":\"My network policy rule\",\"Rules\":
[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection/*\"]}],\"AllowFromPublic
\":true}]",
            "createdDate": 1663691650072,
            "lastModifiedDate": 1663691650072
        }
    ]
}
```

Para ver informações detalhadas sobre uma política específica, use o GetSecurityPolicycomando.

Atualização de políticas de rede

Quando você modifica os endpoints da VPC ou a designação de acesso público para uma rede, todas as coleções associadas são afetadas. Para atualizar uma política de rede no console OpenSearch sem servidor, expanda Políticas de rede, selecione a política a ser modificada e escolha Editar. Faça suas alterações e escolha Salvar.

Para atualizar uma política de rede usando a API OpenSearch Serverless, use o UpdateSecurityPolicycomando. É necessário incluir uma versão da política na solicitação. É possível recuperar a versão da política usando os comandos `ListSecurityPolicies` ou `GetSecurityPolicy`. A inclusão da versão mais recente da política garante que você não anule inadvertidamente uma alteração feita por outra pessoa.

A solicitação a seguir atualiza uma política de rede com um novo documento JSON de política:

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type network \
    --policy-version MTY2MzY5MTY1MDA3Ml8x \
    --policy file://my-new-policy.json
```

Exclusão de políticas de rede

Antes de ser possível excluir uma política de rede, é preciso desvinculá-la de todas as coleções. Para excluir uma política no console OpenSearch sem servidor, selecione a política e escolha Excluir.

Você também pode usar o DeleteSecurityPolicycomando:

```
aws opensearchserverless delete-security-policy --name my-policy --type network
```

Conformidade com FIPS no Amazon Serverless OpenSearch

O Amazon OpenSearch Serverless oferece suporte ao Federal Information Processing Standards (FIPS) 140-2, que é um padrão do governo dos EUA e do Canadá que especifica requisitos de segurança para módulos criptográficos que protegem informações confidenciais. Quando você se conecta a endpoints habilitados para FIPS com o OpenSearch Serverless, as operações criptográficas ocorrem usando bibliotecas criptográficas validadas por FIPS.

OpenSearch Os endpoints FIPS sem servidor estão disponíveis Regiões da AWS onde o FIPS é suportado. Esses endpoints usam o TLS 1.2 ou posteriores e algoritmos criptográficos validados por FIPS em todas as comunicações. Para saber mais, consulte Conformidade com FIPS no Guia do usuário de acesso verificado pela AWS .

Tópicos

- Usando endpoints FIPS com Serverless OpenSearch
- Use endpoints FIPS com AWS SDKs
- Configurar grupos de segurança para endpoints da VPC
- Usar o endpoint FIPS da VPC
- Verificar a conformidade com FIPS
- Resolver problemas de conectividade de endpoint FIPS em zonas hospedadas privadas

Usando endpoints FIPS com Serverless OpenSearch

Regiões da AWS Onde o FIPS é suportado, as coleções OpenSearch sem servidor são acessíveis por meio de terminais padrão e compatíveis com FIPS. Para saber mais, consulte Conformidade com FIPS no Guia do usuário de acesso verificado pela AWS .

Nos exemplos a seguir, substitua *collection_id* e *Região da AWS* por seu ID de coleção e seu Região da AWS.

- Endpoint padrão: **https://*collection_id*.*Região da AWS*.aoss.amazonaws.com**.
- Endpoints conformes com FIPS: **https://*collection_id*.*Região da AWS*.aoss-fips.amazonaws.com**.

Da mesma forma, os OpenSearch painéis podem ser acessados por meio de terminais padrão e compatíveis com FIPS:

- Endpoint padrão do Dashboards: **https://*collection_id*.*Região da AWS*.aoss.amazonaws.com/_dashboards**.
- Endpoint do Dashboards conforme com FIPS: **https://*collection_id*.*Região da AWS*.aoss-fips.amazonaws.com/_dashboards**.

> **Note**
>
> Em regiões habilitadas para FIPS, tanto os endpoints padrão quanto os endpoints conformes com FIPS fornecem criptografia conforme com FIPS. Os endpoints específicos para FIPS ajudam você a atender aos requisitos de conformidade que exigem especificamente o uso de endpoints com FIPS no nome.

Use endpoints FIPS com AWS SDKs

Ao usar AWS SDKs, você pode especificar o endpoint FIPS ao criar o cliente. No exemplo a seguir, substitua *collection_id* e *Região da AWS* por seu ID de coleção e seu Região da AWS.

```
# Python SDK example
from opensearchpy import OpenSearch, RequestsHttpConnection, AWSV4SignerAuth
import boto3
host = '"https://collection_id.Região da AWS.aoss-fips.amazonaws.com"
region = 'us-west-2'
service = 'aoss'
credentials = boto3.Session().get_credentials()
auth = AWSV4SignerAuth(credentials, region, service)
client = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
```

```
    http_auth = auth,
    use_ssl = True,
    verify_certs = True,
    connection_class = RequestsHttpConnection,
    pool_maxsize = 20
)
```

Configurar grupos de segurança para endpoints da VPC

Para garantir a comunicação adequada com seu endpoint Amazon VPC (VPC) compatível com FIPS, crie ou modifique um grupo de segurança para permitir o tráfego HTTPS de entrada (porta TCP 443) dos recursos em sua VPC que precisam acessar o Serverless. OpenSearch Depois, associe esse grupo de segurança ao endpoint da VPC durante a criação ou modificando o endpoint após a criação. Para saber mais, consulte Criar um grupo de segurança no Guia do usuário da Amazon VPC.

Usar o endpoint FIPS da VPC

Depois de criar o VPC endpoint compatível com FIPS, você pode usá-lo para acessar o OpenSearch Serverless a partir de recursos dentro da sua VPC. Para usar o endpoint para operações de API, configure o SDK para usar o endpoint FIPS regional, como descrito na seção Usando endpoints FIPS com Serverless OpenSearch. Para acessar os OpenSearch painéis, use a URL específica dos painéis da coleção, que será roteada automaticamente pelo VPC endpoint compatível com FIPS quando acessada de dentro da sua VPC. Para obter mais informações, consulte Usando OpenSearch painéis com o Amazon Service OpenSearch.

Verificar a conformidade com FIPS

Para verificar se suas conexões com o OpenSearch Serverless estão usando criptografia compatível com FIPS, use para monitorar as chamadas de API feitas AWS CloudTrail para o Serverless. OpenSearch Verifique se o `eventSource` campo nos CloudTrail registros é exibido `aoss-fips.amazonaws.com` para chamadas de API.

Para acessar os OpenSearch painéis, você pode usar as ferramentas do desenvolvedor do navegador para inspecionar os detalhes da conexão TLS e verificar se os conjuntos de criptografia compatíveis com FIPS estão sendo usados.

Resolver problemas de conectividade de endpoint FIPS em zonas hospedadas privadas

Os endpoints FIPS funcionam com coleções Amazon OpenSearch Serverless que têm acesso público. Para coleções da VPC recém-criadas que usam endpoints da VPC recém-criados, os endpoints FIPS funcionam como esperado. Para outras coleções da VPC, talvez seja necessário realizar a configuração manual para garantir que os endpoints FIPS funcionem corretamente.

Para configurar zonas hospedadas privadas FIPS no Amazon Route 53

1. Abra o console do Route 53 em https://console.aws.amazon.com/route53/.
2. Revise as zonas hospedadas:
   a. Localize as zonas hospedadas nas quais Regiões da AWS suas coleções estão.
   b. Verifique os padrões de nomenclatura da zona hospedada:
      - Formato não FIPS: *region*`.aoss.amazonaws.com`.
      - Formato FIPS: *region*`.aoss-fips.amazonaws.com`.
   c. Confirme que Tipo está definido como Zona hospedada privada para todas as zonas hospedadas.
3. Se a zona hospedada privada FIPS estiver faltando:
   a. Selecione a zona hospedada privada não FIPS correspondente.
   b. Copie as VPCs informações associadas. Por exemplo: `vpc-1234567890abcdef0 | us-east-2`.
   c. Encontre o registro de domínio curinga. Por exemplo: `*.us-east-2.aoss.amazonaws.com`.
   d. Copie as informações de Valor/rotear tráfego para. Por exemplo: `uoc1c1qsw7poexampleewjeno1pte3rw.3ym756xh7yj.aoss.searchservices.aws`.
4. Crie a zona hospedada privada FIPS:
   a. Crie uma nova zona hospedada privada com o formato FIPS. Por exemplo: `us-east-2.aoss-fips.amazonaws.com`.
   b. VPCsEm Associado, insira as informações de VPC que você copiou da zona hospedada privada não FIPS.
5. Adicione uma nova regra com as seguintes configurações:

a. Nome do registro: *

b. Tipo de registro: CNAME

c. Valor: insira as informações de Valor/rotear tráfego para que você copiou anteriormente.

Problemas comuns

Se você tiver problemas de conectividade com os endpoints da VPC compatíveis com FIPS, use as informações a seguir para ajudar a resolver o problema.

- Falhas na resolução de DNS: você não consegue resolver o nome de domínio do endpoint FIPS em sua VPC
- Tempos limite de conexão: suas solicitações para o endpoint FIPS atingem o tempo limite
- Erros de acesso negado: ocorre falha na autenticação ou autorização ao usar endpoints FIPS
- Faltam registros de zona hospedada privada para coleções somente VPC

Para solucionar problemas de conectividade de endpoints FIPS

1. Verifique a configuração de zona hospedada privada:

   a. Confirme se existe uma zona hospedada privada para o domínio do endpoint FIPS (`*.region.aoss-fips.amazonaws.com`.

   b. Verifique se a zona hospedada privada está associada à VPC correta.

      Para saber mais, consulte Zonas hospedadas privadas no Guia do desenvolvedor do Amazon Route 53 e Gerenciar nomes do DNS no Guia do AWS PrivateLink .

2. Teste a resolução do DNS:

   a. Conecte-se a uma instância do EC2 na sua VPC.

   b. Execute este comando: .

   ```
   nslookup collection-id.region.aoss-fips.amazonaws.com
   ```

   c. Confirme se a resposta inclui o endereço IP privado do endpoint da VPC.

      Para saber mais, consulte Políticas de endpoint e Atributos de DNS no Guia do usuário da Amazon VPC.

3. Verifique as configurações do grupo de segurança:

   a. Verifique se o grupo de segurança conectado ao endpoint da VPC permite tráfego HTTPS (porta 443) vindo de seus recursos.

   b. Confirme se os grupos de segurança dos recursos permitem tráfego de saída para o endpoint da VPC.

   Para saber mais, consulte Políticas de endpoint no Guia da AWS PrivateLink e Grupos de segurança no Guia do usuário da Amazon VPC.

4. Revise a configuração de ACL da rede:

   - Verifique se a rede ACLs permite tráfego entre seus recursos e o VPC endpoint.

     Para obter mais informações, consulte Rede ACLs no Guia do usuário da Amazon VPC.

5. Revise a política de endpoint:

   - Verifique se a política de VPC endpoint permite as ações necessárias em seus recursos sem servidor. OpenSearch

     Para saber mais, consulte Permissões necessárias de endpoint da VPC e Política de endpoints no Guia do AWS PrivateLink .

> **Tip**
>
> Se você usa resolvedores de DNS personalizados em sua VPC, configure-os para encaminhar solicitações de `*.amazonaws.com` domínios para os servidores. AWS

Controle de acesso a dados para Amazon OpenSearch Serverless

Com o controle de acesso aos dados no Amazon OpenSearch Serverless, você pode permitir que os usuários acessem coleções e índices, independentemente do mecanismo de acesso ou da fonte de rede. É possível fornecer acesso a perfis do IAM e identidades de SAML.

Você gerencia as permissões de acesso por meio de políticas de acesso a dados que se aplicam às coleções e aos recursos de índice. As políticas de acesso a dados ajudam você a gerenciar coleções em grande escala atribuindo automaticamente permissões de acesso a coleções e índices que correspondam a um padrão específico. Várias políticas de acesso a dados podem ser aplicadas a

um único recurso. Observe que você deve ter uma política de acesso a dados para sua coleção para acessar a URL do seu OpenSearch painel.

Tópicos

- Políticas de acesso a dados versus políticas do IAM
- Permissões do IAM necessárias para configurar as políticas de acesso a dados
- Sintaxe da política
- Permissões de políticas com suporte
- Exemplos de conjuntos de dados em painéis OpenSearch
- Criação de políticas de acesso a dados (console)
- Criação de políticas de acesso a dados (AWS CLI)
- Exibição de políticas de acesso a dados
- Atualização de políticas de acesso a dados
- Exclusão de políticas de acesso a dados
- Acesso a dados entre contas

Políticas de acesso a dados versus políticas do IAM

As políticas de acesso aos dados são logicamente separadas das políticas AWS Identity and Access Management (IAM). As permissões do IAM controlam o acesso às operações da API do Sem Servidor, como `CreateCollection` e `ListAccessPolicies`. As políticas de acesso a dados controlam o acesso às OpenSearch operações suportadas pelo OpenSearch Serverless, como PUT `<index>` ou. `GET _cat/indices`

As permissões do IAM que controlam o acesso às operações da API da política de acesso a dados, como `aoss:CreateAccessPolicy` e `aoss:GetAccessPolicy` (descritas na próxima seção), não afetam a permissão especificada em uma política de acesso a dados.

Por exemplo, suponha que uma política do IAM impeça que um usuário crie políticas de acesso a dados para `collection-a`, mas permita que ele crie políticas de acesso a dados para todas as coleções (*):

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aoss:collection": "collection-a"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Se o usuário criar uma política de acesso a dados que permita certa permissão para todas as coleções (`collection/*` ou `index/*/*`), a política será aplicada a todas as coleções, incluindo a coleção A.

> **Important**
>
> Receber permissões em uma política de acesso a dados não é suficiente para acessar os dados em sua coleção OpenSearch Serverless. Uma entidade principal associada também deve ter acesso às permissões do IAM `aoss:APIAccessAll` e `aoss:DashboardsAccessAll`. Ambas as permissões concedem acesso total aos recursos da coleção, enquanto a permissão Painéis também fornece acesso aos OpenSearch Painéis. Se uma entidade principal não tiver essas duas permissões do IAM, receberá erros 403 ao tentar enviar solicitações para a coleção. Para saber mais, consulte the section called “Usando operações OpenSearch de API”.

Permissões do IAM necessárias para configurar as políticas de acesso a dados

O controle de acesso a dados para OpenSearch Serverless usa as seguintes permissões do IAM. É possível especificar condições do IAM para restringir os usuários a nomes de políticas de acesso específicas.

- `aoss:CreateAccessPolicy`: criar uma política de acesso.
- `aoss:ListAccessPolicies`: listar todas as políticas de acesso.
- `aoss:GetAccessPolicy`: exibir detalhes sobre uma política de acesso específica.
- `aoss:UpdateAccessPolicy`: modificar uma política de acesso.
- `aoss:DeleteAccessPolicy`: excluir uma política de acesso.

A seguinte política de acesso baseada em identidade permite que um usuário exiba todas as políticas de acesso e atualize as políticas que contenham o padrão de recursos `collection/logs`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "aoss:ListAccessPolicies",
                "aoss:GetAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "aoss:UpdateAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "logs"
                    ]
```

```
                }
            }
        }
    ]
}
```

> **Note**
>
> Além disso, o OpenSearch Serverless exige `aoss:DashboardsAccessAll` permissões `aoss:APIAccessAll` e permissões para recursos de coleta. Para obter mais informações, consulte the section called “Usando operações OpenSearch de API”.

Sintaxe da política

Uma política de acesso a dados inclui um conjunto de regras, cada uma com os seguintes elementos:

Elemento	Description
`ResourceType`	O tipo de recurso (coleção ou índice) ao qual as permissões se aplicam. As permissões de alias e modelo estão no nível da coleção, enquanto as permissões para criar, modificar e pesquisar dados estão no nível do índice. Para saber mais, consulte Permissões de políticas com suporte.
`Resource`	Uma lista de and/or padrões de nomes de recursos. Os padrões são prefixos seguidos por um curinga (*), que permitem que as permissões associadas sejam aplicadas a vários recursos. • As coleções assumem o formato `collection/` *`<name|pattern>`* . • Os índices assumem o formato `index/`*`<collection-name|pattern>`* */*`<index-name|pattern/>` .
`Permission`	Uma lista de permissões a serem concedidas para os recursos especific ados Para obter uma lista completa de permissões e as operações da API que elas permitem, consulte the section called “Operações e permissões de OpenSearch API suportadas”.

Elemento	Description
`Principal`	Uma lista de uma ou mais entidades principais às quais conceder acesso. Os principais podem ser a função do IAM ARNs ou as identidades do SAML. Essas entidades principais devem estar dentro da Conta da AWS atual. As políticas de acesso a dados não oferecem suporte direto ao acesso entre contas, mas você pode incluir uma função em sua política que um usuário de outro usuário Conta da AWS possa assumir na conta proprietária da coleção. Para obter mais informações, consulte the section called “Acesso a dados entre contas”.

O exemplo de política a seguir concede permissões de alias e modelo à coleção chamada `autopartsinventory`, bem como a quaisquer coleções iniciadas pelo prefixo `sales*`. Ele também concede permissões de leitura e gravação a todos os índices da coleção `autopartsinventory` e a todos os índices da coleção `salesorders` iniciados pelo prefixo `orders*`.

```
[
   {
      "Description": "Rule 1",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/autopartsinventory",
               "collection/sales*"
            ],
            "Permission":[
               "aoss:CreateCollectionItems",
               "aoss:UpdateCollectionItems",
               "aoss:DescribeCollectionItems"
            ]
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/autopartsinventory/*",
               "index/salesorders/orders*"
            ],
            "Permission":[
```

```
                "aoss:*"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/Dale",
         "arn:aws:iam::123456789012:role/RegulatoryCompliance",
         "saml/123456789012/myprovider/user/Annie",
         "saml/123456789012/anotherprovider/group/Accounting"
      ]
   }
]
```

Você não pode negar explicitamente o acesso em uma política. Dessa forma, todas as permissões de política são aditivas. Por exemplo, se uma política conceder a um usuário `aoss:ReadDocument` e outra conceder `aoss:WriteDocument`, o usuário terá ambas as permissões. Se uma terceira política conceder ao mesmo usuário `aoss:*`, o usuário poderá realizar todas as ações no índice associado; permissões mais restritivas não substituem as menos restritivas.

Permissões de políticas com suporte

Há suporte para as permissões a seguir nas políticas de acesso a dados. Para ver as operações de OpenSearch API que cada permissão permite, consultethe section called “Operações e permissões de OpenSearch API suportadas”.

Permissões de coleção

- `aoss:CreateCollectionItems`
- `aoss:DeleteCollectionItems`
- `aoss:UpdateCollectionItems`
- `aoss:DescribeCollectionItems`
- `aoss:*`

Permissões de índice

- `aoss:ReadDocument`
- `aoss:WriteDocument`
- `aoss:CreateIndex`

- `aoss:DeleteIndex`
- `aoss:UpdateIndex`
- `aoss:DescribeIndex`
- `aoss:*`

Exemplos de conjuntos de dados em painéis OpenSearch

OpenSearch Os painéis fornecem conjuntos de dados de amostra que vêm com visualizações, painéis e outras ferramentas para ajudá-lo a explorar os painéis antes de adicionar seus próprios dados. Para criar índices a partir desses dados de amostra, você precisa de uma política de acesso a dados que forneça permissões para o conjunto de dados com o qual você deseja trabalhar. A política a seguir usa um caractere curinga (*) para fornecer permissões aos três conjuntos de dados de amostra.

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/<collection-name>/opensearch_dashboards_sample_data_*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:DescribeIndex",
          "aoss:ReadDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::<account-id>:user/<user>"
    ]
  }
]
```

Criação de políticas de acesso a dados (console)

É possível criar uma política de acesso a dados usando o editor visual, ou no formato JSON. Qualquer nova coleção que corresponda a um dos padrões definidos na política receberá as permissões correspondentes quando você criar a coleção.

Para criar uma política de OpenSearch acesso a dados sem servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, expanda Tecnologia sem servidor e em Segurança, escolha Políticas de acesso a dados.
3. Selecione Criar política de acesso.
4. Forneça um nome e uma descrição para a política.
5. Forneça um nome para a primeira regra em sua política. Por exemplo, “Acesso à coleção de logs”.
6. Escolha Adicionar entidades principais e selecione um ou mais perfis do IAM, ou usuários e grupos de SAML aos quais fornecer acesso aos dados.

   > **Note**
   >
   > Para selecionar entidades principais nos menus suspensos, é necessário ter as permissões `iam:ListUsers` e `iam:ListRoles` (para entidades principais do IAM) e a permissão `aoss:ListSecurityConfigs` (para identidades de SAML).

7. Escolha Conceder e selecione o alias, o modelo e as permissões de índice para conceder às entidades principais associadas. Para obter uma lista completa de permissões e o acesso que elas permitem, consulte the section called “Operações e permissões de OpenSearch API suportadas”.
8. (Opcional) Configure regras adicionais para a política.
9. Escolha Criar. Pode haver cerca de um minuto de atraso entre a criação da política e o momento em que as permissões são aplicadas. Se demorar mais de 5 minutos, entre em contato com o Suporte.

> **Important**
>
> Se sua política incluir apenas permissões de indexação (e nenhuma permissão de coleção), talvez você ainda veja uma mensagem sobre coleções correspondentes informando o seguinte: `Collection cannot be accessed yet. Configure data access policies so that users can access the data within this collection.` Você pode ignorar esse aviso. As entidades principais autorizadas ainda podem realizar suas operações relacionadas ao índice atribuídas na coleção.

Criação de políticas de acesso a dados (AWS CLI)

Para criar uma política de acesso a dados usando a API OpenSearch Serverless, use o `CreateAccessPolicy` comando. O comando aceita tanto políticas em linha quanto arquivos .json. As políticas em linha devem ser codificadas como uma string JSON com escape.

A solicitação a seguir cria uma política de acesso a dados:

```
aws opensearchserverless create-access-policy \
    --name marketing \
    --type data \
    --policy "[{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"],\"Permission\":
[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",\"Resource\":
[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],\"Permission
\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]"
```

Para fornecer a política em um arquivo .json, use o formato `--policy file://my-policy.json`.

Os diretores incluídos na política agora podem usar as OpenSearch operações às quais receberam acesso.

Exibição de políticas de acesso a dados

Antes de criar uma coleção, talvez você queira pré-visualizar as políticas de acesso a dados existentes em sua conta para ver qual delas tem um padrão de recurso que corresponda ao nome da sua coleção. A ListAccessPoliciessolicitação a seguir lista todas as políticas de acesso a dados em sua conta:

```
aws opensearchserverless list-access-policies --type data
```

A solicitação retorna informações sobre todas as políticas de acesso a dados configuradas. Para visualizar as regras de padrões definidas em uma política específica, encontre as informações sobre políticas no conteúdo do elemento `accessPolicySummaries` na resposta. Observe o `name` final `type` desta política e use essas propriedades em uma GetAccessPolicysolicitação para receber uma resposta com os seguintes detalhes da política:

```
{
    "accessPolicyDetails": [
        {
```

```
            "type": "data",
            "name": "my-policy",
            "policyVersion": "MTY2NDA1NDE4MDg1OF8x",
            "description": "My policy",
            "policy": "[{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection/autopartsinventory\",\"collection/sales*\"],
\"Permission\":[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",
\"Resource\":[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],
\"Permission\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]",
            "createdDate": 1664054180858,
            "lastModifiedDate": 1664054180858
        }
    ]
}
```

É possível incluir filtros de recursos para limitar os resultados às políticas que contenham coleções ou índices específicos:

```
aws opensearchserverless list-access-policies --type data --resource
 "index/autopartsinventory/*"
```

Para ver detalhes sobre uma política específica, use o GetAccessPolicycomando.

Atualização de políticas de acesso a dados

Quando você atualiza uma política de acesso a dados, todas as coleções associadas são afetadas. Para atualizar uma política de acesso a dados no console OpenSearch sem servidor, escolha Controle de acesso a dados, selecione a política a ser modificada e escolha Editar. Faça suas alterações e escolha Salvar.

Para atualizar uma política de acesso a dados usando a API OpenSearch Serverless, envie uma `UpdateAccessPolicy` solicitação. É necessário incluir uma versão da política, que pode ser recuperada usando os comandos `ListAccessPolicies` ou `GetAccessPolicy`. A inclusão da versão mais recente da política garante que você não anule inadvertidamente uma alteração feita por outra pessoa.

A UpdateAccessPolicysolicitação a seguir atualiza uma política de acesso a dados com um novo documento JSON de política:

```
aws opensearchserverless update-access-policy \
```

```
    --name sales-inventory \
    --type data \
    --policy-version MTY2NDA1NDE4MDg10F8x \
    --policy file://my-new-policy.json
```

Pode haver alguns minutos de atraso entre a atualização da política e o momento em que as novas permissões são aplicadas.

Exclusão de políticas de acesso a dados

Quando você exclui uma política de acesso a dados, todas as coleções associadas perdem o acesso definido na política. Certifique-se de que seus usuários do IAM e do SAML tenham o acesso apropriado à coleção antes de excluir uma política. Para excluir uma política no console OpenSearch sem servidor, selecione a política e escolha Excluir.

Você também pode usar o DeleteAccessPolicycomando:

```
aws opensearchserverless delete-access-policy --name my-policy --type data
```

Acesso a dados entre contas

Embora não seja possível criar uma política de acesso a dados com identidade entre contas ou coleções entre contas, você pode configurar o acesso entre contas com a opção de assumir perfil. Por exemplo, se a *account-a* possuir uma coleção à qual a *account-b* precisa ter acesso, o usuário da *account-b* poderá assumir um perfil na *account-a*. O perfil deve ter as permissões do IAM `aoss:APIAccessAll` e `aoss:DashboardsAccessAll`, e estar incluído na política de acesso a dados na *account-a*.

Acesso ao plano de dados por meio de AWS PrivateLink

O Amazon OpenSearch Serverless oferece suporte a dois tipos de AWS PrivateLink conexões para operações de plano de controle e plano de dados. As operações do plano de controle incluem a criação e exclusão de coleções e o gerenciamento de políticas de acesso. As operações do plano de dados servem para indexar e consultar dados em uma coleção. Esta página aborda os endpoints VPC do plano de dados. Para obter informações sobre os AWS PrivateLink pontos finais do plano de controle, consultethe section called “Endpoint VPC do plano de controle”.

Você pode usar AWS PrivateLink para criar uma conexão privada entre sua VPC e o Amazon OpenSearch Serverless. Você pode acessar o OpenSearch Serverless como se estivesse em

sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão. Direct Connect As instâncias em sua VPC não precisam de endereços IP públicos para acessar OpenSearch o Serverless. Para obter mais informações sobre o acesso à rede VPC, consulte Padrões de conectividade de rede para Amazon OpenSearch Serverless.

Você estabelece essa conexão privada criando um endpoint de interface, alimentado pelo AWS PrivateLink. Criamos uma interface de rede de endpoint em cada sub-rede que você habilitar para o endpoint de interface. Essas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado ao Serverless. OpenSearch

Para saber mais, consulte Acessar os Serviços da AWS pelo AWS PrivateLink no Guia do AWS PrivateLink .

Tópicos

- Resolução de DNS dos endpoints de coleta
- VPCs e políticas de acesso à rede
- VPCs e políticas de endpoint
- Considerações
- Permissões obrigatórias
- Crie um endpoint de interface para Serverless OpenSearch
- Configuração de VPC compartilhada para Amazon Serverless OpenSearch

Resolução de DNS dos endpoints de coleta

Quando você cria um endpoint VPC do plano de dados por meio do console OpenSearch Serverless, o serviço cria uma nova zona hospedada Amazon Route 53 privada e a anexa à VPC. Essa zona hospedada privada consiste em um registro para resolver o registro DNS curinga para coleções OpenSearch sem servidor (`*.us-east-1.aoss.amazonaws.com`) para os endereços de interface usados para o endpoint. Você só precisa de um OpenSearch VPC endpoint sem servidor em uma VPC para acessar todas e quaisquer coleções e painéis em cada uma. Região da AWS Cada VPC com um endpoint para OpenSearch Serverless tem sua própria zona hospedada privada anexada.

O endpoint da interface OpenSearch sem servidor também cria um registro DNS curinga público do Route 53 para todas as coleções na região. O nome DNS é resolvido para os endereços IP OpenSearch públicos sem servidor. Clientes VPCs que não têm um endpoint OpenSearch VPC sem servidor ou clientes em redes públicas podem usar o resolvedor público do Route 53 e acessar as coleções e os painéis com esses endereços IP. O tipo de endereço IP (IPv4 IPv6, ou Dualstack) do

VPC endpoint é determinado com base nas sub-redes fornecidas quando você cria um endpoint de interface para Serverless. OpenSearch

> **Note**
>
> OpenSearch O Serverless cria uma zona hospedada privada `<region>.opensearch.amazonaws.com` (`) adicional do Amazon Route 53 para OpenSearch uma resolução de domínio de serviço. Você pode atualizar seu IPv4 VPC endpoint existente para o Dualstack usando o comando no. update-vpc-endpoint AWS CLI

O endereço do resolvedor DNS de uma determinada VPC é o segundo endereço IP do CIDR da VPC. Qualquer cliente na VPC precisa usar esse resolvedor para obter o endereço do endpoint da VPC para qualquer coleção. O resolvedor usa uma zona hospedada privada criada pelo OpenSearch Serverless. É suficiente usar esse resolvedor para todas as coleções em qualquer conta. Também é possível usar o resolvedor da VPC para alguns endpoints de coleção e o resolvedor público para outros, embora isso normalmente não seja necessário.

VPCs e políticas de acesso à rede

Para conceder permissão de rede OpenSearch APIs e painéis para suas coleções, você pode usar políticas de acesso à rede OpenSearch sem servidor. Você pode controlar esse acesso à rede a partir dos seus endpoints da VPC ou de Internet pública. Como sua política de rede controla apenas as permissões de tráfego, você também deve configurar uma política de acesso a dados que especifique a permissão para operar com os dados em uma coleção e seus índices. Pense em um endpoint OpenSearch VPC sem servidor como um ponto de acesso ao serviço, uma política de acesso à rede como o ponto de acesso em nível de rede para coleções e painéis e uma política de acesso a dados como o ponto de acesso para controle de acesso refinado para qualquer operação com dados na coleção.

Como você pode especificar vários VPC endpoints IDs em uma política de rede, recomendamos criar um VPC endpoint para cada VPC que precisa acessar uma coleção. Eles VPCs podem pertencer a AWS contas diferentes da conta que possui a política de rede e coleção OpenSearch Serverless. Não recomendamos que você crie uma solução de VPC-to-VPC emparelhamento ou outra solução de proxy entre duas contas para que a VPC de uma conta possa usar o VPC endpoint de outra conta. Isso é menos seguro e econômico do que cada VPC ter seu próprio endpoint. A primeira VPC não será facilmente visível para o administrador da outra VPC, que configurou o acesso ao endpoint dessa VPC na política de rede.

VPCs e políticas de endpoint

O Amazon OpenSearch Serverless oferece suporte a políticas de endpoint para. VPCs Uma política de endpoint é uma política baseada em recursos do IAM que você anexa a um VPC endpoint para controlar quais AWS entidades principais podem usar o endpoint para acessar seu serviço. AWS Para saber mais, consulte Controlar o acesso a endpoints de VPC usando políticas de endpoint.

Para usar uma política de endpoint, primeiro você deve criar um endpoint de interface. Você pode criar um endpoint de interface usando o console OpenSearch Serverless ou a API Serverless. OpenSearch Depois de criar seu endpoint de interface, você precisará adicionar a política de endpoint a esse endpoint. Para obter mais informações, consulte the section called “Como criar um endpoint de interface”.

> **Note**
>
> Você não pode definir uma política de endpoint diretamente no console OpenSearch de serviço.

Uma política de endpoint não substitui políticas baseadas em recursos, políticas de rede nem políticas de acesso a dados que você possa ter configurado. Para obter informações sobre como atualizar sua política de endpoint de VPC, consulte Controlar o acesso a endpoints da VPC usando políticas de endpoint.

Por padrão, uma política de endpoint concede acesso total ao seu endpoint de VPC.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Embora a política padrão de endpoint de VPC conceda acesso total ao endpoint, você pode configurar uma política de endpoint de VPC para permitir acesso a perfis e usuários específicos. Para fazer isso, veja o exemplo a seguir:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "123456789012",
                    "987654321098"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Você pode especificar uma coleção OpenSearch Serverless para ser incluída como um elemento condicional na sua política de VPC endpoint. Para fazer isso, veja o exemplo a seguir:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "coll-abc"
                    ]
                }
            }
        }
```

```
    ]
}
```

`aoss:CollectionId` é compatível.

```
Condition": {
         "StringEquals": {
               "aoss:CollectionId": "collection-id"
          }
}
```

Você pode usar identidades SAML em sua política de endpoint de VPC para determinar o acesso ao endpoint de VPC. Você deve usar um caractere curinga (*) na seção principal da sua política de endpoint de VPC. Para fazer isso, veja o exemplo a seguir:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "saml:cn": [
                        "saml/111122223333/idp123/group/football",
                        "saml/111122223333/idp123/group/soccer",
                        "saml/111122223333/idp123/group/cricket"
                    ]
                }
            }
        }
    ]
}
```

Além disso, você pode configurar sua política de endpoint para incluir uma política de entidade principal de SAML específica. Para isso, veja o seguinte:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/Department": [
                        "Engineering"]
                    }
                }
            }
        ]
    }
```

Para obter mais informações sobre o uso da autenticação SAML com o Amazon OpenSearch Serverless, consulte Autenticação SAML para Amazon Serverless. OpenSearch

Você também pode incluir usuários do IAM e do SAML na mesma política de endpoint de VPC. Para fazer isso, veja o exemplo a seguir:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
```

```
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "saml:cn": [
                        "saml/111122223333/idp123/group/football",
                        "saml/111122223333/idp123/group/soccer",
                        "saml/111122223333/idp123/group/cricket"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "111122223333"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Você também pode acessar uma coleção Amazon OpenSearch Serverless do Amazon EC2 por meio da interface VPC endpoints. Para obter mais informações, consulte Acesse uma coleção OpenSearch sem servidor do Amazon EC2 (via interface VPC endpoints).

Considerações

Antes de configurar um endpoint de interface para OpenSearch Serverless, considere o seguinte:

- OpenSearch O Serverless oferece suporte para fazer chamadas para todas as operações de OpenSearch API suportadas (não operações de API de configuração) por meio do endpoint da interface.
- Depois de criar um endpoint de interface para OpenSearch Serverless, você ainda precisa incluí-lo nas políticas de acesso à rede para que ele acesse coleções sem servidor.
- Por padrão, o acesso total ao OpenSearch Serverless é permitido por meio do endpoint da interface. Você pode associar um grupo de segurança às interfaces de rede do endpoint para controlar o tráfego para o OpenSearch Serverless por meio do endpoint da interface.
- Um único Conta da AWS pode ter no máximo 50 endpoints OpenSearch VPC sem servidor.

- Se você habilitar o acesso público à API ou aos painéis da sua coleção em uma política de rede, sua coleção poderá ser acessada por qualquer VPC e pela Internet pública.
- Se você estiver no local e fora da VPC, não poderá usar um resolvedor de DNS diretamente para a resolução do endpoint da VPC OpenSearch sem servidor. Se você precisar de acesso à VPN, a VPC precisará de um resolvedor de proxy DNS para ser usado por clientes externos. O Route 53 fornece uma opção de endpoint de entrada que você pode usar para resolver consultas ao DNS à VPC, originadas na rede no local (on-premises) ou em outra VPC.
- A zona hospedada privada que o OpenSearch Serverless cria e anexa à VPC é gerenciada pelo serviço, mas aparece nos seus Amazon Route 53 recursos e é cobrada na sua conta.
- Para outras considerações, consulte Considerações no Guia do AWS PrivateLink .

Permissões obrigatórias

O acesso à VPC para OpenSearch Serverless usa as seguintes permissões AWS Identity and Access Management (IAM). É possível especificar as condições do IAM para restringir os usuários a coleções específicas.

- `aoss:CreateVpcEndpoint`: criar um endpoint da VPC.
- `aoss:ListVpcEndpoints`: listar todos os endpoints da VPC.
- `aoss:BatchGetVpcEndpoint`: veja detalhes sobre um subconjunto de endpoints da VPC.
- `aoss:UpdateVpcEndpoint`: modificar um endpoint da VPC.
- `aoss:DeleteVpcEndpoint`: excluir um endpoint da VPC.

Além disso, as seguintes permissões do Amazon EC2 e do Route 53 são necessárias para criar um endpoint de VPC.

- `ec2:CreateTags`
- `ec2:CreateVpcEndpoint`
- `ec2:DeleteVpcEndPoints`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcEndpoints`
- `ec2:DescribeVpcs`

- `ec2:ModifyVpcEndPoint`
- `route53:AssociateVPCWithHostedZone`
- `route53:ChangeResourceRecordSets`
- `route53:CreateHostedZone`
- `route53:DeleteHostedZone`
- `route53:GetChange`
- `route53:GetHostedZone`
- `route53:ListHostedZonesByName`
- `route53:ListHostedZonesByVPC`
- `route53:ListResourceRecordSets`

Crie um endpoint de interface para Serverless OpenSearch

Você pode criar um endpoint de interface para OpenSearch Serverless usando o console ou a OpenSearch API Serverless.

Para criar um endpoint de interface para uma coleção sem OpenSearch servidor

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Expanda Sem Servidor no painel de navegação à esquerda e escolha Endpoints da VPC.
3. Escolha Criar endpoint da VPC.
4. Forneça um nome para o endpoint.
5. Para VPC, selecione a VPC a partir da qual você acessará o Serverless. OpenSearch
6. Em Sub-redes, selecione uma sub-rede a partir da qual você OpenSearch acessará o Serverless.
    - O endereço IP e o tipo de DNS do endpoint são baseados no tipo de sub-rede
        - Dualstack: se todas as sub-redes tiverem ambos os intervalos de endereços IPv4 IPv6
        - IPv6: Se todas as sub-redes forem IPv6 somente sub-redes
        - IPv4: Se todas as sub-redes tiverem intervalos de endereços IPv4
7. Em Grupos de segurança, selecione os grupos de segurança para associar às interfaces de rede do endpoint. Essa é uma etapa crítica em que você limita as portas, protocolos e fontes do tráfego de entrada que você está autorizando em seu endpoint. Certifique-se de que as regras

do grupo de segurança permitam que os recursos que usarão o VPC endpoint se comuniquem com o OpenSearch Serverless se comuniquem com a interface de rede do endpoint.

8. Escolha Criar endpoint.

Para criar um VPC endpoint usando a API OpenSearch Serverless, use o comando. `CreateVpcEndpoint`

> **Note**
>
> Depois de criar um endpoint, anote seu ID (por exemplo, `vpce-abc123def4EXAMPLE`. Para fornecer ao endpoint acesso às suas coleções, será necessário incluir esse ID em uma ou mais políticas de acesso à rede.

Depois de criar um endpoint da interface, você deverá fornecer a ele acesso às coleções por meio de políticas de acesso à rede. Para obter mais informações, consulte the section called “Acesso à rede”.

Configuração de VPC compartilhada para Amazon Serverless OpenSearch

Você pode usar a Amazon Virtual Private Cloud (VPC) para compartilhar sub-redes VPC com outras pessoas Contas da AWS em sua organização, bem como compartilhar a infraestrutura de rede, como uma VPN, entre vários recursos. Contas da AWS

Atualmente, o Amazon OpenSearch Serverless não oferece suporte à criação de uma AWS PrivateLink conexão em uma VPC compartilhada, a menos que você seja proprietário dessa VPC. AWS PrivateLink também não suporta o compartilhamento de conexões entre Contas da AWS.

No entanto, com base na arquitetura flexível e modular do OpenSearch Serverless, você ainda pode configurar uma VPC compartilhada. Isso ocorre porque a infraestrutura de rede OpenSearch sem servidor é separada da infraestrutura de coleção individual (OpenSearch Serviço). Portanto, você pode criar um AWS PrivateLink VPCe endpoint para uma conta onde está localizada uma VPC e, em seguida, usar VPCe uma ID na política de rede de outras contas para restringir o tráfego proveniente somente dessa VPC compartilhada.

Os procedimentos a seguir se referem a uma conta de proprietário e a uma conta de consumidor.

Uma conta de proprietário funciona como uma conta de rede comum, na qual você configura e compartilha uma VPC com outras contas. As contas de consumidor são aquelas contas que criam

e mantêm suas coleções OpenSearch sem servidor na VPC compartilhadas com elas pela conta do proprietário.

Pré-requisitos

Certifique-se de que os seguintes requisitos sejam atendidos antes de configurar a VPC compartilhada:

- A conta de proprietário em questão já deve ter configurada uma VPC, sub-redes, uma tabela de rotas e os outros recursos necessários na Amazon Virtual Private Cloud. Para saber mais, consulte o Manual do usuário da Amazon VPC.
- A conta de proprietário e as contas de consumidor em questão devem pertencer à mesma organização do AWS Organizations. Para obter mais informações, consulte o Guia do usuário do AWS Organizations.

Para configurar uma VPC compartilhada em uma conta de account/common rede do proprietário.

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Siga as etapas em the section called “Como criar um endpoint de interface”. Ao fazer isso, faça as seguintes seleções:
   - Selecione uma VPC e as sub-redes que são compartilhadas com as contas de consumidores da sua organização.
3. Depois de criar o endpoint, anote a VPCe ID gerada e forneça-a aos administradores que realizarão a tarefa de configuração nas contas de consumidores.

   VPCe IDs estão no formato`vpce-abc123def4EXAMPLE`.

Para configurar uma VPC compartilhada em uma conta de consumidor

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Use as informações em the section called “Gerenciar coleções” para criar uma coleção, se ainda não tiver uma.
3. Use as informações em the section called “Criação de políticas de rede (console)” para criar uma política de rede. Ao fazer isso, faça as seleções a seguir.

> **Note**
>
> Também é possível atualizar uma política de rede existente para essa finalidade.

a. Em Tipo de acesso, selecione VPC (recomendado).

b. Para acessar os endpoints VPC, escolha a VPCe ID fornecida pela conta do proprietário, no formato. `vpce-abc123def4EXAMPLE`

c. Na área Tipo de recurso, faça o seguinte:

- Selecione a caixa Habilitar acesso ao OpenSearch endpoint e, em seguida, selecione o nome da coleção ou o padrão de coleção a ser usado para habilitar o acesso a partir dessa VPC compartilhada.
- Selecione a caixa Habilitar acesso ao OpenSearch painel e, em seguida, selecione o nome da coleção ou o padrão de coleção a ser usado para habilitar o acesso a partir dessa VPC compartilhada.

4. Para uma nova política, escolha Criar. Para uma política existente, escolha Atualizar.

Controle o acesso ao avião por meio de AWS PrivateLink

O Amazon OpenSearch Serverless oferece suporte a dois tipos de AWS PrivateLink conexões para operações de plano de controle e plano de dados. As operações do plano de controle incluem a criação e exclusão de coleções e o gerenciamento de políticas de acesso. As operações do plano de dados servem para indexar e consultar dados em uma coleção. Esta página aborda o AWS PrivateLink ponto final do plano de controle. Para obter informações sobre os endpoints VPC do plano de dados, consulte. the section called “Endpoint VPC do plano de dados”

Criando um ponto AWS PrivateLink final do plano de controle

Você pode melhorar a postura de segurança da sua VPC OpenSearch configurando o Serverless para usar uma interface VPC endpoint. Os endpoints de interface são alimentados por AWS PrivateLink. Essa tecnologia permite que você acesse de forma privada o OpenSearch Serverless APIs sem um gateway de internet, dispositivo NAT, conexão VPN ou conexão Direct AWS Connect.

Para obter mais informações sobre AWS PrivateLink endpoints de VPC, consulte VPC endpoints no Guia do usuário da Amazon VPC.

Considerações

- Os endpoints de VPC são compatíveis somente na mesma região.
- Os endpoints da VPC oferecem suporte somente a DNS fornecidos pela Amazon por meio do Amazon Route 53.
- Os endpoints VPC oferecem suporte a políticas de endpoint para controlar o acesso a coleções, OpenSearch políticas e. VpcEndpoints
- OpenSearch O Serverless oferece suporte somente a endpoints de interface. Endpoints de gateway não são aceitos.

Criação do VPC endpoint

Para criar o endpoint VPC do plano de controle para o Amazon OpenSearch Serverless, use o procedimento Acesse um serviço AWS usando uma interface de VPC endpoint no Amazon VPC Developer Guide. Crie o seguinte endpoint:

- `com.amazonaws.region.aoss`

Para criar um endpoint VPC do plano de controle usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints.
3. Escolha Create Endpoint (Criar endpoint).
4. Em Service category (Categoria de serviço), escolha Serviços da AWS.
5. Para Serviços, escolha`com.amazonaws.region.aoss`. Por exemplo, `.com.amazonaws.us-east-1.aoss`
6. Em VPC, escolha a VPC na qual deseja criar o endpoint.
7. Em Subnets (Sub-redes), selecione as sub-redes (zonas de disponibilidade) nas quais você deseja criar as interfaces de rede do endpoint.
8. Em Grupos de segurança, escolha os grupos de segurança a serem associados às interfaces de rede do endpoint. Verifique se o HTTPS (porta 443) está permitido.
9. Em Política, escolha Acesso total para permitir todas as operações ou escolha Personalizado para anexar uma política personalizada.
10. Escolha Criar endpoint.

Criar uma política de endpoint

Você pode anexar uma política de endpoint ao seu VPC endpoint que controla o acesso ao Amazon Serverless. OpenSearch Essa política especifica as seguintes informações:

- A entidade principal que pode realizar ações.
- As ações que podem ser realizadas.
- Os recursos aos quais as ações podem ser aplicadas.

Para mais informações, consulte Controlar o acesso a serviços com VPC endpoints no Guia do usuário da Amazon VPC.

Example Política de VPC endpoint para servidores sem servidor OpenSearch

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "aoss:ListCollections",
        "aoss:BatchGetCollection"
      ],
      "Resource": "*"
    }
  ]
}
```

Example Política restritiva que permite somente operações de lista

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "aoss:ListCollections",
      "Resource": "*"
    }
  ]
```

```
}
```

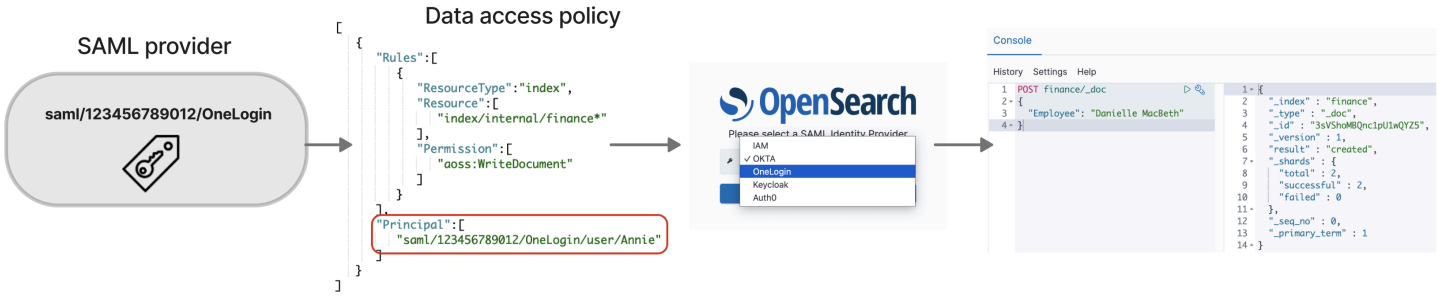
Autenticação SAML para Amazon Serverless OpenSearch

Com a autenticação SAML para Amazon OpenSearch Serverless, você pode usar seu provedor de identidade existente para oferecer login único (SSO) para os endpoints do Dashboards de coleções sem servidor. OpenSearch

A autenticação SAML permite que você use provedores de identidade terceirizados para entrar nos OpenSearch painéis para indexar e pesquisar dados. OpenSearch O Serverless oferece suporte a provedores que usam o padrão SAML 2.0, como IAM Identity Center, Okta, Keycloak, Active Directory Federation Services (AD FS) e Auth0. Você pode configurar o IAM Identity Center para sincronizar usuários e grupos de outras fontes de identidade OneLogin, como Okta e Microsoft Entra ID. Para ver uma lista das fontes de identidades compatíveis com o IAM Identity Center e as etapas para configurá-las, consulte os Tutoriais de introdução no Guia do usuário do IAM.

> **Note**
>
> A autenticação SAML serve apenas para acessar OpenSearch painéis por meio de um navegador da web. Usuários autenticados só podem fazer solicitações às operações da OpenSearch API por meio de ferramentas de desenvolvimento em OpenSearch painéis. Suas credenciais SAML não permitem que você faça solicitações HTTP diretas às operações da OpenSearch API.

Para configurar a autenticação SAML, primeiro é necessário configurar um provedor de identidade (IdP) SAML. Em seguida, você inclui um ou mais usuários desse IdP em uma política de acesso a dados. Essa política concede certas permissões aos and/or índices de coleções. Em seguida, um usuário pode entrar nos OpenSearch painéis e realizar as ações permitidas na política de acesso a dados.



Tópicos

- Considerações
- Permissões obrigatórias
- Criação de provedores de SAML (console)
- Acessando OpenSearch painéis
- Concessão de acesso de identidades do SAML a dados de coleções
- Criação de provedores de SAML (AWS CLI)
- Exibição de provedores de SAML
- Atualização de provedores de SAML
- Exclusão de provedores de SAML

Considerações

Considere o seguinte ao configurar a autenticação SAML:

- Não há suporte para solicitações assinadas e criptografadas.
- Não há suporte para declarações criptografadas.
- Não há suporte para autenticação e desconexão iniciadas pelo IdP.
- As Políticas de Controle de Serviços (SCP) não serão aplicáveis nem avaliadas no caso de identidades que não sejam do IAM (como SAML no OpenSearch Amazon Serverless e SAML e autorização básica de usuário interno para o Amazon Service). OpenSearch

Permissões obrigatórias

A autenticação SAML para OpenSearch Serverless usa as seguintes permissões AWS Identity and Access Management (IAM):

- `aoss:CreateSecurityConfig`: criar um provedor de SAML.
- `aoss:ListSecurityConfig`: listar todos os provedores de SAML na conta atual.
- `aoss:GetSecurityConfig`: exibir as informações do provedor de SAML.
- `aoss:UpdateSecurityConfig`: modificar uma determinada configuração do provedor de SAML, incluindo os metadados XML.
- `aoss:DeleteSecurityConfig`: excluir um provedor de SAML.

A seguinte política de acesso baseada em identidade permite que um usuário gerencie todas as configurações do IdP:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "aoss:CreateSecurityConfig",
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Observe que o elemento `Resource` deve ser um caractere curinga.

Criação de provedores de SAML (console)

Estas etapas explicam como criar provedores de SAML. Isso permite a autenticação SAML com autenticação iniciada pelo provedor de serviços (SP) para OpenSearch painéis. Não há suporte para autenticação iniciada pelo IdP.

Para habilitar a autenticação SAML para painéis OpenSearch

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, expanda Sem Servidor e escolha Autenticação SAML.
3. Escolha Adicionar provedor de SAML.
4. Forneça um nome e uma descrição para o provedor.

> **Note**
>
> O nome que você especificar pode ser acessado publicamente e aparecerá em um menu suspenso quando os usuários entrarem OpenSearch nos Painéis. Certifique-se de que o nome seja facilmente reconhecível e não revele informações confidenciais sobre seu provedor de identidade.

5. Em Configurar seu IdP, copie o URL do Assertion Consumer Service (ACS).
6. Use o URL do ACS que você acabou de copiar para configurar seu provedor de identidade. A terminologia e as etapas variam de acordo com o provedor. Consulte a documentação do seu provedor.

   No Okta, por exemplo, você cria uma “aplicação Web SAML 2.0” e especifica o URL do ACS como URL de login único, URL do destinatário e URL de destino. Para Auth0, você o especifica em Allowed Callback URLs.
7. Forneça a restrição de público se seu IdP possuir um campo para isso. A restrição de público é um valor dentro da declaração do SAML que especifica a quem a declaração se destina. Com o OpenSearch Serverless, você pode fazer o seguinte. Certifique-se de substituir o código *`content`* no exemplo a seguir pelo seu próprio Conta da AWS ID:

   1. Use a restrição de público padrão `:opensearch:`*`111122223333`*.
   2. (Opcional) configure uma restrição de público personalizada usando a AWS CLI. Para obter mais informações, consulte Criação de provedores de SAML (AWS CLI).

   O nome do campo de restrição de público varia de acordo com o provedor. Para o Okta, é URI do público, ID de entidade do SP. Para o IAM Identity Center, é Público de SAML da aplicação.
8. Se você estiver usando o IAM Identity Center, você também precisará especificar o seguinte mapeamento de atributos: `Subject=${user:name}`, com um formato `unspecified`.
9. Aopis você configurar o provedor de identidade, ele gera um arquivo de metadados IdP. Esse arquivo XML contém informações sobre o provedor, como um certificado TLS, endpoints de acesso único e o ID de entidade do provedor de identidade.

   Copie o texto no arquivo de metadados do IdP e cole-o no campo Fornecer metadados do seu IdP. Alternativamente, escolha Importar de arquivo XML e carregue o arquivo. O arquivo de metadados deve ser semelhante ao seguinte:

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>s
    </md:KeyDescriptor>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

10. Mantenha o campo Atributo de ID do usuário personalizado vazio para usar o elemento `NameID` da declaração do SAML para o nome do usuário. Se sua asserção não usar este elemento padrão e, em vez disso, incluir o nome de usuário como um atributo personalizado, especifique esse atributo aqui. Os atributos diferenciam maiúsculas de minúsculas. Só há suporte para um único atributo de usuário.

    O exemplo a seguir mostra um atributo de substituição para `NameID` na declaração do SAML:

    ```
    <saml2:Attribute Name="UserId" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
    format:basic">
      <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="xs:string">annie</saml2:AttributeValue>
    </saml2:Attribute>
    ```

11. (Opcional) Especifique um atributo personalizado no campo Atributo do grupo, como `role` ou `group`. Só há suporte para um único atributo de grupo. Não há atributo de grupo padrão.

Se você não especificar uma, suas políticas de acesso a dados só poderão conter entidades principais de usuários.

O exemplo a seguir mostra um atributo de grupo na declaração do SAML:

```
<saml2:Attribute Name="department"
 NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="xs:string">finance</saml2:AttributeValue>
</saml2:Attribute>
```

12. Por padrão, os OpenSearch painéis desconectam os usuários após 24 horas. Você pode configurar esse valor para qualquer número entre 1 e 12 horas (15 e 720 minutos) especificando o tempo limite dos OpenSearch painéis. Se você tentar definir um tempo limite igual ou inferior a 15 minutos, sua sessão será redefinida para uma hora.
13. Escolha Criar provedor de SAML.

Acessando OpenSearch painéis

Depois de configurar um provedor SAML, todos os usuários e grupos associados a esse provedor podem navegar até o endpoint do OpenSearch Dashboards. O URL do Dashboards tem o formato *collection-endpoint*/_dashboards/ para todas as coleções.

Se você tiver o SAML ativado, selecionar o link no Console de gerenciamento da AWS direcionará você para a página de seleção do IdP, na qual você poderá fazer login usando suas credenciais do SAML. Primeiro, use o menu suspenso para selecionar um provedor de identidade:



Em seguida, faça login usando suas credenciais do IdP.

Se você não tiver o SAML ativado, selecionar o link no Console de gerenciamento da AWS direcionará você a fazer login como usuário ou função do IAM, sem opção para SAML.

Concessão de acesso de identidades do SAML a dados de coleções

Depois de criar um provedor de SAML, você ainda precisa conceder aos usuários e grupos subjacentes acesso aos dados em suas coleções. Você concede acesso por meio de políticas de acesso a dados. Até que você forneça acesso aos usuários, eles não poderão ler, gravar ou excluir nenhum dado de suas coleções.

Para conceder acesso, crie uma política de acesso a dados e especifique seu and/or grupo de usuários do SAML IDs na `Principal` declaração:

```
[
```

```
   {
      "Rules":[
       ...
      ],
      "Principal":[
         "saml/987654321098/myprovider/user/Shaheen",
         "saml/987654321098/myprovider/group/finance"
      ]
   }
]
```

É possível conceder acesso a coleções, índices ou ambos. Se você quiser que usuários diferentes tenham permissões diferentes, crie várias regras. Para obter uma lista das permissões disponíveis, consulte Permissões de políticas com suporte. Para obter informações sobre como formatar uma política de acesso, consulte Sintaxe das políticas.

Criação de provedores de SAML (AWS CLI)

Para criar um provedor SAML usando a API OpenSearch Serverless, envie uma solicitação: CreateSecurityConfig

```
aws opensearchserverless create-security-config \
    --name myprovider \
    --type saml \
    --saml-options file://saml-auth0.json
```

Especifique `saml-options`, incluindo o XML de metadados, como um mapa de chave-valor em um arquivo .json. O XML de metadados deve ser codificado como uma string de escape JSON.

```
{
   "sessionTimeout": 70,
   "groupAttribute": "department",
   "userAttribute": "userid",
   "openSearchServerlessEntityId": "aws:opensearch:111122223333:app1",
   "metadata": "EntityDescriptor xmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ... ... ... IDPSSODescriptor\r\n\/EntityDescriptor"
}
```

> **Note**
>
> (Opcional) configure uma restrição de público personalizada usando a AWS CLI. Para obter mais informações, consulte Criação de provedores de SAML (AWS CLI).

Exibição de provedores de SAML

A ListSecurityConfigssolicitação a seguir lista todos os provedores de SAML em sua conta:

```
aws opensearchserverless list-security-configs --type saml
```

A solicitação retorna informações sobre todos os provedores de SAML existentes, incluindo os metadados completos do IdP que seu provedor de identidade gera:

```
{
   "securityConfigDetails": [
      {
         "configVersion": "MTY2NDA1MjY4NDQ5M18x",
         "createdDate": 1664054180858,
         "description": "Example SAML provider",
         "id": "saml/111122223333/myprovider",
         "lastModifiedDate": 1664054180858,
         "samlOptions": {
            "groupAttribute": "department",
            "metadata": "EntityDescriptorxmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ...... ...IDPSSODescriptor\r\n/EntityDescriptor",
            "sessionTimeout": 120,
            "openSearchServerlessEntityId": "aws:opensearch:111122223333:app1",
            "userAttribute": "userid"
         }
      }
   ]
}
```

Para exibir detalhes sobre um provedor específico, inclusive a `configVersion` para futuras atualizações, envie uma solicitação `GetSecurityConfig`.

Atualização de provedores de SAML

Para atualizar um provedor SAML usando o console OpenSearch Serverless, escolha a autenticação SAML, selecione seu provedor de identidade e escolha Editar. É possível modificar todos os campos, incluindo os metadados e os atributos personalizados.

Para atualizar um provedor por meio da API OpenSearch Serverless, envie uma UpdateSecurityConfigsolicitação e inclua o identificador da política a ser atualizada. Também é necessário incluir uma versão da configuração, que pode ser recuperada usando os comandos `ListSecurityConfigs` ou `GetSecurityConfig`. A inclusão da versão mais recente garante que você não anule inadvertidamente uma alteração feita por outra pessoa.

A solicitação a seguir atualiza as opções do SAML para um provedor:

```
aws opensearchserverless update-security-config \
    --id saml/123456789012/myprovider \
    --type saml \
    --saml-options file://saml-auth0.json \
    --config-version MTY2NDA1MjY4NDQ5M18x
```

Especifique suas opções de configuração do SAML como um mapa de chave-valor em um arquivo .json.

> **Important**
>
> As atualizações nas opções do SAML não são incrementais. Se você não especificar um valor para um parâmetro no objeto `SAMLOptions` ao fazer uma atualização, os valores existentes serão substituídos por valores vazios. Por exemplo, se a configuração atual contiver um valor para `userAttribute`, e você fizer uma atualização em seguida e não incluir esse valor, o valor será removido da configuração. Certifique-se de saber quais são os valores existentes antes de fazer uma atualização chamando a operação `GetSecurityConfig`.

Exclusão de provedores de SAML

Quando você exclui um provedor de SAML, quaisquer referências a usuários e grupos associados em suas políticas de acesso a dados não funcionam mais. Para evitar confusão, sugerimos que você remova todas as referências ao endpoint em suas políticas de acesso antes de excluir o endpoint.

Para excluir um provedor SAML usando o console OpenSearch sem servidor, escolha Autenticação, selecione o provedor e escolha Excluir.

Para excluir um provedor por meio da API OpenSearch Serverless, envie uma DeleteSecurityConfigsolicitação:

```
aws opensearchserverless delete-security-config --id saml/123456789012/myprovider
```

Validação de conformidade para Amazon OpenSearch Serverless

Auditores terceirizados avaliam a segurança e a conformidade do Amazon OpenSearch Serverless como parte de vários AWS programas de conformidade. Esses programas incluem SOC, PCI e HIPAA.

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a documentação AWS de segurança.

Marcação de coleções Amazon OpenSearch Serverless

As tags permitem que você atribua informações arbitrárias a uma coleção Amazon OpenSearch Serverless para que você possa categorizar e filtrar essas informações. Uma tag é um rótulo de metadados que você atribui ou AWS atribui a um AWS recurso.

Cada tag consiste em uma chave e um valor. Em tags atribuídas por você, você mesmo define a chave e o valor. Por exemplo, você pode definir a chave como `stage` e o valor de um atributo como `test`.

Com as tags, você pode identificar e organizar seus AWS recursos. Muitos AWS serviços oferecem suporte à marcação, então você pode atribuir a mesma tag a recursos de serviços diferentes para

indicar que os recursos estão relacionados. Por exemplo, você pode atribuir a mesma tag a uma coleção OpenSearch Serverless que você atribui a um domínio do Amazon OpenSearch Service.

No OpenSearch Serverless, o recurso principal é uma coleção. Você pode usar o console de OpenSearch serviço AWS CLI, as operações da API OpenSearch Serverless ou o AWS SDKs para adicionar, gerenciar e remover tags de uma coleção.

Permissões obrigatórias

OpenSearch O Serverless usa as seguintes permissões AWS Identity and Access Management Access Analyzer (IAM) para marcar coleções:

- `aoss:TagResource`
- `aoss:ListTagsForResource`
- `aoss:UntagResource`

Aplicar tags em coleções (console)

O console é a maneira mais simples de aplicar tags em uma coleção.

Para criar uma tag (console)

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Expanda Serverless (Sem Servidor) no painel de navegação à esquerda e escolha Collections (Coleções).
3. Selecione a coleção na qual você deseja aplicar tags e vá para guia Tags.
4. Escolha Gerenciar e Adicionar nova tag.
5. Insira uma chave de tag e um valor opcional.
6. Escolha Salvar.

Para excluir uma tag, siga as mesmas etapas e escolha Remover na página Gerenciar tags.

Para saber mais sobre como usar o console para trabalhar com tags, consulte Editor de tags no Guia de conceitos básicos do Console de Gerenciamento da AWS .

Aplicar tags em coleções (AWS CLI)

Para marcar uma coleção usando o AWS CLI, envie uma TagResourcesolicitação:

```
aws opensearchserverless tag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tags Key=service,Value=aoss Key=source,Value=logs
```

Visualize as tags existentes para uma coleção com o ListTagsForResourcecomando:

```
aws opensearchserverless list-tags-for-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
```

Remova as tags de uma coleção usando o UntagResourcecomando:

```
aws opensearchserverless untag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tag-keys service
```

Operações e plug-ins compatíveis no Amazon OpenSearch Serverless

O Amazon OpenSearch Serverless oferece suporte a uma variedade de OpenSearch plug-ins, bem como a um subconjunto das operações de API de indexação, pesquisa e metadados disponíveis em. OpenSearch É possível incluir as permissões na coluna à esquerda da tabela nas políticas de acesso a dados para limitar o acesso a determinadas operações.

Tópicos

- Operações e permissões de OpenSearch API suportadas
- OpenSearch Plugins compatíveis

Operações e permissões de OpenSearch API suportadas

A tabela a seguir lista as operações de API suportadas pelo OpenSearch Serverless, junto com as permissões correspondentes da política de acesso a dados:

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:CreateIndex`	PUT <index>	Criar índices. Para saber mais, consulte Criar índices. **Note** Essa permissão também se aplica à criação de índices com os dados de amostra nos OpenSearch painéis.
`aoss:DescribeIndex`	• GET <index> • GET <index>/_mapping • GET <index>/_mappings • GET <index>/_setting • GET <index>/_setting/<setting> • GET <index>/_settings • GET <index>/_settings/<setting> • GET _cat/indices • GET _mapping • GET _mappings • GET _resolve/index/<index> • CABEÇALHO <index>	Descreve índices. Para saber mais, consulte os seguintes recursos: • Obter índice • Obter um mapeamento • Obter configurações • O índice existe • Índices CAT (a resposta não inclui os `health` ou `status`.)
`aoss:WriteDocument`	• EXCLUIR <index>/_doc/ <id> • POST <index>/_bulk	Escreve e atualiza documentos. Para saber mais, consulte os seguintes recursos:

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
	• POST <index>/_create/<id> (somente para tipos de coleção de pesquisa) • POST <index>/_doc • POST <index>/_update/<id> (somente para tipos de coleção de pesquisa) • POST _bulk • PUT <index>/_create/<id> (somente para tipos de coleção de pesquisa) • PUT <index>/_doc/<id> (somente para tipos de coleção de pesquisa)	• Em massa • Dados de índice **Note** Algumas operações só são permitidas para coleções do tipo SEARCH. Para saber mais, consulte the section called “Escolha de um tipo de coleção”.

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:ReadDocument`	• GET <index>/_analyz • GET <index>/_doc/<id> • GET <index>/_explain/<id> • GET <index>/_mget • GET <index>/_source/<id> • GET <index>/_count • GET <index>/_field_caps • GET <index>/_msearch • GET <index>/_rank_eval • GET <index>/_search • GET <index>/_validate/<query> • GET _analyze • GET _field_caps • GET _mget • GET _search • GET /_search/point_in_time/_all • HEAD <index>/_doc/<id> • HEAD <index>/_source/<id> • POST /_plugins/_sql • POST /_plugins/_ppl • POST /_plugins/_sql/_explain • POST /_plugins/_ppl/_explain • POST /_plugins/_ppl/_close • POST <index>/_analyze • POST /_search/point_in_time • POST <index>/_explain/<id> • POST <index>/_count • POST <index>/_field_caps	Lê documentos. Para saber mais, consulte os seguintes recursos: • Realizar análise de texto • Obter documento • Contagem • Consultar DSL • Avaliação de classific ação • API de análise • Explicar • Ponto no tempo • SQL e PPL

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
	• POST <index>/_rank_eval • POST <index>/_search • POST _analyze • POST _field_caps • POST _search • DELETE /_search/point_in_time/_all • DELETE /_search/point_in_time	
`aoss:DeleteIndex`	DELETE <target>	Excluir índices. Para saber mais, consulte Excluir índice.
`aoss:UpdateIndex`	• POST _mapping • POST <index>/_mapping/ • POST <index>/_mappings/ • POST <index>/_setting • POST <index>/_settings • POST _setting • POST _settings • PUT _mapping • PUT <index>/_mapping • PUT <index>/_mappings/ • PUT <index>/_setting • PUT <index>/_settings • PUT _setting • PUT _settings	Atualizar configurações de índice. Para saber mais, consulte os seguintes recursos: • Mapeamento • Atualizar configurações

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:CreateCollectionItems`	• POST _aliases • POST /_plugins/_flow_framework/workflow • * POST /_plugins/_flow_framework/workflow/<workflow_id>/_provision • PUT _ingest/pipeline/<pipeline-id> • PUT _search/pipeline/<pipeline-id>	• Crie aliases de índice, pipelines e modelos. Para saber mais, consulte Criar aliases. • * Modelos de provisionamento ou reprovisionamento. Os serviços ML Commons Client e OpenSearch Serverless gerenciam políticas dependentes.

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:DescribeCollectionItems`	• GET <index>/_alias/<alias> • GET _alias • GET _alias/<alias> • GET _cat/aliases • GET _cat/templates • GET _cat/templates/<template_name> • GET _component_template • GET _component_template/<component-template> • GET _index_template • GET _index_template/<index-template> • GET _ingest/pipeline/<pipeline-id> • GET _ingest/pipeline/_simulate • GET /_plugins/_flow_framework/workflow/<workflow-id> • GET /_plugins/_flow_framework/workflow/_search • GET /_plugins/_flow_framework/workflow/<workflow-id>/_status • OBTENHA /_plugins/_flow_ /_search framework/workflow/state • GET /_plugins/_flow_framework/workflow/_steps • GET /_plugins/_flow_framework/workflow/_step?workflow_step=<step_name> • GET _search/pipeline/<pipeline-id>	Descreve como trabalhar com aliases, modelos de índice e estrutura, e pipelines. Para saber mais, consulte os seguintes recursos: • Gerenciar aliases • Modelos de índices

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
	• HEAD _alias/<alias> • HEAD _component_template/<component-template> • HEAD _index_template/<name> • HEAD <index>/_alias/<alias> • POST _ingest/pipeline/_simulate • POST /_plugins/_flow_framework/workflow/_search • POST /_plugins/_flow_ /_search framework/workflow/state	
`aoss:UpdateCollectionItems`	• POST <index>/_alias/<alias> • POST <index>/_aliases/<alias> • POST _component_template/<component-template> • POST _index_template/<index-template> • * POST /_plugins/_flow_framework/workflow/<workflow_id>/_deprovision • PUT <index>/_alias/<alias> • PUT <index>/_aliases/<alias> • PUT _component_template/<component-template> • PUT _index_template/<index-template> • PUT /_plugins/_flow_framework/workflow/<workflow_id>	Atualize aliases, modelos de índice e modelos de estrutura. Para saber mais, consulte os seguintes recursos: • Aliases de índice • Modelos de índices * A API para desprovisionar modelos. Os serviços ML Commons Client e OpenSearch Serverless gerenciam políticas dependentes.

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:DeleteCollectionItems`	• DELETE <index>/_alias/<alias> • DELETE _component_template/<component-template> • DELETE _index_template/<index-template> • DELETE <index>/_aliases/<alias> • DELETE _search/pipeline/<pipeline-id> • DELETE _ingest/pipeline/<pipeline-id> • DELETE /_plugins/_flow_framework/workflow/<workflow_id>	Exclua aliases, modelos de índice e estrutura, e pipelines. Para saber mais, consulte os seguintes recursos: • Excluir aliases • Excluir um modelo

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:DescribeMLResource`	• GET /_plugins/_ml/models/<model_id> • GET /_plugins_ml/models/_search • GET /_plugins/_ml/model_groups/<model_group_id> • GET /_plugins/_ml/model_groups/_search • GET /_plugins/_ml/connectors/<connector_id> • GET /_plugins/_ml/connectors/_search • OBTENHA /_plugins/_/ml/profile/tasks<task_id> • POST /_plugins/_ml/models/_search • POST /_plugins/_ml/model_groups/_search • POST /_plugins/_ml/connectors/_search	Descreve o GET e APIs a pesquisa para recuperar informações sobre modelos e conectores.
`aoss:CreateMLResource`	• POST /_plugins/_ml/models/_register • POST /_plugins/_ml/model_groups/_register • POST /_plugins/_ml/connectors/_create	Fornece permissão para criar recursos de ML.

Permissão da política de acesso a dados	OpenSearch Operações de API	Descrição e advertências
`aoss:UpdateMLResource`	• PUT /_plugins/_ml/models/<model_id> • POST /_plugins/_ml/models/<model_id>/_deploy • POST /_plugins/_ml/models/<model_id>/_undeploy • PUT /_plugins/_ml/model_groups/<model_group_id> • PUT /_plugins/_ml/connectors/<connector_id>	Concede permissão para atualizar os recursos de ML existentes.
`aoss:DeleteMLResource`	• DELETE /_plugins/_ml/models/<model_id> • DELETE _plugins/_ml/model_groups/<model_group_id> • DELETE /_plugins/_ml/connectors/<connector_id> • DELETE /_plugins/_ml/tasks/<task_id>	Fornece permissão para excluir recursos de ML.
`aoss:ExecuteMLResource`	• POST /_plugins/_ml/models/<model_id>/_predict	Fornece permissão para executar modelos.

OpenSearch Plugins compatíveis

OpenSearch As coleções sem servidor vêm pré-embaladas com os seguintes plug-ins da comunidade. OpenSearch O Serverless (Sem Servidor) implanta e gerencia automaticamente os plug-ins para você.

Plug-ins de análise

- ICU Analysis
- Japanese (kuromoji) Analysis

- Análise de coreano (Nori)
- Phonetic Analysis
- Smart Chinese Analysis
- Stempel Polish Analysis
- Ukrainian Analysis

Plug-ins do Mapper

- Mapper Size
- Mapper Murmur3
- Texto anotado do Mapper

Plug-ins de script

- Painless
- Expressão
- Mustache

Além disso, o OpenSearch Serverless inclui todos os plug-ins fornecidos como módulos.

Monitorando o Amazon OpenSearch Serverless

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho do Amazon OpenSearch Serverless e de suas outras AWS soluções. AWS fornece as seguintes ferramentas de monitoramento para monitorar o OpenSearch Serverless, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- A Amazon CloudWatch monitora seus AWS recursos e os aplicativos nos quais você executa AWS em tempo real. Você pode coletar e rastrear métricas, criar painéis personalizados e definir alarmes que o notificam ou que realizam ações quando uma métrica especificada atinge um limite definido.

  Por exemplo, você pode CloudWatch rastrear o uso da CPU ou outras métricas de suas instâncias do Amazon EC2 e iniciar automaticamente novas instâncias quando necessário. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.

- O AWS CloudTrail captura chamadas de API e eventos relacionados feitos por sua conta da Conta da AWS ou em nome dela. Ele disponibiliza os arquivos de log para um bucket do Amazon S3 especificado por você. Você pode identificar quais usuários e contas ligaram AWS, o endereço IP de origem a partir do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte o Guia do usuário do AWS CloudTrail.
- A Amazon EventBridge fornece um fluxo quase em tempo real de eventos do sistema que descrevem as mudanças em seus domínios OpenSearch de serviço. Você pode criar regras que observem determinados eventos e acionem ações automatizadas em outros Serviços da AWS quando esses eventos ocorrerem. Para obter mais informações, consulte o Guia EventBridge do usuário da Amazon.

Monitoramento OpenSearch sem servidor com a Amazon CloudWatch

Você pode monitorar o Amazon OpenSearch Serverless usando CloudWatch, que coleta dados brutos e os processa em métricas legíveis, quase em tempo real. Essas estatísticas são mantidas por 15 meses, de maneira que você possa acessar informações históricas e ter uma perspectiva melhor de como o aplicativo web ou o serviço está se saindo.

Você também pode definir alarmes que observam determinados limites e enviam notificações ou realizam ações quando esses limites são atingidos. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.

OpenSearch O Serverless relata as seguintes métricas no namespace. `AWS/AOSS`

Métrica	Description
`ActiveCollection`	Indica se uma coleção está ativa. Um valor de 1 significa que a coleção está em um estado `ACTIVE`. Esse valor é emitido após a criação com êxito de uma coleção, e permanece como 1 até que você exclua a coleção. A métrica não pode ter um valor de 0. Estatísticas relevantes: máx. Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos

Métrica	Description
`DeletedDocuments`	O número total de documentos excluídos. Estatísticas relevantes: média, soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos
`IndexingOCU`	O número de unidades de OpenSearch computação (OCUs) usadas para ingerir dados de coleta. Esta métrica aplica-se no nível da conta. Representa o uso somente para coleções que não fazem parte de nenhum grupo de coleções. Estatísticas relevantes: soma Dimensões: `ClientId` Frequência: 60 segundos
`IndexingOCU`	O número de unidades de OpenSearch computação (OCUs) usadas para ingerir dados de coleta. Essa métrica se aplica ao nível do grupo de coleta. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionGroupId` , `CollectionGroupName` Frequência: 60 segundos

Métrica	Description
`IngestionDataRate`	A taxa de indexação em GiB por segundo para uma coleção ou índice. Esta métrica aplica-se apenas às solicitações de indexação em massa. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos
`IngestionDocumentErrors`	O número total de erros de documentos durante a ingestão de uma coleção ou índice. Depois de uma solicitação de indexação em massa com êxito, os gravadores processam a solicitação e emitem erros para todos os documentos que falharam na solicitação. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos
`IngestionDocumentRate`	A taxa por segundo na qual os documentos estão sendo ingeridos em uma coleção ou índice. Esta métrica aplica-se apenas às solicitações de indexação em massa. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos

Métrica	Description
`IngestionRequestErrors`	O número total de erros de solicitação de indexação em massa para uma coleção. OpenSearch O Serverless emite essa métrica quando uma solicitação de indexação em massa falha por qualquer motivo, como um problema de autenticação ou disponibilidade. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos
`IngestionRequestLatency`	A latência, em segundos, para operações de gravação em massa em uma coleção. Estatísticas relevantes: mínimo, máximo, média Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos
`IngestionRequestRate`	O número total de operações de gravação em massa recebidas por uma coleção. Estatísticas relevantes: mínimo, máximo, média Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos

Métrica	Description
`IngestionRequestSuccess`	O número total de operações de indexação para uma coleção. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos
`SearchableDocuments`	O número total de documentos pesquisáveis em uma coleção ou no índice. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos
`SearchRequestErrors`	O número total de erros de consulta por minuto para uma coleção. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos

Métrica	Description
`SearchRequestLatency`	O tempo médio necessário, em milissegundos, para que uma operação de pesquisa seja concluída em uma coleção. Estatísticas relevantes: mínimo, máximo, média Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos
`Search0CU`	O número de unidades de OpenSearch computação (OCUs) usadas para pesquisar dados de coleta. Esta métrica aplica-se no nível da conta. Representa o uso somente para coleções que não fazem parte de nenhum grupo de coleções. Estatísticas relevantes: soma Dimensões: `ClientId` Frequência: 60 segundos
`Search0CU`	O número de unidades de OpenSearch computação (OCUs) usadas para pesquisar dados de coleta. Essa métrica se aplica ao nível do grupo de coleta. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionGroupId` , `CollectionGroupName` Frequência: 60 segundos

Métrica	Description
`SearchRequestRate`	O número total de solicitações de pesquisa por minuto para uma coleção. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos
`StorageUsedInS3`	A quantidade, em bytes, do armazenamento do Amazon S3 usado. OpenSearch O Serverless armazena dados indexados no Amazon S3. Você deve selecionar o período em um minuto para receber um valor preciso. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Frequência: 60 segundos
`VectorIndexBuildAccelerationOCU`	O número de unidades de OpenSearch computação (OCUs) usadas para acelerar a indexação vetorial. Essa métrica se aplica no nível da coleção. Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` Frequência: 60 segundos

Métrica	Description
`2xx, 3xx, 4xx, 5xx`	O número de solicitações para a coleção que resultaram no determinado código de resposta HTTP (2xx, 3xx, 4xx, 5xx). Estatísticas relevantes: soma Dimensões: `ClientId`, `CollectionId` , `CollectionName` Frequência: 60 segundos

Registrando OpenSearch chamadas de API sem servidor usando AWS CloudTrail

O Amazon OpenSearch Serverless é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no Serverless.

CloudTrail captura todas as chamadas de API para OpenSearch Serverless como eventos. As chamadas capturadas incluem chamadas da seção Serverless do console de OpenSearch serviço e chamadas de código para as operações da API OpenSearch Serverless.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para OpenSearch Serverless. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita à OpenSearch Serverless, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o Guia AWS CloudTrail do usuário.

OpenSearch Informações sem servidor em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre no OpenSearch Serverless, essa atividade é registrada em um CloudTrail evento junto com outros eventos de AWS serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos

recentes no seu Conta da AWS. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em seu Conta da AWS, incluindo eventos para OpenSearch Serverless, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as Regiões da AWS.

A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para saber mais, consulte:

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações OpenSearch sem servidor são registradas CloudTrail e documentadas na referência da API sem OpenSearch servidor. Por exemplo, chamadas para as `DeleteCollection` ações `CreateCollectionListCollections`, e geram entradas nos arquivos de CloudTrail log.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário root ou AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte Elemento userIdentity do CloudTrail .

OpenSearch Eventos de dados sem servidor em CloudTrail

Os eventos de dados fornecem informações sobre as operações de recursos realizadas em ou em um recurso (por exemplo, pesquisa ou indexação em uma coleção OpenSearch sem servidor).

Também são conhecidas como operações de plano de dados. Os eventos de dados costumam ser atividades de alto volume. Por padrão, CloudTrail não registra eventos de dados. O histórico de CloudTrail eventos não registra eventos de dados.

Há cobranças adicionais para eventos de dados. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços.

Você pode registrar eventos de dados para os tipos de `AWS::AOSS::Collection` recursos usando o CloudTrail console ou AWS CLI as operações CloudTrail da API. Para saber mais sobre como registrar eventos de dados em log, consulte Registrar eventos de dados com o Console de gerenciamento da AWS e Registrar eventos de dados com a AWS Command Line Interface no Guia do usuário do AWS CloudTrail .

É possível configurar seletores de eventos avançados para filtrar os campos `eventName`, `readOnly` e `resources.ARN` para registrar em log somente os eventos que são importantes para você. Para saber mais sobre esses campos, consulte AdvancedFieldSelector na Referência de API do AWS CloudTrail .

Entendendo as entradas de eventos de dados OpenSearch sem servidor

No seguinte exemplo:

- O campo `requestParameters` contém detalhes sobre a chamada de API feita para a coleção. Ele inclui o caminho básico da solicitação (sem parâmetros de consulta).
- O campo `responseElements` inclui um código de status que indica o resultado da solicitação ao modificar recursos. Esse código de status ajuda você a saber se suas alterações foram processadas com sucesso ou se precisam de atenção.
- OpenSearch O Serverless registra eventos CloudTrail de dados somente para solicitações que concluíram com êxito a autenticação do IAM.

```
        "accessKeyId": "access-key",
        "userName": "",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AROA123456789EXAMPLE",
            "arn": "arn:aws:iam::111122223333:role/Admin",
            "accountId": "111122223333",
            "userName": "Admin"
          },
          "attributes": {
            "creationDate": "2025-08-15T22:57:38Z",
            "mfaAuthenticated": "false"
          },
          "sourceIdentity": "",
          "ec2RoleDelivery": "",
          "assumedRoot": ""
        },
        "identityProvider": "",
        "credentialId": ""
      },
      "eventTime": "2025-08-15T22:58:00Z",
      "eventSource": "aoss.amazonaws.com",
      "eventName": "Search",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "AWS Internal",
      "userAgent": "python-requests/2.32.3",
      "requestParameters": {
        "pathPrefix": "/_search"
      },
      "responseElements": null,
      "requestID": "2cfee788-EXAM-PLE1-8617-4018cEXAMPLE",
      "eventID": "48d43617-EXAM-PLE1-9d9c-f7EXAMPLE",
      "readOnly": true,
      "resources": [
        {
          "type": "AWS::AOSS::Collection",
          "ARN": "arn:aws:aoss:us-east-1:111122223333:collection/aab9texampletu45xh77"
        }
      ],
      "eventType": "AwsApiCall",
      "managementEvent": false,
      "recipientAccountId": "111122223333",
      "eventCategory": "Data"
    }
  ]
}
```

Compreendendo as entradas de eventos de gerenciamento OpenSearch sem servidor

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log.

Um evento representa uma solicitação única de qualquer fonte. Ele inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir exibe uma entrada de CloudTrail registro que demonstra a `CreateCollection` ação.


```
{
   "eventVersion":"1.08",
   "userIdentity":{
      "type":"AssumedRole",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/test-user",
      "accountId":"123456789012",
      "accessKeyId":"access-key",
      "sessionContext":{
         "sessionIssuer":{
            "type":"Role",
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:role/Admin",
            "accountId":"123456789012",
            "userName":"Admin"
         },
         "webIdFederationData":{

         },
         "attributes":{
            "creationDate":"2022-04-08T14:11:34Z",
            "mfaAuthenticated":"false"
         }
      }
   },
   "eventTime":"2022-04-08T14:11:49Z",
   "eventSource":"aoss.amazonaws.com",
   "eventName":"CreateCollection",
```

```
   "awsRegion":"us-east-1",
   "sourceIPAddress":"AWS Internal",
   "userAgent":"aws-cli/2.1.30 Python/3.8.8 Linux/5.4.176-103.347.amzn2int.x86_64 exe/
x86_64.amzn.2 prompt/off command/aoss.create-collection",
   "errorCode":"HttpFailureException",
   "errorMessage":"An unknown error occurred",
   "requestParameters":{
      "accountId":"123456789012",
      "name":"test-collection",
      "description":"A sample collection",
      "clientToken":"d3a227d2-a2a7-49a6-8fb2-e5c8303c0718"
   },
   "responseElements": null,
   "requestID":"12345678-1234-1234-1234-987654321098",
   "eventID":"12345678-1234-1234-1234-987654321098",
   "readOnly":false,
   "eventType":"AwsApiCall",
   "managementEvent":true,
   "recipientAccountId":"123456789012",
   "eventCategory":"Management",
   "tlsDetails":{
      "clientProvidedHostHeader":"user.aoss-sample.us-east-1.amazonaws.com"
   }
}
```


Monitoramento de eventos OpenSearch sem servidor usando a Amazon EventBridge

O Amazon OpenSearch Service se integra EventBridge à Amazon para notificá-lo sobre determinados eventos que afetam seus domínios. Os eventos dos AWS serviços são entregues quase EventBridge em tempo real. Os mesmos eventos também são enviados para a Amazon CloudWatch Events, a antecessora da Amazon EventBridge. Você pode escrever regras para indicar quais eventos são do seu interesse, e as ações automatizadas a serem tomadas quando um evento corresponder à regra. Exemplos de ações que você pode ativar automaticamente incluem o seguinte:

- Invocando uma função AWS Lambda
- Invocação de um Run Command do Amazon EC2
- Transmitir o evento Amazon Kinesis Data Streams
- Ativando uma máquina de estado AWS Step Functions
- Notificar um tópico do Amazon SNS ou uma fila do Amazon SQS

Para obter mais informações, consulte Comece a usar a Amazon EventBridge no Guia EventBridge do usuário da Amazon.

Configuração de notificações

Você pode usar as Notificações AWS do usuário para receber notificações quando ocorrer um evento OpenSearch sem servidor. Um evento é um indicador de uma mudança no ambiente OpenSearch sem servidor, como quando você atinge o limite máximo de uso da OCU. Amazon EventBridge recebe o evento e encaminha uma notificação para a Central de Console de gerenciamento da AWS Notificações e os canais de entrega escolhidos. Você recebe uma notificação quando um evento corresponde a uma regra especificada.

OpenSearch Eventos de unidades de computação (OCU)

OpenSearch O Serverless envia eventos para EventBridge quando um dos seguintes eventos relacionados à OCU ocorrer.

OCU usage approaching maximum limit (Uso de OCU próximo do limite máximo)

OpenSearch O Serverless envia esse evento quando o uso da OCU de pesquisa ou indexação atinge 75% do seu limite de capacidade. O uso de OCU é calculado com base no limite de capacidade configurado e no consumo atual de OCU.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
```

```
}
```

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
```

OCU usage reached maximum limit (O uso de OCU atingiu o limite máximo)

OpenSearch O Serverless envia esse evento quando o uso da OCU de pesquisa ou indexação atinge 100% do seu limite de capacidade. O uso de OCU é calculado com base no limite de capacidade configurado e no consumo atual de OCU.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage has reached the configured maximum limit."
```

```
  }
}
```

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage has reached the configured maximum limit."
  }
}
```

Criação e gerenciamento de domínios OpenSearch do Amazon Service

Este capítulo descreve como criar e gerenciar domínios do Amazon OpenSearch Service. Um domínio é o equivalente AWS provisionado de um cluster de código aberto. OpenSearch Ao criar um um domínio, você especifica as configurações, os tipos de instâncias, as contagens de instâncias e a alocação de armazenamento. Para obter mais informações sobre clusters de código aberto, consulte Como criar um cluster na OpenSearch documentação.

Diferentemente das breves instruções apresentadas no Tutorial de introdução, este capítulo descreve todas as opções e fornece informações de referência relevantes. Você pode concluir cada procedimento usando as instruções do console OpenSearch de serviço, do AWS Command Line Interface (AWS CLI) ou do AWS SDKs.

Criação OpenSearch de domínios de serviço

Esta seção descreve como criar domínios OpenSearch de serviço usando o console OpenSearch de serviços ou usando o AWS CLI com o `create-domain` comando.

Criação OpenSearch de domínios de serviço (console)

Use o procedimento a seguir para criar um domínio de OpenSearch serviço usando o console.

Para criar um domínio OpenSearch de serviço (console)

1. Acesse https://aws.amazon.come escolha Entrar no console.
2. Em Analytics, escolha Amazon OpenSearch Service.
3. Escolha Criar domínio.
4. Em Nome de domínio, insira um nome de domínio O nome deve atender aos seguintes critérios:
   - Exclusivo para sua conta e Região da AWS
   - Iniciar com letra minúscula.
   - Conter de 3 a 28 caracteres.
   - Conter apenas letras minúsculas a-z, números de 0-9 e hífen (-).
5. Como método de criação de domínio, escolha Criação padrão.
6. Em Modelos, escolha a opção que melhor corresponde à finalidade do seu domínio:

- Domínios de produção para workload que precisam de alta disponibilidade e desempenho. Os domínios usam Multi-AZ (com ou sem standby) e nós principais dedicados para uma maior disponibilidade.
- Dev/test para desenvolvimento ou teste. Esses domínios podem usar Multi-AZ (com ou sem modo de espera) ou uma única zona de disponibilidade.

> **Important**
>
> Diferentes tipos de implantação apresentam diferentes opções em páginas subsequentes. Essas etapas incluem todas as opções.

7. Para Opções de implantação, escolha Domínio com modo de espera para configurar um domínio 3-AZ, com os nós em uma das zonas reservados como modo de espera. Essa opção aplica várias práticas recomendadas, como contagem especificada de nós de dados, contagem de nós principais, tipo de instância, contagem de réplicas e configurações de atualização de software.
8. Em Versão, escolha a versão OpenSearch ou o Elasticsearch OSS legado a ser usado. Recomendamos que você escolha a versão mais recente do OpenSearch. Para obter mais informações, consulte the section called “Versões compatíveis do Elasticsearch e OpenSearch”.

   (Opcional) Se você escolher uma OpenSearch versão para seu domínio, selecione Ativar modo de compatibilidade para OpenSearch reportar sua versão como 7.10, o que permite que determinados clientes e plug-ins do Elasticsearch OSS que verificam a versão antes de se conectar continuem trabalhando com o serviço.
9. Em Tipo de instância escolha um tipo de instância para os nós de dados. Para saber mais, consulte the section called “Tipos de instâncias compatíveis”.

   > **Note**
   >
   > Nem todas as zonas de disponibilidade são compatíveis com todos os tipos de instância. Se você escolher Multi-AZ com ou sem standby, é recomendável selecionar tipos de instância da geração atual, como R5 ou I3.
10. Em Número de nós, selecione o número de nós de dados.

    Para valores máximos, consulte Cotas OpenSearch de domínio e instância do serviço. Os clusters de nó único são excelentes para desenvolvimento e testes, mas não devem ser

usados para workloads de produção. Para obter mais orientações, consulte the section called “Dimensionamento de domínios” e the section called “Configuração de um domínio Multi-AZ”.

> **Note**
>
> (Opcional) Os nós coordenadores dedicados oferecem suporte a todas OpenSearch as versões e ElasticSearch versões 6.8 a 7.10. Os nós coordenadores dedicados estão disponíveis para uso com domínios que têm um gerenciador de cluster dedicado ativado. Para habilitar nós coordenadores dedicados, você selecionará o tipo e a contagem de instâncias. Como prática recomendada, você deve manter a família de instâncias do seu nó coordenador dedicado igual aos seus nós de dados (instâncias baseadas em Intel ou Graviton).

11. Em Tipo de armazenamento, selecione Amazon EBS. Os tipos de volume disponíveis na lista dependem do tipo de instância escolhido. Para obter orientações sobre a criação de domínios especialmente grandes, consulte the section called “Escala de petabytes”.
12. Em armazenamento EBS, configure as opções a seguir. A depender do tipo de volume escolhido, algumas configurações poderão não aparecer.

Configuração	Description
Tipo de volume do EBS	Escolha entre Finalidade geral (SSD): gp3 e Finalidade geral (SSD): gp2 ou IOPS provisionadas (SSD) e Magnético (padrão) da geração anterior.
Tamanho de armazenamento do EBS por nó	Insira o tamanho do volume do EBS que você deseja anexar a cada nó de dados. EBS volume size é por nó. Você pode calcular o tamanho total do cluster para o domínio OpenSearch Service multiplicando o número de nós de dados pelo tamanho do volume do EBS. O tamanho mínimo e máximo de um volume do EBS depende tanto do tipo de volume do EBS especificado quanto do tipo da instância à qual ele está anexado. Para saber mais, consulte Limites de tamanhos de volume do EBS.

Configuração	Description
IOPS provisionadas	Se você selecionou um tipo de volume SSD de IOPS provision adas, insira o número de I/O operações por segundo (IOPS) que o volume pode suportar.

13. (Opcional) Se você selecionou um tipo de `gp3` volume, expanda Configurações avançadas e especifique IOPS adicionais (até 16.000 para cada tamanho de volume de 3 TiB provisionado por nó de dados) e taxa de transferência (até 1.000 para MiB/s cada tamanho de volume de 3 TiB provisionado por nó de dados) além do que está incluído no preço do armazenamento, por um custo adicional. Para obter mais informações, consulte os preços do Amazon OpenSearch Service.
14. (Opcional) Para ativar o UltraWarm armazenamento, escolha Ativar nós UltraWarm de dados. Cada tipo de instância tem uma quantidade máxima de armazenamento que ele pode processar. Multiplique essa quantidade pelo número de nós de dados de alta atividade pelo total de armazenamento de alta atividade endereçável.
15. (Opcional) Para habilitar o armazenamento de baixa atividade, escolha Habilitar armazenamento de baixa atividade. Você deve habilitar UltraWarm para habilitar o armazenamento a frio.
16. Se você usa o multi-AZ com modo de espera, três nós principais dedicados já estão habilitados. Escolha o tipo de nós principais que você deseja. Se você escolheu um domínio Multi-AZ sem modo de espera, selecione Habilitar nós principais dedicados e escolha o tipo e o número de nós principais que você deseja. Os nós principais dedicados aumentam a estabilidade do cluster e são necessários para domínios com contagem de instâncias superior a 10. Recomendamos três nós principais dedicados para domínios de produção.

> **Note**
>
> Você pode escolher diferentes tipos de instâncias para seus nós principais dedicados e nós de dados. Por exemplo, você pode selecionar instâncias de uso geral ou de armazenamento otimizado para os nós de dados e instâncias otimizadas para computação para os nós principais dedicados.

17. (Opcional) Para domínios que executam o Elasticsearch OpenSearch 5.3 e versões posteriores, a configuração do Snapshot é irrelevante. Para saber mais sobre snapshots automatizados, consulte the section called “Criação de snapshots de índices”.
18. Se você quiser usar um endpoint personalizado em vez do padrão `https://search-`*`mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a`*`.`*`us-east-1`*`.es.amazonaws.com`,

escolha Habilitar endpoints personalizados e forneça um nome e um certificado. Para saber mais, consulte the section called “Criar um endpoint personalizado”.

19. Na seção Rede, escolha Acesso via VPC ou Acesso público. Se você selecionar Acesso público, vá para a próxima etapa. Se escolher Acesso à VPC, certifique-se de atender aos pré-requisitos e defina as seguintes configurações:

Configuração	Description
VPC	Escolha o ID da nuvem privada virtual (VPC) que deseja usar. A VPC e o domínio devem estar no mesmo lugar Região da AWS, e você deve selecionar uma VPC com a locação definida como Padrão. OpenSearch O serviço ainda não oferece suporte a VPCs esse uso de locação dedicada.
Sub-rede	Escolha uma sub-rede. Se você ativou o Multi-AZ, deverá escolher duas ou três sub-redes. OpenSearch O serviço colocará um endpoint VPC e interfaces de rede elástica nas sub-redes. Você deve reservar endereços IP suficientes para as interfaces de rede em toda sub-rede. Para saber mais, consulte Reserva de endereços IP em uma sub-rede da VPC.
Grupos de segurança	Escolha um ou mais grupos de segurança de VPC que permitam que seu aplicativo necessário alcance o domínio do OpenSearch Serviço nas portas (80 ou 443) e protocolos (HTTP ou HTTPS) expostos pelo domínio. Para obter mais informações, consulte the section called “Suporte à VPC”.
IAM Role	Mantenha a função padrão. OpenSearch O serviço usa essa função predefinida (também conhecida como função vinculada ao serviço) para acessar sua VPC e colocar um endpoint de VPC e interfaces de rede na sub-rede da VPC. Para saber mais, consulte Função vinculada ao serviço para acesso à VPC.
Tipo de endereço IP	Escolha pilha dupla ou IPv4 como seu tipo de endereço IP. A pilha dupla permite que você compartilhe recursos de domínio IPv4 e tipos de IPv6 endereço, e é a opção recomendada. Se você definir o tipo de endereço

Configuração	Description
	IP como pilha dupla, não poderá alterar o tipo de endereço posterior mente.

20. Habilite ou desabilite controle de acesso refinado:

    - Se você quiser usar o IAM para o gerenciamento de usuários, escolha Definir ARN do IAM como usuário primário e especifique o ARN para uma função do IAM.
    - Se quiser usar o banco de dados de usuário interno, escolha Criar usuário primário e especifique um nome de usuário e senha.

    Qualquer que seja a opção escolhida, o usuário principal pode acessar todos os índices do cluster e tudo mais. OpenSearch APIs Para obter orientações sobre qual opção escolher, consulte the section called “Principais conceitos”.

    Se você desabilitar o controle de acesso refinado, ainda assim poderá controlar o acesso ao seu domínio, colocando-o em uma VPC, aplicando uma política de acesso restritiva ou ambos. Você deve habilitar a node-to-node criptografia e a criptografia em repouso para usar um controle de acesso refinado.

    > **Note**
    >
    > Recomendamos enfaticamente habilitar o controle de acesso refinado para proteger os dados do seu domínio. O controle de acesso refinado fornece segurança nos níveis de cluster, índice, documento e campo.

21. (Opcional) Se você quiser usar a autenticação SAML para OpenSearch painéis, escolha Habilitar autenticação SAML e configure as opções de SAML para o domínio. Para instruções, consulte the section called “Autenticação SAML para painéis OpenSearch”.
22. (Opcional) Se você quiser usar a autenticação do Amazon Cognito para OpenSearch painéis, escolha Habilitar a autenticação do Amazon Cognito. Em seguida, escolha o grupo de usuários e o grupo de identidades do Amazon Cognito que você deseja usar para autenticação de OpenSearch painéis. Para obter orientações sobre a criação desses recursos, consulte the section called “Autenticação do Amazon Cognito para painéis OpenSearch ”.
23. (Opcional) Se você quiser usar a autenticação do IAM Identity Center (IDC) para conectar sua fonte de identidade existente e dar aos AWS aplicativos uma visão comum dos seus usuários,

escolha Habilitar o acesso à API autenticado com o IAM Identity Center. Para saber mais, consulte Visão geral da propagação de identidades de confiança no Guia do usuário do IAM Identity Center.

24. (Opcional) Na seção Recursos avançados, deixe Habilitar geração de consultas em linguagem natural e os atributos do Amazon Q Developer selecionado, se quiser usar esses atributos.

    a. Escolha Habilitar o S3 Vectors como uma opção de mecanismo para obter opções avançadas de pesquisa vetorial. Para obter mais informações, consulte Recursos avançados de pesquisa com um mecanismo vetorial Amazon S3.

    b. Escolha Ativar aceleração de GPU para opções aprimoradas de pesquisa vetorial. Para obter mais informações, consulte Aceleração por GPU para indexação vetorial.

25. Para Política de acesso, escolha uma política de acesso ou configure uma das suas próprias políticas. Se você optar por criar uma política personalizada, poderá configurá-la você mesmo ou importar uma política de outro domínio. Para saber mais, consulte the section called “Gerenciamento de Identidade e Acesso”.

    > **Note**
    >
    > Se você ativou o acesso à VPC, não poderá usar políticas baseadas em IP. Em vez disso, você poderá usar grupos de segurança para controlar quais endereços IP poderão acessar o domínio. Para saber mais, consulte the section called “Sobre políticas de acesso em domínios da VPC”.

26. (Opcional) Para exigir que todas as solicitações ao domínio sejam recebidas por HTTPS, selecione Exigir HTTPS para todo o tráfego do domínio. Para ativar a node-to-node criptografia, selecione ode-to-nodeCriptografia N. Para obter mais informações, consulte the section called “Node-to-node criptografia”. Para habilitar criptografia de dados em repouso, selecione Ativar criptografia em repouso. Essas opções são pré-selecionadas se você escolher a opção de implantação multi-AZ com modo de espera.

27. (Opcional) Selecione AWS Usar chave própria para que o OpenSearch Serviço crie uma chave de AWS KMS criptografia em seu nome (ou use a que já foi criada). Caso contrário, escolha sua própria chave do KMS. Para obter mais informações, consulte the section called “Criptografia em repouso”.

28. Para a janela fora do pico, selecione um horário de início para agendar atualizações do software de serviço e otimizações do Auto-Tune que exijam uma implantação. blue/green As atualizações

fora do horário de pico ajudam a minimizar a sobrecarga nos nós principais dedicados de um cluster durante períodos de tráfego intenso.

29. Para o Auto-Tune, escolha se deseja permitir que o OpenSearch Serviço sugira alterações de configuração relacionadas à memória para seu domínio para melhorar a velocidade e a estabilidade. Para obter mais informações, consulte the section called “Auto-Tune”.

    (Opcional) Selecione Janela fora do horário de pico para agendar uma janela recorrente durante a qual o Auto-Tune atualizará o domínio.
30. (Opcional) Selecione Atualização automática de software para habilitar atualizações automáticas de software.
31. (Opcional) Adicione tags para descrever seu domínio para que você possa categorizar e filtrar essas informações. Para saber mais, consulte the section called “Marcação de domínios”.
32. (Opcional) Expanda e defina as Configurações avançadas de cluster. Para obter um resumo dessas opções, consulte the section called “Configurações avançadas do cluster”.
33. Escolha Criar.

Criação OpenSearch de domínios de serviço ()AWS CLI

Em vez de criar um domínio de OpenSearch serviço usando o console, você pode usar AWS CLI o. Para obter a sintaxe, consulte Amazon OpenSearch Service na referência de comandos da AWS CLI a.

Exemplos de comando

Este primeiro exemplo demonstra a seguinte configuração do domínio OpenSearch de serviço:

- Cria um domínio OpenSearch de serviço chamado mylogs com a OpenSearch versão 1.2
- Preenche o domínio com duas instâncias do tipo `r6g.large.search`
- Utilização de um volume `gp3` de Finalidade geral (SSD) do EBS de 100 GiB como armazenamento para cada nó de dados
- Permite acesso anônimo, mas apenas de endereço IP único: 192.0.2.0/32.

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version OpenSearch_1.2 \
```

```
    --cluster-config  InstanceType=r6g.large.search,InstanceCount=2 \
    --ebs-options
 EBSEnabled=true,VolumeType=gp3,VolumeSize=100,Iops=3500,Throughput=125 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [{"Action":
 "es:*", "Principal":"*","Effect": "Allow", "Condition": {"IpAddress":{"aws:SourceIp":
["192.0.2.0/32"]}}}]}'
```

O próximo exemplo demonstra a seguinte configuração do domínio OpenSearch de serviço:

- Cria um domínio OpenSearch de serviço chamado mylogs com a versão 7.10 do Elasticsearch
- Preenche o domínio com seis instâncias do tipo `r6g.large.search`
- Utilização de um volume gp2 de Finalidade geral (SSD) do EBS de 100 GiB como armazenamento para cada nó de dados
- Restringe o acesso ao serviço a um único usuário, identificado pelo Conta da AWS ID do usuário: 555555555555
- Distribui as instâncias em três zonas de disponibilidade

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version Elasticsearch_7.10 \
    --cluster-config
 InstanceType=r6g.large.search,InstanceCount=6,ZoneAwarenessEnabled=true,ZoneAwarenessConfig={A
 \
    --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": {"AWS": "arn:aws:iam::555555555555:root" }, "Action":"es:*",
 "Resource": "arn:aws:es:us-east-1:555555555555:domain/mylogs/*" } ] }'
```

O próximo exemplo demonstra a seguinte configuração do domínio OpenSearch de serviço:

- Cria um domínio OpenSearch de serviço chamado mylogs com a OpenSearch versão 1.0
- Preenche o domínio com 10 instâncias do tipo `r6g.xlarge.search`
- Preenche o domínio com três instâncias do tipo `r6g.large.search` para funcionar como nós principais dedicados
- Usa um volume de EBS de IOPS provisionadas de 100 GiB como armazenamento, configurado com performance de referência de 1.000 IOPS para cada nó de dados.
- Restringe o acesso a um único usuário e a um único sub-recurso, a API `_search`

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version OpenSearch_1.0 \
    --cluster-config
 InstanceType=r6g.xlarge.search,InstanceCount=10,DedicatedMasterEnabled=true,DedicatedMasterTyp
 \
    --ebs-options EBSEnabled=true,VolumeType=io1,VolumeSize=100,Iops=1000 \
    --access-policies '{"Version": "2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": "arn:aws:iam::555555555555:root" }, "Action": "es:*",
 "Resource": "arn:aws:es:us-east-1:555555555555:domain/mylogs/_search" } ] }'
```

> **Note**
>
> Se você tentar criar um domínio OpenSearch de serviço e já existir um domínio com o mesmo nome, a CLI não relatará um erro. Em vez disso, ela retornará detalhes do domínio existente.

Criação OpenSearch de domínios de serviço ()AWS SDKs

O AWS SDKs (exceto o Android e o iOS SDKs) suporta todas as ações definidas na Amazon OpenSearch Service API Reference, inclusive`CreateDomain`. Para obter o código de exemplo, consulte the section called “Usando o AWS SDKs”. Para obter mais informações sobre como instalar e usar o AWS SDKs, consulte Kits AWS de desenvolvimento de software.

Criação OpenSearch de domínios de serviço ()AWS CloudFormation

OpenSearch O serviço é integrado com AWS CloudFormation, um serviço que ajuda você a modelar e configurar seus AWS recursos para que você possa gastar menos tempo criando e gerenciando seus recursos e infraestrutura. Você cria um modelo que descreve o OpenSearch domínio que você deseja criar e CloudFormation provisiona e configura o domínio para você. Para obter mais informações, incluindo exemplos de modelos JSON e YAML para OpenSearch domínios, consulte a referência do tipo de recurso do Amazon OpenSearch Service no Guia do AWS CloudFormation usuário.

Configuração de políticas de acesso

O Amazon OpenSearch Service oferece várias maneiras de configurar o acesso aos seus domínios do OpenSearch Serviço. Para obter mais informações, consulte the section called “Gerenciamento de Identidade e Acesso” e the section called “Controle de acesso refinado”.

O console fornece políticas de acesso pré-configuradas que você pode personalizar de acordo com as necessidades específicas de seu domínio. Você também pode importar políticas de acesso de outros domínios do OpenSearch Serviço. Para obter informações sobre como essas políticas de acesso interagem com o acesso à VPC, consulte the section called “Sobre políticas de acesso em domínios da VPC”.

Para configurar políticas de acesso (console)

1. Vá para https://aws.amazon.com, e escolha Sign In to the Console (Faça login no Console).
2. Em Analytics, escolha Amazon OpenSearch Service.
3. No painel de navegação, em Domínios, escolha o domínio que deseja atualizar.
4. Escolha Ações e Editar configuração de segurança.
5. Edite a política de acesso JSON ou importe uma opção pré-configurada.
6. Escolha Salvar alterações.

Migração para o OpenSearch serviço com o Migration Assistant

O Migration Assistant for Amazon OpenSearch Service é uma solução abrangente que simplifica o processo de migração do Elasticsearch ou clusters autogerenciados para o Service. OpenSearch OpenSearch Esse kit de ferramentas lida com a complexidade operacional das migrações e, ao mesmo tempo, garante a integridade dos dados e valida a performance após a migração.

Visão geral do

Se você estiver configurando uma proof-of-concept entrada AWS, fazendo a transição de cargas de trabalho de produção ou atualizando para OpenSearch as versões mais recentes, o Assistente de Migração fornece step-by-step orientação, melhores práticas e ferramentas para aproveitar todo o potencial das migrações. OpenSearch

Estes são os principais benefícios do Migration Assistant:

- Migração de metadados: migra metadados de cluster, incluindo configurações de índice, mapeamentos de tipos, modelos de índice e aliases
- Migração de dados - migra dados existentes de clusters legados para domínios OpenSearch de serviço
- Tratamento de tráfego ao vivo - intercepta e redireciona o tráfego ao vivo de clusters autogerenciados para domínios de OpenSearch serviço com latência mínima
- Replicação de tráfego: replica o tráfego de produção nos clusters de destino para validar precisão e performance
- Teste de performance: simula o tráfego do mundo real capturando e reproduzindo padrões de solicitações para ajustar a performance do sistema
- Disponibilidade global - Implantações AWS nas regiões mais comuns para alcance e escalabilidade globais

O Migration Assistant oferece suporte à migração das versões 6.x e 7.x e 1.x e 2.x do Elasticsearch. OpenSearch Para saber mais, consulte Caminhos de migração compatíveis.

> **Note**
>
> Observe as informações a seguir sobre o Migration Assistant.
>
> - A ferramenta oferece suporte a migrações de vários saltos (por exemplo, migração do Elasticsearch 5.x para o OpenSearch Service 3.x em um único salto).
> - É possível reverter uma migração.
> - Em alguns casos de uso, a ferramenta requer pouco ou nenhum tempo de inatividade.
> - A ferramenta fornece backfill de alta performance sem afetar o cluster de origem.

Cenários de migração

O Migration Assistant foi projetado para lidar com os seguintes cenários de migração:

Migração de metadados

Migra metadados do cluster, como configurações de índice, aliases e modelos do seu cluster de origem para o domínio do Serviço de destino OpenSearch .

Migração de backfill

Migra dados existentes ou históricos de um cluster de origem para um domínio de OpenSearch serviço de destino, garantindo que todos os seus dados importantes sejam preservados durante a transição.

Migração de tráfego ao vivo

Replica o tráfego ativo contínuo do seu cluster de origem para o domínio de OpenSearch serviço de destino, permitindo que você mantenha a disponibilidade do serviço durante a migração.

> **Important**
>
> As estratégias de migração não são universalmente aplicáveis. O Migration Assistant fornece orientação com base nas práticas recomendadas de engenharia, mas você deve avaliar seus requisitos específicos e testar cuidadosamente antes de migrar workloads de produção.

Introdução ao Migration Assistant

O Migration Assistant for Amazon OpenSearch Service está disponível como uma AWS solução com documentação abrangente, modelos de implantação e código-fonte. Para começar com o Migration Assistant:

Para acessar os recursos do Migration Assistant

1. Consulte a documentação completa da solução na visão geral da solução Migration Assistant for Amazon OpenSearch Service.
2. Entenda os custos e os requisitos lendo o guia de planejamento de implantação.
3. Implante a solução usando as instruções de implantação e siga o guia de uso para realizar a migração.

Para desenvolvedores e usuários avançados, o código-fonte e a documentação adicional do Assistente de Migração estão disponíveis no GitHub repositório de OpenSearch migrações.

Arquitetura da solução

Quando implantado no AWS, o Assistente de Migração usa vários AWS serviços para fornecer uma solução de migração abrangente:

- AWS CloudFormation: fornece modelos de infraestrutura como código (IaC) para implantar e configurar o Migration Assistant
- OpenSearch Serviço - O serviço de destino para suas cargas de trabalho migradas de pesquisa e análise
- Amazon Managed Streaming para Apache Kafka: fornece recursos de processamento de fluxos para armazenamento durável e reutilização do tráfego HTTP
- Amazon Elastic Container Service: executa o Migration Management Console e o Traffic Replayer em contêineres seguros escaláveis
- Amazon Elastic File System: fornece armazenamento persistente escalável para dados de solicitações e respostas dos clusters de origem e de destino
- Amazon Simple Storage Service: armazena snapshots para tarefas históricas de backfill e conteúdo infraestrutura como código

Para obter informações detalhadas, consulte Detalhes da arquitetura do Migration Assistant.

Configurações avançadas do cluster

Use as opções avançadas para configurar o seguinte:

Índices em corpos de solicitações

Especifica se são permitidas referências explícitas aos índices dentro do corpo das solicitações HTTP. A definição dessa propriedade como `false` impede que os usuários ignorem o controle de acesso para sub-recursos. Por padrão, o valor é `true`. Para saber mais, consulte the section called “Opções avançadas e considerações sobre a API”.

Alocação de cache de dados de campo

Especifica a porcentagem de espaço do heap do Java alocada a dados de campo. Por padrão, essa configuração é 20% do heap JVM.

> **Note**
>
> Muitos clientes consultam índices alternados diariamente. Recomenda-se começar a realizar um teste de comparação com `indices.fielddata.cache.size` configurado como 40% do heap de JVM para a maioria desses casos de uso. Para índices muito grandes, talvez um cache de dados de campo grande seja necessário.

Contagem máxima de cláusulas

Especifica o número máximo de cláusulas permitidas em uma consulta booliana no Lucene. O padrão é 1.024. Consultas que ultrapassam o número permitido de cláusulas geram o erro `TooManyClauses`. Para saber mais, consulte a documentação do Lucene.

Enriquecimento semântico automático para Amazon Service OpenSearch

Introdução

O Amazon OpenSearch Service usa word-to-word correspondência (pesquisa léxica) para encontrar resultados, semelhante a outros mecanismos de pesquisa tradicionais. Essa abordagem funciona bem para consultas específicas, como códigos de produtos ou números de modelos, mas tem dificuldades com pesquisas abstratas, nas quais entender a intenção do usuário se torna crucial. Por exemplo, quando você pesquisa por “sapatos para a praia”, a pesquisa léxica corresponde às palavras individuais “sapatos”, “praia”, “para” e “o” nos itens do catálogo, possivelmente ausentes de produtos relevantes, como “sandálias resistentes à água” ou “calçados de surfe”, que não contêm os termos de pesquisa exatos.

O enriquecimento semântico automático resolve essa limitação considerando as correspondências de palavras-chave e o significado contextual por trás das pesquisas. Esse recurso compreende a intenção da pesquisa e melhora a relevância da pesquisa em até 20%. Ative esse recurso para campos de texto em seu índice para aprimorar os resultados da pesquisa.

> **Note**
>
> O enriquecimento semântico automático está disponível para domínios de OpenSearch serviço que executam a versão 2.19 ou posterior. Além disso, os domínios com a OpenSearch versão 2.19 também precisam estar na atualização mais recente da versão do software de serviço. Atualmente, o recurso está disponível para domínios públicos, e os domínios VPC não são compatíveis.

Detalhes do modelo e benchmark de desempenho

Embora esse recurso lide com as complexidades técnicas nos bastidores sem expor o modelo subjacente, fornecemos transparência por meio de uma breve descrição do modelo e resultados de benchmark para ajudá-lo a tomar decisões informadas sobre a adoção de recursos em suas cargas de trabalho críticas.

O enriquecimento semântico automático usa um modelo esparso pré-treinado e gerenciado por serviços que funciona de forma eficaz sem exigir ajustes personalizados. O modelo analisa os campos que você especifica, expandindo-os em vetores esparsos com base em associações aprendidas de diversos dados de treinamento. Os termos expandidos e seus pesos de significância são armazenados no formato nativo do índice Lucene para uma recuperação eficiente. Otimizamos esse processo usando o modo somente documentos, em que a codificação ocorre somente durante a ingestão de dados. As consultas de pesquisa são meramente tokenizadas em vez de processadas por meio de um modelo esparso, tornando a solução econômica e eficiente.

Nossa validação de desempenho durante o desenvolvimento do recurso usou o conjunto de dados de recuperação de passagens MS MARCO, com passagens com média de 334 caracteres. Para pontuação de relevância, medimos a média de ganho cumulativo com desconto normalizado (NDCG) para os primeiros 10 resultados de pesquisa (ndcg @10) no benchmark BEIR para conteúdo em inglês e a média de ndcg @10 no MIRACL para conteúdo multilíngue. Avaliamos a latência por meio de medições do percentil 90 (p90) do lado do cliente e a resposta de pesquisa p90 obteve valores. Esses benchmarks fornecem indicadores básicos de desempenho para a relevância da pesquisa e os tempos de resposta. Aqui estão os principais números de referência -

- Idioma inglês - Melhoria de relevância de 20% em relação à pesquisa léxica. Também reduziu a latência de pesquisa do P90 em 7,7% em relação à pesquisa lexical (BM25 é 26 ms e o enriquecimento semântico automático é 24 ms).
- Multilíngue - Melhoria de relevância de 105% em relação à pesquisa lexical, enquanto a latência da pesquisa P90 aumentou 38,4% em relação à pesquisa lexical (BM25 é 26 ms e o enriquecimento semântico automático é 36 ms).

Dada a natureza exclusiva de cada carga de trabalho, recomendamos que você avalie esse recurso em seu ambiente de desenvolvimento usando seus próprios critérios de benchmarking antes de tomar decisões de implementação.

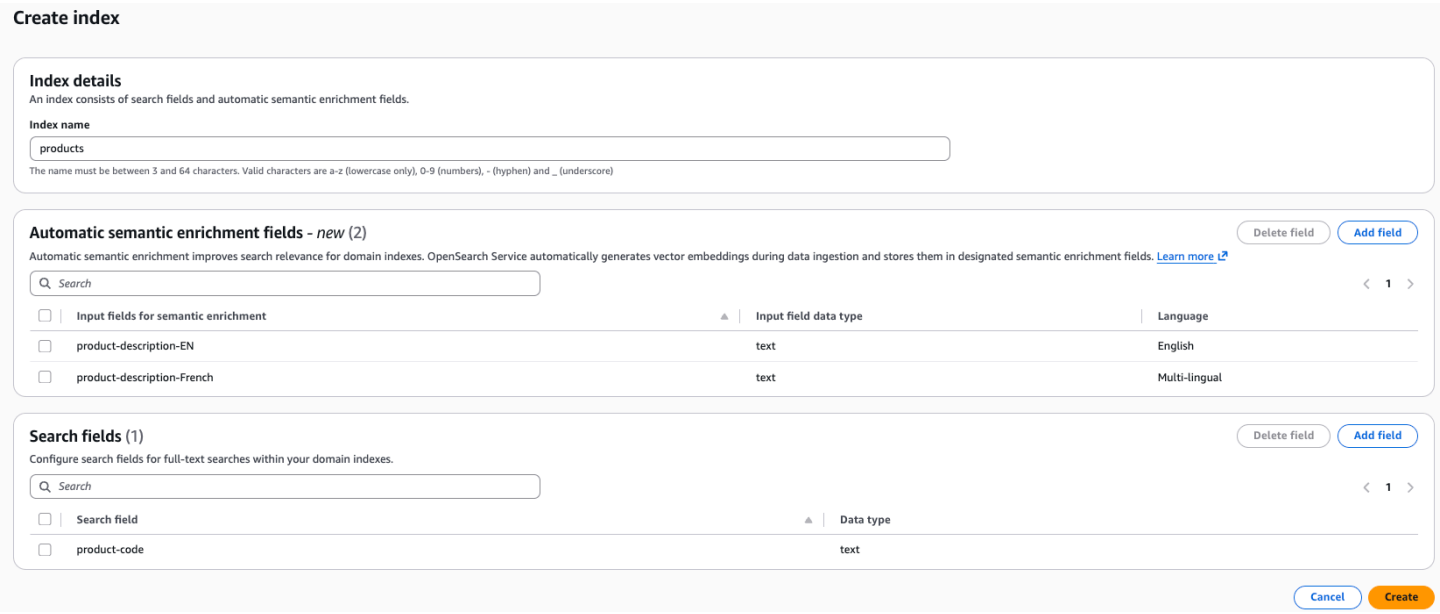
Idiomas suportados

O recurso oferece suporte ao inglês. Além disso, o modelo também oferece suporte para árabe, bengali, chinês, finlandês, francês, hindi, indonésio, japonês, coreano, persa, russo, espanhol, suaíli e telugu.

Configure um índice automático de enriquecimento semântico para domínios

É fácil configurar um índice com o enriquecimento semântico automático ativado para seus campos de texto e você pode gerenciá-lo por meio do console e de CloudFormation modelos durante a criação do novo índice. APIs Para habilitá-lo para um índice existente, você precisa recriar o índice com o enriquecimento semântico automático ativado para campos de texto.

Experiência de console - O AWS console permite criar facilmente um índice com campos de enriquecimento semântico automático. Depois de selecionar um domínio, você encontrará o botão criar índice na parte superior do console. Depois de clicar no botão criar índice, você encontrará opções para definir campos de enriquecimento semântico automático. Em um índice, você pode ter combinações de enriquecimento semântico automático para inglês e multilíngue, bem como campos lexicais.



Experiência de API - Para criar um índice automático de enriquecimento semântico usando a interface de linha de AWS comando (AWS CLI), use o comando create-index:

```
aws opensearch create-index \
```

```
--domain-name [domain_name] \
--index-name [index_name] \
--index-schema [index_body] \
```

No exemplo de esquema de índice a seguir, o campo title_semantic tem um tipo de campo definido como texto e o parâmetro semantic_enrichment definido como status ENABLED. Definir o parâmetro semantic_enrichment ativa o enriquecimento semântico automático no campo title_semantic. Você pode usar o campo language_options para especificar inglês ou multilíngue.

```
    aws opensearch create-index \
    --id XXXXXXXXX \
    --index-name 'product-catalog' \
    --index-schema '{
    "mappings": {
        "properties": {
            "product_id": {
                "type": "keyword"
            },
            "title_semantic": {
                "type": "text",
                "semantic_enrichment": {
                    "status": "ENABLED",
                    "language_options": "english"
                }
            },
            "title_non_semantic": {
                "type": "text"
            }
        }
    }
}'
```

Para descrever o índice criado, use o seguinte comando:

```
aws opensearch get-index \
--domain-name [domain_name] \
--index-name [index_name] \
```

Ingestão e pesquisa de dados

Depois de criar um índice com o enriquecimento semântico automático ativado, o recurso funciona automaticamente durante o processo de ingestão de dados, sem necessidade de configuração adicional.

Ingestão de dados: quando você adiciona documentos ao seu índice, o sistema automaticamente:

- Analisa os campos de texto que você designou para enriquecimento semântico
- Gera codificações semânticas usando o modelo esparso gerenciado por OpenSearch serviços
- Armazena essas representações enriquecidas junto com seus dados originais

Esse processo usa conectores OpenSearch de ML e pipelines de ingestão integrados, que são criados e gerenciados automaticamente nos bastidores.

Pesquisa: os dados de enriquecimento semântico já estão indexados, então as consultas são executadas com eficiência sem invocar o modelo de ML novamente. Isso significa que você obtém maior relevância de pesquisa sem sobrecarga adicional de latência de pesquisa.

Configurar permissões para enriquecimento semântico automático

Antes de criar um índice com enriquecimento semântico automático, você precisa configurar as permissões necessárias. Esta seção explica as permissões necessárias para diferentes operações de indexação e como configurá-las tanto para cenários de controle de acesso AWS Identity and Access Management (IAM) quanto para cenários de controle de acesso refinados.

permissões do IAM

As seguintes permissões do IAM são necessárias para operações automáticas de enriquecimento semântico. Essas permissões variam de acordo com a operação de indexação específica que você deseja realizar.

CreateIndex Permissões de API

Para criar um índice com enriquecimento semântico automático, você precisa das seguintes permissões do IAM:

- `es:CreateIndex`— Crie um índice com recursos de enriquecimento semântico.

- `es:ESHttpHead`— Execute solicitações HEAD para verificar a existência do índice.
- `es:ESHttpPut`— Execute solicitações PUT para criação de índice.
- `es:ESHttpPost`— Execute solicitações POST para operações de indexação.

UpdateIndex Permissões de API

Para atualizar um índice existente com enriquecimento semântico automático, você precisa das seguintes permissões do IAM:

- `es:UpdateIndex`— Atualize as configurações e mapeamentos do índice.
- `es:ESHttpPut`— Execute solicitações PUT para atualizações de índices.
- `es:ESHttpGet`— Execute solicitações GET para recuperar informações do índice.
- `es:ESHttpPost`— Execute solicitações POST para operações de indexação.

GetIndex Permissões de API

Para recuperar informações sobre um índice com enriquecimento semântico automático, você precisa das seguintes permissões do IAM:

- `es:GetIndex`— Recupere informações e configurações do índice.
- `es:ESHttpGet`— Execute solicitações GET para recuperar dados do índice.

DeleteIndex Permissões de API

Para excluir um índice com enriquecimento semântico automático, você precisa das seguintes permissões do IAM:

- `es:DeleteIndex`— Exclua um índice e seus componentes de enriquecimento semântico.
- `es:ESHttpDelete`— Execute solicitações DELETE para remoção do índice.

Exemplo de política do IAM

O exemplo de política de acesso baseado em identidade a seguir fornece as permissões necessárias para que um usuário gerencie índices com enriquecimento semântico automático:

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSemanticEnrichmentIndexOperations",
            "Effect": "Allow",
            "Action": [
                "es:CreateIndex",
                "es:UpdateIndex",
                "es:GetIndex",
                "es:DeleteIndex",
                "es:ESHttpHead",
                "es:ESHttpGet",
                "es:ESHttpPut",
                "es:ESHttpPost",
                "es:ESHttpDelete"
            ],
            "Resource": "arn:aws:es:aws-region:111122223333:domain/domain-name"
        }
    ]
}
```

Substitua *aws-region111122223333*,, e *domain-name* por seus valores específicos. Você pode restringir ainda mais o acesso especificando padrões de índice específicos no ARN do recurso.

Permissões de acesso refinado

Se o seu domínio do Amazon OpenSearch Service tiver um controle de acesso refinado ativado, você precisará de permissões adicionais além das permissões do IAM. As permissões a seguir são necessárias para cada operação de indexação.

CreateIndex Permissões de API

Quando o controle de acesso refinado é ativado, as seguintes permissões adicionais são necessárias para criar um índice com enriquecimento semântico automático:

- `indices:admin/create`— Crie operações de indexação.
- `indices:admin/mapping/put`— Crie e atualize mapeamentos de índice.
- `cluster:admin/opensearch/ml/create_connector`— Crie conectores de aprendizado de máquina para processamento semântico.
- `cluster:admin/opensearch/ml/register_model`— Registre modelos de aprendizado de máquina para enriquecimento semântico.

- `cluster:admin/ingest/pipeline/put`— Crie canais de ingestão para processamento de dados.
- `cluster:admin/search/pipeline/put`— Crie canais de pesquisa para processamento de consultas.

UpdateIndex Permissões de API

Quando o controle de acesso refinado é ativado, as seguintes permissões adicionais são necessárias para atualizar um índice com enriquecimento semântico automático:

- `indices:admin/get`— Recupere informações do índice.
- `indices:admin/settings/update`— Atualizar as configurações do índice.
- `indices:admin/mapping/put`— Atualizar mapeamentos de índice.
- `cluster:admin/opensearch/ml/create_connector`— Crie conectores de aprendizado de máquina.
- `cluster:admin/opensearch/ml/register_model`— Registre modelos de aprendizado de máquina.
- `cluster:admin/ingest/pipeline/put`— Crie canais de ingestão.
- `cluster:admin/search/pipeline/put`— Crie canais de busca.
- `cluster:admin/ingest/pipeline/get`— Recupere as informações do pipeline de ingestão.
- `cluster:admin/search/pipeline/get`— Recupere informações do pipeline de pesquisa.

GetIndex Permissões de API

Quando o controle de acesso refinado é ativado, as seguintes permissões adicionais são necessárias para recuperar informações sobre um índice com enriquecimento semântico automático:

- `indices:admin/get`— Recupere informações do índice.
- `cluster:admin/ingest/pipeline/get`— Recupere as informações do pipeline de ingestão.
- `cluster:admin/search/pipeline/get`— Recupere informações do pipeline de pesquisa.

DeleteIndex Permissões de API

Quando o controle de acesso refinado é ativado, a seguinte permissão adicional é necessária para excluir um índice com enriquecimento semântico automático:

- `indices:admin/delete`— Excluir operações de indexação.

Regravações de consultas

O enriquecimento semântico automático converte automaticamente suas consultas de "correspondência" existentes em consultas de pesquisa semântica sem exigir modificações na consulta. Se uma consulta de correspondência fizer parte de uma consulta composta, o sistema percorrerá sua estrutura de consulta, localizará consultas de correspondência e as substituirá por consultas neurais esparsas. Atualmente, o recurso suporta apenas a substituição de consultas "correspondentes", seja uma consulta autônoma ou parte de uma consulta composta. "multi_match" não é suportado. Além disso, o recurso oferece suporte a todas as consultas compostas para substituir suas consultas de correspondência aninhadas. As consultas compostas incluem: bool, boosting, constant_score, dis_max, function_score e hybrid.

Limitações do enriquecimento semântico automático

A pesquisa semântica automática é mais eficaz quando aplicada a campos small-to-medium de tamanho contendo conteúdo em linguagem natural, como títulos de filmes, descrições de produtos, avaliações e resumos. Embora a pesquisa semântica aumente a relevância para a maioria dos casos de uso, ela pode não ser ideal para determinados cenários. Considere as seguintes limitações ao decidir se deve implementar o enriquecimento semântico automático para seu caso de uso específico.

- Documentos muito longos — O modelo esparso atual processa apenas os primeiros 8.192 tokens de cada documento em inglês. Para documentos multilíngues, são 512 tokens. Para artigos longos, considere implementar a fragmentação de documentos para garantir o processamento completo do conteúdo.
- Cargas de trabalho de análise de registros — O enriquecimento semântico aumenta significativamente o tamanho do índice, o que pode ser desnecessário para a análise de registros, onde a correspondência exata normalmente é suficiente. O contexto semântico adicional raramente melhora a eficácia da pesquisa de registros o suficiente para justificar o aumento dos requisitos de armazenamento.
- O enriquecimento semântico automático não é compatível com o recurso Fonte Derivada.
- Limitação — As solicitações de inferência de indexação estão atualmente limitadas a 200 TPS para domínios de serviço. OpenSearch Esse é um limite flexível; entre em contato com o AWS Support para obter limites mais altos.

Preços

O Amazon OpenSearch Service fatura o enriquecimento semântico automático com base nas unidades de OpenSearch computação (OCUs) consumidas durante a geração de vetores esparsos no momento da indexação. Você é cobrado somente pelo uso real durante a indexação. Você pode monitorar esse consumo usando a SemanticSearch OCU CloudWatch métrica da Amazon. Para obter detalhes específicos sobre os limites do token do modelo, a taxa de transferência de volume por OCU e um exemplo de cálculo de amostra, visite Preços de OpenSearch serviços.

Fazendo alterações de configuração no Amazon OpenSearch Service

O Amazon OpenSearch Service usa um processo de implantação azul/verde ao atualizar domínios. Uma blue/green implantação cria um ambiente ocioso para atualizações de domínio que copia o ambiente de produção e direciona os usuários para o novo ambiente após a conclusão dessas atualizações. Em uma blue/green implantação, o ambiente azul é o ambiente de produção atual. O ambiente verde é o ambiente inativo.

Os dados são migrados do ambiente azul para o ambiente verde. Quando o novo ambiente estiver pronto, o OpenSearch Serviço alterna os ambientes para promover o ambiente verde como o novo ambiente de produção. A transição ocorre sem perda de dados. Essa prática minimiza o tempo de inatividade e mantém o ambiente original caso a implantação no novo ambiente resulte em erro.

Tópicos

- Mudanças que geralmente causam blue/green implantações
- Mudanças que geralmente não causam blue/green implantações
- Opções de implantação azul/verde
- Determinar se uma alteração causará uma blue/green implantação
- Rastrear uma alteração de configuração
- Etapas de uma alteração de configuração
- Impacto no desempenho das blue/green implantações
- Cobranças para alterações de configuração
- Solução de problemas de erros de validação

Mudanças que geralmente causam blue/green implantações

As seguintes operações causam blue/green implantações:

- Alterar o tipo de instância
- Habilitar o controle de acesso detalhado
- Atualizações de software de serviço
- Habilitar ou desabilitar os nós principais dedicados
- Ativar ou desativar o Multi-AZ sem modo de espera
- Alterar o tipo de armazenamento, o tipo do volume ou o tamanho do volume
- Escolher diferentes sub-redes da VPC
- Adicionar ou remover os grupos de segurança da VPC
- Adicionar e remover nós coordenadores dedicados
- Ativar ou desativar a autenticação do Amazon Cognito para painéis OpenSearch
- Escolha de outro grupo de usuários ou grupo de identidades do Amazon Cognito
- Modificar configurações avançadas
- Atualização para uma nova OpenSearch versão (os OpenSearch painéis podem estar indisponíveis durante parte ou toda a atualização)
- Habilitando a criptografia de dados em repouso ou node-to-node criptografia
- Ativando ou desativando nosso UltraWarm armazenamento a frio
- Desabilitação do Auto-Tune e reversão de suas alterações
- Associar um plug-in opcional a um domínio e dissociar um plug-in opcional de um domínio
- Aumento da contagem de nós principais dedicados para domínios Multi-AZ com dois nós principais dedicados
- Diminuição de tamanho do volume do EBS
- Alteração do tamanho do volume, IOPS ou throughput do EBS, se a última alteração feita estiver em andamento ou tiver ocorrido há menos de 6 horas
- Habilitando a publicação de registros de auditoria para CloudWatch.

Para domínios multi-AZ com modo de espera, você só pode fazer uma solicitação de alteração por vez. Se uma alteração já estiver em andamento, a nova solicitação será rejeitada. Você pode verificar o status da alteração atual com a API da `DescribeDomainChangeProgress`.

Mudanças que geralmente não causam blue/green implantações

Na maioria dos casos, as seguintes operações não causam blue/green implantações:

- Modificar a política de acesso
- Como modificar o endpoint personalizado
- Alterar política do Transport Layer Security (TLS)
- Alterar o horário do snapshot automatizado
- Habilitar ou desabilitar a opção Exigir HTTPS
- Habilitação do Auto-Tune ou desabilitação sem reverter suas alterações
- Se seu domínio tiver nós mestres dedicados, alterando o nó de dados ou a contagem de UltraWarm nós
- Se seu domínio tiver nós principais dedicados, altere o tipo de instância principal dedicada ou a contagem (exceto para domínios Multi-AZ com dois nós principais dedicados)
- Ativar ou desativar a publicação de registros de erros ou registros lentos no CloudWatch
- Desabilitando a publicação de registros de auditoria no CloudWatch
- Aumentar o tamanho do volume, o IOPS ou a taxa de transferência para o tipo gp3 EBS.

> **Note**
>
> Antes de 10 de março de 2026, os aumentos de volume no local eram suportados somente para volumes de até 3 TiB para gp3. Em 10 de março de 2026, essa limitação foi removida, permitindo aumentos de volume no local além de 3 TiB. No entanto, se seu cluster tinha um tamanho de volume acima de 3 TiB antes que essa limitação fosse removida, o primeiro aumento de volume exigirá uma blue/green deployment. All subsequent volume increases for that cluster will be performed as in-place updates and will not require a blue/green implantação.

- Adicionar ou remover tags

> **Note**
>
> Há algumas exceções, dependendo da versão do software de serviço. Se você quiser ter certeza de que uma alteração não causará uma blue/green implantação, faça um dry run

antes de atualizar seu domínio, se essa opção estiver disponível. Algumas mudanças não oferecem a opção de simulação. Geralmente, recomendamos que você faça alterações em seu cluster fora dos horários de pico de tráfego.

Opções de implantação azul/verde

Selecione uma estratégia de implantação para controlar como seu cluster lida com implantações quando a capacidade suficiente não está disponível no momento da atualização.

1. Full Swap Blue/Green — O comportamento padrão de implantação. Requer capacidade total da instância desde o início, garantindo a implantação mais rápida quando a capacidade está disponível. A implantação não prosseguirá se a capacidade suficiente não puder ser alocada.
2. Capacidade otimizada — Recomendado para clusters com mais de 30 nós de dados. Tenta primeiro uma blue/green troca completa e, se a capacidade for insuficiente, prossegue com a implantação em lotes. Garante que as implantações possam ser concluídas mesmo quando a capacidade é limitada. O tempo de conclusão pode aumentar, pois a implantação será feita em lotes.

Console

Para editar fluxo de domínio:

1. Abra o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/home.
2. No painel de navegação, em Domínios, escolha o nome do domínio para abrir a Configuração do Cluster.
3. Clique no botão Editar no lado direito na guia Configuração do cluster.
4. Em Opções de estratégia de implantação, escolha as configurações necessárias para atualização do domínio:

    1. Full Swap Blue/Green — O comportamento padrão de implantação. Requer capacidade total da instância desde o início, garantindo a implantação mais rápida quando a capacidade está disponível. A implantação não prosseguirá se a capacidade suficiente não puder ser alocada.
    2. Capacidade otimizada — Recomendado para clusters com mais de 30 nós de dados. Tenta primeiro uma blue/green troca completa e, se a capacidade for insuficiente,

prossegue com a implantação em lotes. Garante que as implantações possam ser concluídas mesmo quando a capacidade é limitada. O tempo de conclusão pode aumentar, pois a implantação será feita em lotes.

5. Escolha Salvar alterações.

Para criar fluxo de domínio:

1. Abra o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/home.
2. No painel de navegação à esquerda, selecione Domínios.
3. Clique no botão Criar domínio.
4. Selecione todas as configurações necessárias para o domínio.
5. Em Opções de estratégia de implantação, escolha as configurações necessárias para atualização do domínio:
    1. Full Swap Blue/Green — O comportamento padrão de implantação. Requer capacidade total da instância desde o início, garantindo a implantação mais rápida quando a capacidade está disponível. A implantação não prosseguirá se a capacidade suficiente não puder ser alocada.
    2. Capacidade otimizada — Recomendado para clusters com mais de 30 nós de dados. Tenta primeiro uma blue/green troca completa e, se a capacidade for insuficiente, prossegue com a implantação em lotes. Garante que as implantações possam ser concluídas mesmo quando a capacidade é limitada. O tempo de conclusão pode aumentar, pois a implantação será feita em lotes.
6. Clique no botão Criar no painel Resumo do Domínio no lado direito.

API

Você pode configurar a estratégia de implantação usando a UpdateDomainConfigAPI.

Opção de capacidade otimizada

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
config
{
   "DeploymentStrategyOptions": {
```

```
    "DeploymentStrategy": "CapacityOptimized"
   }
}
```

Opção padrão

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
config
{
   "DeploymentStrategyOptions": {
    "DeploymentStrategy": "Default"
   }
}
```

Determinar se uma alteração causará uma blue/green implantação

Você pode testar alguns tipos de alterações de configuração planejadas para determinar se elas causarão uma blue/green implantação, sem precisar se comprometer com essas alterações. Antes de iniciar uma alteração de configuração, use o console ou uma API para executar uma verificação de validação para garantir que o seu domínio seja qualificação para uma atualização.

Console

Para validar uma alteração de configuração

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, selecione Domínios.
3. Selecione o domínio para o qual deseja fazer uma alteração de configuração. Isso abre a página de detalhes do domínio. Selecione o menu suspenso Ações e escolha Editar configuração do cluster.
4. Faça alterações no domínio, como alterar o tipo de instância ou o número de nós.
5. Em Análise de simulação, escolha Executar. O dry run valida sua alteração de configuração em busca de erros e determina se ela requer uma blue/green implantação.
6. Quando a simulação é concluída, os resultados são exibidos automaticamente no fim da página, junto com um ID da simulação. A análise indica se a alteração na configuração requer ou não uma blue/green implantação.

Cada simulação substitui a anterior. Para reter os detalhes de cada simulação, salve o ID da simulação. As simulações ficam disponíveis por 90 dias ou até que seja feita uma atualização de configuração.

7. Para continuar com a atualização de configuração, escolha Salvar alterações. Caso contrário, escolha Cancelar. Qualquer uma das opções levará você de volta à guia Configuração do cluster . Nessa guia, você pode escolher Detalhes da simulação para ver os detalhes da última simulação. Essa página também inclui uma side-by-side comparação entre a configuração antes da operação a seco e a configuração da operação a seco.

API

Você pode executar uma validação de simulação por meio da API de configuração. Para testar suas alterações com a API, defina `DryRun ´`como `true` e `DryRunMode` como `Verbose`. O modo detalhado executa uma verificação de validação, além de determinar se a alteração iniciará uma implantação azul/verde. Por exemplo, essa UpdateDomainConfigsolicitação testa o tipo de implantação resultante da ativação de UltraWarm:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
config
{
   "ClusterConfig": {
    "WarmCount": 3,
    "WarmEnabled": true,
    "WarmType": "ultrawarm1.large.search"
   },
   "DryRun": true,
   "DryRunMode": "Verbose"
}
```

A solicitação executa uma verificação de validação e retorna o tipo de implantação que a alteração causará, mas na verdade não executa a atualização:

```
{
   "ClusterConfig": {
     ...
    },
   "DryRunResults": {
      "DeploymentType": "Blue/Green",
      "Message": "This change will require a blue/green deployment."
```

```
    }
}
```

Os possíveis tipos de implantação são:

- `Blue/Green`: a alteração causará uma implantação azul/verde.
- `DynamicUpdate`: a alteração não causará uma implantação azul/verde.
- `Undetermined`: o domínio ainda está em um estado de processamento, portanto, não é possível determinar o tipo de implantação.
- `None`: nenhuma alteração de configuração.

Se a validação falhar, ela retornará uma lista de falhas de validação.

```
{
   "ClusterConfig":{
      "..."
   },
   "DryRunProgressStatus":{
      "CreationDate":"2023-01-12T01:14:33.847Z",
      "DryRunId":"db00ca39-48b2-4774-bbd3-252cf094d205",
      "DryRunStatus":"failed",
      "UpdateDate":"2023-01-12T01:14:33.847Z",
      "ValidationFailures":[
         {
            "Code":"Cluster.Index.WriteBlock",
            "Message":"Cluster has index write blocks."
         }
      ]
   }
}
```

Se o status persistir`pending`, você poderá usar o ID de execução seca em sua UpdateDomainConfig resposta em DescribeDryRunProgresschamadas subsequentes para verificar o status da validação.

```
GET https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
dryRun?dryRunId=my-dry-run-id
{
    "DryRunConfig": null,
```

```
    "DryRunProgressStatus": {
        "CreationDate": "2023-01-12T01:14:42.998Z",
        "DryRunId": "db00ca39-48b2-4774-bbd3-252cf094d205",
        "DryRunStatus": "succeeded",
        "UpdateDate": "2023-01-12T01:14:49.334Z",
        "ValidationFailures": null
    },
    "DryRunResults": {
        "DeploymentType": "Blue/Green",
        "Message": "This change will require a blue/green deployment."
    }
}
```

Para executar uma análise de simulação sem uma verificação de validação, defina `DryRunMode` como `Basic` quando usar a API de configuração.

Python

O código Python a seguir usa a UpdateDomainConfigAPI para realizar uma verificação de validação de execução seca e, se a verificação for bem-sucedida, chama a mesma API sem uma execução seca para iniciar a atualização. Se a verificação falhar, o script imprimirá o erro e será interrompido.

```
import time
import boto3

client = boto3.client('opensearch')

response = client.UpdateDomainConfig(
    ClusterConfig={
        'WarmCount': 3,
        'WarmEnabled': True,
        'WarmCount': 123,
    },
    DomainName='test-domain',
    DryRun=True,
    DryRunMode='Verbose'
)

dry_run_id = response.DryRunProgressStatus.DryRunId

retry_count = 0
```

```
while True:

    if retry_count == 5:
        print('An error occured')
        break

    dry_run_progress_response = client.DescribeDryRunProgress('test-domain',
 dry_run_id)
    dry_run_status = dry_run_progress_response.DryRunProgressStatus.DryRunStatus

    if dry_run_status == 'succeeded':
        client.UpdateDomainConfig(
            ClusterConfig={
            'WarmCount': 3,
            'WarmEnabled': True,
            'WarmCount': 123,
        })
        break

    elif dry_run_status == 'failed':
        validation_failures_list =
 dry_run_progress_response.DryRunProgressStatus.ValidationFailures
        for item in validation_failures_list:
            print(f"Code: {item['Code']}, Message: {item['Message']}")
        break

    retry_count += 1
    time.sleep(30)
```

Rastrear uma alteração de configuração

É possível solicitar uma alteração de configuração de cada vez ou agrupar várias alterações em uma única solicitação. Use os campos Status de processamento de domínio e Status de alteração de configuração no console para rastrear as alterações de configuração. Espere o status do domínio se tornar `Active` antes de solicitar qualquer outras alterações.

Um domínio pode ter os seguintes status de processamento:

- `Active`: nenhuma alteração de configuração está em andamento. Você pode enviar uma nova solicitação de alteração de configuração.
- `Creating`: o domínio está sendo criado.

- `Modifying`: alterações de configuração, como a adição de novos nós de dados, EBS, gp3, provisionamento de IOPS ou configuração de chaves do KMS, estão em andamento.
- `Upgrading engine version`: uma atualização de versão do mecanismo está em andamento.
- `Updating service software`: uma atualização de software do serviço está em andamento.
- `Deleting`: o domínio está sendo excluído.
- `Isolated`: o domínio está suspenso.

Um domínio pode ter os seguintes status de alteração de configuração:

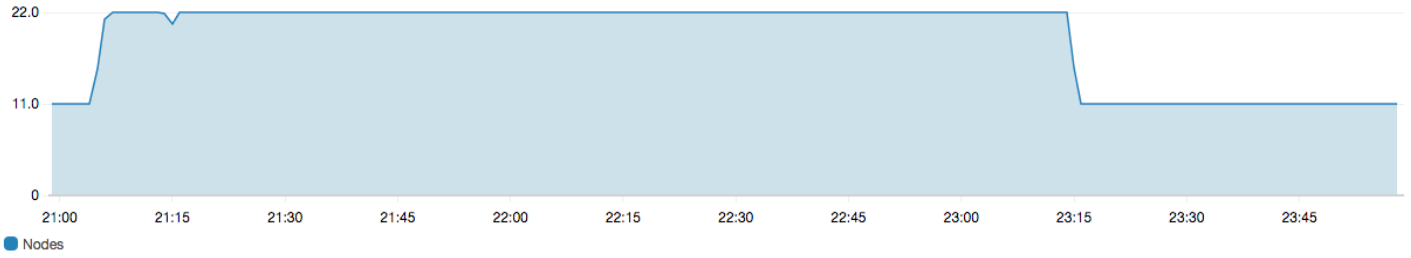
- `Pending`: uma solicitação de alteração de configuração foi enviada.
- `Initializing`: o serviço está inicializando alteração de configuração.
- `Validating`: o serviço está validando as alterações solicitadas e os recursos necessários.
- `Awaiting user inputs`: o serviço espera que alterações de configuração, como alteração de tipo de instância, continuem. Você pode editar as alterações de configuração.
- `Applying changes`: o serviço está aplicando as alterações de configuração solicitadas.
- `Cancelled`: a alteração de configuração foi cancelada. Escolher Cancelar reverte todas as alterações.
- `Completed`: as alterações de configuração solicitadas foram concluídas com êxito.
- `Validation failed`: as alterações de configuração solicitadas não puderam ser concluídas. Nenhuma alteração de configuração foi aplicada.

> **Note**
>
> As falhas de validação podem ser o resultado de índices vermelhos presentes no domínio, indisponibilidade de um tipo de instância escolhido ou falta de espaço em disco. Para obter uma lista de erros de validação, consulte the section called “Solução de problemas de erros de validação”. Durante um evento de falha na validação, você pode cancelar, tentar novamente ou editar as alterações de configuração.

Quando as alterações de configuração são concluídas, o estado do domínio volta a ser `Active`.

Você pode analisar a integridade do cluster e CloudWatch as métricas da Amazon e ver que o número de nós no cluster aumenta temporariamente, geralmente dobrando, enquanto a atualização

do domínio ocorre. No exemplo a seguir, você pode ver o número de nós que dobram de 11 para 22 durante uma alteração de configuração e que retornam para 11 quando a atualização é concluída.



Esse aumento temporário pode sobrecarregar os nós principais dedicados do cluster, que repentinamente poderão ter muito mais nós para gerenciar. Também pode aumentar as latências de pesquisa e indexação à medida que o OpenSearch Service copia dados do cluster antigo para o novo. É importante manter capacidade suficiente no cluster para lidar com a sobrecarga associada a essas blue/green implantações.

> **Important**
>
> Não há nenhuma cobrança adicional nas alterações de configuração e na manutenção do serviço. Você será cobrado apenas pelo número de nós que solicitar para seu cluster. Para obter detalhes, consulte the section called “Cobranças para alterações de configuração”.

Para evitar a sobrecarga de nós mestres dedicados, você pode monitorar o uso com as CloudWatch métricas da Amazon. Para obter os valores máximos recomendados, consulte the section called “ CloudWatch Alarmes recomendados”.

Etapas de uma alteração de configuração

Depois de iniciar uma alteração na configuração, o OpenSearch Service passa por uma série de etapas para atualizar seu domínio. Você pode visualizar o progresso da alteração de configuração em Status da alteração de configuração no console. As etapas exatas para a realização de uma atualização depende do tipo de alteração que você está fazendo. Você também pode monitorar uma alteração de configuração usando a operação da API DescribeDomainChangeProgress.

A seguir, estão as possíveis etapas de uma atualização durante uma alteração de configuração:

Nome da etapa	Description
Validação	Validação se o domínio está qualificado para uma atualização e identificação de problemas de validação, se necessário.
Criação de um novo ambiente	Concluindo os pré-requisitos necessários e criando os recursos necessários para iniciar a implantação. blue/green
Provisionamento de novos nós	Criando um novo conjunto de instâncias no novo ambiente.
Roteamento de tráfego em novos nós	Redirecionamento do tráfego para os nós de dados recém-criados.
Roteamento de tráfego em nós antigos	Desabilitação tráfego em

Nome da etapa	Description
	nós de dados antigos.
Preparação dos nós para remoção	Preparação para a remoção de nós. Esta etapa só ocorre quando você reduz a escala do seu domínio (por exemplo, de 8 nós para 6 nós).
Cópia de fragmentos para novos nós	Transferência de fragmentos dos nós antigos para os novos nós.
Encerramento de nós	Encerramento e exclusão de nós antigos após a remoção dos fragmentos.

Nome da etapa	Description
Exclusão de recursos mais antigos	Exclusão de recursos associados ao ambiente antigo (por exemplo, o balanceador de carga).
Atualização dinâmica	Exibido quando a atualização não exige uma blue/green implantação e pode ser aplicada dinamicamente.
Aplicando alterações dedicadas relacionadas à entidade principal	Exibido quando o tipo ou a contagem de instâncias principais dedicadas são alterados.
Aplicar alterações relacionadas ao volume	Exibido quando o tamanho, o tipo, o IOPS e o throughput do volume são alterados.

Impacto no desempenho das blue/green implantações

Durante a blue/green implantação, seu cluster do Amazon OpenSearch Service está disponível para receber solicitações de pesquisa e indexação. No entanto, você pode enfrentar os seguintes problemas de desempenho:

- Aumento temporário no uso nos nós principais, pois os clusters têm mais nós para gerenciar.
- Maior latência de pesquisa e indexação à medida que o OpenSearch Serviço copia dados de nós antigos para novos nós.
- Aumento das rejeições de solicitações recebidas à medida que a carga do cluster aumenta durante blue/green as implantações.
- Para evitar problemas de latência e rejeições de solicitações, você deve executar blue/green implantações quando o cluster estiver íntegro e houver pouco tráfego de rede.

Cobranças para alterações de configuração

Se você alterar a configuração de um domínio, o OpenSearch Service criará um novo cluster conforme descrito emthe section called “Alterações de configuração”. Durante a migração do antigo para o novo, você é cobrado pelos seguintes encargos:

- Se você alterar o tipo de instância, será cobrado por ambos os clusters para a primeira hora. Após a primeira hora, você será cobrado apenas pelo novo cluster. Os volumes do EBS não são cobrados duas vezes porque fazem parte do cluster. Portanto, o faturamento segue o faturamento da instância.

  Exemplo: Você altera a configuração de três instâncias `m3.xlarge` para quatro instâncias `m4.large`. Na primeira hora, você é cobrado por ambos os clusters (3 * `m3.xlarge` + 4 * `m4.large`). Após a primeira hora, você será cobrado apenas pelo novo cluster (4 * `m4.large`).
- Se você não alterar o tipo de instância, será cobrado apenas pelo cluster maior para a primeira hora. Após a primeira hora, você será cobrado apenas pelo novo cluster.

  Exemplo: Você altera a configuração de seis instâncias `m3.xlarge` para três instâncias `m3.xlarge`. Para a primeira hora, você será cobrado pelo cluster maior (6 * `m3.xlarge`). Após a primeira hora, você será cobrado apenas pelo novo cluster (3 * `m3.xlarge`).

Solução de problemas de erros de validação

Quando você inicia uma alteração na configuração ou realiza uma OpenSearch atualização de versão do Elasticsearch, o OpenSearch Service primeiro executa uma série de verificações de validação para garantir que seu domínio esteja qualificado para uma atualização. Se alguma dessas verificações falhar, você receberá uma notificação no console contendo os problemas específicos que deverão ser corrigidos antes da atualização do domínio.

OpenSearch O serviço isola seu domínio se ele permanecer inutilizável por mais de 60 dias. Enviaremos lembretes de notificação para resolver esses problemas. Se você não corrigir os problemas dentro desse período, o OpenSearch Serviço excluirá seu domínio e seus dados.

A tabela a seguir lista os possíveis problemas de domínio que o OpenSearch Serviço pode surgir e as etapas para resolvê-los.

Problema	Código de erro	Etapas de solução de problemas
Grupo de segurança não encontrado	`SecurityGroupNotFound`	O grupo de segurança associado ao seu domínio de OpenSearch serviço não existe. Para resolver esse problema, crie um grupo de segurança com o nome especificado.
Sub-rede não encontrada	`SubnetNotFound`	A sub-rede associada ao seu domínio OpenSearch de serviço não existe. Para resolver esse problema, crie uma sub-rede na sua VPC.
Função vinculada ao serviço não configurada	`SLRNotConfigured`	A função vinculada ao OpenSearch serviço para Serviço não está configurada. A função vinculada ao serviço é predefinida pelo OpenSearch Serviço e inclui todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome. Se a função não existir, talvez seja necessário criá-la manualmente.
Não há endereços IP suficientes	`InsufficientFreeIPsForSubnets`	Uma ou mais sub-redes da VPC não têm endereços IP suficientes para atualizar seu domínio. Para calcular quantos endereços IP são necessários, consulte the section called “Reserva de endereços IP em uma sub-rede da VPC”.

Problema	Código de erro	Etapas de solução de problemas
O grupo de usuários do Cognito não existe	`CognitoUserPoolNotFound`	OpenSearch O serviço não consegue encontrar o grupo de usuários do Amazon Cognito. Confirme se você criou um e se tem o ID correto. Para encontrar o ID, você pode usar o console do Amazon Cognito ou o seguinte comando da AWS CLI : `aws cognito-idp list-user-pools --max-results 60 --region us-east-1`
O grupo de identidades do Cognito não existe	`CognitoIdentityPoolNotFound`	OpenSearch O serviço não consegue encontrar o pool de identidade do Cognito. Confirme se você criou um e se tem o ID correto. Para encontrar o ID, você pode usar o console do Amazon Cognito ou o seguinte comando da AWS CLI : `aws cognito-identity list-identity-pools --max-results 60 --region us-east-1`
Domínio do Cognito não encontrado para grupo de usuários	`CognitoDomainNotFound`	O grupo de usuários não tem um nome de domínio. Você pode configurar um usando o console do Amazon Cognito ou o seguinte comando: AWS CLI `aws cognito-idp create-user-pool-domain --domain my-domain --user-pool-id id`
Função do Cognito não configurada	`CognitoRoleNotConfigured`	A função do IAM que OpenSearch concede permissão ao Serviço para configurar os grupos de usuários e identidades do Amazon Cognito e usá-los para autenticação não está configurada. Configure a função com um conjunto de permissões e uma relação de confiança apropriados. Você pode usar o console, que cria a CognitoAccessForAmazonOpenSearchfunção padrão para você, ou pode configurar manualmente uma função usando o AWS CLI ou o AWS SDK.

Problema	Código de erro	Etapas de solução de problemas
Não é possível descrever o grupo de usuários	`UserPoolNotDescribable`	A função especificada do Amazon Cognito não tem permissão para descrever o grupo de usuários associado ao seu domínio. Verifique se a política de permissões da função permite a ação `cognito-identity:DescribeUserPool`. Consulte the section called "Sobre a CognitoAccessForAmazonOpenSearch função" para ver a política de permissões completa.
Não é possível descrever o grupo de identidades	`IdentityPoolNotDescribable`	A função especificada do Amazon Cognito não tem permissão para descrever o grupo de identidades associado ao seu domínio. Verifique se a política de permissões da função permite a ação `cognito-identity:DescribeIdentityPool`. Consulte the section called "Sobre a CognitoAccessForAmazonOpenSearch função" para ver a política de permissões completa.
Não é possível descrever os grupos usuários e de identidades	`CognitoPoolsNotDescribable`	A função especificada do Amazon Cognito não tem permissão para descrever os grupos de usuários e de identidades associados ao seu domínio. Verifique se a política de permissões da função permite as ações `cognito-identity:DescribeIdentityPool` e `cognito-identity:DescribeUserPool`. Consulte the section called "Sobre a CognitoAccessForAmazonOpenSearch função" para ver a política de permissões completa.
A chave do KMS não está habilitada	`KMSKeyNotEnabled`	A chave AWS Key Management Service (AWS KMS) usada para criptografar seu domínio está desativada. Reative a chave imediatamente.

Problema	Código de erro	Etapas de solução de problemas
O certificado personalizado não está no estado ISSUED (EMITIDO)	`InvalidCertificate`	Se o seu domínio usa um endpoint personalizado, você o protege gerando um certificado SSL no AWS Certificate Manager (ACM) ou importando um de sua preferência. O status do certificado deve ser Emitido. Ao receber esse erro, verifique o status do certificado no console do ACM. Se o status for Expired (Expirado), Failed (Com falha), Inactive (Inativo) ou Pending validation (Validação pendente), consulte a documentação de solução de problemas do ACM para resolver o problema.
Capacidade insuficiente para iniciar o tipo de instância escolhido	`InsufficientInstanceCapacity`	A capacidade do tipo de instância solicitada não está disponível. Por exemplo, você pode ter solicitado cinco `i3.16xlarge.search` nós, mas o OpenSearch Serviço não tem `i3.16xlarge.search` hosts suficientes disponíveis, então a solicitação não pode ser atendida. Verifique os tipos de instância compatíveis em OpenSearch Service e escolha um tipo de instância diferente.
Índices vermelhos no cluster	`RedCluster`	Um ou mais índices em seu cluster têm um status vermelho, o que leva a um status geral de cluster vermelho. Para solucionar e corrigir esse problema, consulte the section called “Status de cluster vermelho”.
Disjuntor de memória, excesso de solicitações	`TooManyRequests`	Há muitas solicitações de pesquisa e gravação em seu domínio, então o OpenSearch Serviço não pode atualizar sua configuração. É possível reduzir o número de solicitações, aumentar instâncias na vertical até 64 GiB de RAM ou aumentar a escala na horizontal adicionando instâncias.

Problema	Código de erro	Etapas de solução de problemas
A nova configuração não pode acomodar os dados (pouco espaço em disco)	`InsufficientStorageCapacity`	O tamanho de armazenamento configurado não é capaz de acomodar todos os dados no seu domínio. Para resolver esse problema, escolha um volume maior, exclua índices não utilizados ou aumente o número de nós no cluster para liberar espaço em disco imediatamente.

Fragmentos fixados em nós específicos	`ShardMovementBlocked`	Um ou mais índices em seu domínio estão anexados a nós específicos e não podem ser reatribuídos. Isso provavelmente aconteceu porque você configurou a filtragem de alocação de fragmentos, que permite especificar quais nós têm permissão para hospedar os fragmentos de um índice específico.
		
		Para resolver esse problema, remova os filtros de alocação de fragmentos de todos os índices afetados:
		
		`PUT my-index/_settings`
		`{`
		`  "settings": {`
		`  "index.routing.allocation.require._name": null`
		`  }`
		`}`

Problema	Código de erro	Etapas de solução de problemas
A nova configuração não pode conter todos os fragmentos (contagem de fragmentos)	`TooManyShards`	A contagem de fragmentos em seu domínio é muito alta, o que impede que o OpenSearch Serviço os mova para a nova configuração. Para resolver esse problema, dimensione seu domínio horizontalmente adicionando nós do mesmo tipo de configuração que os nós de cluster atuais. Observe que o tamanho máximo do volume do EBS depende do tipo de instância do nó. Para evitar esse problema no futuro, consulte the section called "Como escolher o número de fragmentos" e defina uma estratégia de fragmentação que seja apropriada para o seu caso de uso.
A sub-rede associada ao seu domínio não suporta endereços IPv4	`ResultCodeIPv4BlockNotExists`	Para resolver esse problema, crie uma sub-rede ou atualize a sub-rede existente na sua VPC, de acordo com o tipo de endereço IP configurado do domínio. Se seu domínio usa um IPv4único tipo de endereço, use uma IPv4 sub-rede somente. Se o domínio usa o modo de pilha dupla, use uma sub-rede de pilha dupla.
A sub-rede associada ao seu domínio não suporta endereços IPv6	`ResultCodeIPv6BlockNotExists`	Para resolver esse problema, crie uma sub-rede ou atualize a sub-rede existente na sua VPC, de acordo com o tipo de endereço IP configurado do domínio. Se seu domínio usa um IPv4único tipo de endereço, use uma IPv4 sub-rede somente. Se o domínio usa o modo de pilha dupla, use uma sub-rede de pilha dupla.

Atualizações do software de serviço no Amazon OpenSearch Service

> **Note**
>
> Consulte as notas de versão para obter explicações sobre as alterações e adições feitas em cada atualização principal do software do serviço (sem patch).

O Amazon OpenSearch Service lança regularmente atualizações de software de serviços que adicionam recursos ou melhoram seus domínios. O painel Notificações no console é a maneira mais fácil de ver se uma atualização está disponível ou verificar o status de uma atualização. Cada notificação inclui detalhes sobre a atualização do software de serviço. Todas as atualizações de software de serviço usam implantações azul/verde para minimizar o tempo de inatividade.

As atualizações do software do OpenSearch serviço são diferentes das atualizações da versão do serviço. Para obter informações sobre a atualização para uma versão posterior do OpenSearch Serviço, consultethe section called “Atualização de domínios”.

OpenSearch O serviço exige que você aplique as atualizações de software de serviço necessárias dentro de 30 dias de sua disponibilidade. Essas atualizações são essenciais para manter a conformidade de segurança.

Se você não aplicar as atualizações necessárias dentro do período de 30 dias, receberá notificações de lembrete a cada 15 dias durante 30 dias. Após esse período sem conformidade, seu domínio será isolado com os seguintes efeitos:

- Todo o acesso de rede ao seu domínio é removido
- O status do domínio muda para isolado
- O domínio permanece inutilizável até que você aplique as atualizações necessárias

Durante o isolamento, você continuará recebendo notificações de lembretes a cada 15 dias por 60 dias. Se você não aplicar as atualizações necessárias dentro desse período, seu domínio do OpenSearch Serviço e todos os dados associados serão excluídos permanentemente. Para obter mais informações, consulte Solução de problemas de erros de validação.

Atualizações opcionais x obrigatórias

OpenSearch O serviço tem duas grandes categorias de atualizações de software de serviço:

Atualizações opcionais

As atualizações opcionais do software de serviço geralmente incluem aprimoramentos e suporte para novos atributos ou funcionalidades. As atualizações opcionais não são aplicadas aos seus domínios e não há um prazo fixo para instalá-las. A disponibilidade da atualização é comunicada por e-mail e uma notificação no console. Você pode optar por aplicar a atualização imediatamente ou reagendá-la para uma data e hora mais convenientes. Você também pode programá-la durante a janela fora do horário de pico do domínio. A maioria das atualizações de software é opcional.

Independentemente de você agendar ou não uma atualização, se você fizer uma alteração no domínio que cause uma implantação azul/verde, o OpenSearch Service atualizará automaticamente o software do serviço para você.

Você pode configurar seu domínio para aplicar automaticamente atualizações opcionais fora do horário de pico. Quando essa opção está ativada, o OpenSearch Serviço espera pelo menos 13 dias a partir do momento em que uma atualização opcional está disponível e, em seguida, agenda a atualização após sete dias. Você recebe uma notificação do console quando a atualização é agendada e pode optar por reagendá-la para uma data posterior.

Para ativar as atualizações automáticas de software, selecione Habilitar atualização automática de software ao criar ou atualizar seu domínio. Para definir a mesma configuração usando o AWS CLI, `--software-update-options` defina como `true` quando você cria ou atualiza seu domínio.

Atualizações necessárias

As atualizações obrigatórias de software de serviço geralmente incluem correções críticas de segurança ou outras atualizações indispensáveis para garantir a integridade e a funcionalidade contínuas do seu domínio. Exemplos de atualizações necessárias são as vulnerabilidades e exposições comuns do Log4j (CVEs) e a aplicação do serviço de metadados de instância versão 2 (). IMDSv2 O número de atualizações obrigatórias em um ano geralmente é menor que três.

OpenSearch O serviço agenda automaticamente essas atualizações e notifica você sete dias antes da atualização agendada por e-mail e uma notificação no console. Você pode optar por aplicar a atualização imediatamente ou reprogramá-la para uma data e hora mais convenientes dentro do prazo permitido. Você também pode programá-la durante a próxima janela fora do horário de pico do domínio. Se você não tomar nenhuma ação em relação a uma atualização necessária e não

fizer nenhuma alteração no domínio que cause uma blue/green implantação, o OpenSearch Serviço poderá iniciar a atualização a qualquer momento além do prazo especificado (normalmente 14 dias a partir da disponibilidade), dentro da janela fora de pico do domínio.

Independentemente de quando a atualização está agendada, se você fizer uma alteração no domínio que cause uma implantação azul/verde, o OpenSearch Service atualizará automaticamente seu domínio para você.

Atualizações de patch

Versões de software de serviço que terminam em “-P” e um número, como R20211203-, são lançamentos de patches. *P4* É provável que os patches incluam melhorias de performance, pequenas correções de bugs e correções de segurança ou melhorias de postura. As versões de patch não incluem novos atributos ou alterações significativas e geralmente não têm um impacto direto ou perceptível para os usuários. A notificação do software de serviço informa se a versão de um patch é opcional ou obrigatória.

Considerações

Considere o seguinte ao decidir se deseja atualizar seu domínio:

- A atualização manual do seu domínio permite aproveitar os novos recursos mais rapidamente. Quando você escolhe Atualizar, o OpenSearch Serviço coloca a solicitação em uma fila e inicia a atualização quando tiver tempo.
- Quando você inicia uma atualização do software do OpenSearch serviço, o Serviço envia uma notificação quando a atualização é iniciada e concluída.
- As atualizações de software usam blue/green implantações para minimizar o tempo de inatividade. As atualizações podem sobrecarregar temporariamente os nós principais dedicados de um cluster. Por isso, certifique-se de manter capacidade suficiente para lidar com a sobrecarga associada.
- Normalmente, as atualizações são concluídas em minutos, mas também podem levar várias horas ou até dias se o sistema estiver lidando com muita carga. Considere atualizar seu domínio durante a janela fora do horário de pico para evitar longos períodos de atualização.

Iniciar uma atualização do software de serviço

Você pode solicitar uma atualização do software de OpenSearch serviço por meio do console de serviço AWS CLI, do ou de um dos SDKs.

Console

Solicitar uma atualização de software de serviço

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Escolha Ações, Atualizar e selecione uma das seguintes opções:

   - Aplicar a atualização agora: programa a ação para acontecer imediatamente, se houver capacidade disponível. Se a capacidade não estiver disponível, outros slots de horários disponíveis serão sugeridos.
   - Agendar fora do horário de pico: disponível somente se a janela fora do horário de pico estiver ativada para o domínio. Agenda a atualização para ocorrer durante a janela fora do horário de pico configurada do domínio. Não há garantia de que a atualização ocorrerá durante a próxima janela imediata. Dependendo da capacidade, isso pode acontecer nos dias subsequentes. Para saber mais, consulte the section called “Janelas fora do horário de pico”.
   - Agendar para data e hora específicas agenda a atualização para ocorrer em uma data e hora específicas. Se o horário especificado não estiver disponível por motivos de capacidade, você poderá selecionar um slot de horário diferente.

   Se você agendar a atualização para uma data posterior (dentro ou fora da janela de horário de pico do domínio), poderá reagendá-la a qualquer momento. Para instruções, consulte the section called “Ações de reagendamento”.
4. Selecione a opção Confirmar.

AWS CLI

Envie uma start-service-software-update AWS CLI solicitação para iniciar uma atualização do software de serviço. Este exemplo adiciona a atualização à fila imediatamente:

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "NOW"
```

Resposta:

```
{
```

```
    "ServiceSoftwareOptions": {
        "CurrentVersion": "R20220928-P1",
        "NewVersion": "R20220928-P2",
        "UpdateAvailable": true,
        "Cancellable": true,
        "UpdateStatus": "PENDING_UPDATE",
        "Description": "",
        "AutomatedUpdateDate": "1969-12-31T16:00:00-08:00",
        "OptionalDeployment": true
    }
}
```

> **Tip**
>
> Depois de solicitar uma atualização, você tem um período de tempo limitado para cancelá-la. A duração desse `PENDING_UPDATE` estado pode variar muito e depende de você Região da AWS e do número de atualizações simultâneas que o OpenSearch Serviço está executando. Para cancelar uma atualização, use o console ou o `cancel-service-software-update` AWS CLI comando.

Se a solicitação falhar com uma `BaseException`, isso significa que o horário especificado não está disponível por motivos de capacidade e você deve especificar um horário diferente. OpenSearch O serviço fornece sugestões alternativas de slots disponíveis na resposta.

AWS SDKs

Esse exemplo de script Python usa os métodos describe_domain e start_service_software_update do AWS SDK para Python (Boto3) para verificar se um domínio está qualificado para uma atualização do software de serviço e, em caso afirmativo, inicia a atualização. Você deve fornecer um valor para `domain_name`:

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
```

```
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)

domain_name = ''  # The name of the domain to check and update

client = boto3.client('opensearch', config=my_config)


def getUpdateStatus(client):
    """Determines whether the domain is eligible for an update"""
    response = client.describe_domain(
        DomainName=domain_name
    )
    sso = response['DomainStatus']['ServiceSoftwareOptions']
    if sso['UpdateStatus'] == 'ELIGIBLE':
        print('Domain [' + domain_name + '] is eligible for a service software update
 from version ' +
              sso['CurrentVersion'] + ' to version ' + sso['NewVersion'])
        updateDomain(client)
    else:
        print('Domain is not eligible for an update at this time.')


def updateDomain(client):
    """Starts a service software update for the eligible domain"""
    response = client.start_service_software_update(
        DomainName=domain_name
    )
    print('Updating domain [' + domain_name + '] to version ' +
          response['ServiceSoftwareOptions']['NewVersion'] + '...')
    waitForUpdate(client)


def waitForUpdate(client):
    """Waits for the domain to finish updating"""
    response = client.describe_domain(
        DomainName=domain_name
    )
    status = response['DomainStatus']['ServiceSoftwareOptions']['UpdateStatus']
    if status == 'PENDING_UPDATE' or status == 'IN_PROGRESS':
        time.sleep(30)
        waitForUpdate(client)
    elif status == 'COMPLETED':
```

```
        print('Domain [' + domain_name +
              '] successfully updated to the latest software version')
    else:
        print('Domain is not currently being updated.')

def main():
    getUpdateStatus(client)
```

Agendamento de atualizações do software fora do horário de pico

Cada domínio do OpenSearch Serviço criado após 16 de fevereiro de 2023 tem uma janela diária de 10 horas entre 22h e 8h, horário local, que consideramos a janela fora de pico. OpenSearch O serviço usa essa janela para agendar atualizações do software do serviço para o domínio. As atualizações fora do pico ajudam a minimizar a sobrecarga nos nós principais dedicados de um cluster durante períodos de maior tráfego. OpenSearch O serviço não pode iniciar atualizações fora dessa janela de 10 horas sem o seu consentimento.

- Para atualizações opcionais, o OpenSearch Serviço notifica você sobre a disponibilidade da atualização e solicita que você agende a atualização durante um próximo período fora de pico.
- Para as atualizações necessárias, o OpenSearch Serviço agenda automaticamente a atualização durante uma próxima janela fora de pico e notifica você com três dias de antecedência. Você pode reagendar a atualização (dentro ou fora da janela de pico), mas somente dentro do prazo necessário para que a atualização seja concluída.

Para cada domínio, você pode optar por substituir o horário de início padrão das 22h por um horário personalizado. Para instruções, consulte the section called “Configurar uma janela personalizada fora do horário de pico”.

Console

Como agendar uma atualização durante uma próxima janela fora do horário de pico

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Escolha Ações, Atualizar.
4. Selecione Agendar em uma janela fora do horário de pico.
5. Selecione a opção Confirmar.

Você pode visualizar a ação agendada na guia Janela fora do horário de pico e reagendá-la a qualquer momento. Consulte the section called “Exibir ações agendadas”.

CLI

Para agendar uma atualização durante uma próxima janela fora de pico usando o AWS CLI, envie uma StartServiceSoftwareUpdatesolicitação e especifique `OFF_PEAK_WINDOW` o `--schedule-at` parâmetro:

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "OFF_PEAK_WINDOW"
```

Monitoramento das atualizações de software de serviço

OpenSearch O serviço envia uma notificação quando uma atualização do software do serviço está disponível, é necessária, iniciada, concluída ou falha. Você pode ver essas notificações no painel Notificações do console OpenSearch de serviço. A gravidade da notificação será `Informational` se a atualização for opcional e `High` se ela for necessária.

OpenSearch O serviço também envia eventos de software de serviço para a Amazon EventBridge. Você pode usar EventBridge para configurar regras que enviam um e-mail ou executam uma ação específica quando um evento é recebido. Para ver um exemplo de apresentação, consultethe section called “Tutorial: Envio de alertas do SNS para atualizações disponíveis”.

Para ver o formato de cada evento de software de serviço enviado para a Amazon EventBridge, consultethe section called “Eventos de atualização de software de serviço”.

Quando os domínios não são elegíveis para uma atualização

Seu domínio poderá ser inelegível para um serviço de atualização de software se ele estiver em qualquer um dos seguintes estados:

Estado	Description
Domínio no processamento	O domínio está no meio de uma mudança de configuração. Verifique a qualificação da atualização após a conclusão da operação.

Estado	Description
Status de cluster vermelho	Um ou mais índices no cluster estão vermelhos. Para obter etapas sobre a solução de problemas, consulte the section called “Status de cluster vermelho”.
Alta taxa de erros	O OpenSearch cluster está retornando um grande número de 5 erros xx ao tentar processar solicitações. Geralmente, esse problema é resultado de muitas solicitações de leitura ou gravação simultâneas. Considere reduzir o tráfego para o cluster ou dimensionar seu domínio.
Cérebro dividido	Cérebro dividido significa que seu OpenSearch cluster tem mais de um nó principal e se dividiu em dois clusters que nunca se reunirão sozinhos. Você pode evitar dividir o cérebro usando o número recomendado de nós principais dedicados. Para ajudar na recuperação do cérebro dividido, entre em contato com Suporte.
Problema de integração do Amazon Cognito	Seu domínio usa autenticação para OpenSearch painéis, e o OpenSearch Service não consegue encontrar um ou mais recursos do Amazon Cognito. Este problema normalmente ocorre quando o grupo de usuários do Amazon Cognito está ausente. Para corrigir o problema, recrie o recurso ausente e configure o domínio do OpenSearch serviço para usá-lo.
Outro problema de serviço do	Problemas com o OpenSearch serviço em si podem fazer com que seu domínio seja exibido como inelegível para uma atualização. Se nenhuma das condições anteriores se aplicar ao seu domínio e o problema persistir por mais de um dia, entre em contato com o Suporte.

Definindo períodos fora de pico para o Amazon Service OpenSearch

Ao criar um domínio do Amazon OpenSearch Service, você define uma janela diária de 10 horas que é considerada fora do horário de pico. OpenSearch O serviço usa essa janela para agendar atualizações de software de serviço e otimizações de ajuste automático que exigem uma implantação azul/verde durante períodos de tráfego comparativamente mais baixos, sempre que

possível. Blue/green refere-se ao processo de criação de um novo ambiente para atualizações de domínio e roteamento de usuários para o novo ambiente após a conclusão dessas atualizações.

Embora blue/green as implantações não causem interrupções, para minimizar qualquer impacto potencial no desempenho enquanto os recursos estão sendo consumidos para uma blue/green implantação, recomendamos que você agende essas implantações durante a janela fora de pico configurada do domínio. Atualizações como substituições de nós ou que precisem ser implantadas no domínio imediatamente não usam a janela fora do horário de pico.

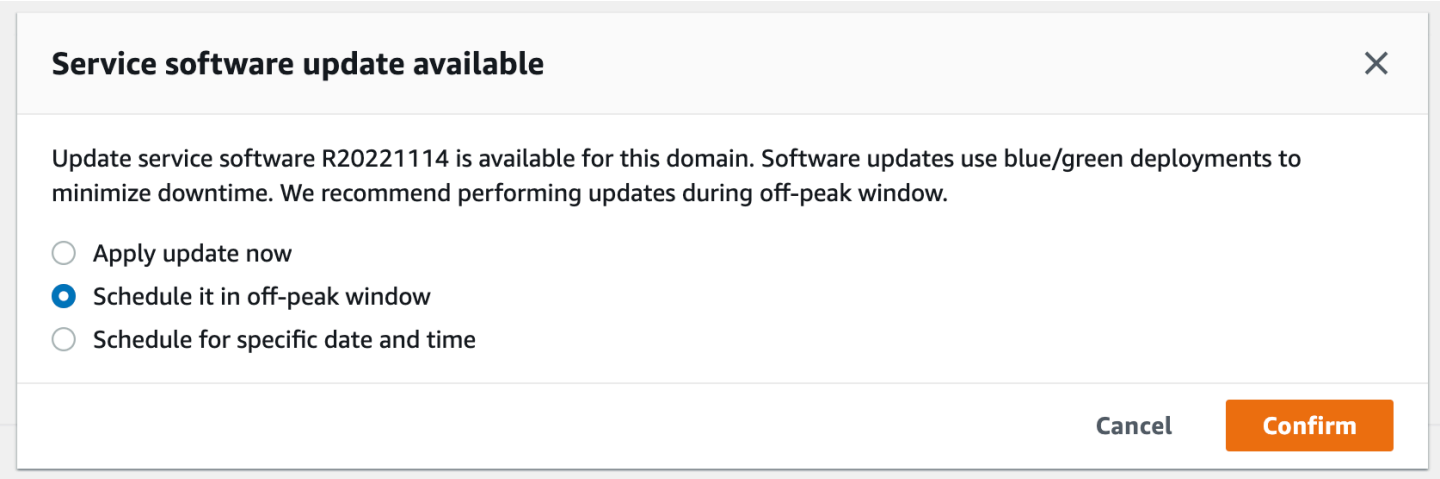
Você pode modificar a hora de início da janela fora do horário de pico, mas não pode modificar o comprimento da janela.

> **Note**
>
> As janelas fora do horário de pico foram introduzidas em 16 de fevereiro de 2023. Todos os domínios criados antes dessa data têm a janela fora do horário de pico desativada por padrão. Você deve ativar e configurar manualmente a janela fora do horário de pico para esses domínios. Todos os domínios criados após essa data terão a janela fora do horário de pico ativada por padrão. Você não pode desativar a janela fora do horário de pico de um domínio depois que ela for ativada.

Atualizações de software de serviço fora do horário de pico

OpenSearch O serviço tem duas grandes categorias de atualizações de software de serviço — opcionais e obrigatórias. Ambos os tipos exigem blue/green implantações. As atualizações opcionais não são aplicadas em seus domínios, enquanto as atualizações obrigatórias são instaladas automaticamente se você não realizar nenhuma ação antes do prazo especificado (normalmente duas semanas após a disponibilidade). Para saber mais, consulte the section called “Atualizações opcionais x obrigatórias”.

Ao iniciar uma atualização opcional, você tem a opção de aplicá-la imediatamente, programá-la para uma janela subsequente fora do horário de pico ou especificar uma data e hora personalizadas.



Para as atualizações necessárias, o OpenSearch Serviço agenda automaticamente uma data e hora fora do horário de pico para realizar a atualização. Você recebe uma notificação três dias antes da atualização agendada e pode optar por reagendá-la para uma data e hora posteriores dentro do período de implantação necessário. Para instruções, consulte the section called “Ações de reagendamento”.

Otimizações do Auto-Tune fora do horário de pico

Anteriormente, o Auto-Tune usava janelas de manutenção para programar mudanças que exigiam uma blue/green implantação. Os domínios que já tinham o ajuste automático e as janelas de manutenção ativadas antes da introdução das janelas fora do horário de pico continuarão usando janelas de manutenção para essas atualizações, a menos que você os migre para usar a janela fora do horário de pico.

Recomendamos que você migre seus domínios para usar a janela fora do horário de pico, pois ela é usada para agendar outras atividades no domínio, como atualizações de software de serviço. Para instruções, consulte the section called “Migração das janelas de manutenção do Auto-Tune”. Você não pode voltar a usar as janelas de manutenção depois de migrar seu domínio para a janela fora do horário de pico.

Todos os domínios criados após 16 de fevereiro de 2023 usarão a janela fora do pico, em vez das janelas de manutenção legadas, para agendar blue/green implantações. Você não pode desativar a janela fora do horário de pico de um domínio. Para obter uma lista de otimizações do Auto-Tune que exigem implantações azul/verde, consulte the section called “Tipos de alterações”.

Ativar a janela fora do horário de pico

Todos os domínios criados antes de 16 de fevereiro de 2023 (quando os períodos fora do horário de pico foram introduzidos) têm o atributo desativado por padrão. Você deve habilitá-lo manualmente para esses domínios. Você não pode desativar a janela fora do horário de pico depois de ativada.

Console

Para ativar a janela fora do horário de pico de um domínio

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Navegue até a guia Janela fora do horário de pico e escolha Editar.
4. Especifique o horário de início customizado em Tempo Universal Coordenado (UTC). Por exemplo, para configurar um horário de início às 23h30 na região Oeste dos EUA (Oregon), especifique 07h30
5. Escolha Salvar alterações.

CLI

Para modificar a janela fora do pico usando o AWS CLI, envie uma UpdateDomainConfigsolicitação:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'Enabled=true,
 OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Se você não especificar um horário de início de janela personalizado, o padrão será 0h UTC.

Configurar uma janela personalizada fora do horário de pico

Você especifica uma janela personalizada fora do horário de pico para o domínio de acordo com o fuso horário UTC (Tempo Universal Coordenado). Por exemplo, se você quiser que o período comece às 23h para um domínio na região leste dos EUA (Norte da Virgínia), você deverá especificar às 04h UTC.

Console

Para modificar a janela fora do horário de pico de um domínio

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Navegue até a guia Janela fora do horário de pico. Você pode ver a janela fora do horário de pico configurada, além de uma lista das próximas ações agendadas para o domínio.
4. Escolha Editar e especifique um novo horário de início em UTC. Por exemplo, para configurar um horário de início às 21h na região leste dos EUA (Norte da Virginia), especifique 02h UCT.
5. Escolha Salvar alterações.

CLI

Para configurar uma janela personalizada fora do horário de pico usando o AWS CLI, envie uma UpdateDomainConfigsolicitação e especifique a hora e o minuto no formato de 24 horas.

Por exemplo, a solicitação a seguir altera o horário de início da janela para 2h da manhã UTC:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Se você não especificar o horário de início da janela, o padrão é 22h, horário local, na Região da AWS aonde o domínio foi criado.

Exibir ações agendadas

Você pode ver todas as ações agendadas, em andamento ou pendentes atualmente para cada um dos seus domínios. As ações podem ter uma severidade de `HIGH`, `MEDIUM` e `LOW`.

As ações podem ter os seguintes status:

- `Pending update`: a ação está na fila para ser processada.
- `In progress`: a ação está em andamento.
- `Failed` – a operação não foi concluída.
- `Completed` – a ação foi concluída com êxito.

- `Not eligible`: somente para atualizações de software de serviço. A atualização não pode ser continuada porque o cluster não está íntegro.
- `Eligible`: somente para atualizações de software de serviço. O domínio está qualificado para uma atualização.

Console

O console OpenSearch de serviço exibe todas as ações agendadas na configuração do domínio, junto com a gravidade e o status atual de cada ação.

Como ver ações agendadas para um domínio

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Navegue até a guia Janela fora do horário de pico.
4. Em Ações agendadas, visualize todas as ações atualmente agendadas, em andamento ou pendentes no domínio.

CLI

Para ver as ações agendadas usando o AWS CLI, envie uma ListScheduledActionssolicitação:

```
aws opensearch list-scheduled-actions \
  --domain-name my-domain
```

Resposta:

```
{
    "ScheduledActions": [
        {
            "Cancellable": true,
            "Description": "The Deployment type is : BLUE_GREEN.",
            "ID": "R20220721-P13",
            "Mandatory": false,
            "Severity": "HIGH",
            "ScheduledBy": "CUSTOMER",
            "ScheduledTime": 1.673871601E9,
            "Status": "PENDING_UPDATE",
```

```
            "Type": "SERVICE_SOFTWARE_UPDATE",
        },
        {
            "Cancellable": true,
            "Description": "Amazon Opensearch will adjust the young generation JVM
 arguments on your domain to improve performance",
            "ID": "Auto-Tune",
            "Mandatory": true,
            "Severity": "MEDIUM",
            "ScheduledBy": "SYSTEM",
            "ScheduledTime": 1.673871601E9,
            "Status": "PENDING_UPDATE",
            "Type": "JVM_HEAP_SIZE_TUNING",
        }
    ]
}
```

Ações de reagendamento

OpenSearch O serviço notifica você sobre atualizações programadas do software de serviço e otimizações do Auto-Tune. Você pode optar por aplicar a alteração imediatamente ou reprogramá-la para uma data e hora posteriores.

> **Note**
>
> OpenSearch O serviço pode agendar a ação dentro de uma hora a partir do horário selecionado. Por exemplo, se você optar por aplicar uma atualização às 17h, ela poderá acontecer entre 17h e 18h.

Console

Como reagendar uma ação

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o nome do domínio para abrir a configuração.
3. Navegue até a guia Janela fora do horário de pico.
4. Selecione Ações agendadas, escolha a ação e, depois, escola Reagendar.
5. Escolha uma das seguintes opções:

- Aplicar a atualização agora: programa a ação para acontecer imediatamente, se houver capacidade disponível. Se a capacidade não estiver disponível, outros slots de horários disponíveis serão sugeridos.
- Programar para fora do horário de pico: agenda a ação para ser iniciada durante uma próxima janela fora do horário de pico. Não há garantia de que a alteração será implementada imediatamente na próxima janela. Dependendo da capacidade, isso pode acontecer nos dias subsequentes.
- Reagendar esta atualização: permite especificar uma data e hora personalizadas para aplicar a alteração. Se o horário especificado não estiver disponível por motivos de capacidade, você poderá selecionar um slot de horário diferente.
- Cancelar atualização agendada: cancela a atualização. Essa opção só estará disponível para atualizações opcionais de software de serviço. Ela não está disponível para ações de ajuste automático ou atualizações obrigatórias de software.

6. Escolha Salvar alterações.

CLI

Para reagendar uma ação usando o AWS CLI, envie uma solicitação. UpdateScheduledAction Para recuperar o ID da ação, envie uma solicitação `ListScheduledActions`.

A solicitação a seguir reagenda uma atualização do software de serviço para uma data e hora específicas:

```
aws opensearch update-scheduled-action \
  --domain-name my-domain \
  --action-id R20220721-P13 \
  --action-type "SERVICE_SOFTWARE_UPDATE" \
  --desired-start-time 1677348395000 \
  --schedule-at TIMESTAMP
```

Resposta:

```
{
   "ScheduledAction": {
      "Cancellable": true,
      "Description": "Cluster status is updated.",
      "Id": "R20220721-P13",
```

```
      "Mandatory": false,
      "ScheduledBy": "CUSTOMER",
      "ScheduledTime": 1677348395000,
      "Severity": "HIGH",
      "Status": "PENDING_UPDATE",
      "Type": "SERVICE_SOFTWARE_UPDATE"
   }
}
```

Se a solicitação falhar com um`SlotNotAvailableException`, isso significa que o horário especificado não está disponível por motivos de capacidade e você deve especificar um horário diferente. OpenSearch O serviço fornece sugestões alternativas de slots disponíveis na resposta.

Migração das janelas de manutenção do Auto-Tune

Se um domínio foi criado antes de 16 de fevereiro de 2023, ele poderia usar janelas de manutenção para agendar otimizações de ajuste automático que exigem uma implantação. blue/green Em vez disso, você pode migrar seus domínios do Auto-Tune existentes para usar a janela fora do horário de pico.

> **Note**
>
> Você não pode voltar a usar janelas de manutenção depois de migrar seu domínio para usar janelas fora do horário de pico.

Console

Como migrar um domínio para usar a janela fora do horário de pico

1. No console do Amazon OpenSearch Service, selecione o nome do domínio para abrir sua configuração.
2. Vá até a guia Auto-Tune e escolha Editar.
3. Selecione Migrar para a janela fora do horário de pico.
4. Em Hora de início (UTC), forneça uma hora de início diária para a janela fora do horário de pico de acordo com o Horário Universal Coordenado (UTC).
5. Escolha Salvar alterações.

CLI

Para migrar de uma janela de manutenção do Auto-Tune para a janela fora de pico usando o AWS CLI, envie uma solicitação: UpdateDomainConfig

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=[]
```

A janela fora do horário de pico deve estar ativada para que você possa migrar um domínio da janela de manutenção do Auto-Tune para a janela fora do horário de pico. Você pode ativar a janela fora do horário de pico em uma solicitação separada ou na mesma solicitação. Para instruções, consulte the section called “Ativar a janela fora do horário de pico”.

Notificações no Amazon OpenSearch Service

As notificações no Amazon OpenSearch Service contêm informações importantes sobre o desempenho e a integridade de seus domínios. OpenSearch O serviço notifica você sobre atualizações de software de serviço, aprimoramentos do Auto-Tune, eventos de integridade do cluster e erros de domínio. As notificações estão disponíveis para todas as versões do OpenSearch Elasticsearch OSS.

Você pode ver as notificações no painel Notificações do console OpenSearch de serviço. Todas as notificações do OpenSearch Serviço também são exibidas na Amazon EventBridge. Para obter uma lista completa de notificações e exemplos de eventos, consulte the section called “Monitoramento de eventos”.

Conceitos básicos das notificações

As notificações são ativadas automaticamente quando você cria um domínio. Acesse o painel Notificações do console de OpenSearch serviço para monitorar e reconhecer as notificações. Cada notificação inclui informações como a hora em que foi publicada, o domínio ao qual se relaciona, um nível de gravidade e status e uma breve explicação. Você pode exibir notificações históricas por até 90 dias no console.

Depois de acessar o painel Notifications (Notificações) ou confirmar uma notificação, você pode receber uma mensagem de erro sobre não ter permissões para executar

`es:ListNotificationsV2` ou `es:UpdateNotificationStatus`. Para resolver esse problema, dê ao usuário ou função as seguintes permissões no IAM:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "es:DescribeDomain",
      "es:ListDomainNames"
    ],
    "Resource": "arn:aws:es:*:111122223333:domain/*"
  }]
}
```

O console do IAM gera um erro (“O IAM não reconhece uma ou mais ações.”) que você pode ignorar com segurança. Você também pode restringir a ação `es:UpdateNotificationStatus` a determinados domínios. Para saber mais, consulte the section called “Referência de elementos da política”.

Gravidades das notificações

As notificações no OpenSearch Serviço podem ser informativas, relacionadas a qualquer ação que você já tenha realizado ou às operações do seu domínio, ou acionáveis, que exigem que você execute ações específicas, como a aplicação de um patch de segurança obrigatório. Cada notificação tem uma gravidade associada a ela, que pode ser `Informational`, `Low`, `Medium`, `High` ou `Critical`. A tabela a seguir resume cada gravidade:

Gravidade	Description	Exemplos
`Informational`	Informações relacionadas à operação do seu domínio.	• Atualização do software de serviço disponível • Auto-Tune iniciado
`Low`	Uma ação recomendada, mas que não tem impacto	• Auto-Tune cancelado

Gravidade	Description	Exemplos
	negativo na disponibilidade ou na performance do domínio se nenhuma ação for tomada.	• Aviso de alta contagem de fragmentos
`Medium`	Poderá haver um impacto se a ação recomendada não for executada, mas oferece uma janela de tempo estendida para que a ação seja executada.	• Falha na atualização do software de serviço • Limite de contagem de fragmentos excedido
`High`	Uma ação urgente é necessária para evitar impactos adversos.	• Atualização do software de serviço necessári a • Chave do KMS inacessível
`Critical`	Uma ação imediata é necessária para evitar impactos adversos ou se recuperar deles.	Nenhum disponível no momento

Exemplo de EventBridge evento

O exemplo a seguir mostra um evento OpenSearch de notificação de serviço enviado para a Amazon EventBridge. A notificação tem gravidade de `Informational` porque a atualização é opcional:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
```

```
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] available."
  }
}
```

Configurando um domínio Multi-AZ no Amazon Service OpenSearch

Para evitar a perda de dados e minimizar o tempo de inatividade do cluster Amazon OpenSearch Service em caso de interrupção do serviço, você pode distribuir nós em duas ou três zonas de disponibilidade na mesma região, uma configuração conhecida como Multi-AZ. As zonas de disponibilidade são locais isolados em cada AWS região.

Para domínios que executam workloads de produção, recomendamos a opção de implantação multi-AZ com modo de espera, que cria a seguinte configuração:

- Domínio implementado em três zonas.
- Tipos de instância da geração atual para os nós principais dedicados e nós de dados.
- Três nós principais dedicados e três (ou um múltiplo de três) nós de dados.
- Pelo menos duas réplicas para cada índice no seu domínio ou um múltiplo de três cópias de dados (incluindo nós primários e réplicas).

O restante desta seção fornece explicações e contexto para estas configurações.

Multi-AZ com modo de espera

O Multi-AZ with Standby é uma opção de implantação para domínios do Amazon OpenSearch Service que oferece disponibilidade de 99,99%, desempenho consistente para cargas de trabalho de produção e configuração e gerenciamento simplificados de domínio. Quando você usa o multi-AZ com modo de espera, os domínios são resilientes a falhas de infraestrutura, sem impacto no desempenho ou na disponibilidade. Essa opção de implantação atinge esse padrão ao exigir várias práticas recomendadas, como uma contagem especificada de nós de dados, contagem de nós principais, tipo de instância, contagem de réplicas, configurações de atualização de software e ajuste automático ativado.

> **Note**
>
> O Multi-AZ com Standby está disponível para a OpenSearch versão 1.3 e superior e requer regiões com pelo menos três zonas de disponibilidade.

Quando você usa o Multi-AZ com o Standby, o OpenSearch Service cria um domínio em três zonas de disponibilidade, com cada zona contendo uma cópia completa dos dados e com os dados distribuídos igualmente em cada uma das zonas. Seu domínio reserva nós em uma dessas zonas como modo de espera, o que significa que eles não atendem a solicitações de pesquisa. Quando o OpenSearch Serviço detecta uma falha na infraestrutura subjacente, ele ativa automaticamente os nós em espera em menos de um minuto. O domínio continua atendendo às solicitações de indexação e pesquisa, e qualquer impacto é limitado ao tempo necessário para realizar o failover. Não há redistribuição de dados ou recursos, o que resulta em desempenho inalterado do cluster e sem risco de redução da disponibilidade. O multi-AZ com modo de espera está disponível sem custo adicional.

Você tem duas opções para criar um domínio com modo de espera no Console de gerenciamento da AWS. Primeiro, você pode criar um domínio com o método de criação fácil, e o OpenSearch Serviço usará automaticamente uma configuração predeterminada, que inclui o seguinte:

- Três zonas de disponibilidade, com uma atuando como reserva
- Três nós principais e nós de dados dedicados
- Ajuste automático ativado no domínio
- GP3 armazenamento para os nós de dados

Você também pode escolher o método Criação padrão e selecionar Domínio com modo de espera como sua opção de implantação. Isso permite que você personalize seu domínio e, ao mesmo tempo, exija os principais atributos do modo de espera, como três zonas e três nós principais. Recomendamos escolher uma contagem de nós de dados que seja múltipla de três (o número de zonas de disponibilidade).

Depois de criar seu domínio, você pode navegar até as páginas de detalhes do domínio e, na guia Configuração do cluster, confirmar se 3-AZ com espera aparece em Zona(s) de Disponibilidade.

Se você tiver problemas ao migrar um domínio existente para o multi-AZ com modo de espera, consulte Erro ao migrar para o multi-AZ com modo de espera no guia de solução de problemas.

Limitações

Ao configurar um domínio com multi-AZ com modo de espera, considere as seguintes limitações:

- O número total de fragmentos em um nó não pode exceder 1.000, o número total de fragmentos em um cluster não pode exceder 75.000 e o tamanho de um único fragmento não pode exceder 65 GB.
- O Multi-AZ com Standby funciona somente com os `c5` tipos`m5`,`r5`,`r6g`,`c6g`,`m6g`,`r6gd`,`r7g`,`c7g`, `m7g``r7gd`, e de `i3` instância. Para saber mais sobre instâncias compatíveis, consulte Tipos de instância compatíveis.
- Você só pode usar SSD provisionado, IOPs SSD de uso geral (GP3) ou armazenamento baseado em instância com espera.
- Se você habilitar UltraWarmem um domínio Multi-AZ com Standby, o número de nós quentes deverá ser um múltiplo do número de zonas de disponibilidade que estão sendo usadas.

Multi-AZ sem modo de espera

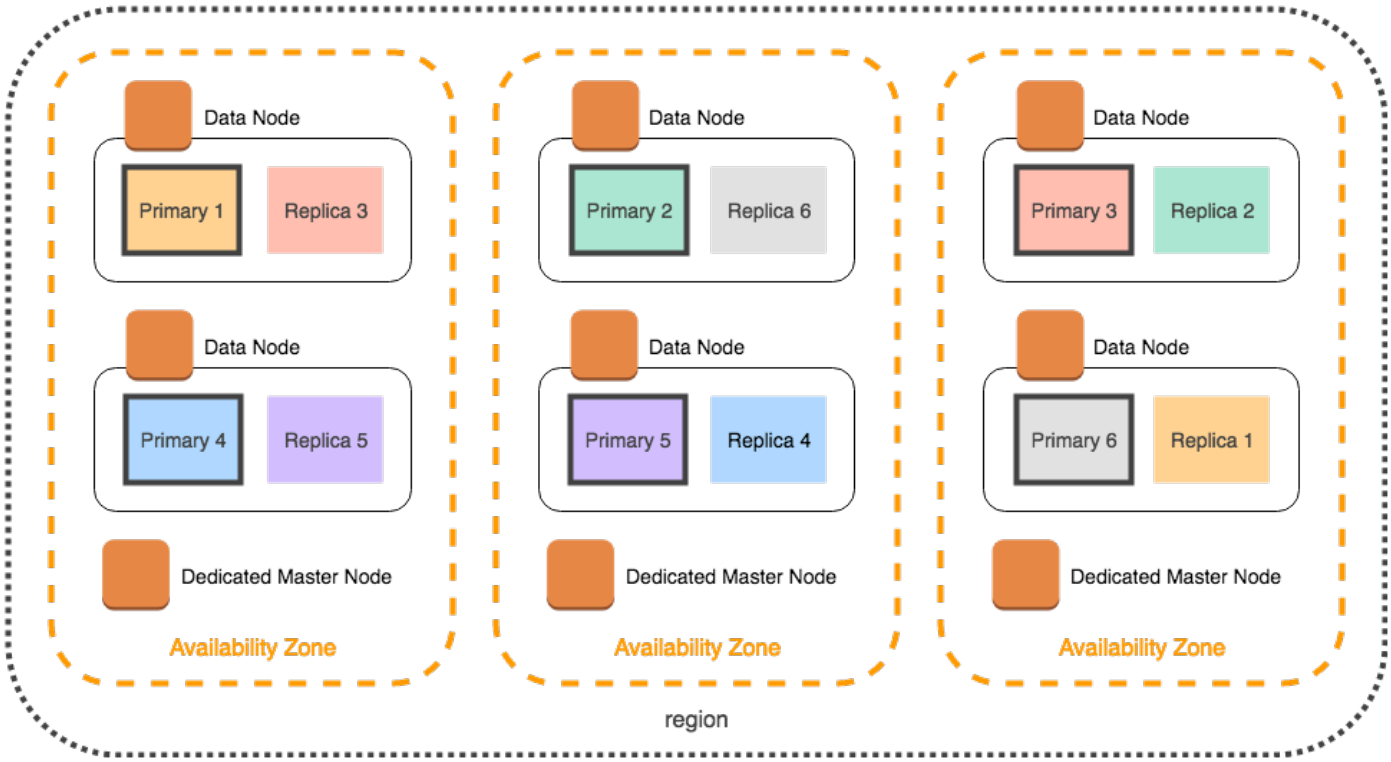
OpenSearch O serviço ainda oferece suporte ao Multi-AZ sem o modo de espera, o que oferece 99,9% de disponibilidade. Os nós são distribuídos em zonas de disponibilidade, e a disponibilidade depende do número de zonas de disponibilidade e cópias dos dados. Enquanto no modo de espera você precisa configurar seu domínio com as melhores práticas, sem o modo de espera você pode escolher seu próprio número de zonas de disponibilidade, nós e réplicas. Não recomendamos essa opção, a menos que você tenha fluxos de trabalho existentes que seriam interrompidos pela criação de domínios em espera.

Se você escolher essa opção, ainda recomendamos que você selecione três zonas de disponibilidade para permanecer resiliente a falhas de nó, disco e single-AZ. Quando ocorre uma falha, o cluster redistribui os dados pelos recursos restantes para manter a disponibilidade e a redundância. Essa movimentação de dados aumenta o uso de recursos no cluster e pode ter um impacto no desempenho. Se o cluster não for dimensionado adequadamente, ele poderá ter uma disponibilidade reduzida, o que, em grande parte, anula o propósito do multi-AZ.

A única maneira de configurar um domínio sem espera no Console de gerenciamento da AWS é escolher o método de criação padrão e selecionar Domínio sem espera como sua opção de implantação.

Distribuição de fragmentos

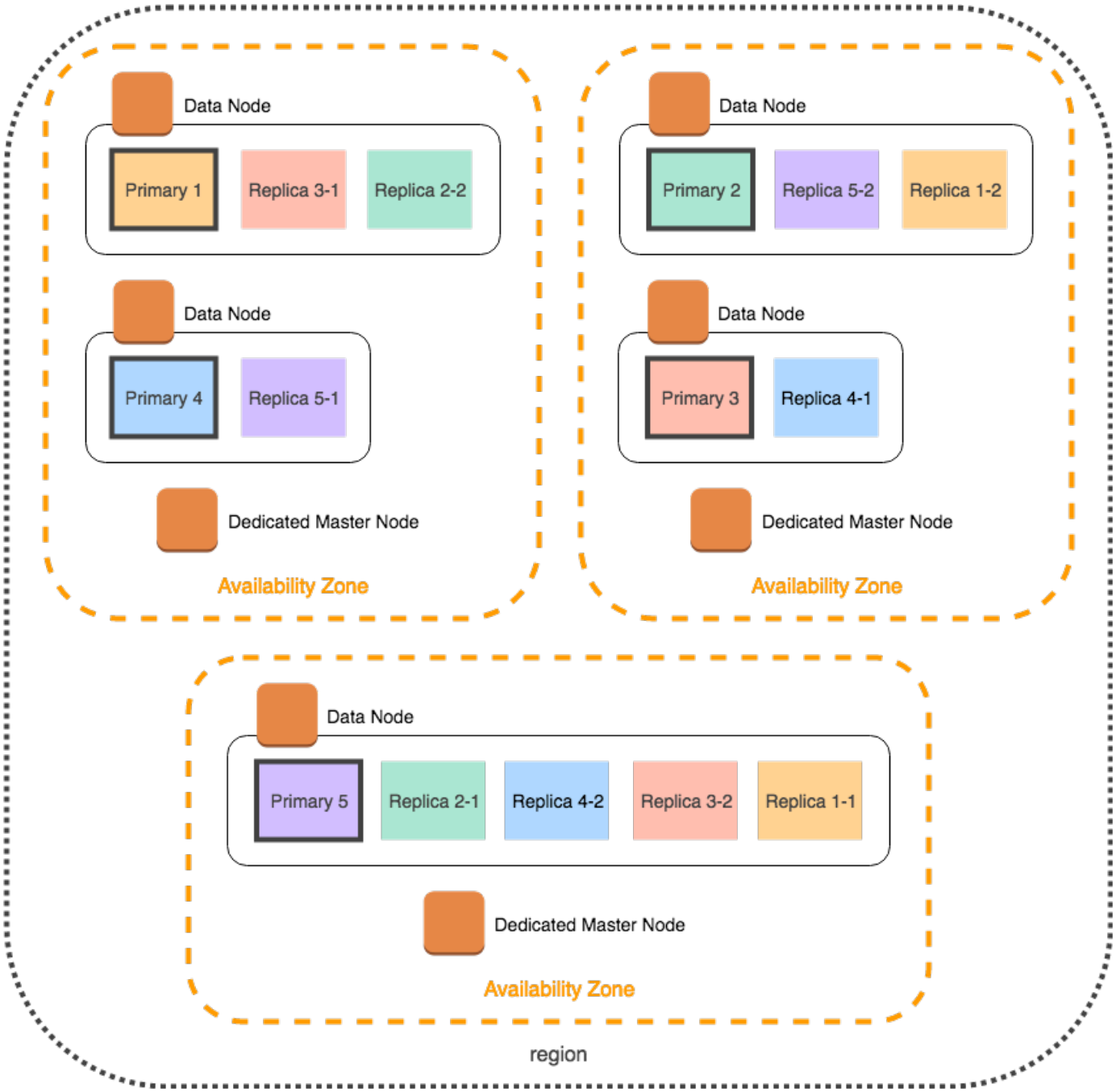
Se habilitar Multi-AZ sem standby, você deverá ter pelo menos uma réplica para cada índice no cluster. Sem réplicas, o OpenSearch Serviço não pode distribuir cópias dos seus dados para outras zonas de disponibilidade. Felizmente, a configuração padrão para qualquer índice é uma contagem de réplica de 1. Como mostra o diagrama a seguir, o OpenSearch Service se esforça ao máximo para distribuir os fragmentos primários e seus fragmentos de réplica correspondentes em diferentes zonas.



Além de distribuir fragmentos por zona de disponibilidade, o OpenSearch Service os distribui por nó. Ainda assim, determinadas configurações de domínio podem resultar em contagens de fragmentos desequilibradas. Considere o seguinte domínio:

- 5 nós de dados
- 5 fragmentos principais
- 2 réplicas
- 3 zonas de disponibilidade

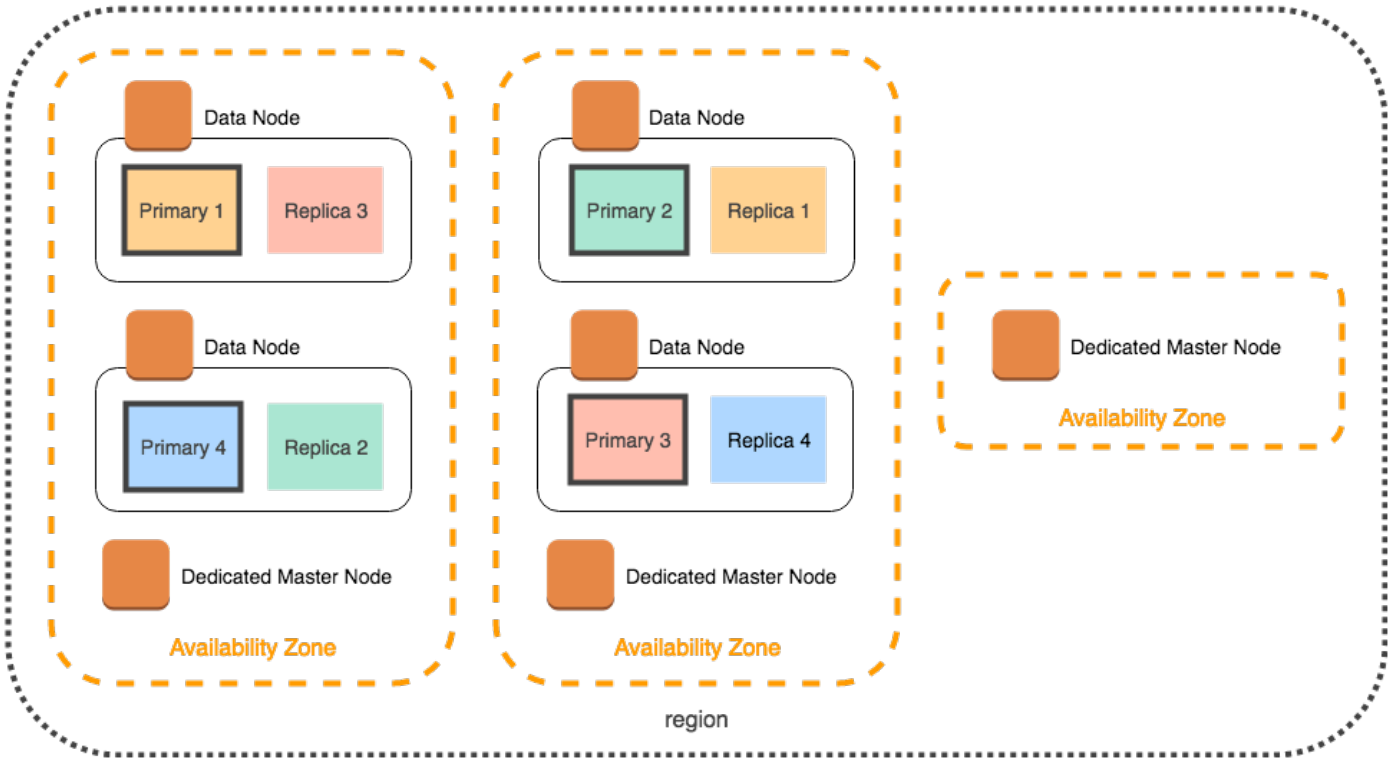
Nessa situação, o OpenSearch serviço precisa sobrecarregar um nó para distribuir os fragmentos primários e de réplica pelas zonas, conforme mostrado no diagrama a seguir.



Para evitar esses tipos de situações, que podem sobrecarregar nós individuais e afetar a performance, recomendamos selecionar multi-AZ com modo de espera ou uma contagem de instâncias que seja um múltiplo de três quando você planejar ter duas ou mais réplicas por índice.

Distribuição de nó principal dedicado

Mesmo que você selecione duas zonas de disponibilidade ao configurar seu domínio, o OpenSearch serviço distribui automaticamente nós mestres dedicados em três zonas de disponibilidade. Essa distribuição ajuda a evitar tempo de inatividade do cluster se uma zona sofrer uma interrupção de serviço. Se você usar os três nós principais dedicados recomendados e uma zona de disponibilidade ficar inativa, seu cluster ainda terá um quorum (2) de nós principais dedicados e poderá selecionar um novo principal. O diagrama a seguir demonstra essa configuração.



Se você escolher um tipo de instância de gerações anteriores que não esteja disponível nas três zonas de disponibilidade, os seguintes cenários se aplicam:

- Se você escolher três zonas de disponibilidade para o domínio, o OpenSearch serviço gerará um erro. Escolha um tipo de instância diferente e tente novamente.
- Se você escolher duas zonas de disponibilidade para o domínio, o OpenSearch Serviço distribuirá os nós principais dedicados em duas zonas.

Interrupções na zona de disponibilidade

As interrupções na zona de disponibilidade são raras, mas ocorrem. A tabela a seguir relaciona diferentes configurações de Multi-AZ e comportamentos durante uma interrupção. A última linha na tabela se aplica ao multi-AZ com modo de espera, enquanto todas as outras linhas têm configurações que se aplicam somente ao multi-AZ sem modo de espera.

Número de zonas de disponibilidade em uma região	Número de zonas de disponibilidade que você escolheu	Número de nós principais dedicados	Comportamento se uma zona de disponibilidade apresentar uma interrupção
2 ou mais	2	0	Tempo de inatividade. Seu cluster perde metade dos seus nós de dados e deve substituir pelo menos um na zona de disponibilidade restante antes que possa escolher um principal.
2	2	3	50/50 de chance de inatividade. OpenSearch O serviço distribui dois nós principais dedicados em uma zona de disponibilidade e um na outra: • Se a zona de disponibilidade com um nó principal dedicado tiver uma interrupção, os dois nós principais dedicados na zona de disponibilidade restante podem escolher um principal. • Se a zona de disponibilidade com dois nós principais dedicados apresentar uma interrupção, o cluster permanecerá indisponível até que a zona de disponibilidade se recupere.
3 ou mais	2	3	Sem tempo de inatividade. OpenSearch O serviço distribui automaticamente os nós principais dedicados em três zonas de

Número de zonas de disponibilidade em uma região	Número de zonas de disponibilidade que você escolheu	Número de nós principais dedicados	Comportamento se uma zona de disponibilidade apresentar uma interrupção
			disponibilidade, para que os dois nós principais dedicados restantes possam eleger um mestre.
3 ou mais	3	0	Sem tempo de inatividade. Aproximadamente, dois terços dos seus nós de dados ainda estão disponíveis para escolher um principal.
3 ou mais	3	3	Sem tempo de inatividade. Os dois nós principais dedicados restantes podem escolher um principal.

Em todas as configurações, independentemente da causa, as falhas nos nós podem fazer com que os nós de dados restantes do cluster passem por um período de maior carga, enquanto o OpenSearch Serviço configura automaticamente novos nós para substituir os que estão faltando.

Por exemplo, no caso de uma falha na zona de disponibilidade em uma configuração de três zonas, dois terços dos nós de dados terão que processar várias solicitações para o cluster. Conforme eles processam essas solicitações, os nós restantes também estão replicando fragmentos para novos nós à medida que ficam online, o que pode afetar ainda mais a performance. Se a disponibilidade for essencial para sua workload, considere a adição de recursos ao seu cluster para diminuir essa preocupação.

> **Note**
>
> OpenSearch O serviço gerencia domínios Multi-AZ de forma transparente, para que você não possa simular manualmente interrupções na zona de disponibilidade.

Lançamento de seus domínios OpenSearch do Amazon Service em uma VPC

Você pode lançar AWS recursos, como domínios do Amazon OpenSearch Service, em uma nuvem privada virtual (VPC). Uma VPC é uma rede virtual dedicada à sua. Conta da AWSÉ logicamente isolado de outras redes virtuais na AWS nuvem. A colocação OpenSearch de um domínio de serviço em uma VPC permite a comunicação segura entre o OpenSearch serviço e outros serviços dentro da VPC sem a necessidade de um gateway de internet, dispositivo NAT ou conexão VPN. Todo o tráfego permanece seguro na nuvem. AWS

> **Note**
>
> Se você colocar seu domínio OpenSearch de serviço em uma VPC, seu computador deverá ser capaz de se conectar à VPC. Essa conexão geralmente assume a forma de VPN, gateway de transito, rede gerenciada ou servidor de proxy. Você não pode acessar seus domínios diretamente de fora da VPC.

VPC versus domínios públicos

A seguir estão algumas das maneiras pelas quais os domínios da VPC diferem dos domínios públicos. Cada diferença é descrita posteriormente em mais detalhes.

- Devido ao seu isolamento lógico, os domínios que residem em uma VPC contam com uma camada adicional de segurança se comparados aos domínios que utilizam endpoints públicos.
- Embora os domínios públicos sejam acessíveis a partir de qualquer dispositivo conectado à Internet, os domínios da VPC exigem alguma forma de VPN ou proxy.
- Em comparação com domínios públicos, domínios VPC exibem menos informações no console do . Especificamente, a guia Cluster health (Integridade do cluster) não inclui informações de fragmentos, e a guia Indexes (Índices) não está presente.
- Os endpoits de domínio assumem formas diferentes (`https://search-domain-name` vs. `https://vpc-domain-name`).
- Não é possível aplicar políticas de acesso baseadas em IP aos domínios que residem em uma VPC porque o grupo de segurança já impõe políticas de acesso baseadas em IP.

Limitações

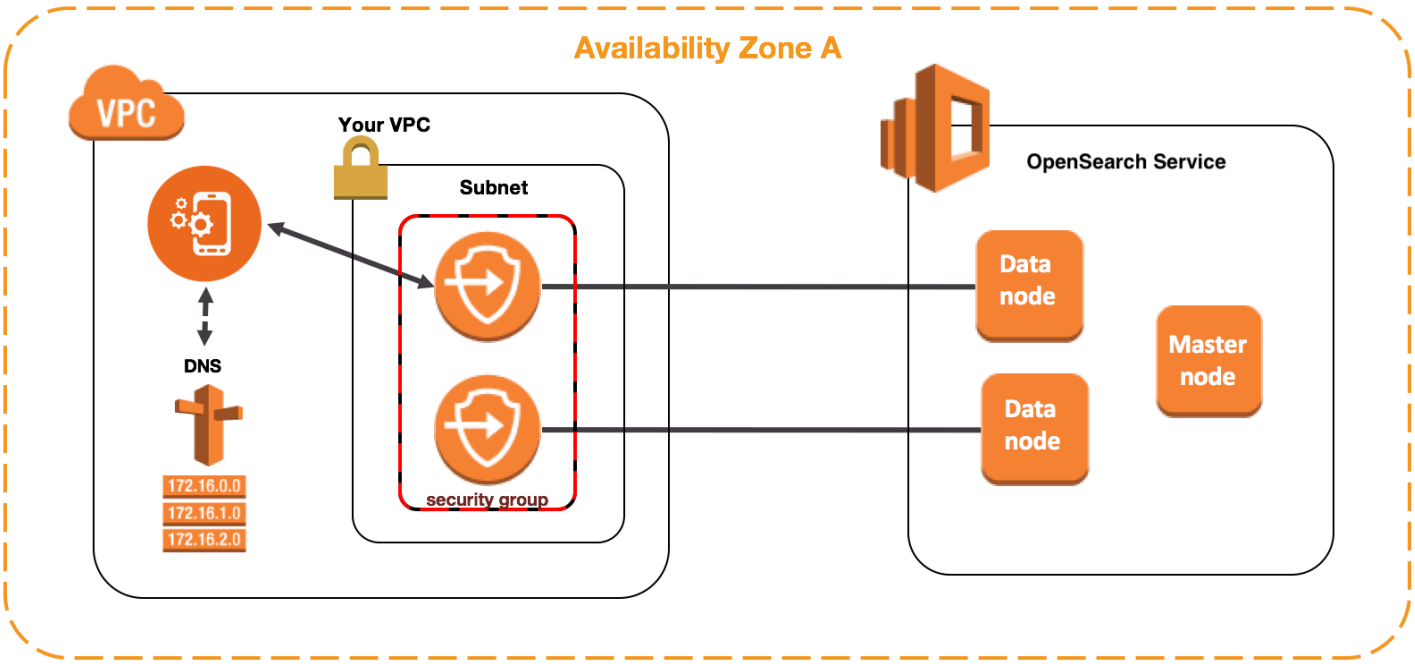
Operar um domínio de OpenSearch serviço em uma VPC tem as seguintes limitações:

- Se você executar um novo domínio de uma VPC, não será possível alternar posteriormente para um endpoint público. O inverso também é verdadeiro: se você criar um domínio com um endpoint público, não será possível colocá-lo em uma VPC. Em vez disso, é necessário criar um novo domínio e migrar seus dados.
- Você pode iniciar seu domínio de uma VPC ou usar um endpoint público, mas não pode fazer ambos. Você deve escolher uma opção ou outra ao criar seu domínio.
- Você não pode iniciar seu domínio em uma VPC que usa locação dedicada. É necessário usar uma VPC com locação definida como Padrão.
- Após colocar um domínio dentro de uma VPC, não será possível movê-lo para uma VPC diferente, mas será possível alterar as sub-redes e as configurações do grupo de segurança.
- Para acessar a instalação padrão dos OpenSearch painéis para um domínio que reside em uma VPC, os usuários devem ter acesso à VPC. Esse processo varia de acordo com a configuração de rede, mas geralmente envolve a conexão a uma VPN ou rede gerenciada ou o uso de um servidor de proxy ou gateway de trânsito. Para saber mais, consulte the section called “Sobre políticas de acesso em domínios da VPC”, o Manual do usuário da Amazon VPC e o the section called “Controlar acesso ao Dashboards”.

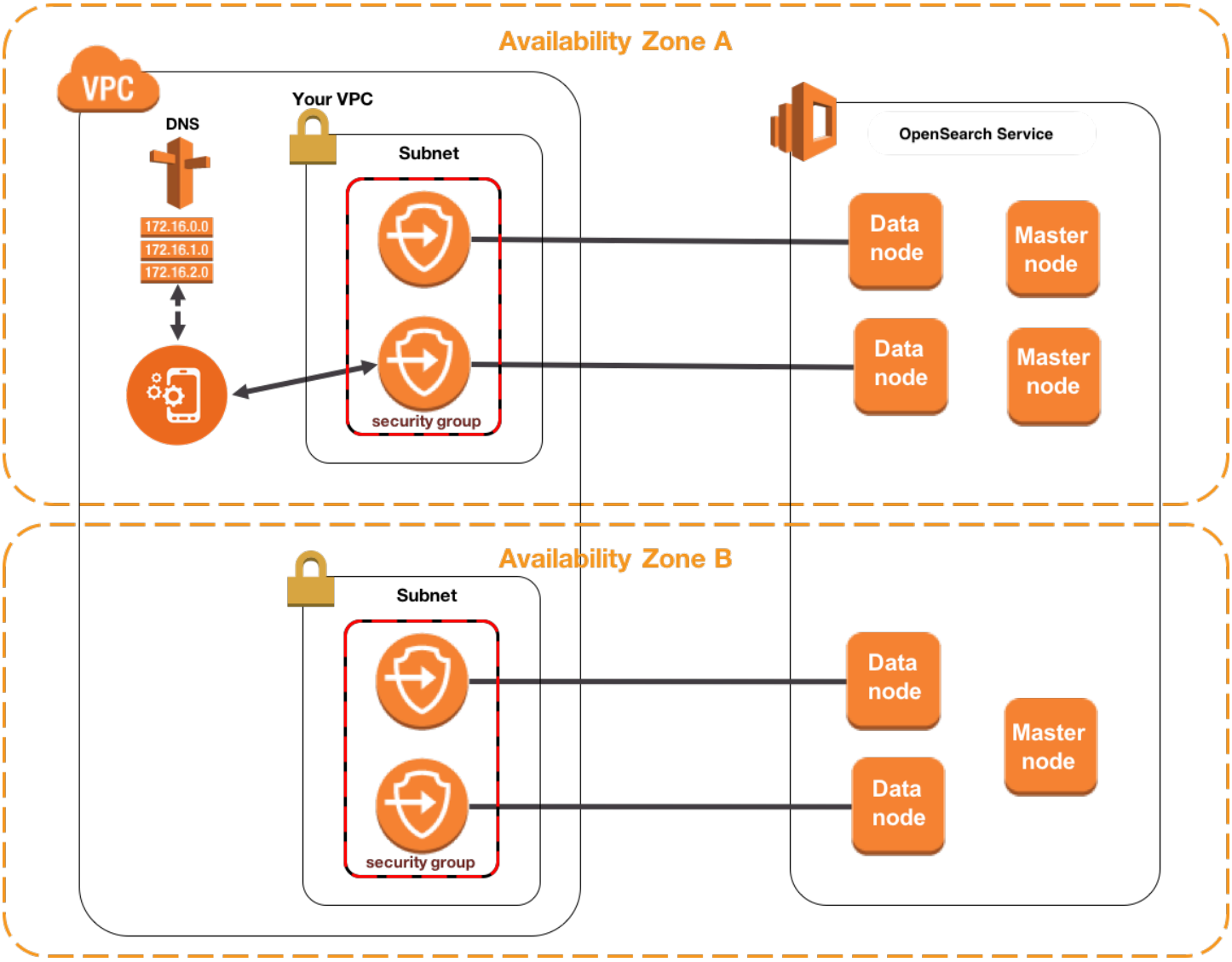
Arquitetura

Para dar suporte VPCs, o OpenSearch Service coloca um endpoint em uma, duas ou três sub-redes da sua VPC. Se você habilitar várias zonas de disponibilidade para seu domínio, cada sub-rede deverá estar em uma zona de disponibilidade diferente na mesma região. Se você usar apenas uma zona de disponibilidade, o OpenSearch Service colocará um endpoint em apenas uma sub-rede.

A ilustração a seguir mostra a arquitetura da VPC para uma zona de disponibilidade:



A ilustração a seguir mostra a arquitetura da VPC para duas zonas de disponibilidade:



OpenSearch O serviço também coloca uma interface de rede elástica (ENI) na VPC para cada um dos seus nós de dados. OpenSearch O serviço atribui a cada ENI um endereço IP privado do intervalo de IPv4 endereços da sua sub-rede. O serviço também atribui um nome de host DNS público (que é o endpoint de domínio) aos endereços IP. Você deve usar um serviço de DNS público para resolver o endpoint (que é um nome de host DNS) para os endereços IP apropriados dos nós de dados:

- Se sua VPC usar o servidor DNS fornecido pela Amazon definindo a `enableDnsSupport` opção como `true` (o valor padrão), a resolução para o endpoint do OpenSearch serviço será bem-sucedida.

- Se sua VPC usa um servidor DNS privado e o servidor pode acessar os servidores DNS públicos autoritativos para resolver nomes de host DNS, a resolução para o endpoint de serviço também será bem-sucedida. OpenSearch

Como os endereços IP podem mudar, você deve resolver o endpoint do domínio periodicamente para que sempre possa acessar os nós de dados corretos. Recomendamos que você defina o intervalo de resolução do DNS para um minuto. Se você estiver usando um cliente, também deve garantir que o cache do DNS no cliente seja limpo.

Migração do acesso público para o acesso via VPC

Ao criar um domínio, você especifica se deve haver um endpoint público ou residir em uma VPC. Após ter sido criado, você não poderá mudar de um para o outro. Em vez disso, você deve criar um novo domínio e reindexar ou migrar manualmente seus dados. Os snapshots representam uma maneira conveniente de migração de dados. Para obter informações sobre a realização e restauração de snapshots, consulte the section called “Criação de snapshots de índices”.

Sobre políticas de acesso em domínios da VPC

Colocar seu domínio de OpenSearch serviço em uma VPC fornece uma camada de segurança forte e inerente. Quando você cria um domínio com acesso público, o endpoint é composto da seguinte forma:

```
https://search-domain-name-identifier.region.es.amazonaws.com
```

Como o rótulo “público” sugere, esse endpoint é acessível de qualquer dispositivo conectado à Internet, embora você possa (e deva) controlar o acesso a ele. Se você acessar o endpoint em um navegador da Web, poderá receber uma mensagem `Not Authorized`, mas a solicitação atingirá o domínio.

Quando você cria um domínio com acesso à VPC, o endpoint se assemelha a um endpoint público:

```
https://vpc-domain-name-identifier.region.es.amazonaws.com
```

Se você tentar acessar o endpoint em um navegador da Web, no entanto, poderá descobrir que a solicitação está ultrapassando o tempo limite. Para executar até mesmo solicitações `GET` básicas, seu computador deve ser capaz de se conectar à VPC. Essa conexão geralmente assume a forma de VPN, gateway de transito, rede gerenciada ou servidor de proxy. Para obter detalhes sobre as várias formas que podem ser apresentadas, consulte Exemplos de VPC no Manual do usuário da

Amazon VPC. Para obter um exemplo focalizado em desenvolvimento, consulte the section called "Teste dos domínios da VPC".

Além desse requisito de conectividade, VPCs permita que você gerencie o acesso ao domínio por meio de grupos de segurança. Para muitos casos de uso, essa combinação de recursos de segurança é suficiente e pode ser conveniente aplicar uma política de acesso aberta ao domínio.

Operar com uma política de acesso aberto não significa que qualquer pessoa na Internet possa acessar o domínio do OpenSearch Serviço. Em vez disso, significa que, se uma solicitação chegar ao domínio do OpenSearch Serviço e os grupos de segurança associados permitirem, o domínio aceitará a solicitação. A única exceção é no caso de você estar usando o controle de acesso refinado ou uma política de acesso que especifique perfis do IAM. Nessas situações, para que o domínio aceite uma solicitação, os grupos de segurança devem permiti-la e assiná-la com credenciais válidas.

> **Note**
>
> Como os grupos de segurança já aplicam políticas de acesso baseadas em IP, você não pode aplicar políticas de acesso baseadas em IP aos domínios de OpenSearch serviço que residem em uma VPC. Se você usa o acesso público, as políticas baseadas em IP ainda estão disponíveis.

Antes de começar: pré-requisitos de acesso à VPC

Antes de habilitar uma conexão entre uma VPC e seu novo domínio de OpenSearch serviço, você deve fazer o seguinte:

- Criar uma VPC

  Para criar sua VPC, você pode usar o console Amazon VPC, a AWS CLI ou uma das. AWS SDKs Para obter mais informações, consulte Trabalhando com VPCs no Guia do usuário da Amazon VPC. Se você já tiver uma VPC, ignore esta etapa.
- Reservar endereços IP

  OpenSearch O serviço permite a conexão de uma VPC a um domínio colocando interfaces de rede em uma sub-rede da VPC. Cada interface de rede está associada a um endereço IP. Você deve reservar um número suficiente de endereços IP na sub-rede para as interfaces de rede. Para saber mais, consulte Reserva de endereços IP em uma sub-rede da VPC.

Teste dos domínios da VPC

A segurança avançada de uma VPC pode tornar a conexão com seu domínio e a execução de testes básicos um desafio. Se você já tem um domínio OpenSearch Service VPC e prefere não criar um servidor VPN, tente o seguinte processo:

1. Para a política de acesso do domínio, escolha Only use fine-grained access control (Use somente o controle de acesso refinado). Sempre é possível atualizar essa configuração depois de concluir o teste.
2. Crie uma instância Amazon Linux Amazon EC2 na mesma VPC, sub-rede e grupo de segurança do seu domínio de serviço. OpenSearch

   Como essa instância é para fins de teste e precisa fazer muito pouco trabalho, escolha um tipo de instância de custo reduzido, como o `t2.micro`. Atribua um endereço IP público à instância e crie um novo par de chaves ou escolha um existente. Se você criar uma nova chave, faça download dela em seu diretório `~/.ssh`.

   Para saber mais sobre a criação de instâncias, consulte Conceitos básicos de instâncias do Amazon EC2 Linux.
3. Adicione um gateway da Internet à VPC.
4. Na tabela de rotas da VPC, adicione uma nova rota. Em Destination (Destino), especifique um bloco CIDR que contém o endereço IP público do computador. Em Target (Destino), especifique o gateway da Internet que você acabou de criar.

   Por exemplo, você pode especificar `123.123.123.123/32` somente para seu computador ou `123.123.123.0/24` para vários computadores.
5. Para o grupo de segurança, especifique duas regras de entrada:

Tipo	Protocolo	Intervalo de portas	Fonte
SSH (22)	TCP (6)	22	*your-cidr-block*
HTTPS (443)	TCP (6)	443	*your-security-group-id*

   A primeira regra permite que você use SSH em sua instância do EC2. A segunda permite que a instância do EC2 se comunique com o domínio do OpenSearch serviço por HTTPS.

6. No terminal, execute o comando a seguir:

```
ssh -i ~/.ssh/your-key.pem ec2-user@your-ec2-instance-public-ip -N -L
 9200:vpc-domain-name-identifier.region.es.amazonaws.com:443
```

   Esse comando cria um túnel SSH que encaminha solicitações para https://localhost:9200 para seu domínio de OpenSearch serviço por meio da instância EC2. Especificar a porta 9200 no comando simula uma OpenSearch instalação local, mas use a porta que você quiser. OpenSearch O serviço só aceita conexões pela porta 80 (HTTP) ou 443 (HTTPS).

   O comando não fornece comentários e é executado indefinidamente. Para interrompê-lo, pressione `Ctrl + C`.

7. Navegue até https://localhost:9200/_dashboards/ em seu navegador da web. Talvez você precise confirmar uma exceção de segurança.

   Como alternativa, você pode enviar solicitações para https://localhost:9200 usando curl, Postman ou a linguagem de programação de sua preferência.

> **Tip**
>
> Se você encontrar erros de curl devido a uma incompatibilidade de certificado, tente o sinalizador `--insecure`.

Reserva de endereços IP em uma sub-rede da VPC

OpenSearch O serviço conecta um domínio a uma VPC colocando interfaces de rede em uma sub-rede da VPC (ou em várias sub-redes da VPC se você habilitar várias zonas de disponibilidade). Cada interface de rede está associada a um endereço IP. Antes de criar seu domínio OpenSearch de serviço, você deve ter um número suficiente de endereços IP disponíveis em cada sub-rede para acomodar as interfaces de rede.

Aqui está a fórmula básica: o número de endereços IP que o OpenSearch serviço reserva em cada sub-rede é três vezes o número de nós de dados, dividido pelo número de zonas de disponibilidade.

Exemplos

- Se um domínio tiver nove nós de dados por três zonas de disponibilidade, a quantidade de IPs por sub-rede será 9 * 3 / 3 = 9.

- Se um domínio tiver oito nós de dados por duas zonas de disponibilidade, a quantidade de IPs por sub-rede será 8 * 3 / 2 = 12.
- Se um domínio tiver seis nós de dados por uma zona de disponibilidade, a quantidade de IPs por sub-rede será 6 * 3 / 1 = 18.

Quando você cria o domínio, o OpenSearch Serviço reserva os endereços IP, usa alguns para o domínio e reserva o restante para implantações azul/verde. Você pode ver as interfaces de rede e seus endereços IP associados na seção Network Interfaces (Interfaces de rede) do console do Amazon EC2. A coluna Descrição mostra a qual domínio OpenSearch de serviço a interface de rede está associada.

> **Tip**
>
> Recomendamos que você crie sub-redes dedicadas para os endereços IP reservados do OpenSearch Serviço. Ao usar sub-redes dedicadas, você evita a sobreposição com outros aplicativos e serviços e garante a possibilidade de reservar endereços IP adicionais se precisar escalar seu cluster no futuro. Para saber mais, consulte Criação de uma sub-rede na VPC.

Você também pode considerar o provisionamento de nós coordenadores dedicados para reduzir o número de reservas de endereços IP privados necessárias para seu domínio VPC. OpenSearchanexa uma interface de rede elástica (ENI) aos seus nós coordenadores dedicados em vez dos seus nós de dados. Os nós coordenadores dedicados geralmente representam cerca de 10% do total de nós de dados. Como resultado, um número menor de endereços IP privados será reservado para domínios da VPC.

Função vinculada ao serviço para acesso à VPC

Uma função vinculada ao serviço é um tipo exclusivo de função do IAM que delega permissões para um serviço de forma que ele possa criar e gerenciar recursos em seu nome. OpenSearch O serviço requer uma função vinculada ao serviço para acessar sua VPC, criar o endpoint de domínio e colocar interfaces de rede em uma sub-rede da sua VPC.

OpenSearch O Service cria automaticamente a função quando você usa o console do OpenSearch Service para criar um domínio em uma VPC. Para que essa criação automática seja bem-sucedida, você precisa ter permissões para a ação `iam:CreateServiceLinkedRole`. Para saber mais, consulte Permissões de funções vinculadas ao serviço no Manual do usuário do IAM.

Depois que o OpenSearch Service criar a função, você poderá visualizá-la (`AWSServiceRoleForAmazonOpenSearchService`) usando o console do IAM.

Para saber mais sobre as permissões dessa função e como excluí-la, consulte the section called "Uso de perfis vinculados ao serviço".

Criação de instantâneos de índice no Amazon Service OpenSearch

Os snapshots no Amazon OpenSearch Service são backups dos índices e do estado de um cluster. O estado inclui configurações do cluster, informações de nó, configurações de índice e alocação de fragmentos.

OpenSearch Os instantâneos do serviço vêm nas seguintes formas:

- Os snapshots automatizados são apenas para recuperação de cluster. Você pode usá-los para restaurar seu domínio em caso de status de cluster vermelho ou perda de dados. Para obter mais informações, consulte Restauração de instantâneos abaixo. OpenSearch O serviço armazena instantâneos automatizados em um bucket pré-configurado do Amazon S3 sem custo adicional.
- Os snapshots manuais são usados na recuperação de clusters ou na movimentação de dados de um cluster para outro. Você precisa iniciar os snapshots manuais. Esses snapshots são armazenados no seu próprio bucket do Amazon S3, e cobranças padrão do S3 são aplicáveis. Se você tiver um instantâneo de um OpenSearch cluster autogerenciado, poderá usar esse instantâneo para migrar para um domínio de serviço. OpenSearch Para obter mais informações, consulte Migração para o Amazon OpenSearch Service.

Todos os domínios OpenSearch de serviço tiram instantâneos automatizados, mas a frequência é diferente das seguintes formas:

- Para domínios que executam o Elasticsearch OpenSearch 5.3 e versões posteriores, o OpenSearch Service tira instantâneos automatizados de hora em hora e retém até 336 deles por 14 dias. Os snapshots por hora são menos disruptivos em função de sua natureza incremental. Eles também fornecem um ponto de recuperação mais recente, caso haja problemas em domínios.
- Para domínios que executam o Elasticsearch 5.1 e versões anteriores, o OpenSearch Service tira instantâneos automatizados diariamente durante a hora especificada, retém até 14 deles e não retém nenhum dado instantâneo por mais de 30 dias.

Se o cluster entrar no status vermelho, todos os snapshots automatizados falharão enquanto o status do cluster persistir. Se você não corrigir o problema em até duas semanas, poderá perder permanentemente os dados do cluster. Para obter etapas sobre a solução de problemas, consulte the section called “Status de cluster vermelho”.

Pré-requisitos

Para criar os snapshots manualmente, é necessário trabalhar com o IAM e o Amazon S3. Verifique se você atende aos seguintes pré-requisitos antes de tentar criar um snapshot:

Pré-requisito	Description
Bucket do S3	Crie um bucket S3 para armazenar instantâneos manuais para seu domínio de OpenSearch serviço. Para obter instruções, consulte Criar um bucket de uso geral no Guia do usuário do Amazon Simple Storage Service. Lembre-se do nome do bucket para usá-lo nos seguintes locais: • Na instrução `Resource` da política do IAM que está anexada à função do IAM • O cliente Python usado para registrar um repositório de snapshots (se você usa esse método) **Important** Não aplique uma regra de ciclo de vida do Amazon Glacier a esse bucket. Snapshots manuais não são compatíveis com a classe de armazenamento do Amazon Glacier.
perfil do IAM	Crie uma função do IAM para delegar permissões ao OpenSearch Serviço. Para obter instruções, consulte Criação de funções do IAM (console) no Manual do usuário do IAM. O restante deste capítulo se refere a essa função como `TheSnapshotRole` . Anexar uma política do IAM Anexe a política a seguir ao `TheSnapshotRole` para permitir acesso ao bucket do S3:

Pré-requisito	Description

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3::: amzn-s3-demo-bucket  "
      ]
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3::: amzn-s3-demo-bucket  /*"
      ]
    }
  ]
}
```

Para obter instruções para anexar uma política gerenciada a um perfil, consulte Adicionar permissões de identidade do IAM (console) no Guia do usuário do IAM.

Editar a relação de confiança

Edite a relação de confiança de `TheSnapshotRole` para especificar o OpenSearch Serviço na `Principal` declaração, conforme mostrado no exemplo a seguir:

```
{
  "Version":"2012-10-17",
```

Pré-requisito	Description

	"Statement": [{
	"Sid": "",
	"Effect": "Allow",
	"Principal": {
	"Service": "es.amazonaws.com"
	},
	"Action": "sts:AssumeRole"
	}]
	
	}
	
	Para obter instruções para editar a relação de confiança, consulte Atualizar a política de confiança de um perfil no Guia do usuário do IAM.

Pré-requisito	Description

Permissões	Para registrar o repositório de instantâneos, você precisa ser capaz de passar `TheSnapshotRole` para OpenSearch o Serviço. Você também precisa de acesso à ação `es:ESHttpPut`. Para conceder ambas as permissões, anexe a seguinte política ao perfil do IAM cujas credenciais estão sendo usadas para assinar a solicitação:
	
	`{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "iam:PassRole", "Resource": "arn:aws:iam:: 123456789012 :role/TheSnapshotRole " }, { "Effect": "Allow", "Action": "es:ESHttpPut", "Resource": "arn:aws:es: us-east-1 :123456789012 :domain/domain-name /*" } ] }`
	
	Se seu usuário ou função não tiver permissões `iam:PassRole` para passar `TheSnapshotRole`, talvez você encontre o seguinte erro comum ao tentar registrar um repositório na próxima etapa:
	
	`$ python register-repo.py {"Message":"User: arn:aws:iam:: 123456789012 :user/MyUserAccount is not authorized to perform: iam:PassRole on resource: arn:aws:iam:: 123456789012 :role/TheSnapshotRole "}`

Registro de um repositório de snapshots manuais

Você precisa registrar um repositório de instantâneos no OpenSearch Service antes de poder tirar instantâneos de índice manuais. Essa operação única exige que você assine sua AWS solicitação com credenciais de acesso permitido`TheSnapshotRole`, conforme descrito em. the section called “Pré-requisitos”

Etapa 1: mapear a função de instantâneo nos OpenSearch painéis (se estiver usando controle de acesso refinado)

O controle de acesso refinado introduz uma etapa adicional ao registrar um repositório. Mesmo que você use a autenticação básica HTTP para todos os outros fins, será necessário mapear o perfil `manage_snapshots` para o seu perfil do IAM que tem permissões `iam:PassRole` para passar `TheSnapshotRole`.

1. Navegue até o plug-in OpenSearch Dashboards do seu domínio OpenSearch de serviço. Você pode encontrar o endpoint Dashboards no painel do seu domínio no console de OpenSearch serviços.
2. No menu principal, escolha Segurança, Funções e selecione a função manage_snapshots.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Adicione o ARN do perfil que tenha permissões para aprovar `TheSnapshotRole`. Coloque a função ARNs em Funções de back-end.

   ```
   arn:aws:iam::123456789123:role/role-name
   ```

5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Etapa 2: Registrar um repositório

A guia Snapshots a seguir demonstra como registrar um diretório de snapshots. Para opções específicas para criptografar e registrar um snapshot manual após a migração para um novo domínio, consulte as guias relevantes.

Snapshots

Para registrar um repositório de snapshots, envie uma solicitação PUT para o endpoint do domínio OpenSearch Service. Você pode usar o curl, o cliente do Python de exemplo, Postman ou outro método para enviar uma solicitação assinada a fim de registrar o repositório de

snapshot. Observe que você não pode usar uma solicitação PUT no console OpenSearch Dashboards para registrar o repositório.

A solicitação assume o seguinte formato:

```
PUT domain-endpoint/_snapshot/my-snapshot-repo-name
{
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "region",
    "role_arn": "arn:aws:iam::123456789012:role/TheSnapshotRole"
  }
}
```

> **Note**
>
> Os nomes dos repositórios não podem começar com “cs-”. Além disso, você não deve gravar no mesmo repositório a partir de vários domínios. Apenas um domínio deve ter acesso de gravação ao repositório.

Se o domínio residir em uma nuvem privada virtual (VPC), o computador deverá estar conectado à VPC para que a solicitação registre o repositório de snapshots com êxito. O acesso a uma VPC varia de acordo com a configuração de rede, mas geralmente requer uma conexão com VPN ou rede corporativa. Para verificar se você pode acessar o domínio do OpenSearch Serviço, navegue até `https://your-vpc-domain.region.es.amazonaws.com` em um navegador da Web e verifique se você recebeu a resposta JSON padrão.

Quando seu bucket do Amazon S3 estiver em outro lugar Região da AWS que não seja seu OpenSearch domínio, adicione o parâmetro `"endpoint": "s3.amazonaws.com"` à solicitação.

Encrypted snapshots

Atualmente, você não pode usar chaves AWS Key Management Service (KMS) para criptografar instantâneos manuais, mas pode protegê-los usando criptografia do lado do servidor (SSE).

Para ativar a SSE com chaves gerenciadas pelo S3 para o bucket que você usa como repositório de snapshots, adicione `"server_side_encryption": true` ao bloco `"settings"` da

solicitação PUT. Para saber mais, consulte Usar criptografia do lado do servidor com chaves gerenciadas pelo Amazon S3 (SSE-S3) no Guia do usuário do Amazon Simple Storage Service.

Como alternativa, você pode usar AWS KMS chaves para criptografia do lado do servidor no bucket do S3 que você usa como repositório de instantâneos. Se você usar essa abordagem, certifique-se de fornecer `TheSnapshotRole` permissão para a AWS KMS chave usada para criptografar o bucket do S3. Para saber mais, consulte Usar políticas de chaves no AWS KMS.

Domain migration

O registro de um repositório de snapshots é uma operação única. No entanto, para migrar de um domínio para outro, é necessário registrar o repositório de snapshots no domínio antigo e no novo. O nome do repositório é arbitrário.

Considere as seguintes diretrizes ao migrar para um novo domínio ou registrar o mesmo repositório com vários domínios:

- Ao registrar o repositório no novo domínio, adicione `"readonly": true` para o bloco `"settings"` da solicitação PUT. Essa configuração impede que você sobrescreva acidentalmente dados do domínio antigo. Apenas um domínio deve ter acesso de gravação ao repositório.
- Se estiver migrando dados para um domínio em uma Região da AWS diferente (por exemplo, de um domínio antigo e um bucket localizado em us-east-2 para um novo domínio em us-west-2), substitua `"region": "region"` por `"endpoint": "s3.amazonaws.com"` na instrução de PUT e tente novamente a solicitação.

Uso do cliente Python de exemplo

O cliente Python é mais fácil de automatizar do que uma simples solicitação HTTP, além de ser mais fácil reutilizá-lo. Se você optar por usar esse método para registrar um repositório de snapshots, salve o seguinte código de exemplo Python como um arquivo Python. Por exemplo, `register-repo.py`. O cliente requer o AWS SDK para Python (Boto3), solicitações e pacotes requests-aws4auth. O cliente contém exemplos comentados para outras operações de snapshot.

Atualize as seguintes variáveis no código de exemplo: `host`, `region`, `path` e `payload`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth
```

```
host = '' # domain endpoint
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository

path = '/_snapshot/my-snapshot-repo-name' # the OpenSearch API endpoint
url = host + path

payload = {
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "us-west-1",
    "role_arn": "arn:aws:iam::123456789012:role/snapshot-role"
  }
}

headers = {"Content-Type": "application/json"}

r = requests.put(url, auth=awsauth, json=payload, headers=headers)

print(r.status_code)
print(r.text)

# # Take snapshot
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot'
# url = host + path
#
# r = requests.put(url, auth=awsauth)
#
# print(r.text)
#
# # Delete index
#
# path = 'my-index'
# url = host + path
#
# r = requests.delete(url, auth=awsauth)
```

```
#
# print(r.text)
#
# # Restore snapshot (all indexes except Dashboards and fine-grained access control)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {
#   "indices": "-.kibana*,-.opendistro_security,-.opendistro-*",
#   "include_global_state": False
# }
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
#
# # Restore snapshot (one index)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {"indices": "my-index"}
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
```

Obtenção manual de snapshots

Os snapshots não são instantâneos. Eles demoram para serem concluídos e não representam uma point-in-time visão perfeita do cluster. Enquanto um snapshot está em andamento, você ainda pode indexar documentos e fazer outras solicitações ao cluster, mas novos documentos e atualizações em documentos existentes geralmente não são incluídos no snapshot. O instantâneo inclui fragmentos primários conforme existiam quando o instantâneo OpenSearch foi iniciado. Dependendo do tamanho do grupo de threads de snapshot, diferentes fragmentos podem ser incluídos no snapshot em momentos um pouco diferentes. Para ver as práticas recomendadas de snapshots, consulte the section called “Melhore a performance do snapshot”.

> **Warning**
>
> Os instantâneos manuais não incluem dados armazenados em níveis de armazenamento frio UltraWarm ou em camadas. Se o seu domínio usa UltraWarm ou armazenamento frio, migre esses índices para o armazenamento dinâmico antes de tirar um instantâneo manual se precisar reter esses dados.

Armazenamento e performance de snapshots

OpenSearch os instantâneos são incrementais, o que significa que eles armazenam somente os dados que foram alterados desde o último instantâneo bem-sucedido. Essa natureza incremental significa que a diferença no uso de disco entre snapshots frequentes e infrequentes normalmente é mínima. Ou seja, criar snapshots por hora por uma semana (em um total de 168 snapshots) pode não usar muito mais espaço em disco do que criar um único snapshot no final da semana. Além disso, quanto maior a frequência da criação de snapshots, menos tempo eles demoram para serem concluídos. Por exemplo, snapshots diários podem levar de 20 a 30 minutos para serem concluídos, enquanto os snapshots por hora podem ser concluídos em poucos minutos. Alguns OpenSearch usuários tiram fotos a cada meia hora.

Faça um snapshot

Ao criar um snapshot, você especifica as seguintes informações:

- O nome do repositório de snapshots
- Um nome para o snapshot

Os exemplos neste capítulo usam curl, um cliente HTTP comum, por conveniência e brevidade. Para passar um nome de usuário e uma senha para sua solicitação de curl, consulte o Tutorial de introdução.

Se as políticas de acesso especificarem usuários ou perfis, você deverá assinar suas solicitações de snapshot. Para o curl, você pode usar a opção `--aws-sigv4` com a versão 7.75.0 ou posterior. Você também pode usar os exemplos comentados no exemplo de cliente Python para fazer solicitações HTTP assinadas para os mesmos endpoints usados pelos comandos curl.

Para obter um snapshot manual, faça o seguinte:

1. Você não poderá obter um snapshot se houver um em andamento no momento. Para verificar, execute o seguinte comando:

   ```
   curl -XGET 'domain-endpoint/_snapshot/_status'
   ```

2. Execute o comando a seguir para obter um snapshot manual:

   ```
   curl -XPUT 'domain-endpoint/_snapshot/repository-name/snapshot-name'
   ```

   Para incluir ou excluir determinados índices e especificar outras configurações, adicione um corpo de solicitação. Para a estrutura da solicitação, consulte Tirar instantâneos na OpenSearch documentação.

> **Note**
>
> O tempo necessário para tirar um instantâneo aumenta com o tamanho do domínio do OpenSearch Serviço. As operações de snapshot de longa duração, às vezes, encontram o seguinte erro: `504 GATEWAY_TIMEOUT`. Normalmente, você pode ignorar esses erros e esperar até que a operação seja concluída com êxito. Execute o comando a seguir para verificar o estado de todos os snapshots de seu domínio:
>
> ```
> curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
> ```

Restauração de snapshots

Antes de restaurar um snapshot, certifique-se de que o domínio de destino não use Multi-AZ com modo de espera. Ter o modo de espera habilitado faz com que a operação de restauração falhe.

> **Warning**
>
> Se você usar aliases de índice, você deve interromper as solicitações de gravação para um alias ou mudar o alias para outro índice antes de excluir seu índice. Parar as solicitações de gravação ajuda a evitar o seguinte cenário:
>
> 1. Você exclui um índice, que também exclui seu alias.

2. Uma solicitação de gravação com erro para o alias recém-excluído cria um novo índice com o mesmo nome do alias.
3. Você não pode mais usar o alias devido a um conflito de nomes com o novo índice. Se você alternou o alias para outro índice, especifique `"include_aliases": false` ao restaurar a partir de um snapshot.

Para restaurar um snapshot

1. Identifique o snapshot que deseja restaurar. Assegure-se de que todas as configurações desse índice, como pacotes de análise personalizados ou configurações de requisitos de alocação, sejam compatíveis com o domínio. Para ver todos os repositórios de snapshots, execute o comando a seguir:

   ```
   curl -XGET 'domain-endpoint/_snapshot?pretty'
   ```

   Após identificar o repositório, execute o comando a seguir para ver todos os snapshots:

   ```
   curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
   ```

   > **Note**
   >
   > A maioria dos snapshots automatizados é armazenada no repositório `cs-automated`. Se o seu domínio criptografa dados em repouso, eles são armazenados no repositório `cs-automated-enc`. Se não encontrar o repositório de snapshots manuais que estava buscando, confirme se você o registrou no domínio.

2. (Opcional) Exclua ou renomeie um ou mais índices no domínio OpenSearch Service se você tiver conflitos de nomenclatura entre os índices no cluster e os índices no snapshot. Você não pode restaurar um snapshot dos seus índices em um OpenSearch cluster que já contém índices com os mesmos nomes.

   Você terá as seguintes opções em caso de conflitos de nomenclatura de índice:

   - Exclua os índices no domínio de OpenSearch serviço existente e, em seguida, restaure o snapshot.

- Renomeie os índices à medida que os restaura no snapshot e reindexe-os mais tarde. Para saber como renomear índices, consulte esse exemplo de solicitação na OpenSearch documentação.
- Restaure o instantâneo em um domínio OpenSearch de serviço diferente (possível somente com instantâneos manuais).

O seguinte comando exclui todos os índices existentes em um domínio:

```
curl -XDELETE 'domain-endpoint/_all'
```

No entanto, se você não planeja restaurar todos os índices, pode simplesmente excluir um:

```
curl -XDELETE 'domain-endpoint/index-name'
```

3. Para restaurar um snapshot, execute o seguinte comando:

```
curl -XPOST 'domain-endpoint/_snapshot/repository-name/snapshot-name/_restore'
```

Devido às permissões especiais nos OpenSearch painéis e aos índices de controle de acesso refinados, as tentativas de restaurar todos os índices podem falhar, especialmente se você tentar restaurar a partir de um instantâneo automatizado. O exemplo a seguir restaura apenas um índice `my-index` de `2020-snapshot` no repositório de snapshots `cs-automated`:

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "my-index"}' \
-H 'Content-Type: application/json'
```

Como alternativa, é possível restaurar todos os índices, exceto os índices de controle de acesso refinado e o Dashboards:

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "-.kibana*,-.opendistro*"}' \
-H 'Content-Type: application/json'
```

> **Note**
>
> Dependendo da sua OpenSearch versão, índices adicionais do sistema também podem precisar ser excluídos, como `-.opensearch-observability*` e. `-.plugins-ml-config*` A operação de restauração pode falhar se esses índices já existirem no domínio de destino. Para excluí-los, adicione-os à lista de `indices` exclusão. Por exemplo: `"indices": "-.kibana*,-.opendistro*,-.opensearch-observability*,-.plugins-ml-config*"`.

Você pode restaurar um snapshot sem excluir seus dados usando os parâmetros `rename_pattern` e `rename_replacement`. Para obter mais informações sobre esses parâmetros, consulte os campos de solicitação da API Restore Snapshot e o exemplo de solicitação na OpenSearch documentação.

> **Note**
>
> Se nem todos os fragmentos principais estiverem disponíveis para os índices envolvidos, o `state` do snapshot poderá ser `PARTIAL`. Esse valor indica que os dados de pelo menos um fragmento não foram armazenados com êxito. Mesmo assim é possível restaurar por meio de um snapshot parcial, mas pode ser necessário usar snapshots mais antigos para restaurar índices ausentes.

Excluir snapshots manuais

Para excluir um snapshot manual, execute o seguinte comando:

```
DELETE _snapshot/repository-name/snapshot-name
```

Automação de snapshots com o Snapshot Management

Você pode configurar uma política de gerenciamento de instantâneos (SM) nos OpenSearch painéis para automatizar a criação e a exclusão periódicas de instantâneos. O SM pode capturar um snapshot de um grupo de índices, enquanto o Index State Management só pode tirar um snapshot por índice. Para usar o SM in OpenSearch Service, você precisa registrar seu próprio repositório

Amazon S3. Para obter instruções sobre como registrar seu repositório, consulte Registrar um repositório manual de snapshots.

> **Note**
>
> O Snapshot Management é suportado a partir da OpenSearch versão 2.5. Os domínios que executam versões anteriores não oferecem suporte a esse recurso.

Antes do SM, o OpenSearch Service oferecia um recurso de captura instantânea gratuito e automatizado que ainda está ativado por padrão. Esse atributo envia snapshots para o repositório mantido pelo serviço `cs-*`. Para desativar o atributo, entre em contato com o Suporte.

Para obter mais informações sobre o recurso SM, consulte Gerenciamento de instantâneos na OpenSearch documentação.

Atualmente, o SM não oferece suporte à criação de snapshots em vários tipos de índice. Por exemplo, se você tentar criar um snapshot em vários índices * e alguns índices estiverem na camada de maior atividade, a criação do snapshot falhará. Se você precisar que seu snapshot contenha vários tipos de índice, use a ação de snapshot do ISM até que o SM ofereça suporte a essa opção.

Configurar permissões do

Se você estiver atualizando para 2.5 de uma versão anterior do domínio OpenSearch de serviço, as permissões de segurança do gerenciamento de instantâneos podem não estar definidas no domínio. Os usuários não administradores deverão ser mapeados nessa função para usar o gerenciamento de snapshot usando o controle de aceso detalhado. Para criar manualmente o perfil de gerenciamento de snapshot, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Group name	Permissões
`snapshot_managemen`	• `cluster:admin/opensearch/snapshot_management/*` • `cluster:admin/opensearch/notifications/feature/publish`

Group name	Permissões
`t_full_access`	• `cluster:admin/repository/*` • `cluster:admin/snapshot/*`
`snapshot_management_read_access`	• `cluster:admin/opensearch/snapshot_management/policy/get` • `cluster:admin/opensearch/snapshot_management/policy/search` • `cluster:admin/opensearch/snapshot_management/policy/explain` • `cluster:admin/repository/get` • `cluster:admin/snapshot/get`

3. Escolha Funções e, em seguida, Criar função.
4. Nomeie o perfil snapshot_management_role.
5. Para Permissões de cluster, selecione `snapshot_management_full_access` ou `snapshot_management_read_access`.
6. Escolha Criar.
7. Depois de criar a função, mapeie-a em qualquer função de usuário ou de backend que gerencie snapshots.

Considerações

Considere o seguinte ao configurar o gerenciamento de snapshots:

- É permitida uma política por repositório.
- São permitidos até 400 snapshots para uma política.
- Esse atributo não será executado se seu domínio tiver um status vermelho, estiver sob alta pressão da JVM (85% ou mais) ou tiver uma função de captura instantânea bloqueada. Quando o desempenho geral de indexação e pesquisa do seu cluster é afetado, o SM também pode ser afetado.
- Uma operação de snapshot só é iniciada após a conclusão da operação anterior, de forma que nenhuma operação simultânea de snapshot seja ativada por uma política.

- Várias políticas com o mesmo cronograma podem causar um pico de recursos. Se os índices de captura instantânea das políticas se sobrepõem, as operações de captura instantânea em nível de fragmento só podem ser executadas sequencialmente, o que pode causar um problema de desempenho em cascata. Se as políticas compartilharem um repositório, haverá um pico de operações de gravação nesse repositório.
- Recomendamos que você agende a automação das operações de snapshot para não mais do que uma vez por hora, a menos que tenha um caso de uso especial.

Automação de snapshots com o Gerenciamento de estados de índices

Você pode usar a operação snapshot do Index State Management (ISM) para acionar automaticamente snapshots de índices com base em suas alterações de idade, tamanho ou número de documentos. O ISM é melhor quando você precisa de um snapshot por índice. Se você precisar capturar um snapshot de um grupo de índices, consulte Automação de snapshots com o Snapshot Management.

Para usar o SM in OpenSearch Service, você precisa registrar seu próprio repositório Amazon S3. Para obter um exemplo de política do ISM usando a operação `snapshot`, consulte Políticas de exemplo.

Uso do Curator para snapshots

Se o ISM não funcionar para o gerenciamento de índices e snapshots, você poderá usar o Curator. Ele oferece funcionalidade de filtragem avançada que pode ajudar a simplificar tarefas de gerenciamento em clusters complexos. Use o pip para instalar o Curator:

```
pip install elasticsearch-curator
```

Você pode usar o Curator como uma interface de linha de comando (CLI) ou API do Python. Se você usar a API do Python, deverá usar a versão 7.13.4 ou anterior do cliente elasticsearch-py herdado. Ele não oferece suporte a um cliente opensearch-py.

Se você usar a CLI, exporte suas credenciais na linha de comando e configure o `curator.yml` da seguinte maneira:

```
client:
  hosts: search-my-domain.us-west-1.es.amazonaws.com
```

```
  port: 443
  use_ssl: True
  aws_region: us-west-1
  aws_sign_request: True
  ssl_no_validate: False
  timeout: 60

logging:
  loglevel: INFO
```

Atualizando domínios do Amazon OpenSearch Service

> **Note**
>
> OpenSearch e as atualizações de versão do Elasticsearch diferem das atualizações do software de serviço. Para obter informações sobre como atualizar o software de serviço para seu domínio OpenSearch de serviço, consultethe section called “Atualizações de software de serviço”.

O Amazon OpenSearch Service oferece atualizações no local para domínios que executam OpenSearch 1.0 ou posterior, ou Elasticsearch 5.1 ou posterior. Se você usa serviços como Amazon Data Firehose ou Amazon CloudWatch Logs para transmitir dados para o OpenSearch Service, verifique se esses serviços são compatíveis com a versão mais recente do antes de OpenSearch migrar.

Caminhos de atualização com suporte

Atualmente, o OpenSearch Service oferece suporte aos seguintes caminhos de atualização:

Da versão	Para a versão
OpenSearch 1.3, 2. x ou 3. x	OpenSearch 3. x Se você estiver executando OpenSearch 1.3 ou 2.x, você deve primeiro atualizar para OpenSearch 2.19 antes de atualizar para 3.x. OpenSearch Ao atualizar da versão OpenSearch 2.19 para a OpenSearch 3.x, talvez seja necessário resolver as seguintes configurações de índice incompatíveis. Elas se

Da versão	Para a versão
	tornaram obsoletas na versão 2.x e causarão uma falha na validação da verificação de atualização se não forem removidas. • index.knn.algo_param.ef_construction • index.knn.algo_param.m • index.knn.space_type • index.store.hybrid.mmap.extensions Para saber mais sobre configurações de índice obsoletas e etapas para corrigi-las, consulte Configurações de índice. Para obter uma lista completa das alterações mais importantes da OpenSearch versão 3.x, consulte 3.0.0. Se o seu domínio contém índices criados em OpenSearch 1.3, Elasticsearch 7.10 ou versões anteriores, você deve reindexá-los antes de atualizar para 3.x. OpenSearch OpenSearch O 3.x não é compatível com esses índices mais antigos, independentemente de estarem em armazenamento quente ou frio. UltraWarm Para reindexar índices incompatíveis UltraWarm ou frios: 1. Migre os índices para o armazenamento dinâmico. 2. Reindexe os dados. 3. Migre os índices de volta para um armazenamento quente ou frio. Como alternativa, exclua os índices se você não precisar mais deles.

Da versão	Para a versão
OpenSearch 1.3 ou 2. x	OpenSearch 2. x OpenSearch 2.17 habilitará a pesquisa simultânea de segmentos por padrão com o modo automático se o domínio atender às seguintes condições: • Nenhuma configuração anterior de pesquisa simultânea está explicitamente definida. • Todas as instâncias de dados (quentes e mornos) são do tipo 2.xl ou mais. • A utilização média da CPU p90 em instâncias de dados (quentes e mornos) por mais de uma semana é abaixo de 45%. Para obter mais detalhes sobre as configurações de pesquisa simultâneos de segmentos aqui, consulte Pesquisa simultânea de segmentos. A versão 2.3 tem as seguintes alterações importantes: • O `type` parâmetro foi removido de todos os endpoints OpenSearch da API na versão 2.0. Para saber mais, consulte alterações que podem causar interrupções. • Se seu domínio contiver algum índice (quente ou frio) que tenha sido originalmente criado no Elasticsearch 6.8, esses índices não são compatíveis com 2.3. UltraWarm OpenSearch Antes de atualizar para a versão 2.3, será necessário reindexar os índices incompatíveis. Para índices incompatíveis UltraWarm ou frios, migre-os para o armazenamento ativo, reindexe os dados e, em seguida, migre-os de volta para o armazenamento quente ou frio. Também é possível excluir os índices quando eles não são mais necessários. Se você, acidentalmente, atualizar seu domínio para a versão 2.3 sem executar essas etapas primeiro, não poderá migrar os índices incompatíveis do nível de armazenamento atual. Sua única opção será excluí-los.
OpenSearch 1. x	OpenSearch 1. x

Da versão	Para a versão
Elasticsearch 7.x	Elasticsearch 7. x ou OpenSearch 1. x **Important** OpenSearch 1. x introduz várias mudanças significativas. Para obter detalhes, consulte Renomeação OpenSearch do Amazon Service.
Elasticsearch 6.8	Elasticsearch 7. x ou OpenSearch 1. x **Important** O Elasticsearch 7.0 e OpenSearch 1.0 incluem várias mudanças importantes. Antes de iniciar uma atualização no local, recomendamos tirar um instantâneo manual do 6. domínio x, restaurando-o em um teste 7. x ou OpenSearch 1. domínio x e usando esse domínio de teste para identificar possíveis problemas de atualização. Para alterações significativas na OpenSearch versão 1.0, consulteRenomeação OpenSearch do Amazon Service. Assim como o Elasticsearch 6.x, os índices só podem conter um tipo de mapeamento, mas esse tipo agora deve ser chamado de `_doc`. Como resultado, alguns APIs não exigem mais um tipo de mapeamento no corpo da solicitação (como a `_bulk` API). Para novos índices, o Elasticsearch 7 auto-hospedado. x e OpenSearch 1. x têm uma contagem de fragmentos padrão de um. OpenSearch Domínios de serviço no Elasticsearch 7. x e posteriores mantêm o padrão anterior de cinco.
Elasticsearch 6.x	Elasticsearch 6.x

Da versão	Para a versão
Elasticsearch 5.6	Elasticsearch 6.x **Important** Os índices criados na versão 6.x não são mais compatíveis com vários tipos de mapeamento. Índices criados na versão 5.x ainda são compatíve is com vários tipos de mapeamento quando restaurados em um cluster 6.x. Verifique se o seu código de cliente cria apenas um único tipo de mapeamento por índice. Para minimizar o tempo de inatividade durante a atualização do Elasticsearch 5.6 para 6. x, o OpenSearch serviço reindexa o `.kibana` índice`.kibana-6` , exclui`.kibana`, cria um alias chamado `.kibana` e mapeia o novo índice para o novo alias.
Elasticsearch 5.x	Elasticsearch 5.x

O processo de atualização consiste em três etapas:

1. Verificações de pré-atualização — O OpenSearch serviço verifica se há problemas que podem bloquear uma atualização e não prossegue para a próxima etapa, a menos que essas verificações sejam bem-sucedidas.
2. Snapshot — O OpenSearch serviço tira um snapshot do cluster OpenSearch ou do Elasticsearch e não passa para a próxima etapa, a menos que o snapshot seja bem-sucedido. Se a atualização falhar, o OpenSearch Service usará esse snapshot para restaurar o cluster ao estado original. Para obter mais informações, consulte the section called “Não é possível reverter para a versão anterior após a atualização.”.
3. Atualização — O OpenSearch serviço inicia a atualização, que pode levar de 15 minutos a várias horas para ser concluída. OpenSearch Os painéis podem estar indisponíveis durante parte ou toda a atualização.

Atualizar um domínio (console)

O processo de atualização é irreversível e não pode ser pausado nem cancelado. Durante uma atualização, não é possível fazer alterações de configuração no domínio. Antes de iniciar uma atualização, verifique novamente se deseja prosseguir. Você pode usar essas mesmas etapas para executar a verificação de pré-atualização sem realmente iniciar uma atualização.

Se o cluster tiver nós principais dedicados, as OpenSearch atualizações serão concluídas sem tempo de inatividade. Caso contrário, o cluster poderá não responder durante vários segundos após a atualização enquanto elege um nó principal.

Para atualizar um domínio para uma versão posterior do OpenSearch ou Elasticsearch

1. Crie um snapshot manual do seu domínio. Esse instantâneo serve como um backup que você pode restaurar em um novo domínio se quiser voltar a usar a OpenSearch versão anterior.
2. Acesse https://aws.amazon.come escolha Entrar no console.
3. Em Analytics, escolha Amazon OpenSearch Service.
4. No painel de navegação, em Domínios, escolha o domínio que deseja atualizar.
5. Escolha Ações e Atualizar.
6. Selecione a versão para a qual deseja atualizar. Se você estiver atualizando para uma OpenSearch versão, a opção Ativar modo de compatibilidade será exibida. Se você habilitar essa configuração, OpenSearch reporta sua versão como 7.10 para permitir que clientes e plug-ins do Elasticsearch OSS, como o Logstash, continuem trabalhando com o Amazon Service. OpenSearch Você poderá desabilitar essa configuração posteriormente.
7. Escolha Atualizar.
8. Marque Status no painel do domínio para monitorar o status da atualização.

Atualizar um domínio (CLI)

Você pode usar as seguintes operações para identificar a versão correta do OpenSearch ou do Elasticsearch para seu domínio, iniciar uma atualização no local, realizar a verificação de pré-atualização e ver o progresso:

- `get-compatible-versions` (`GetCompatibleVersions`)
- `upgrade-domain` (`UpgradeDomain`)

- `get-upgrade-status` (`GetUpgradeStatus`)
- `get-upgrade-history` (`GetUpgradeHistory`)

Para obter mais informações, consulte a referência de comandos da AWS CLI e a Referência da API do Amazon OpenSearch Service.

Atualizar um domínio (SDK)

Este exemplo usa o cliente Python de OpenSearchServicebaixo nível AWS SDK para Python (Boto) do para verificar se um domínio está qualificado para atualização para uma versão específica, atualizá-lo e verificar continuamente o status do upgrade.

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default Region.

DOMAIN_NAME = ''  # The name of the domain to upgrade
TARGET_VERSION = ''  # The version you want to upgrade the domain to. For example,
 OpenSearch_1.1

my_config = Config(
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)
client = boto3.client('opensearch', config=my_config)


def check_versions():
    """Determine whether domain is eligible for upgrade"""
    response = client.get_compatible_versions(
        DomainName=DOMAIN_NAME
    )
    compatible_versions = response['CompatibleVersions']
    for i in range(len(compatible_versions)):
        if TARGET_VERSION in compatible_versions[i]["TargetVersions"]:
            print('Domain is eligible for upgrade to ' + TARGET_VERSION)
            upgrade_domain()
            print(response)
```

```
        else:
            print('Domain not eligible for upgrade to ' + TARGET_VERSION)


def upgrade_domain():
    """Upgrades the domain"""
    response = client.upgrade_domain(
        DomainName=DOMAIN_NAME,
        TargetVersion=TARGET_VERSION
    )
    print('Upgrading domain to ' + TARGET_VERSION + '...' + response)
    time.sleep(5)
    wait_for_upgrade()


def wait_for_upgrade():
    """Get the status of the upgrade"""
    response = client.get_upgrade_status(
        DomainName=DOMAIN_NAME
    )
    if (response['UpgradeStep']) == 'UPGRADE' and (response['StepStatus']) ==
 'SUCCEEDED':
        print('Domain successfully upgraded to ' + TARGET_VERSION)
    elif (response['StepStatus']) == 'FAILED':
        print('Upgrade failed. Please try again.')
    elif (response['StepStatus']) == 'SUCCEEDED_WITH_ISSUES':
        print('Upgrade succeeded with issues')
    elif (response['StepStatus']) == 'IN_PROGRESS':
        time.sleep(30)
        wait_for_upgrade()


def main():
    check_versions()


if __name__ == "__main__":
    main()
```

Solução de problemas de falha de validação

Quando você inicia uma atualização da versão OpenSearch ou do Elasticsearch, o OpenSearch Service primeiro executa uma série de verificações de validação para garantir que seu domínio

esteja qualificado para uma atualização. Se alguma dessas verificações falhar, você receberá uma notificação no console contendo os problemas específicos que deverão ser corrigidos antes da atualização do domínio. Para obter uma lista de possíveis problemas e as etapas para resolvê-los, consulte the section called “Solução de problemas de erros de validação”.

Solução de problemas em uma atualização

As atualizações do no local exigem domínios íntegros. Seu domínio pode não estar qualificado para uma atualização ou não ser atualizado por vários motivos. A tabela a seguir mostra os problemas mais comuns.

Problema	Description
Plug-in opcional não compatível	Quando você atualiza um domínio com plug-ins opcionais, o OpenSearch Service também atualiza automaticamente os plug-ins. Portanto, a versão de destino do seu domínio também deve oferecer suporte a esses plug-ins opcionais. Se o domínio tiver um plug-in opcional instalado que não esteja disponível para a versão de destino, a solicitação de upgrade falhará.
Muitos fragmentos por nó	OpenSearch, bem como 7. As versões x do Elasticsearch têm uma configuração padrão de no máximo 1.000 fragmentos por nó. Se um nó em seu cluster atual exceder essa configuração, o OpenSearch Serviço não permitirá que você faça o upgrade. Consulte the section called “Limite máximo de fragmentos excedido” para obter opções de solução de problemas.
Domínio no processamento	O domínio está no meio de uma mudança de configuração. Verifique a qualificação da atualização após a conclusão da operação.
Status de cluster vermelho	Um ou mais índices no cluster estão vermelhos. Para obter etapas sobre a solução de problemas, consulte the section called “Status de cluster vermelho”.
Alta taxa de erros	O cluster está retornando um grande número de erros 5xx ao tentar processar solicitações. Geralmente, esse problema é resultado de muitas solicitações de leitura ou gravação simultâneas. Considere reduzir o tráfego para o cluster ou dimensionar seu domínio.

Problema	Description
Cérebro dividido	Cérebro dividido significa que o cluster tem mais de um nó principal e foi dividido em dois clusters que nunca se juntarão por conta própria. Você pode evitar dividir o cérebro usando o número recomendado de nós principais dedicados. Para ajudar na recuperação do cérebro dividido, entre em contato com Suporte.
Nó principal não encontrado	OpenSearch O serviço não consegue encontrar o nó principal do cluster. Se o domínio usa multi-AZ, uma falha da zona de disponibilidade pode ter causado a perda de quorum do cluster e a incapacidade de escolher um novo nó principal. Se o problema não se resolver, entre em contato com Suporte.
Muitas tarefas pendentes	O nó principal está sob carga pesada e tem muitas tarefas pendentes. Considere reduzir o tráfego para o cluster ou dimensionar seu domínio.
Volume de armazenamento prejudicado	O volume de disco de um ou mais nós não está funcionando corretame nte. Esse problema geralmente ocorre junto com outros problemas , como uma alta taxa de erros ou muitas tarefas pendentes. Se o problema ocorrer isoladamente e não se resolver, entre em contato com Suporte.
Problema de chave do KMS	A chave do KMS usada para criptografar o domínio está inacessível ou ausente. Para saber mais, consulte the section called “Monitoramento de domínios que criptografam dados em repouso”.
Snapshot em andamento	O domínio está tirando um snapshot no momento. Verifique a qualifica ção da atualização após a conclusão do snapshot. Além disso, verifique se é possível listar repositórios de snapshots manuais, listar snapshots nesses repositórios e obter snapshots manuais. Se o OpenSearch Serviço não conseguir verificar se um snapshot está em andamento, as atualizações podem falhar.
Tempo limite ou falha de snapshot	O snapshot de pré-atualização demorou muito para ser concluído ou falhou. Verifique o status do cluster e tente novamente. Se o problema continuar, entre em contato com o Suporte.

Problema	Description
Índices incompatíveis	Um ou mais índices são incompatíveis com a versão de destino. Esse problema pode ocorrer se você migrou os índices de uma versão mais antiga do Elasticsearch OpenSearch . Reindexe os índices e tente novamente.
Uso elevado do disco	O uso de disco para o cluster está acima de 90%. Exclua os dados ou dimensione o domínio e tente novamente.
Uso elevado do JVM	A pressão de memória JVM está acima de 75%. Reduza o tráfego para o cluster ou dimensione o domínio e tente novamente.
OpenSearch Problema de alias de painéis	`.dashboards` já está configurado como um alias e mapeia para um índice incompatível, provavelmente de uma versão anterior do Dashboards. OpenSearch Reindexe e tente novamente.
Status de painéis vermelhos	OpenSearch O status dos painéis é vermelho. Tente usar o Dashboard s quando a atualização for concluída. Se o status vermelho persistir, resolva-o manualmente e tente novamente.
Compatibilidade entre clusters	Você só pode atualizar se a compatibilidade entre clusters for mantida entre os domínios de origem e de destino após a atualização. Durante o processo de atualização, todas as conexões incompatíveis são identific adas. Para prosseguir, atualize o domínio remoto ou excluir as conexões incompatíveis. Observe que, se a replicação estiver ativa no domínio, você não poderá retomá-la depois de excluir a conexão.
Outro problema de OpenSearch serviço	Problemas com o OpenSearch serviço em si podem fazer com que seu domínio seja exibido como inelegível para um upgrade. Se nenhuma das condições anteriores se aplicar ao seu domínio e o problema persistir por mais de um dia, entre em contato com Suporte.

Como usar um snapshot para migrar dados

As atualizações no local são a maneira mais fácil, rápida e confiável de atualizar um domínio para uma versão posterior OpenSearch ou do Elasticsearch. Os snapshots são uma boa opção se você

precisa migrar de uma versão anterior a 5.1 do Elasticsearch ou deseja migrar para um cluster totalmente novo.

A tabela a seguir mostra como usar snapshots para migrar dados para um domínio que usa uma versão diferente OpenSearch ou uma versão do Elasticsearch. Para saber mais sobre a criação e a restauração de snapshots, consulte the section called “Criação de snapshots de índices”.

Da versão	Para a versão	Processo de migração

OpenSearch 1.3 ou 2. x	OpenSearch 2. x	1. Analise as alterações mais recentes da versão OpenSearch 2.3 para ver se você precisa fazer ajustes em seus índices ou aplicativos.
		2. Crie um snapshot manual do domínio 1.3 ou 2.x.
		3. Crie um domínio 2.x que seja uma versão superior ao seu domínio 1.3 ou 2.x original.
		4. Restaure o snapshot do domínio original para o domínio 2.x. Durante a operação, talvez seja necessári o restaurar o índice do `.opensearch` com um novo nome:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`"indices": "*",`
		`"ignore_unavailable": true,`
		`"rename_pattern": ".opensearch",`
		`"rename_replacement": ".backup-opensearc h"`
		`}`
		Em seguida, você pode reindexar o `.backup-o pensearch` no novo domínio e definir `.opensear ch` como seu alias. Observe que a chamada REST `_restore` não inclui `include_global_state` porque a entrada padrão `_restore` é falsa. Como resultado, o domínio de teste não incluirá nenhum modelo de índice e não terá o estado completo do backup.

Da versão	Para a versão	Processo de migração
		5. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

OpenSearch 1. x	OpenSearch 1. x	1. Crie um snapshot manual do domínio 1.x.
		2. Crie um domínio 1.x que seja uma versão superior ao seu domínio 1.x original.
		3. Restaure o snapshot do domínio original para o domínio 1.x. Durante a operação, talvez seja necessário restaurar o índice do `.opensearch` com um novo nome:
		
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".opensearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		
		Em seguida, você pode reindexar o `.backup-opensearch` no novo domínio e definir `.opensearch` como seu alias. Observe que a chamada REST `_restore` não inclui `include_global_state` porque a entrada padrão `_restore` é falsa. Como resultado, o domínio de teste não incluirá nenhum modelo de índice e não terá o estado completo do backup.
		4. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Da versão	Para a versão	Processo de migração

Elasticsearch 6.x ou 7.x	OpenSearch 1. x	1. Analise as alterações mais recentes da OpenSearch versão 1.0 para ver se você precisa fazer ajustes em seus índices ou aplicativos.
		2. Crie um snapshot manual do domínio do Elasticsearch 7.x ou 6.x.
		3. Crie um OpenSearch 1. domínio x.
		4. Restaure o snapshot do domínio Elasticsearch para o domínio. OpenSearch Durante a operação, talvez seja necessário restaurar o índice do `.elasticsearch` com um novo nome:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		Em seguida, você pode reindexar o `.backup-opensearch` no novo domínio e definir `.elasticsearch` como seu alias. Observe que a chamada REST `_restore` não inclui `include_global_state` porque a entrada padrão `_restore` é falsa. Como resultado, o domínio de teste não incluirá nenhum modelo de índice e não terá o estado completo do backup.
		5. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Da versão	Para a versão	Processo de migração

Elasticsearch 6.x	Elasticsearch 7.x	1. Revise as alterações que podem causar falhas na versão 7.0 para verificar se é necessário ajustar os índices ou as aplicações.
		2. Crie um snapshot manual do domínio 6.x.
		3. Crie um domínio 7.x.
		4. Restaure o snapshot do domínio original para o domínio 7.x. Durante a operação, você provavelmente precisará restaurar o índice do `.opensearch` com um novo nome:
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-elasticsearch"`
		`}`
		Em seguida, você pode reindexar o `.backup-elasticsearch` no novo domínio e definir `.elasticsearch` como seu alias. Observe que a chamada REST `_restore` não inclui `include_global_state` porque a entrada padrão `_restore` é falsa. Como resultado, o domínio de teste não incluirá nenhum modelo de índice e não terá o estado completo do backup.
		5. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Da versão	Para a versão	Processo de migração
Elasticsearch 6.x	Elasticsearch 6.8	1. Crie um snapshot manual do domínio 6.x. 2. Crie um domínio 6.8. 3. Restaure o snapshot do domínio original para o domínio 6.8. 4. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.
Elasticsearch 5.x	Elasticsearch 6.x	1. Revise as alterações que podem causar interrupções na versão 6.0 para verificar se você precisa fazer ajustes em seus índices ou aplicações. 2. Crie um snapshot manual do domínio 5.x. 3. Crie um domínio 6.x. 4. Restaure o snapshot do domínio original para o domínio 6.x. 5. Se você não precisar mais do domínio 5.x, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.
Elasticsearch 5.x	Elasticsearch 5.6	1. Crie um snapshot manual do domínio 5.x. 2. Crie um domínio 5.6. 3. Restaure o snapshot do domínio original para o domínio 5.6. 4. Se você não precisar mais do domínio original, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Da versão	Para a versão	Processo de migração
Elasticsearch 2.3	Elasticsearch 6.x	Os snapshots do Elasticsearch 2.3 não são compatíveis com o 6.x. Para migrar os dados diretamente da versão 2.3 para a 6.x, você terá que recriar manualmente os índices no novo domínio. Como alternativa, você pode seguir as etapas da atualização da versão 2.3 para a 5.x nesta tabela, realizar as operações de `_reindex` no novo domínio 5.x para converter os índices 2.3 em índices 5.x e depois seguir as etapas da atualização da versão 5.x para a 6.x.
Elasticsearch 2.3	Elasticsearch 5.x	1. Revise as alterações que podem causar falhas na versão 5.0 para verificar se é necessário ajustar os índices ou as aplicações. 2. Crie um snapshot manual do domínio 2.3. 3. Crie um domínio 5.x. 4. Restaure o snapshot do domínio 2.3 para o 5.x. 5. Se você não precisar mais do domínio 2.3, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Da versão	Para a versão	Processo de migração
Elasticsearch 1.5	Elasticsearch 5.x	Os snapshots do Elasticsearch 1.5 não são compatíveis com o 5.x. Para migrar os dados da versão 1.5 para a 5.x, você terá que recriar manualmente os índices no novo domínio. **Important** Os instantâneos 1.5 são compatíveis com 2.3, mas os domínios do OpenSearch Service 2.3 não suportam a `_reindex` operação. Como você não pode reindexá-los, os índices originados em um domínio da versão 1.5 ainda não podem ser restaurados de snapshots da 2.3 para domínios da 5.x.
Elasticsearch 1.5	Elasticsearch 2.3	1. Use o plug-in de migração para descobrir se é possível atualizar diretamente para a versão 2.3. Talvez você precise alterar seus dados antes de migrar. a. Em um navegador da web, abra `http://`*`domain-endpoint`*` /_plugin/migration/` . b. Escolha Run checks now. c. Analise os resultados e, se necessário, siga as instruções para fazer alterações em seus dados. 2. Crie um snapshot manual do domínio 1.5. 3. Crie um domínio 2.3. 4. Restaure o snapshot do domínio 1.5 para o 2.3. 5. Se você não precisar mais do domínio 1.5, exclua-o. Do contrário, você continuará a ser cobrado pelo domínio.

Criação de um endpoint personalizado para o Amazon Service OpenSearch

A criação de um endpoint personalizado para seu domínio do Amazon OpenSearch Service facilita a consulta entre você OpenSearch e seus OpenSearch painéis URLs. Você pode incluir a marca da sua empresa ou simplesmente usar um easier-to-remember endpoint mais curto do que o padrão.

Se você precisar alternar para um novo domínio, bastará atualizar seu DNS para apontar para o novo URL e continuar usando o mesmo endpoint de antes.

Você protege endpoints personalizados gerando um certificado no AWS Certificate Manager (ACM) ou importando um dos seus.

Endpoints personalizados para novos domínios

Você pode habilitar um endpoint personalizado para um novo domínio OpenSearch de serviço usando o console OpenSearch de serviço ou a API de configuração. AWS CLI

Para personalizar o endpoint (console)

1. No console de OpenSearch serviço, escolha Criar domínio e forneça um nome para o domínio.
2. Em Endpoint personalizado, selecione Habilitar endpoint personalizado.
3. Em Nome de host personalizado, insira o nome de host do endpoint personalizado preferido. O nome de host deve ser um nome de domínio totalmente qualificado (FQDN), como www.seudomínio.com ou exemplo.seudomínio.com.

   > **Note**
   >
   > Caso não tenha um certificado curinga, você deverá obter um novo certificado para seus subdomínios de endpoint personalizados.

4. Em Certificado AWS , escolha o certificado SSL que deseja usar para o domínio. Se nenhum certificado estiver disponível, você poderá importar um para o ACM ou usar o ACM para provisionar um. Para saber mais, consulte Emissão e gerenciamento de certificados no Manual do usuário do AWS Certificate Manager.

> **Note**
>
> O certificado deve ter o nome do endpoint personalizado e estar na mesma conta do seu domínio OpenSearch de serviço. O status do certificado deve ser ISSUED (EMITIDO).

- Siga o restante das etapas para criar seu domínio e escolha Criar.
- Selecione o domínio quando terminar o processamento para visualizar seu endpoint personalizado.

Para usar a CLI ou API de configuração, use as operações `CreateDomain` e `UpdateDomainConfig`. Para obter mais informações, consulte a Referência de AWS CLI Comandos e a Referência da API do Amazon OpenSearch Service.

Endpoints personalizados para domínios existentes

Para adicionar um endpoint personalizado a um domínio de OpenSearch serviço existente, escolha Editar e execute as etapas 2 a 4 acima.

Mapeamento CNAME

Depois de habilitar um endpoint personalizado para seu domínio de OpenSearch serviço, você pode criar um mapeamento CNAME no Amazon Route 53 (ou no seu provedor de serviços DNS preferido). A criação de um mapeamento CNAME permitirá rotear o tráfego para o endpoint personalizado e seus subdomínios. Sem esse mapeamento, não será possível rotear o tráfego para o endpoint personalizado. Para ver as etapas necessárias para criar esse mapeamento no Route 53, consulte Configuração do roteamento de DNS para um novo domínio e Criação de uma zona hospedada para um subdomínio. Para outros provedores, consulte a respectiva documentação.

Crie o registro CNAME apontando o endpoint personalizado para o endpoint de domínio gerado automaticamente. Se seu domínio for de pilha dupla, você poderá apontar seu registro CNAME para qualquer um dos dois endpoints gerados pelo serviço. A capacidade de pilha dupla do seu endpoint personalizado depende do endpoint gerado pelo serviço para o qual você direciona o registro CNAME. O nome de host do endpoint personalizado é o nome do registro CNAME, e o nome de host do endpoint do domínio é o valor do registro CNAME.

Se você usar a autenticação SAML para OpenSearch painéis, deverá atualizar seu IdP com a nova URL de SSO.

Você pode usar o Amazon Route 53 para criar um tipo de registro de alias para apontar o endpoint personalizado de domínio para um endpoint de pesquisa de pilha dupla. Para criar um tipo de registro de alias, você deve configurar seu domínio para usar o tipo de endereço IP de pilha dupla. Você pode fazer isso usando a API do Route 53.

Para criar um tipo de registro de alias usando a API do Route 53, especifique o destino do alias do seu domínio. Você pode encontrar o alias de destino do seu domínio no campo Zona hospedada (pilha dupla) na seção de endpoint personalizado do console de OpenSearch serviço ou usando a `DescribeDomain` API e copiando o valor do. `DomainEndpointV2HostedZoneId`

Auto-Tune para Amazon Service OpenSearch

O Auto-Tune no Amazon OpenSearch Service usa métricas de desempenho e uso do seu OpenSearch cluster para sugerir alterações de configuração relacionadas à memória, incluindo tamanhos de fila e cache e configurações de máquina virtual Java (JVM) em seus nós. Essas alterações opcionais melhoram a velocidade e a estabilidade do cluster.

Algumas alterações são implantadas imediatamente, enquanto outras são agendadas durante o período fora do horário de pico do seu domínio. Você pode voltar às configurações padrão do OpenSearch Serviço a qualquer momento. À medida que o Auto-Tune reúne e analisa as métricas de desempenho do seu domínio, você pode ver suas recomendações no console de OpenSearch serviço na página Notificações.

O Auto-Tune está disponível comercialmente Regiões da AWS em domínios que executam qualquer OpenSearch versão, ou Elasticsearch 6.7 ou posterior, com um tipo de instância compatível.

Tipos de alterações

O Auto-Tune tem duas categorias de alterações amplas:

- Alterações sem interrupções aplicadas à medida em que o cluster é executado.
- Alterações que exigem uma implantação azul/verde, que se aplicam durante a janela fora do horário de pico do domínio.

Com base nas métricas de performance do seu domínio, o Auto-Tune pode sugerir ajustes nas seguintes configurações:

Alterar tipo	Categoria	Description
Tamanho do heap do JVM	Azul/verde	Por padrão, o OpenSearch Service usa 50% da RAM de uma instância para o heap da JVM, até um tamanho de heap de 32 GiB. Aumentar essa porcentagem fornece OpenSearch mais memória, mas deixa menos para o sistema operacional e outros processos. Valores maiores podem diminuir o número de pausas de coleta de resíduos, mas aumentar o comprimento dessas pausas.
Configurações de geração jovem do JVM	Azul/verde	As configurações de “geração jovem” do JVM afetam a frequência de coletas de resíduos secundárias. Coleções secundárias mais frequentes podem diminuir o número de coleções principais e pausas.
Tamanho da fila	Sem interrupções	Por padrão, o tamanho da fila de pesquisa é `1000` e o tamanho da fila de gravação é `10000`. O Auto-Tune dimensiona automaticamente as filas de pesquisa e gravação quando há heap adicional disponível para lidar com solicitações.
Tamanho do cache	Sem interrupções	O cache de campo monitora estruturas de dados no heap. Por isso, é importante monitorar o uso do cache. O Auto-Tune dimensiona o tamanho do cache de dados de campo para evitar problemas de falta de memória e interruptores de circuito. O cache de solicitação de fragmento é gerenciado em nível de nó e tem um tamanho máximo padrão de 1% do heap. O Auto-Tune dimensiona o tamanho do cache de solicitação de fragmentos para aceitar mais solicitações de pesquisa e índice do que o cluster configurado é capaz de manipular.
Dimensão da solicitação	Sem interrupções	Por padrão, quando o tamanho agregado das solicitações em andamento ultrapassa 10% do total da JVM (2% para tipos de `t2` instância e 1% para`t3.small`), limita todas OpenSearch as solicitações novas `_search` e `_bulk` até que as solicitações existentes sejam concluídas.

Alterar tipo	Categoria	Description
		O Auto-Tune ajusta esse limite de forma automática, que costuma ser entre 5 e 15%, de acordo com a quantidade da JVM ocupada atualmente no sistema. Por exemplo, se a pressão de memória da JVM estiver alta, o Auto-Tune poderá reduzir o limite para 5%. Se for o caso, talvez você veja mais rejeições até o cluster se estabiliz ar e o limite aumentar.

Habilitação ou desabilitação do Auto-Tune

OpenSearch O serviço ativa o Auto-Tune por padrão em novos domínios. Para habilitar ou desabilitar o Auto-Tune em domínios existentes, recomendamos utilizar o console, o que simplifica o processo. Ativar o Auto-Tune não causa uma blue/green implantação.

No momento, não é possível habilitar ou desabilitar o Ajuste automático usando o AWS CloudFormation.

Console

Como habilitar o Auto-Tune em um domínio existente

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, em Domínios, escolha o nome do domínio para abrir a configuração do cluster.
3. Escolha Ativar se o Auto-Tune ainda não estiver ativado.
4. Opcionalmente, selecione Janela fora do pico para agendar otimizações que exijam uma blue/green implantação durante a janela fora do pico configurada do domínio. Para obter mais informações, consulte the section called “Agendamento de melhorias no Auto-Tune”.
5. Escolha Salvar alterações.

CLI

Para ativar o Auto-Tune usando o AWS CLI, envie uma UpdateDomainConfigsolicitação:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
```

```
  --auto-tune-options DesiredState=ENABLED
```

Agendamento de melhorias no Auto-Tune

Antes de 16 de fevereiro de 2023, o Auto-Tune usava janelas de manutenção para programar mudanças que exigiam uma blue/green implantação. As janelas de manutenção agora estão obsoletas em favor da janela fora do horário de pico, que é um período diário de 10 horas durante o qual seu domínio normalmente tem pouco tráfego. Você pode modificar a hora de início padrão para a janela fora do horário de pico, mas não pode alterar a duração dela.

Todos os domínios do Auto-Tune que tinham as janelas de manutenção ativadas antes da introdução das janelas fora do horário de pico em 16 de fevereiro de 2023 podem continuar usando as janelas de manutenção antigas, sem interrupção. No entanto, recomendamos a migração dos seus domínios existentes para usar a janela fora do horário de pico para a manutenção do domínio. Para instruções, consulte the section called “Migração das janelas de manutenção do Auto-Tune”.

Console

Como agendar ações do Auto-Tune na janela fora do horário de pico

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, em Domínios, escolha o nome do domínio para abrir a configuração do cluster.
3. Vá até a guia Auto-Tune e escolha Editar.
4. Escolha Ativar se o Auto-Tune ainda não estiver ativado.
5. Em Programar otimizações durante a janela fora do pico, selecione Janela fora do horário de pico.
6. Escolha Salvar alterações.

CLI

Para configurar seu domínio para agendar ações de ajuste automático durante a janela configurada fora de pico, inclua `UseOffPeakWindow` UpdateDomainConfigna solicitação:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=null
```

Monitoramento de alterações no Auto-Tune

Você pode monitorar as estatísticas do Auto-Tune em Amazon CloudWatch. Para obter uma lista completa de métricas, consulte the section called “Métricas do Auto-Tune”.

OpenSearch O serviço envia eventos do Auto-Tune para a Amazon EventBridge. Você pode usar EventBridge para configurar regras que enviam um e-mail ou executam uma ação específica quando um evento é recebido. Para ver o formato de cada evento do Auto-Tune enviado para EventBridge, consultethe section called “Auto-Tune de eventos”.

Marcação de domínios do Amazon OpenSearch Service

As tags permitem que você atribua informações arbitrárias a um domínio do Amazon OpenSearch Service para que você possa categorizar e filtrar essas informações. Uma tag é um par de valores-chave que você define e associa a um domínio de OpenSearch serviço. Você pode usar essas tags para rastrear custos agrupando despesas para recursos com tags semelhantes. AWS não aplica nenhum significado semântico às suas tags. Tags são interpretadas estritamente como sequências de caracteres. Todas as tags têm os elementos a seguir:

Elemento da tag	Description	Obrigatório
Chave de tag	A chave de tags é o nome da tag. A chave deve ser exclusiva para o domínio do OpenSearch serviço ao qual ela está vinculada. Para obter uma lista de restrições básicas a chaves e valores de tag, consulte Restrições a tags definidas pelo usuário.	Sim
Valor da tag	O valor da tag é o valor da string da tag. Os valores de tag podem ser `null` e não precisam ser exclusivos em um conjunto de tags. Por exemplo, você pode ter um par de valores-chave em um conjunto de tags de. project/Trinity and cost-center/Trinity Para obter uma lista de restrições básicas a chaves e valores de tag, consulte Restrições a tags definidas pelo usuário.	Não

Cada domínio OpenSearch de serviço tem um conjunto de tags, que contém todas as tags atribuídas a esse domínio OpenSearch de serviço. AWS não atribui automaticamente nenhuma tag aos domínios OpenSearch de serviço. Um conjunto de tags pode conter entre 0 e 50 tags. Se você

adicionar uma tag a um domínio que tenha a mesma chave que uma tag existente, o novo valor sobrescreverá o antigo.

Exemplos de marcação com tags

Você pode usar uma chave para definir uma categoria, e o valor da tag pode ser um item nessa categoria. Por exemplo, você pode definir uma chave de tag de `project` e um valor de tag de`Salix`, indicando que o domínio do OpenSearch serviço está atribuído ao projeto Salix. Você também pode usar tags para designar domínios de OpenSearch serviço como sendo usados para teste ou produção usando uma chave como `environment=test` ou. `environment=production` Tente usar um conjunto consistente de chaves de tag para facilitar o rastreamento de metadados associados aos domínios OpenSearch de serviço.

Você também pode usar etiquetas para organizar sua AWS fatura de forma a refletir sua própria estrutura de custos. Para fazer isso, inscreva-se para receber sua Conta da AWS fatura com os valores-chave da etiqueta incluídos. Organize então suas informações de faturamento de acordo com recursos com os mesmos valores de chave de tag para ver o custo de recursos combinados. Por exemplo, você pode marcar vários domínios de OpenSearch serviço com pares de valores-chave e, em seguida, organizar suas informações de cobrança para ver o custo total de cada domínio em vários serviços. Para saber mais, consulte Como usar tags de alocação de custos na documentação do AWS Billing and Cost Management.

> **Note**
>
> As tags são armazenados em cache para finalidade de autorização. Por esse motivo, as adições e atualizações de tags nos domínios OpenSearch de serviço podem levar alguns minutos até serem disponibilizadas.

Marcação de domínios (console)

O console é a maneira mais simples marcar um domínio com tags.

Para criar uma tag (console)

1. Vá para https://aws.amazon.com, e escolha Sign In to the Console (Faça login no Console).
2. Em Analytics, escolha Amazon OpenSearch Service.
3. Selecione o domínio ao qual você deseja adicionar tags e vá para guia Tags (Etiquetas).

4. Escolha Manage (Gerenciar) e Add new tag (Adicionar nova tag).
5. Insira uma chave de tag e um valor opcional.
6. Escolha Salvar.

Para excluir uma tag, siga as mesmas etapas e escolha Remover na página Gerenciar tags.

Para saber mais sobre como usar o console para trabalhar com tags, consulte Editor de tags no Guia de conceitos básicos do Console de Gerenciamento da AWS .

Marcação de domínios (AWS CLI)

Você pode criar tags de recursos usando o --add-tags comando AWS CLI with the.

Sintaxe

```
add-tags --arn=<domain_arn> --tag-list Key=<key>,Value=<value>
```

Parâmetro	Description
`--arn`	Nome do recurso da Amazon para o domínio do OpenSearch serviço ao qual a tag está anexada.
`--tag-list`	Conjunto de pares de chave/valor separados por espaço no seguinte formato: `Key=<key>,Value=<value>`

Exemplo

O exemplo a seguir cria duas tags para o domínio logs:

```
aws opensearch add-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-list
 Key=service,Value=OpenSearch Key=instances,Value=m3.2xlarge
```

Você pode remover tags de um domínio OpenSearch de serviço usando o --remove-tags comando.

Sintaxe

```
remove-tags --arn=<domain_arn> --tag-keys Key=<key>,Value=<value>
```

Parâmetro	Description
`--arn`	Nome de recurso da Amazon (ARN) para o domínio do OpenSearch serviço ao qual a tag está anexada.
`--tag-keys`	Conjunto de pares de valores-chave separados por espaço que você deseja remover do domínio do Serviço. OpenSearch

Exemplo

O exemplo a seguir remove duas tags do domínio logs que foram criadas no exemplo anterior:

```
aws opensearch remove-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-keys service instances
```

Você pode visualizar as tags existentes para um domínio OpenSearch de serviço com o --list-tags comando:

Sintaxe

```
list-tags --arn=<domain_arn>
```

Parâmetro	Description
`--arn`	Nome de recurso da Amazon (ARN) para o domínio do OpenSearch serviço ao qual as tags estão anexadas.

Exemplo

O exemplo a seguir lista todas as tags de recurso para o domínio logs:

```
aws opensearch list-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs
```

Marcação de domínios (AWS SDKs)

O AWS SDKs (exceto o Android e o iOS SDKs) suporta todas as ações definidas na Amazon OpenSearch Service API Reference`AddTags`, incluindo as `RemoveTags` operações`ListTags`,

e. Para obter mais informações sobre como instalar e usar o AWS SDKs, consulte Kits AWS de desenvolvimento de software.

Python

Este exemplo usa o cliente Python de OpenSearchServicebaixo nível do AWS SDK para Python (Boto) para adicionar uma tag a um domínio, listar a tag anexada ao domínio e remover uma tag do domínio. É necessário fornecer valores para `DOMAIN_ARN`, `TAG_KEY` e `TAG_VALUE`.

```
import boto3
from botocore.config import Config  # import configuration

DOMAIN_ARN = ''  # ARN for the domain. i.e "arn:aws:es:us-east-1:123456789012:domain/
my-domain
TAG_KEY = ''  # The name of the tag key. i.e 'Smileyface'
TAG_VALUE = ''  # The value assigned to the tag. i.e 'Practicetag'

# defines the configurations parameters such as region

my_config = Config(region_name='us-east-1')
client = boto3.client('opensearch', config=my_config)


# defines the client variable

def addTags():
    """Adds tags to the domain"""

    response = client.add_tags(ARN=DOMAIN_ARN,
                               TagList=[{'Key': TAG_KEY,
                                         'Value': TAG_VALUE}])

    print(response)


def listTags():
    """List tags that have been added to the domain"""

    response = client.list_tags(ARN=DOMAIN_ARN)
    print(response)


def removeTags():
```

```
    """Remove tags that have been added to the domain"""

    response = client.remove_tags(ARN=DOMAIN_ARN, TagKeys=[TAG_KEY])

    print('Tag removed')
    return response
```

Execução de ações administrativas em domínios OpenSearch do Amazon Service

O Amazon OpenSearch Service oferece várias opções administrativas que fornecem controle granular se você precisar solucionar problemas com seu domínio. Essas opções incluem a capacidade de reiniciar o OpenSearch processo em um nó de dados e a capacidade de reiniciar um nó de dados.

OpenSearch O serviço monitora os parâmetros de integridade do nó e, quando há anomalias, toma ações corretivas para manter os domínios estáveis. Com as opções administrativas para reiniciar o OpenSearch processo em um nó e reiniciar o próprio nó, você tem controle sobre algumas dessas ações de mitigação.

Você pode usar o Console de gerenciamento da AWS, AWS CLI, ou o AWS SDK para realizar essas ações. As seções a seguir abordam como realizar essas ações com o console.

Reiniciando o OpenSearch processo em um nó no Amazon Service OpenSearch

Para reiniciar o OpenSearch processo em um nó

1. Navegue até o console OpenSearch de serviço emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Domínios. Escolha o nome do domínio que você deseja atualizar.
3. Depois que a página de detalhes do domínio for aberta, navegue até a guia Integridade da instância.
4. Em Nós de dados, selecione o botão ao lado do nó no qual você deseja reiniciar o processo.
5. Selecione o menu suspenso Ações e escolha Reiniciar OpenSearch/Elasticsearch processo.
6. Escolha Confirmar no modal.

7. Para ver o status da ação que você iniciou, selecione o nome do nó. Depois que a página de detalhes do nó for aberta, escolha a guia Eventos abaixo do nome do nó para ver uma lista de eventos associados a esse nó.

Reinicializando um nó de dados no Amazon Service OpenSearch

Como reinicializar um nó de dados

1. Navegue até o console OpenSearch de serviço emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Domínios. Escolha o nome do domínio que você deseja atualizar.
3. Depois que a página de detalhes do domínio for aberta, navegue até a guia Integridade da instância.
4. Em Nós de dados, selecione o botão ao lado do nó no qual você deseja reiniciar o processo.
5. Selecione o menu suspenso Ações e escolha Nó de reinicialização.
6. Escolha Confirmar no modal.
7. Para ver o status da ação que você iniciou, selecione o nome do nó. Depois que a página de detalhes do nó for aberta, escolha a guia Eventos abaixo do nome do nó para ver uma lista de eventos associados a esse nó.

Reiniciando o processo de OpenSearch painéis em um nó no Amazon Service OpenSearch

Você pode reiniciar o processo do OpenSearch Dashboards (anteriormente Kibana) para se recuperar de problemas como interface congelada, falhas de carregamento ou visualizações que não respondem. A opção de reiniciar os OpenSearch painéis só está disponível para o nó que está executando ativamente o processo de painéis. Na maioria dos domínios OpenSearch de serviço, esse processo é executado em nós coordenadores dedicados, não em nós de dados. Por isso, quando você abre a lista suspensa Ações no console, a opção normalmente aparece apenas para nós coordenadores. Para saber mais, consulte the section called “Nodos de coordenação dedicados”.

Esse comportamento depende de como domínio está configurado.

- Domínios com nós coordenadores dedicados: o processo do Dashboards é executado exclusivamente nesses nós e ele mostram apenas a opção de reiniciar.

- Domínios sem nós coordenadores dedicados: em configurações mais simples ou mais antigas, o Dashboards pode ser executado em um nó de dados, e a opção de reiniciar aparece neles.
- Nós principais: esses nós servem exclusivamente para gerenciar o estado e as eleições de cluster. Eles não executam o Dashboards e nunca mostram a opção de reiniciar.

Para determinar qual nó está executando o processo do Dashboards, navegue até a seção Configuração de cluster do domínio e revise os perfis dos nós. A opção de reiniciar está disponível apenas para o nó que hospeda o processo do Dashboards.

Para reiniciar o processo do Dashboards em um nó

1. Navegue até o console OpenSearch de serviço emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Domínios. Escolha o nome do domínio que você deseja atualizar.
3. Depois que a página de detalhes do domínio for aberta, navegue até a guia Integridade da instância.
4. Na seção dos nós que estão executando o processo do Dashboards, selecione o botão ao lado do nó em que você deseja reiniciar o processo.
5. Selecione o menu suspenso Ações e escolha Reiniciar Dashboard/Kibana processo.
6. Escolha Confirmar no modal.
7. Para ver o status da ação que você iniciou, selecione o nome do nó. Depois que a página de detalhes do nó for aberta, escolha a guia Eventos abaixo do nome do nó para ver uma lista de eventos associados a esse nó.

Limitações

As opções administrativas têm as seguintes limitações:

- As opções administrativas são compatíveis com as versões 7.x e superiores do Elasticsearch.
- As opções administrativas não oferecem suporte a domínios com Multi-AZ com modo de espera ativado.
- A reinicialização do processo OpenSearch e do Elasticsearch e a reinicialização do nó de dados são suportadas em domínios com três ou mais nós de dados.
- O suporte ao processo do Dashboards e Kibana é compatível com domínios com dois ou mais nós de dados.

- Para reiniciar o OpenSearch processo em um nó ou reinicializar um nó, o domínio não deve estar no estado vermelho e todos os índices devem ter réplicas configuradas.

Trabalhando com consultas diretas do Amazon OpenSearch Service

Use a consulta direta do Amazon OpenSearch Service para analisar dados no Amazon CloudWatch Logs, no Amazon S3 e no Amazon Security Lake sem criar canais de ingestão. Essa integração sem ETL permite que você consulte dados no local usando OpenSearch SQL ou PPL e os explore no Discover.

Para começar, configure sua fonte de dados no console OpenSearch de serviço. Para o Amazon S3, use conexões de domínio e crie tabelas com SQL no Query Workbench. CloudWatch O Logs e o Security Lake usam fontes e AWS Glue Data Catalog tabelas pré-configuradas.

Preços de consultas diretas

Ao usar consultas diretas do OpenSearch Serviço, você incorrerá em cobranças separadas pelo OpenSearch Serviço e pelos recursos usados para processar e armazenar seus dados no Amazon S3, no Amazon Logs e no CloudWatch Amazon Security Lake. Ao executar consultas diretas, você verá cobranças por unidades de OpenSearch computação (OCUs) por hora, listadas como tipo de uso de DirectQuery OCU em sua fatura.

Existem dois tipos de consultas diretas: consultas interativas e consultas de visão indexada.

- As consultas interativas são usadas para preencher o seletor de dados e realizar análises em seus dados no S3, CloudWatch Logs ou Security Lake.

  Para consultas diretas do Amazon S3, quando você executa uma nova consulta a partir do Discover, o OpenSearch Service inicia uma nova sessão que dura no mínimo três minutos. OpenSearch O serviço mantém essa sessão ativa para garantir que as consultas subsequentes sejam executadas rapidamente.

  Para consultas do CloudWatch Logs e do Security Lake, o OpenSearch Service processa cada consulta com uma tarefa pré-aquecida separada, sem manter uma sessão prolongada.
- As consultas de visualização indexada usam computação para manter visualizações indexadas no Serviço. OpenSearch Essas consultas geralmente demoram mais porque ingerem uma quantidade variável de dados em um índice nomeado. Indexando os dados, você pode acelerar futuras consultas interativas ou desbloquear recursos avançados de analytics, como painéis ou alertas, que exigem um índice para referência.

Para fontes de dados do Amazon S3, os dados indexados são armazenados em um domínio com base no tipo de instância adquirido. Para fontes de dados conectadas ao CloudWatch Logs e ao Security Lake, os dados indexados são armazenados em uma coleção OpenSearch sem servidor, na qual você é cobrado pelos dados indexados (indexingOCU), pelos dados pesquisados (SearchOCU) e pelos dados armazenados em GB.

Para obter mais informações, consulte as seções Direct Query e Serverless no Amazon OpenSearch Service Pricing.

Limitações das consultas diretas

Limitações gerais

As limitações a seguir se aplicam às consultas diretas do OpenSearch Serviço.

- Alguns tipos de dados não são compatíveis. Os tipos de dados compatíveis estão limitados a Parquet, CSV e JSON.
- Se a estrutura de seus dados mudar com o tempo, você precisará atualizar suas visualizações ou out-of-the-box integrações indexadas para considerar as mudanças na estrutura de dados.
- AWS CloudFormation os modelos ainda não são compatíveis.
- OpenSearch As instruções SQL e OpenSearch PPL têm limitações diferentes ao trabalhar com OpenSearch índices em comparação com o uso de consulta direta. A consulta direta oferece suporte a comandos avançados JOINs, como subconsultas e pesquisas, enquanto o suporte a esses comandos em OpenSearch índices é limitado ou inexistente. Para obter mais informações, consulte the section called “Comandos SQL e PPL compatíveis”.

Limitações do Amazon S3

Quando você consulta dados diretamente no Amazon S3, as seguintes limitações adicionais se aplicam:

- A consulta direta para S3 está disponível somente em domínios OpenSearch de serviço que executam a OpenSearch versão 2.13 ou posterior e requer acesso a. AWS Glue Data Catalog AWS Glue Data Catalog As tabelas existentes devem ser recriadas usando SQL no OpenSearch Query Workbench.

- A consulta direta para S3 exige que você especifique um bucket de ponto de verificação no Amazon S3. Esse bucket mantém o estado das visões indexadas, incluindo o horário da última atualização e os últimos dados ingeridos.
- Seu OpenSearch domínio e AWS Glue Data Catalog deve estar no mesmo Conta da AWS. Seu bucket do S3 pode estar em uma conta diferente (requer que a condição seja adicionada à sua política do IAM), mas deve estar no Região da AWS mesmo que seu domínio.
- OpenSearch As consultas diretas de serviço com o S3 oferecem suporte somente às tabelas do Spark geradas a partir do Query Workbench. As tabelas geradas no Athena AWS Glue Data Catalog ou no Athena não são compatíveis com o streaming do Spark, que é necessário para manter as visualizações indexadas.
- OpenSearch os tipos de instância têm limitações de carga útil de rede de 10 MiB ou 100 MiB, dependendo do tipo de instância específico que você escolher.

Limitações do Amazon CloudWatch Logs

Se você estiver consultando dados diretamente no CloudWatch Logs, as seguintes limitações adicionais se aplicam:

- A integração direta de consultas com o CloudWatch Logs está disponível somente nas coleções OpenSearch de serviços e na interface OpenSearch do usuário.
- OpenSearch As coleções sem servidor têm limitações de carga útil em rede de 100 MiB.
- CloudWatch O Logs oferece suporte ao VPC Flow e às integrações de AWS WAF painéis instaladas a partir do console. CloudTrail

Limitações do Amazon Security Lake

Quando você consulta dados diretamente no Security Lake, as seguintes limitações adicionais se aplicam:

- A integração direta de consultas com o Security Lake está disponível somente nas coleções OpenSearch de serviços e na interface OpenSearch do usuário.
- OpenSearch As coleções sem servidor têm limitações de carga útil em rede de 100 MiB.
- O gerenciamento de tabelas do Security Lake é realizado no Lake Formation.
- O Security Lake apenas é compatível com visões materializadas como visões indexadas. Índices abrangentes não são compatíveis.

Recomendações para usar consultas diretas no Amazon Service OpenSearch

Esta página fornece recomendações para o uso de consultas diretas do Amazon OpenSearch Service para analisar dados do CloudWatch Logs, do Amazon S3 e do Amazon Security Lake. Essas práticas recomendadas ajudam a otimizar a performance e garantir consultas eficientes sem necessidade de ingestão ou duplicação de dados.

Tópicos

- Recomendações gerais
- Recomendações para o Amazon S3
- CloudWatch Recomendações de registros
- Recomendações sobre o Security Lake

Recomendações gerais

Recomendamos fazer o seguinte ao usar a consulta direta:

- Use a função `COALESCE SQL` para lidar com colunas ausentes e garantir o retorno dos resultados.

- Use limites em suas consultas para garantir que você não esteja extraindo muitos dados.
- Se você planeja analisar o mesmo conjunto de dados várias vezes, crie uma visualização indexada para ingerir e indexar totalmente os dados OpenSearch e eliminá-los quando tiver concluído a análise.
- Elimine trabalhos de aceleração e índices quando eles não forem mais necessários.
- Consultas contendo nomes de campo idênticos, que diferem somente em maiúsculas e minúsculas (como `field1` e `FIELD1`) não são compatíveis.

  Por exemplo, as seguintes consultas não são compatíveis:

```
Select AWSAccountId, AwsAccountId from LogGroup
Select a.@LogStream, b.@logStream from Table A INNER Join Table B ona.id = b.id
```

  No entanto, a seguinte consulta é compatível porque o nome do campo (@logStream) é idêntico em ambos os grupos de logs:

```
Select a.@logStream, b.@logStream from Table A INNER Join Table B on a.id = b.id
```

- As funções e expressões devem operar em nomes de campo e fazer parte de uma instrução `SELECT` com um grupo de logs especificado na cláusula `FROM`.

  Por exemplo, esta consulta não é compatível:

  ```
  SELECT cos(10) FROM LogGroup
  ```

  Esta consulta é compatível:

  ```
  SELECT cos(field1) FROM LogGroup
  ```

Recomendações para o Amazon S3

Se você estiver usando o Amazon OpenSearch Service para direcionar dados de consulta no Amazon S3, também recomendamos o seguinte:

- Ingira dados no Amazon S3 usando formatos de partição de ano, mês, dia e hora para acelerar as consultas.
- Ao criar índices ignorados, use filtros Bloom para campos com alta cardinalidade e min/max índices para campos com grandes intervalos de valores. Para campos de alta cardinalidade, considere usar uma abordagem baseada em valores para melhorar a eficiência das consultas.
- Use o Index State Management para manter o armazenamento de visões materializadas e índices abrangentes.

CloudWatch Recomendações de registros

Se você estiver usando o Amazon OpenSearch Service para direcionar dados de consulta em CloudWatch Logs, também recomendamos o seguinte:

- Ao pesquisar vários grupos de logs em uma consulta, use a sintaxe adequada. Para saber mais, consulte the section called “Funções de vários grupos de logs”.
- Ao usar comandos SQL ou PPL, coloque certos campos entre crases para consultá-los com êxito. Acentos graves são necessários em campos com caracteres especiais (não alfabéticos e não numéricos). Por exemplo, coloque `@message`, `Operation.Export,` e `Test::Field` entre

acentos graves. Não é necessário colocar entre acentos graves campos com nomes puramente alfabéticos.

Exemplo de consulta com campos simples:

```
SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
LIMIT 1000;
```

Consulta semelhante com acentos graves acrescentados:

```
SELECT `@SessionToken`, `@Operation`, `@StartTime` FROM `LogGroup-A`
LIMIT 1000;
```

Recomendações sobre o Security Lake

Se você estiver usando o Amazon OpenSearch Service para direcionar dados de consulta no Security Lake, também recomendamos o seguinte:

- Verifique o status do Security Lake e garanta que ele esteja funcionando bem, sem nenhum problema. Para obter as etapas detalhadas de solução de problemas, consulte Solução de problemas de status de data lake no Guia do usuário do Amazon Security Lake.
- Verifique seu acesso a consultas:
  - Se você for consultar o Security Lake de uma conta que não seja a conta de administrador delegado do Security Lake, configure um assinante com acesso a consultas no Security Lake.
  - Se você estiver consultando o Security Lake da mesma conta, verifique se há alguma mensagem no Security Lake sobre o registro de seus buckets S3 gerenciados com. LakeFormation
- Explore os modelos de consulta e os painéis pré-compilados para acionar sua análise rapidamente.
- Familiarize-se com o Open Cybersecurity Schema Framework (OCSF) e o Security Lake:
  - Analise exemplos de mapeamento de esquema para AWS fontes no repositório OCSF GitHub
  - Saiba como consultar o Security Lake de forma eficaz visitando as consultas do Security Lake para a versão de AWS origem 2 (OCSF 1.1.0)
  - Melhore a performance das consultas usando partições: `accountid`, `region` e `time_dt`

- Familiarize-se com a sintaxe SQL, com a qual o Security Lake é compatível para consultas. Para saber mais, consulte the section called “Comandos SQL compatíveis”.

Cotas de consultas diretas

Sua conta tem as seguintes cotas relacionadas às consultas diretas do OpenSearch Serviço.

Cotas para o Amazon S3

Cada vez que você inicia uma consulta em uma fonte de dados do Amazon S3 OpenSearch , o Service abre uma sessão e a mantém ativa por pelo menos três minutos. Isso reduz a latência da consulta, eliminando o tempo gasto para iniciar a sessão nas consultas subsequentes.

Description	Máximo	Pode substituir
Conexões por domínio	10	Sim
Fontes de dados por domínio	20	Sim
Índices por domínio	5	Sim
Sessões simultâneas por fonte de dados	10	Sim
Máximo de OCU por consulta	60	Sim
Tempo máximo de execução da consulta (minutos)	30	Sim
Máximo OCUs por aceleração	20	Sim
Armazenamento temporário máximo	20	Sim

Cotas para registros CloudWatch

> **Note**
>
> Se você deseja realizar consultas diretas usando o CloudWatch Logs Insights, consulte. the section called “CloudWatch Informações sobre registros”

Description	Valor	Limite flexível?	Observações
Limite de TPS em nível de conta em consultas diretas APIs	3 TPS	Sim	
Número máximo de fontes de dados	20	Sim	O limite é por Conta da AWS.
Máximo de índices ou visões materiali zadas com atualização automática	30	Sim	O limite é por fonte de dados.
Máximo de consultas simultâneas	30	Sim	O limite é por fonte de dados e se aplica às consultas em estado pendente ou em execução. Inclui consultas interativas (por exemplo, comandos de recuperação de dados como`SELECT`) e consultas de índice (por exemplo, operações como`CREATE`/). `ALTER`
Máximo de OCUs simultâneas por consulta	512	Sim	OpenSearch Unidades de computação (OCU). Limite baseado em 15 executores e 1 driver, cada um com 16 vCPU e 32 GB de memória. Represent a o poder de processamento simultâneo.

Description	Valor	Limite flexível?	Observações
Tempo máximo de execução de consulta em minutos	60	Não	O limite se aplica às consultas OpenSearch PPL/SQL no Logs Insights. CloudWatch
Período para limpar a consulta obsoleta IDs	90 dias	Sim	Esse é o período após o qual o OpenSearch Serviço limpa os metadados da consulta de entradas mais antigas. Por exemplo, ligar GetDirectQuery ou GetDirectQueryResult falhar em consultas com mais de 90 dias.

Cotas para o Security Lake

Description	Valor	Limite flexível?	Observações
Limite de TPS em nível de conta em consultas diretas APIs	3 TPS	Sim	
Número máximo de fontes de dados	20	Sim	O limite é por Conta da AWS.
Máximo de índices ou visões materializadas com atualização automática	30	Sim	O limite se aplica por fonte de dados. Inclui apenas índices e visualizações materializadas (MVs) com atualização automática definida como verdadeira.
Máximo de consultas simultâneas	30	Sim	O limite se aplica a consultas em estado pendente ou em execução. Inclui consultas interativas (por exemplo, comandos de recuperação de dados como

Description	Valor	Limite flexível?	Observações
			SELECT) e consultas de índice (por exemplo, operações como CREATE/ALTER/DROP).
Máximo de OCUs simultâneas por consulta	512	Sim	OpenSearch Unidades de computação (OCU). Limite baseado em 15 executores e 1 driver, cada um com 16 vCPU e 32 GB de memória. Representa o poder de processamento simultâne o.
Tempo máximo de execução de consulta em minutos	30	Não	Aplica-se somente a consultas interativas (por exemplo, comandos de recuperação de dados como SELECT). Para consultas REFRESH, o limite são 6 horas.
Período para limpar a consulta obsoleta IDs	90 dias	Sim	Esse é o período após o qual o OpenSearc h Serviço limpa os metadados da consulta de entradas mais antigas. Por exemplo, ligar GetDirectQuery ou GetDirectQueryResult falhar em consultas com mais de 90 dias.

Suportado Regiões da AWS

O seguinte Regiões da AWS é compatível com consultas diretas de OpenSearch serviço no Amazon S3 CloudWatch , Logs e Security Lake:

Disponível Regiões da AWS para Amazon S3

- Ásia-Pacífico (Hong Kong)
- Ásia-Pacífico (Mumbai)
- Ásia-Pacífico (Seul)
- Ásia-Pacífico (Singapura)
- Ásia-Pacífico (Sydney)
- Ásia-Pacífico (Tóquio)

- Canadá (Central)
- Europa (Frankfurt)
- Europa (Irlanda)
- Europa (Estocolmo)
- Leste dos EUA (Norte da Virgínia)
- Leste dos EUA (Ohio)
- Oeste dos EUA (Oregon)

Disponível Regiões da AWS para CloudWatch registros

- Ásia-Pacífico (Mumbai)
- Ásia Pacífico (Hong Kong)
- Ásia-Pacífico (Osaka)
- Ásia-Pacífico (Seul)
- Ásia-Pacífico (Singapura)
- Ásia-Pacífico (Sydney)
- Ásia-Pacífico (Tóquio)
- Canadá (Central)
- Europa (Frankfurt)
- Europa (Irlanda)
- Europa (Estocolmo)
- Europa (Milão)
- Europa (Espanha)
- Leste dos EUA (Norte da Virgínia)
- Leste dos EUA (Ohio)
- Oeste dos EUA (Oregon)
- Oeste dos EUA (N. da Califórnia)
- Europa (Paris)
- Europa (Londres)
- América do Sul (São Paulo)

Disponível Regiões da AWS para Security Lake

- Ásia-Pacífico (Mumbai)
- Ásia-Pacífico (Singapura)
- Ásia-Pacífico (Sydney)
- Ásia-Pacífico (Tóquio)
- Canadá (Central)
- Europa (Frankfurt)
- Europa (Irlanda)
- Europa (Estocolmo)
- Leste dos EUA (Norte da Virgínia)
- Leste dos EUA (Ohio)
- Oeste dos EUA (Oregon)
- Europa (Paris)
- Europa (Londres)
- América do Sul (São Paulo)

Consultando diretamente os dados do Amazon S3 no Serviço OpenSearch

Esta seção o guiará pelo processo de criação e configuração de uma integração de fonte de dados no Amazon OpenSearch Service, permitindo que você consulte e analise com eficiência seus dados armazenados no Amazon S3.

Nas páginas a seguir, você aprenderá a configurar uma fonte de dados de consulta direta do Amazon S3, navegar pelos pré-requisitos necessários e seguir os step-by-step procedimentos usando a API e a Service API. Console de gerenciamento da AWS OpenSearch Também aborda as próximas etapas importantes, incluindo mapeamento de AWS Glue Data Catalog funções e configuração de controles de acesso em OpenSearch painéis.

Tópicos

- Criação de uma integração de fonte de dados do Amazon S3 no Service OpenSearch

- Configurando e consultando uma fonte de dados do S3 em painéis OpenSearch

Criação de uma integração de fonte de dados do Amazon S3 no Service OpenSearch

Você pode criar uma nova fonte de dados de consulta direta do Amazon S3 para o OpenSearch Serviço por meio da ou da Console de gerenciamento da AWS API. Cada nova fonte de dados usa o AWS Glue Data Catalog para gerenciar tabelas que representam os buckets do Amazon S3.

Tópicos

- Pré-requisitos
- Procedimento
- Próximas etapas
- Mapeie a AWS Glue Data Catalog função
- Recursos adicionais do

Pré-requisitos

Antes de começar, certifique-se de ter revisto a seguinte documentação:

- the section called “Limitações do Amazon S3”
- the section called “Recomendações para o Amazon S3”
- the section called “Cotas para o Amazon S3”

Antes de criar uma fonte de dados, você deve ter os seguintes recursos em seu Conta da AWS:

- Um OpenSearch domínio com a versão 2.13 ou posterior. Essa é a base para configurar a integração de consulta diretas. Para obter instruções para essa configuração, consulte the section called “Criação OpenSearch de domínios de serviço”.
- Um ou mais buckets do S3. Será necessário especificar os buckets que contêm os dados que você deseja consultar e um bucket para armazenar seus pontos de verificação de consulta. Para saber mais sobre como criar um bucket do S3, consulte Criar um bucket no Guia do usuário Amazon S3.
- (Opcional) Uma ou mais AWS Glue tabelas.Para consultar dados no Amazon S3 é necessário que você tenha tabelas configuradas no AWS Glue Data Catalog que apontem para os dados

do S3. Você deve criar as tabelas usando o OpenSearch Query Workbench. As tabelas do Hive existentes não são compatíveis.

Se esta for a primeira vez que você configura uma fonte de dados do Amazon S3, deverá criar uma fonte de dados administrativa para configurar todas as tabelas do AWS Glue Data Catalog . Você pode fazer isso instalando OpenSearch out-of-the-box integrações ou usando o OpenSearch Query Workbench para criar tabelas SQL personalizadas para casos de uso avançados. Para ver exemplos sobre como criar tabelas para registros de VPC e AWS WAF, consulte a documentação sobre GitHub VPC, e. CloudTrail CloudTrailAWS WAF Depois de criar as tabelas, você pode criar novas fontes de dados do Amazon S3 e restringir o acesso a tabelas limitadas.

- (Opcional) Um perfil do IAM criado manualmente.Você pode usar esse perfil para gerenciar o acesso à sua fonte de dados. Como alternativa, você pode fazer com que o OpenSearch Service crie uma função para você automaticamente com as permissões necessárias. Se você escolher usar um perfil do IAM criado manualmente, siga as orientações em the section called “Permissões necessárias para perfis do IAM criados manualmente”.

Procedimento

Você pode configurar uma fonte de dados de consulta direta em um domínio com a API Console de gerenciamento da AWS ou a OpenSearch Service API.

Para configurar uma fonte de dados usando o Console de gerenciamento da AWS

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, selecione Domínios.
3. Selecione o domínio para o qual configurar uma nova fonte de dados. Isso abre a página de detalhes do domínio.
4. Escolha a guia Conexões abaixo dos detalhes gerais do domínio e localize a seção Consulta direta.
5. Escolha Configurar fontes de dados.
6. Insira um nome e uma descrição opcional para a nova fonte de dados.
7. Escolha o Amazon S3 com. AWS Glue Data Catalog
8. Em Configurações de permissão de acesso do IAM, escolha como gerenciar o acesso.

    a. Se quiser criar automaticamente um perfil para essa fonte de dados, siga estas etapas:

i. Selecione Criar uma novo perfil.

ii. Insira um nome para a função do IAM.

iii. Selecione um ou mais buckets do S3 que contenham dados que você deseja consultar.

iv. Selecione um bucket do S3 de pontos de verificação para armazenar os pontos de verificação de consulta.

v. Selecione um ou mais AWS Glue bancos de dados ou tabelas para definir quais dados podem ser consultados. Se as tabelas ainda não tiverem sido criadas, forneça acesso ao banco de dados padrão.

b. Se quiser usar um perfil existente gerenciado por você, siga estas etapas:

i. Selecione Usar perfil existente.

ii. Selecione um perfil existente no menu suspenso.

> **Note**
>
> Ao usar seu próprio perfil, confira se ele tem todas as permissões necessárias anexando as políticas necessárias com o console do IAM. Para saber mais, revise o exemplo de política em the section called “Permissões necessárias para perfis do IAM criados manualmente”.

9. Selecione Configurar. Isso abre a tela de detalhes da fonte de dados com uma URL dos OpenSearch painéis. Navegue até esse URL para concluir as próximas etapas.

OpenSearch API de serviço

Use a operação AddDataSourceda API para criar uma nova fonte de dados em seu domínio.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource

{
   "DataSourceType": {
        "S3GlueDataCatalog": {
            "RoleArn": "arn:aws:iam::account-id:role/role-name"
        }
    }
   "Description": "data-source-description",
```

```
    "Name": "my-data-source"
}
```

Próximas etapas

Visite os OpenSearch painéis

Depois de criar uma fonte de dados, o OpenSearch Service fornece um link de OpenSearch painéis. Você pode usar isso para configurar o controle de acesso, definir tabelas, instalar out-of-the-box integrações e consultar seus dados.

Para obter mais informações, consulte the section called “Configurar uma fonte de dados do S3”.

Mapeie a AWS Glue Data Catalog função

Se você ativou o controle de acesso refinado após criar uma fonte de dados, deverá mapear usuários não administradores para uma função do IAM com AWS Glue Data Catalog acesso para executar consultas diretas. Para criar manualmente um perfil do `glue_access` de backend que possa ser mapeado para o perfil do IAM, realize as seguintes etapas:

> **Note**
>
> Índices são usados para qualquer consulta na fonte de dados. Um usuário com acesso para leitura ao índice de solicitações de uma determinada fonte de dados pode ler todas as consultas nessa fonte. Um usuário com acesso para leitura ao índice de resultados pode ler os resultados de todas as consultas nessa fonte de dados.

1. No menu principal em OpenSearch Painéis, escolha Segurança, Funções e Criar funções.
2. Chame o perfil de glue_access.
3. Para Permissões de cluster, selecione `indices:data/write/bulk*`, `indices:data/read/scroll`, `indices:data/read/scroll/clear`.
4. Em Índice, insira os seguintes índices aos quais você deseja conceder acesso ao usuário com o perfil:
   - `.query_execution_request_`*<name of data source>*
   - `query_execution_result_`*<name of data source>*

- `.async-query-scheduler`
- `flint_*`

5. Para Permissões de índice, selecione `indices_all`.
6. Escolha Criar.
7. Escolha Usuários mapeados e Gerenciar mapeamento.
8. Em Perfis de backend, adicione o ARN do perfil do AWS Glue que precisa de permissão para chamar seu domínio.

```
arn:aws:iam::account-id:role/role-name
```

9. Selecione Mapa e confirme se o perfil aparece em Usuários mapeados.

Para saber mais sobre o mapeamento de perfis, consulte the section called “Mapear funções em usuários”.

Recursos adicionais do

Permissões necessárias para perfis do IAM criados manualmente

Ao criar uma fonte de dados para o domínio, você escolhe um perfil do IAM para gerenciar o acesso aos dados. Você tem duas opções:

1. Criar um novo perfil do IAM automaticamente
2. Use um perfil do IAM existente que você criou manualmente

Se usar um perfil criado manualmente, você precisará anexar as permissões corretas a ele. As permissões devem permitir o acesso à fonte de dados específica e permitir que o OpenSearch Serviço assuma a função. Isso é necessário para que o OpenSearch Serviço possa acessar e interagir com seus dados com segurança.

O exemplo de política a seguir demonstra as permissões de privilégio mínimo necessárias para criar e gerenciar uma fonte de dados. Se você tiver permissões mais amplas, como `s3:*` ou a política `AdminstratorAccess`, essas permissões abrangem as permissões de privilégio mínimo na política de amostra.

No exemplo de política a seguir, *`placeholder text`* substitua o por suas próprias informações.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"HttpActionsForOpenSearchDomain",
         "Effect":"Allow",
         "Action":"es:ESHttp*",
"Resource":"arn:aws:es:us-east-1:111122223333:domain/example.com/*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryReadAllS3Buckets",
         "Effect":"Allow",
         "Action":[
            "s3:GetObject",
            "s3:GetObjectVersion",
            "s3:ListBucket"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"111122223333"
            }
         },
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchDirectQueryGlueCreateAccess",
         "Effect":"Allow",
         "Action":[
            "glue:CreateDatabase",
            "glue:CreatePartition",
            "glue:CreateTable",
            "glue:BatchCreatePartition"
         ],
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryModifyAllGlueResources",
         "Effect":"Allow",
         "Action":[
            "glue:DeleteDatabase",
            "glue:DeletePartition",
```

```
            "glue:DeleteTable",
            "glue:GetDatabase",
            "glue:GetDatabases",
            "glue:GetPartition",
            "glue:GetPartitions",
            "glue:GetTable",
            "glue:GetTableVersions",
            "glue:GetTables",
            "glue:UpdateDatabase",
            "glue:UpdatePartition",
            "glue:UpdateTable",
            "glue:BatchGetPartition",
            "glue:BatchDeletePartition",
            "glue:BatchDeleteTable"
         ],
         "Resource":[
            "arn:aws:glue:us-east-1:111122223333:table/*",
            "arn:aws:glue:us-east-1:111122223333:database/*",
            "arn:aws:glue:us-east-1:111122223333:catalog",
            "arn:aws:es:us-east-1:111122223333:domain/domain_name"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"111122223333"
            }
         }
      },
      {
         "Sid":"ReadAndWriteActionsForS3CheckpointBucket",
         "Effect":"Allow",
         "Action":[
            "s3:ListMultipartUploadParts",
            "s3:DeleteObject",
            "s3:GetObject",
            "s3:PutObject",
            "s3:GetBucketLocation",
            "s3:ListBucket"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"111122223333"
            }
         },
         "Resource":[
```

```
            "arn:aws:s3:::amzn-s3-demo-bucket",
            "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
      }
   ]
}
```

Para oferecer suporte a buckets do Amazon S3 em contas diferentes, você precisará incluir uma condição na política do Amazon S3 e adicionar a conta apropriada.

Na condição de amostra a seguir, *placeholder text* substitua o por suas próprias informações.

```
"Condition": {
               "StringEquals": {
                   "aws:ResourceAccount": "{{accountId}}"
               }
```

O perfil também deve ter a seguinte política de confiança, que especifica o ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
       {
          "Effect":"Allow",
          "Principal":{
             "Service": "directquery.opensearchservice.amazonaws.com"
          },
          "Action":"sts:AssumeRole"
       }
     ]
}
```

Para obter instruções sobre como criar o perfil, consulte Criar um perfil usando políticas de confiança personalizadas.

Se você tiver um controle de acesso refinado ativado no OpenSearch Serviço, uma nova função de controle de acesso OpenSearch refinado será criada automaticamente para sua fonte de dados. O

nome da nova função de controle de acesso refinada será `AWS OpenSearchDirectQuery <name of data source>`.

Por padrão, a função tem acesso somente aos índices da fonte de dados de consulta direta. Embora você possa configurar a função para limitar ou conceder acesso à sua fonte de dados, é recomendável não ajustar o acesso dessa função. Se você excluir a fonte de dados, essa função será excluída. Isso removerá o acesso de outros usuários se eles estiverem mapeados para a função.

Configurando e consultando uma fonte de dados do S3 em painéis OpenSearch

Depois de criou a fonte de dados, é possível definir configurações de segurança, suas tabelas do Amazon S3 ou a indexação acelerada de dados. Esta seção mostra vários casos de uso com sua fonte de dados em OpenSearch painéis antes de você consultar seus dados.

Para configurar as seções a seguir, primeiro você deve navegar até sua fonte de dados em OpenSearch Painéis. Na navegação à esquerda, em Gerenciamento, selecione Fontes de dados. Em Gerenciar fontes de dados, selecione o nome da fonte de dados criada no console.

Crie tabelas do Spark usando o Query Workbench

As consultas diretas do OpenSearch Service para o Amazon S3 usam tabelas Spark dentro do. AWS Glue Data Catalog Você pode criar tabelas de dentro do Query Workbench sem precisar sair dos OpenSearch painéis.

Para gerenciar bancos de dados e tabelas existentes na sua fonte de dados ou para criar novas tabelas para usar em consultas diretas, escolha Query Workbench no painel de navegação à esquerda e selecione a fonte de dados do Amazon S3 na lista suspensa de fontes de dados.

Para configurar uma tabela para logs de fluxo da VPC armazenados no S3 no formato Parquet, execute a seguinte consulta:

```
CREATE TABLE
datasourcename.gluedatabasename.vpclogstable (version INT, account_id STRING,
 interface_id STRING,
srcaddr STRING, dstaddr STRING, srcport INT, dstport INT, protocol INT, packets
 BIGINT,
bytes BIGINT, start BIGINT, end BIGINT, action STRING, log_status STRING,
`aws-account-id` STRING, `aws-service` STRING, `aws-region` STRING, year STRING,
```

```
month STRING, day STRING, hour STRING)

USING parquet PARTITIONED BY (aws-account-id, aws-service, aws-region, year, month,
day, hour)

LOCATION "s3://accountnum-vpcflow/AWSLogs"
```

Depois de criar a tabela, execute a consulta a seguir para garantir que ela seja compatível com consultas diretas:

```
MSCK REPAIR TABLE  datasourcename.databasename.vpclogstable
```

Configurar integrações para tipos de AWS log populares

Você pode integrar os tipos de AWS log armazenados no Amazon S3 com OpenSearch o Service. Use OpenSearch painéis para instalar integrações que criam AWS Glue Data Catalog tabelas, consultas salvas e painéis. Essas integrações usam visões indexadas para manter os painéis atualizados.

Para obter instruções sobre como instalar uma integração, consulte Instalação de um ativo de integração na OpenSearch documentação.

Ao selecionar uma integração, certifique-se de que ela tenha a tag `S3 Glue`.

Ao configurar a integração, especifique Conexão do S3 como o tipo de conexão. Depois, selecione a fonte de dados para a integração, a localização dos dados no Amazon S3, o ponto de verificação para gerenciar a indexação de aceleração e os ativos necessários para o seu caso de uso.

> **Note**
>
> Certifique-se de que o bucket S3 para seu ponto de verificação tenha permissões de gravação no local para o ponto de verificação. Sem essas permissões, acelerações da integração não terão sucesso.

Configurar o controle de acesso

Na página de detalhes da fonte de dados, encontre a seção Controles de acesso e escolha Editar. Se o domínio tiver controle de acesso refinado habilitado, escolha Restrito e selecione os perfis

aos quais pretende conceder acesso à nova fonte de dados. Também é possível escolher Somente administrador para que somente o administrador tenha acesso à fonte de dados.

> **Important**
>
> Índices são usados para qualquer consulta na fonte de dados. Um usuário com acesso para leitura ao índice de solicitações de uma determinada fonte de dados pode ler todas as consultas nessa fonte. Um usuário com acesso para leitura ao índice de resultados pode ler os resultados de todas as consultas nessa fonte de dados.

Consultando dados do S3 no Discover OpenSearch

Depois que você configura as tabelas e a aceleração de consulta opcional desejada, pode começar a analisar seus dados. Para consultar os dados, selecione a fonte de dados no menu suspenso. Se você estiver usando o Amazon S3 e os OpenSearch painéis, acesse Discover e selecione o nome da fonte de dados.

Se estiver usando um índice de salto ou não tiver criado um índice, você poderá usar SQL ou PPL para consultar os dados. Se configurou uma visão materializada ou um índice de abrangência, você já tem um índice e pode usar a Dashboards Query Language (DQL) no Dashboards. Você também pode usar o PPL com o plug-in Observability e o SQL com o plug-in Query Workbench. Atualmente, somente os plug-ins Observability e Query Workbench oferecem suporte para PPL e SQL. Para consultar dados usando a API de OpenSearch serviço, consulte a documentação da API assíncrona.

> **Note**
>
> Nem todas as instruções, comandos e funções PPL são compatíveis. Para obter uma lista de comandos compatíveis, consulte the section called “Comandos SQL e PPL compatíveis”. Se você criou uma visão materializada ou um índice abrangente, poderá usar DQL para consultar os dados fornecidos desde que os tenha indexado ali.

Solução de problemas

Pode haver casos em que os resultados não retornem conforme o esperado. Se tiver enfrentando algum problema, lembre-se de seguir as the section called “Recomendações”.

Consultando diretamente os dados do Amazon CloudWatch Logs no Service OpenSearch

Esta seção guiará você pelo processo de criação e configuração de uma integração de fonte de dados no Amazon OpenSearch Service, permitindo que você consulte e analise com eficiência seus dados armazenados no CloudWatch Logs.

Nas páginas a seguir, você aprenderá a configurar uma fonte de dados de consulta direta do CloudWatch Logs, navegar pelos pré-requisitos necessários e seguir os procedimentos usando o. step-by-step Console de gerenciamento da AWS

Tópicos

- Criação de uma integração de fonte de dados do Amazon CloudWatch Logs no OpenSearch Service
- Configurando e consultando uma fonte de dados CloudWatch de registros em painéis OpenSearch

Criação de uma integração de fonte de dados do Amazon CloudWatch Logs no OpenSearch Service

Se você usa o Amazon OpenSearch Serverless para suas necessidades de observabilidade, agora você pode analisar seus Amazon CloudWatch Logs sem copiar ou ingerir os dados no Serviço. OpenSearch Esse recurso aproveita a consulta direta para consultar dados, semelhante à análise de dados no Amazon OpenSearch S3 a partir do Service. Você pode começar criando uma nova fonte de dados conectada no AWS Management Console.

Você pode criar uma nova fonte de dados para analisar dados de CloudWatch registros sem precisar criar o Amazon OpenSearch Serverless para consultar diretamente os registros operacionais no CloudWatch Logs. Isso permite que você analise seus dados operacionais acessados que estão fora do OpenSearch Serviço. Ao consultar o OpenSearch Service e o CloudWatch Logs, você pode começar a analisar os CloudWatch registros no Logs e depois voltar a monitorar as fontes de dados OpenSearch sem precisar trocar de ferramenta.

Para usar esse recurso, você cria uma fonte de dados de consulta direta do CloudWatch Logs para o OpenSearch Service por meio do AWS Management Console.

Tópicos

- Pré-requisitos
- Procedimento
- Próximas etapas
- Recursos adicionais do

Pré-requisitos

Antes de começar, certifique-se de ter revisto a seguinte documentação:

- the section called “Limitações do Amazon CloudWatch Logs”
- the section called “CloudWatch Recomendações de registros”
- the section called “Cotas para registros CloudWatch”

Antes de criar uma fonte de dados, você deve ter os seguintes recursos em seu Conta da AWS:

- Ativar CloudWatch registros. Configure CloudWatch registros para coletar registros da Conta da AWS mesma forma que seu OpenSearch recurso. Para obter instruções, consulte Introdução ao CloudWatch Logs no guia do usuário do Amazon CloudWatch Logs.
- Um ou mais grupos de CloudWatch registros.É possível especificar os grupos de logs que contêm os dados que você deseja consultar. Para obter instruções sobre como criar um grupo de registros, consulte Criar um grupo de CloudWatch registros em Logs no guia do usuário do Amazon CloudWatch Logs.
- (Opcional) Um perfil do IAM criado manualmente.Você pode usar esse perfil para gerenciar o acesso à sua fonte de dados. Como alternativa, você pode fazer com que o OpenSearch Service crie uma função para você automaticamente com as permissões necessárias. Se você escolher usar um perfil do IAM criado manualmente, siga as orientações em the section called “Permissões necessárias para perfis do IAM criados manualmente”.

Procedimento

Você pode configurar uma fonte de dados de consulta no nível da coleção com o Console de gerenciamento da AWS.

Para configurar uma fonte de dados em nível de coleção usando o Console de gerenciamento da AWS

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, vá para Gerenciamento central e escolha Fontes de dados conectadas.
3. Selecione Conectar.
4. Escolha CloudWatchcomo tipo de fonte de dados.
5. Escolha Próximo.
6. Em Detalhes da conexão de dados, insira um nome e uma descrição opcional.
7. Em Perfis do IAM, escolha como gerenciar o acesso aos grupos de logs.

   a. Se quiser criar automaticamente um perfil para essa fonte de dados, siga estas etapas:

      i. Selecione Criar uma novo perfil.
      ii. Insira um nome para a função do IAM.
      iii. Selecione um ou mais grupos de logs para definir os dados que podem ser consultados.

   b. Se quiser usar um perfil existente gerenciado por você, siga estas etapas:

      i. Selecione Usar perfil existente.
      ii. Selecione um perfil existente no menu suspenso.

   > **Note**
   >
   > Ao usar seu próprio perfil, confira se ele tem todas as permissões necessárias anexando as políticas necessárias com o console do IAM. Para obter mais informações, consulte the section called “Permissões necessárias para perfis do IAM criados manualmente”.

8. (Opcional) Em Política de acesso, configure uma política de acesso para a fonte de dados. As políticas de acesso controlam se uma solicitação para a fonte de dados de consulta direta do OpenSearch Serviço é aceita ou rejeitada. Se você não configurar uma política de acesso, somente o proprietário da fonte de dados terá acesso. Você pode configurar a política de acesso para permitir o acesso entre contas, permitindo que diretores de outras pessoas acessem Contas da AWS a fonte de dados.

Você pode criar uma política de acesso usando o editor visual ou fornecendo um documento de política JSON. Com o editor visual, você pode permitir ou negar o acesso especificando um Conta da AWS ID principal, ARN da conta, ARN do usuário do IAM, ARN da função do IAM, endereço IP de origem ou bloco CIDR. O editor visual suporta até 10 elementos. Para definir uma política com mais de 10 elementos, use o editor JSON.

Você também pode escolher Importar política para importar uma política de acesso existente de outra fonte de dados.

9. (Opcional) Em Tags, adicione tags à fonte de dados.
10. Escolha Próximo.
11. Em Configurar OpenSearch, escolha como configurar OpenSearch.

    a. Use as configurações padrão:

    - Revise os nomes de recurso padrão e as configurações de retenção de dados. Sugerimos que você use nomes personalizados.

      Quando você usa as configurações padrão, um novo OpenSearch aplicativo e espaço de trabalho do Essentials são criados para você sem custo adicional. OpenSearch permite que você analise várias fontes de dados. Ele inclui espaços de trabalho, que fornecem experiências personalizadas para casos de uso mais conhecidos. Os espaços de trabalho são compatíveis com controle de acesso, permitindo que você crie espaços privados para seus casos de uso e os compartilhe apenas com seus colaboradores.

    b. Use configurações personalizadas:

    i. Escolha Customize (Personalizar).

    ii. Edite o nome da coleção e as configurações de retenção de dados conforme necessário.

    iii. Selecione o OpenSearch aplicativo e o espaço de trabalho que você deseja usar.

12. Escolha Próximo.
13. Revise suas opções e escolha Editar se precisar fazer alguma alteração.
14. Escolha Conectar para configurar a fonte de dados. Fique nesta página enquanto sua fonte de dados é criada. Quando ela ficar pronta, você será direcionado para a página de detalhes da fonte de dados.

Próximas etapas

Visite os OpenSearch painéis

Depois de criar uma fonte de dados, o OpenSearch Service fornece uma URL de OpenSearch painéis. Você usa isso para configurar o controle de acesso, definir tabelas, configurar painéis baseados no tipo de log para os tipos de log mais conhecidos e consultar seus dados usando SQL ou PPL.

Para saber mais, consulte the section called “Configurando uma fonte de dados de CloudWatch registros”.

Recursos adicionais do

Permissões necessárias para perfis do IAM criados manualmente

Ao criar uma fonte de dados, você escolhe um perfil do IAM para gerenciar o acesso aos dados. Você tem duas opções:

1. Criar um novo perfil do IAM automaticamente
2. Use um perfil do IAM existente que você criou manualmente

Se usar um perfil criado manualmente, você precisará anexar as permissões corretas a ele. As permissões devem permitir o acesso à fonte de dados específica e permitir que o OpenSearch Serviço assuma a função. Isso é necessário para que o OpenSearch Serviço possa acessar e interagir com seus dados com segurança.

O exemplo de política a seguir demonstra as permissões de privilégio mínimo necessárias para criar e gerenciar uma fonte de dados. Se você tiver permissões mais amplas, como `logs:*` ou a política `AdminstratorAccess`, essas permissões abrangem as permissões de privilégio mínimo na política de amostra.

No exemplo de política a seguir, *`placeholder text`* substitua o por suas próprias informações.

JSON

```
{
"Version":"2012-10-17",
"Statement": [
```

```
        {
            "Sid": "AmazonOpenSearchDirectQueryAllLogsAccess",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:StartQuery",
                "logs:GetLogGroupFields"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "111122223333"
                }
            },
            "Resource": [
                "arn:aws:logs:us-east-1:111122223333:log-group:*"
            ]
        }
    ]
}
```

O perfil também deve ter a seguinte política de confiança, que especifica o ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "TrustPolicyForAmazonOpenSearchDirectQueryService",
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:opensearch:us-
east-1:111122223333:datasource/rolename"
                }
            }
        }
```

```
    ]
}
```

Para obter instruções sobre como criar o perfil, consulte Criar um perfil usando políticas de confiança personalizadas.

Por padrão, a função tem acesso somente aos índices da fonte de dados de consulta direta. Embora você possa configurar a função para limitar ou conceder acesso à sua fonte de dados, é recomendável não ajustar o acesso dessa função. Se você excluir a fonte de dados, essa função será excluída. Isso removerá o acesso de outros usuários se eles estiverem mapeados para a função.

Exemplo de política de acesso para uma fonte de dados de consulta direta

As políticas de acesso para fontes de dados de consulta direta seguem a sintaxe da política do IAM. O documento de política deve estar no formato JSON válido. O exemplo de política a seguir concede um Conta da AWS acesso específico à fonte de dados de consulta direta.

No exemplo de política a seguir, *placeholder text* substitua o por suas próprias informações.

```
{
 "Version": "2012-10-17",
 "Statement": [
   {
     "Effect": "Allow",
     "Principal": {
     "AWS": "arn:aws:iam::account-id:root"
     },
     "Action": [
       "opensearch:StartDirectQuery",
       "opensearch:GetDirectQuery",
       "opensearch:CancelDirectQuery",
       "opensearch:GetDirectQueryResult"
     ],
     "Resource": "arn:aws:opensearch:region:account-id:datasource/data-source-name"
   }
 ]
}
```

Se você não configurar uma política de acesso, somente o proprietário da fonte de dados terá acesso à fonte de dados.

Configurando e consultando uma fonte de dados CloudWatch de registros em painéis OpenSearch

Agora que você criou sua fonte de dados, pode começar a usá-la com OpenSearch painéis. Esta seção mostra vários casos de uso com sua fonte de dados em OpenSearch painéis.

Consultar grupos de logs na página do Discover

Na página OpenSearch Descobrir, você pode usar a nova fonte de dados de consulta direta que você configurou para consultar seus grupos de CloudWatch registros de registros. Para isso, escolha Explorar logs e use a barra de pesquisa para criar a consulta usando SQL ou PPL. Você pode filtrar, classificar e visualizar os dados retornados dos grupos de logs. Para entender quais declarações, comandos e limitações são compatíveis com a integração do CloudWatch Logs, consultethe section called “Comandos SQL e PPL compatíveis”.

Criar uma visão em painel para a fonte de dados

Ao usar o OpenSearch Service, você pode analisar rapidamente os tipos de AWS log populares usando modelos de painel predefinidos. Para CloudWatch registros, há modelos para registros de VPC, CloudTrail AWS WAF, e Firewall de Rede. Esses modelos permitem que você crie rapidamente um painel personalizado para os dados específicos. Eles incluem painéis personalizados para o tipo específico de log. Isso permite que você comece rapidamente a analisar essas fontes de AWS log populares, sem precisar criar tudo do zero.

> **Note**
>
> Os painéis usam visualizações indexadas, que ingerem dados de CloudWatch registros usando unidades de OpenSearch computação (OCUs) de consulta direta, bem como indexaçãoOCUs, pesquisa e armazenamento de coleções sem servidor. OCUs

Siga estas etapas para criar um painel usando um desses modelos prontos, para poder começar a explorar e analisar os dados imediatamente.

Para criar uma visão de painel

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.

2. No painel de navegação à esquerda, escolha Gerenciamento central e depois Fontes de dados conectadas.
3. Selecione a fonte de dados para abrir a página de detalhes.
4. Escolha Create dashboard (Criar painel).
5. Escolha o tipo de painel que você deseja criar.
6. Insira um nome para o painel.
7. Insira uma descrição opcional do painel.
8. Selecione um ou mais grupos de logs para visualizar no painel.
9. Escolha com que frequência você deseja atualizar os dados do painel.
10. Escolha qual OpenSearch espaço de trabalho você deseja usar.
    a. Para criar um novo espaço de trabalho, selecione Criar novo espaço de trabalho e insira um nome.
    b. Para usar um espaço de trabalho existente, selecione Selecionar espaço de trabalho existente.
11. Escolha Create dashboard (Criar painel).

Consultando dados de CloudWatch registros no Discover OpenSearch

Para consultar os dados, selecione a fonte de dados no menu suspenso. Se você estiver usando o CloudWatch Logs, navegue até o Discover no seu espaço de trabalho do Essentials e comece a consultar dados usando OpenSearch SQL ou Piped Processing Language (PPL). Para obter uma lista de comandos compatíveis, consulte the section called “Comandos SQL e PPL compatíveis”.

> **Note**
>
> Se você criou uma visão materializada, poderá usar DQL para consultar os dados fornecidos desde que os tenha indexado ali.

Solução de problemas

Pode haver casos em que os resultados não retornem conforme o esperado. Se tiver enfrentando algum problema, lembre-se de seguir as the section called “Recomendações”.

Consultando diretamente os dados do Amazon Security Lake no Service OpenSearch

Esta seção o guiará pelo processo de criação e configuração de uma integração de fonte de dados no Amazon OpenSearch Service, permitindo que você consulte e analise com eficiência seus dados armazenados no Security Lake.

Nas páginas a seguir, você aprenderá como configurar uma fonte de dados de consulta direta do Security Lake, navegar pelos pré-requisitos necessários e seguir os procedimentos usando o. step-by-step Console de gerenciamento da AWS

Tópicos

- Criação de uma integração de fonte de dados do Amazon Security Lake no OpenSearch Service
- Configurando e consultando uma fonte de dados do Security Lake em painéis OpenSearch

Criação de uma integração de fonte de dados do Amazon Security Lake no OpenSearch Service

Você pode usar o Amazon OpenSearch Serverless para consultar dados de segurança diretamente no Amazon Security Lake. Para fazer isso, você cria uma fonte de dados que permite usar recursos de OpenSearch ETL zero nos dados do Security Lake. Ao criar uma fonte de dados, você pode pesquisar diretamente, obter insights e analisar os dados armazenados no Security Lake. Você pode acelerar o desempenho da consulta e usar OpenSearch análises avançadas em conjuntos de dados selecionados do Security Lake usando indexação sob demanda.

Tópicos

- Pré-requisitos
- Procedimento
- Próximas etapas
- Recursos adicionais do

Pré-requisitos

Antes de começar, certifique-se de ter revisto a seguinte documentação:

- the section called “Limitações do Amazon Security Lake”

- the section called “Recomendações sobre o Security Lake”
- the section called “Cotas para o Security Lake”

Antes de criar uma fonte de dados, realize as seguintes ações no Security Lake:

- Habilitar o Security Lake. Configure o Security Lake para coletar registros da Região da AWS mesma forma que seu OpenSearch recurso. Para saber mais, consulte Introdução ao Amazon Security Lake no Guia do usuário do Amazon Security Lake.
- Configure as permissões do Security Lake. Certifique-se de ter aceitado as permissões do perfil vinculado ao serviço para gerenciamento de recursos e que o console não mostre nenhum problema na página Problemas. Para saber mais, consulte Perfil baseado no serviço para o Security Lake no Guia do usuário do Amazon Security Lake.
- Compartilhe as fontes de dados do Security Lake. Ao acessar OpenSearch na mesma conta do Security Lake, certifique-se de que não haja nenhuma mensagem para registrar seus buckets do Security Lake no Lake Formation no console do Security Lake. Para OpenSearch acesso entre contas, configure um assinante de consulta do Lake Formation no console do Security Lake. Use a conta associada ao seu OpenSearch recurso como assinante. Para saber mais, consulte Gerenciamento de usuários no Security Lake no Guia do usuário do Amazon Security Lake.

Além disso, você também deve ter os seguintes recursos em sua Conta da AWS:

- (Opcional) um perfil do IAM criado manualmente. Você pode usar esse perfil para gerenciar o acesso à sua fonte de dados. Como alternativa, você pode fazer com que o OpenSearch Service crie uma função para você automaticamente com as permissões necessárias. Se você escolher usar um perfil do IAM criado manualmente, siga as orientações em the section called “Permissões necessárias para perfis do IAM criados manualmente”.

Procedimento

Você pode configurar uma fonte de dados para se conectar a um banco de dados do Security Lake no Console de gerenciamento da AWS.

Para configurar uma fonte de dados usando o Console de gerenciamento da AWS

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.

2. No painel de navegação à esquerda, vá para Gerenciamento central e escolha Fontes de dados conectadas.
3. Selecione Conectar.
4. Escolha Security Lake como o tipo de fonte de dados.
5. Escolha Próximo.
6. Em Detalhes da conexão de dados, insira um nome e uma descrição opcional.
7. Em Configurações de permissão de acesso do IAM, escolha como gerenciar o acesso à sua fonte de dados.

    a. Se quiser criar automaticamente um perfil para essa fonte de dados, siga estas etapas:

        i. Selecione Criar uma novo perfil.
        ii. Insira um nome para a função do IAM.
        iii. Selecione uma ou mais AWS Glue tabelas para definir quais dados podem ser consultados.

    b. Se quiser usar um perfil existente gerenciado por você, siga estas etapas:

        i. Selecione Usar perfil existente.
        ii. Selecione um perfil existente no menu suspenso.

    > **Note**
    >
    > Ao usar seu próprio perfil, confira se ele tem todas as permissões necessárias anexando as políticas necessárias com o console do IAM. Para saber mais, consulte the section called “Permissões necessárias para perfis do IAM criados manualmente”.

8. (Opcional) Em Tags, adicione tags à fonte de dados.
9. Escolha Próximo.
10. Em Configurar OpenSearch, escolha como configurar OpenSearch.

    - Revise os nomes de recurso padrão e as configurações de retenção de dados.

      Quando você usa as configurações padrão, um novo OpenSearch aplicativo e espaço de trabalho do Essentials são criados para você sem custo adicional. OpenSearch permite que você analise várias fontes de dados. Ele inclui espaços de trabalho, que fornecem experiências personalizadas para casos de uso mais conhecidos. Os espaços de trabalho

são compatíveis com controle de acesso, permitindo que você crie espaços privados para seus casos de uso e os compartilhe apenas com seus colaboradores.

11. Use configurações personalizadas:
    a. Escolha Customize (Personalizar).
    b. Edite o nome da coleção e as configurações de retenção de dados conforme necessário.
    c. Selecione o OpenSearch aplicativo e o espaço de trabalho que você deseja usar.
12. Escolha Próximo.
13. Revise suas opções e escolha Editar se precisar fazer alguma alteração.
14. Escolha Conectar para configurar a fonte de dados. Fique nesta página enquanto sua fonte de dados é criada. Quando ela ficar pronta, você será direcionado para a página de detalhes da fonte de dados.

Próximas etapas

Visite OpenSearch Painéis e crie um painel

Depois de criar uma fonte de dados, o OpenSearch Service fornece uma URL de OpenSearch painéis. Você usa a URL para consultar os dados usando SQL ou PPL. A integração do Security Lake vem com modelos de consulta pré-empacotados para SQL e PPL a fim de ajudar você a começar a analisar os logs.

Para saber mais, consulte the section called “Configurar uma fonte de dados do Security Lake”.

Recursos adicionais do

Permissões necessárias para perfis do IAM criados manualmente

Ao criar uma fonte de dados, você escolhe um perfil do IAM para gerenciar o acesso aos dados. Você tem duas opções:

1. Criar um novo perfil do IAM automaticamente
2. Use um perfil do IAM existente que você criou manualmente

Se usar um perfil criado manualmente, você precisará anexar as permissões corretas a ele. As permissões devem permitir o acesso à fonte de dados específica e permitir que o OpenSearch Serviço assuma a função para que o OpenSearch Serviço possa acessar e interagir com segurança com seus dados. Além disso, conceda LakeFormation permissões à função para quaisquer bancos

de dados e tabelas que você queira consultar. Conceda `DESCRIBE` permissões para a função nos SecurityLake bancos de dados que você deseja consultar a partir da conexão de consulta direta. Conceda permissões de `SELECT and DESCRIBE`, pelo menos, ao perfil da fonte de dados para as tabelas do banco de dados.

O exemplo de política a seguir demonstra as permissões de privilégio mínimo necessárias para criar e gerenciar uma fonte de dados. Se você tiver permissões mais abrangentes, como a política `AdminstratorAccess`, elas incluirão as permissões de privilégio mínimo do exemplo de política.

No exemplo de política a seguir, *`placeholder text`* substitua o por suas próprias informações.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryServerlessAccess",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": "arn:aws:aoss:us-
east-1:111122223333:collection/collectionname/*"
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryGlueAccess",
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:GetTable",
                "glue:GetTableVersions",
                "glue:GetTables",
                "glue:SearchTables",
                "glue:BatchGetPartition"
            ],
            "Resource": [
                "arn:aws:glue:us-east-1:111122223333:table/databasename/*",
```

```
                "arn:aws:glue:us-east-1:111122223333:database/databasename",
                "arn:aws:glue:us-east-1:111122223333:catalog",
                "arn:aws:glue:us-east-1:111122223333:database/default"
            ]
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryLakeFormationAccess",
            "Effect": "Allow",
            "Action": [
                "lakeformation:GetDataAccess"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

O perfil também deve ter a seguinte política de confiança, que especifica o ID de destino.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Para obter instruções sobre como criar o perfil, consulte Criar um perfil usando políticas de confiança personalizadas.

Por padrão, a função tem acesso somente aos índices da fonte de dados de consulta direta. Embora você possa configurar a função para limitar ou conceder acesso à sua fonte de dados, é

recomendável não ajustar o acesso dessa função. Se você excluir a fonte de dados, essa função será excluída. Isso removerá o acesso de outros usuários se eles estiverem mapeados para a função.

Consultar dados do Security Lake criptografados com uma chave gerenciada pelo cliente

Se o bucket do Security Lake associado à conexão de dados for criptografado usando criptografia do lado do servidor com um cliente gerenciado AWS KMS key, você deverá adicionar a função de LakeFormation serviço à política principal. Isso permite que o serviço acesse e leia os dados para suas consultas.

No exemplo de política a seguir, *placeholder text* substitua o por suas próprias informações.

```
{
    "Sid": "Allow LakeFormation to access the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::account:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

Configurando e consultando uma fonte de dados do Security Lake em painéis OpenSearch

Agora que você criou sua fonte de dados, você pode configurá-la em OpenSearch painéis.

Esta seção mostra vários casos de uso com sua fonte de dados em OpenSearch painéis antes de você consultar seus dados. Para começar, você precisa navegar até sua fonte de dados em OpenSearch painéis. No menu à esquerda, em Gerenciamento, escolha Fontes de dados. Em seguida, selecione o nome da fonte de dados que você criou anteriormente no console OpenSearch de serviço.

Consultar tabelas do Security Lake no Discover

Se você criou tabelas com base nos seus registros do Security Lake, agora você pode consultar essas tabelas diretamente do OpenSearch Discover. Isso permite que você acesse e analise facilmente os dados armazenados no Security Lake, diretamente na interface familiar do Discover. Ao consultar o Security Lake diretamente no Discover, você pode evitar a necessidade de extrair, transformar e carregar manualmente os dados em um índice de pesquisa separado. Para que você possa começar rapidamente a analisar seus logs, o Discover inclui um conjunto de consultas salvas em PPL e SQL.

Comece selecionando a fonte de dados que você configurou. Selecione o banco de dados e a tabela associados que você deseja consultar e use a barra de pesquisa para escrever consultas às tabelas. Para entender quais são as instruções, os comandos e as limitações compatíveis com a integração do Security Lake, consulte the section called “Comandos SQL e PPL compatíveis”.

Para aproveitar as consultas pré-criadas que estão disponíveis para o Security Lake, vá para ... no canto superior direito do Discover, escolha Abrir consulta e depois Modelos. Há muitas consultas pré-criadas disponíveis para fontes de log compatíveis com o Security Lake. Pesquise os modelos que correspondem ao seu caso de uso, copie a consulta para usá-la na barra de pesquisa e substitua os campos de modelo (como região e ação) por suas próprias informações.

Acelerar dados do Discover

Para melhorar o desempenho e permitir consultas e análises subsequentes mais rápidas OpenSearch, você pode ingerir os resultados da sua consulta do Discover em uma visualização OpenSearch indexada.

Para criar uma visão indexada

1. No Discover, escolha Criar visão indexada.
2. No editor de consultas, insira a consulta desejada. Você pode criar uma nova consulta aqui ou usar uma de suas pesquisas anteriores já existentes.
3. Especifique um nome para a nova visão indexada. Escolha um nome descritivo que ajudará a identificar a visão posteriormente.
4. Defina as configurações de retenção de dados para a visão indexada. Você pode especificar por quanto tempo os dados devem ser mantidos no índice, o que permite equilibrar a performance com os custos de armazenamento.

5. Crie a visão indexada. Depois de criada, a visão indexada estará disponível para consultas e análises mais rápidas.

Se você já criou visões indexadas, pode acessá-las no Discover.

Para usar uma visão de índice existente

1. No Discover, escolha Selecionar visão indexada para ver uma lista das visões indexadas existentes para o Security Lake.
2. Escolha a visão indexada que você deseja usar. Isso aplicará a visão à sua consulta atual, potencialmente acelerando significativamente a recuperação e a análise de dados.

Criar uma visão em painel para a fonte de dados

Ao usar o OpenSearch Service, você pode analisar tipos de AWS log populares usando modelos de painel pré-criados. Para o Security Lake, existem modelos para registros de VPC e WAF. CloudTrail Esses modelos permitem que você crie um painel ajustado para os dados específicos. Eles incluem consultas e painéis pré-criados, ajustados para o tipo específico de log. Isso permite que você comece rapidamente a analisar essas fontes de AWS log populares, sem precisar criar tudo do zero.

> **Note**
>
> Os painéis usam visões indexadas, que ingerem dados do Security Lake e contribuem para consulta direta e computação de coleção.

Siga estas etapas para criar um painel usando um desses modelos prontos, para poder começar a explorar e analisar os dados imediatamente.

Para criar uma visão de painel

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Gerenciamento central e depois Fontes de dados conectadas.
3. Selecione a fonte de dados para abrir a página de detalhes.
4. Escolha Create dashboard (Criar painel).

5. Escolha o tipo de painel que você deseja criar.
6. Insira um nome para o painel.
7. Insira uma descrição opcional do painel.
8. Selecione uma ou mais tabelas do AWS Glue para ver em seu painel.
9. Escolha com que frequência você deseja atualizar os dados do painel.
10. Escolha qual OpenSearch espaço de trabalho você deseja usar.
    a. Para criar um espaço de trabalho, escolha Criar novo espaço de trabalho.
    b. Para usar um espaço de trabalho existente, selecione Selecionar espaço de trabalho existente.
11. Insira um nome para o espaço de trabalho.
12. Escolha Create dashboard (Criar painel).

Solução de problemas

Pode haver casos em que os resultados não retornem conforme o esperado. Se tiver enfrentando algum problema, lembre-se de seguir as the section called “Recomendações”.

Gerenciando uma fonte de dados no Amazon OpenSearch Service

Gerenciar sua fonte de dados é uma parte importante para manter a confiabilidade, a disponibilidade e o desempenho das fontes de dados de consulta direta e de suas outras AWS soluções. AWS fornece as seguintes ferramentas para monitorar, relatar quando algo está errado e realizar ações automáticas quando apropriado.

Tópicos

- Monitoramento com fontes de dados de CloudWatch métricas
- Habilitação e desabilitação de fontes de dados
- Monitoramento com AWS orçamento
- Excluir uma fonte de dados

Monitoramento com fontes de dados de CloudWatch métricas

Você pode monitorar a consulta direta usando CloudWatch. CloudWatch coleta dados brutos e os processa em métricas legíveis, quase em tempo real. Essas estatísticas são mantidas por 15 meses,

de maneira que você possa acessar informações históricas e ter uma perspectiva melhor de como a aplicação web ou o serviço está se saindo.

Você também pode definir alarmes para monitorar determinados limites e enviar notificações ou realizar ações quando esses limites são atingidos. Para obter mais informações, consulte O que é a Amazon CloudWatch.

O Amazon S3 informa as seguintes métricas:

Métrica	Description
AsyncQueryCreateAPI	O número total de solicitações feitas à API para criar consultas assíncronas. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryGetApiRequestCount	O número total de solicitações feitas à API para recuperar resultados de consultas assíncronas. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryCancelApiRequestCount	O número total de solicitações feitas à API para cancelar consultas assíncronas. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryGetApiFailedRequestCusErrCount	O número de solicitações com falha ao recuperar resultados de consultas assíncronas devido a erros

Métrica	Description
	relacionados ao cliente (por exemplo, ID de consulta inválida). Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryCancelApiFailedRequestCusErrCount	O número de solicitações com falha ao recuperar resultados de consultas assíncronas devido a erros relacionados ao cliente (por exemplo, ID de consulta inválida). Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryCancelApiFailedRequestSysErrCount	O número de solicitações com falha ao criar consultas assíncronas devido a erros relacionados ao cliente. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos
AsyncQueryGetApiFailedRequestSysErrCount	O número de solicitações com falha ao recuperar resultados de consultas assíncronas devido a erros relacionados ao sistema. Estatísticas relevantes: média, máximo, soma Dimensões: `ClientId`, `DomainName` Frequência: 60 segundos

CloudWatch O Logs e o Security Lake relatam as seguintes métricas:

Métrica	Description
DirectQueryRate	A taxa de solicitações feitas às fontes de dados. Estatísticas relevantes: Soma, Máximo, Mínimo, Média Dimensões: `DataSourceName` Frequência: 60 segundos
DirectQueryLatency	A latência observada na execução de consultas nas fontes de dados. Estatísticas relevantes: Média, P90, P99, Total, Mínimo, Máximo Dimensões: `DataSourceName` Frequência: 60 segundos
FailedDirectQueries	O número total de falhas de consulta observadas em consultas à fonte de dados. Estatísticas relevantes: Soma, Máximo, Mínimo, Média Dimensões: `DataSourceName` Frequência: 60 segundos
DirectQueryConsumedOCU	O número OCUs que é consumido para executar as consultas nas fontes de dados. Estatísticas relevantes: Média, P90, P99, Total, Mínimo, Máximo Dimensões: `DataSourceName` Frequência: 60 segundos

Habilitação e desabilitação de fontes de dados

> **Note**
>
> As informações a seguir se aplicam apenas às fontes de dados do Amazon S3.

Nas circunstâncias em que deseja interromper o uso de consultas diretas para uma fonte de dados, você poderá optar por desabilitar a fonte de dados. Desabilitar uma fonte de dados concluirá a execução das consultas existentes e interromperá a execução de todas as consultas novas.

A configuração de acelerações para melhorar a performance das consultas, como índices de salto, visões materializadas, índices abrangentes, será definida como manual quando a fonte de dados estiver desabilitada. Depois que uma fonte de dados for definida como ativa após ser desabilitada, as consultas do usuário serão executadas conforme o esperado. As acelerações previamente configuradas e definidas como manuais precisarão ser configuradas de forma manual para serem executadas novamente de acordo com um cronograma.

Monitoramento com AWS orçamento

O Amazon OpenSearch Service está preenchendo os dados de uso da OCU no nível da conta no Cost Explorer do Billing and Cost Management. Os clientes podem contabilizar o uso de OCUs no nível da conta e definir limites e alertas quando esses limites forem ultrapassados.

O formato do tipo de uso a ser filtrado no Cost Explorer é semelhante a RegionCode - DirectQuery OCU (OCU-hours). Se você quiser ser notificado quando o uso de DirectQuery OCU (horas de OCU) atingir seu limite, você pode criar uma conta de AWS orçamentos e configurar um alerta com base no limite definido. Como alternativa, no Amazon S3, você pode configurar um tópico do Amazon SNS, que desativará uma fonte de dados caso um critério de limite seja atendido.

> **Note**
>
> Os dados de uso nos AWS orçamentos não são em tempo real e podem ser atrasados em até 8 horas.

Excluir uma fonte de dados

Quando você exclui uma fonte de dados, o Amazon OpenSearch Service a remove do seu domínio ou da sua coleção. OpenSearch O serviço também remove os índices associados à fonte de dados. Seus dados transacionais não são excluídos um do outro AWS service (Serviço da AWS), mas o outro AWS service (Serviço da AWS) não envia novos dados para OpenSearch o Serviço.

Você pode excluir uma integração de fonte de dados usando a API Console de gerenciamento da AWS ou a OpenSearch Service API.

Console de gerenciamento da AWS

Para excluir uma fonte de dados do Amazon S3

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Domínios.
3. Selecione o domínio cuja fonte de dados você deseja excluir. Isso abre a página de detalhes do domínio. Escolha a guia Conexões abaixo das informações gerais e localize a seção Consulta direta.
4. Selecione a fonte de dados que você deseja excluir, escolha Excluir e confirme a exclusão.

Para excluir uma fonte de dados do CloudWatch Logs ou do Security Lake

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Gerenciamento central e depois Fontes de dados conectadas.
3. Selecione a fonte de dados que você deseja excluir, escolha Excluir e confirme a exclusão.

OpenSearch API de serviço

Para excluir uma fonte de dados do Amazon S3, use a operação de DeleteDataSourceAPI.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource/data-source-name
```

Para excluir uma fonte de dados do CloudWatch Logs ou do Security Lake, use a operação DeleteDirectQueryDataSourceda API.

Otimizando o desempenho de consultas para fontes de dados OpenSearch do Amazon Service

O desempenho das consultas no Amazon OpenSearch Service pode diminuir quando você está acessando fontes de dados externas. Isso pode ser devido a fatores como latência da rede, transformação de dados ou grandes volumes de dados. Para melhorar a performance, considere indexar quantidades selecionadas de dados, dependendo do caso de uso:

- Acelerar consultas diretas no Amazon S3 (índice de salto)
- Criar visualizações de painel no Security Lake (visões materializadas)
- Ingestão de resultados de consultas usando visões indexadas para análise off-line ou performance aprimorada no Security Lake (visões materializadas)

Para obter a documentação completa sobre consultas aceleradas, incluindo exemplos de consultas, consulte Otimizar o desempenho da consulta usando OpenSearch indexação na documentação de código aberto.

Tópicos

- Índices de salto
- Visões materializadas
- Índices de abrangência

Índices de salto

Um índice de salto ingere apenas os metadados dos dados armazenados no Amazon S3. Quando consulta uma tabela com um índice de salto, o planejador de consultas usa o índice para reescrever a consulta, identificando eficientemente a localização dos dados sem precisar verificar todas as partições e arquivos. Essa abordagem ajuda a determinar a localização exata dos dados armazenados.

Há duas maneiras de criar um índice de salto. A primeira maneira é gerar automaticamente o índice de salto a partir dos detalhes da fonte de dados. A segunda é usar o Query Workbench para criar manualmente o índice de salto com uma instrução SQL.

Para gerar automaticamente um índice de salto na sua fonte de dados, vá para Gerenciamento de painel e Acelerar dados e selecione o banco de dados e a tabela (pode ser necessário fazer uma atualização para obter os bancos de dados e tabelas mais recentes). Depois, você pode escolher Gerar para gerar automaticamente um índice de salto ou selecionar manualmente cada campo que deseja indexar e especificar a aceleração (tipo de índice de salto). Por fim, escolha Criar aceleração para criar um trabalho recorrente que preencha o novo índice de salto.

Índices de salto são compatíveis apenas com fontes de dados do Amazon S3.

Para obter mais informações sobre a configuração de ignorar índices usando o Query Workbench, consulte Ignorando índices na documentação. OpenSearch

Visões materializadas

As visualizações materializadas usam consultas complexas, como agregações, para oferecer suporte OpenSearch às visualizações de painéis. Eles ingerem um subconjunto de seus dados com base na consulta e os armazenam em um OpenSearch índice. Você pode usar esse índice para criar visualizações.

As visões materializadas são compatíveis com as fontes de dados do Amazon S3 e do Security Lake.

Para obter mais informações sobre como configurar visualizações materializadas usando o Query Workbench, consulte Visualizações materializadas na documentação. OpenSearch

Índices de abrangência

Um índice de cobertura ingere dados de uma coluna especificada em uma tabela e OpenSearch cria um novo índice com base nesses dados. Você pode usar esse novo índice para visualizações e outros OpenSearch recursos, como detecção de anomalias ou análise geoespacial.

Índices abrangentes são compatíveis apenas com fontes de dados do Amazon S3.

Para obter mais informações sobre como configurar índices de cobertura, consulte Índices de cobertura na OpenSearch documentação.

Comandos SQL e PPL compatíveis

OpenSearch SQL e OpenSearch Pipeline Processing Language (PPL) são linguagens para consultar, analisar e processar dados no OpenSearch CloudWatch Logs Insights e no Security Lake.

Você pode usar OpenSearch SQL e OpenSearch PPL no OpenSearch Discover para consultar dados no CloudWatch Logs, no Amazon S3 ou no Security Lake. CloudWatch O Logs Insights também oferece suporte às linguagens de consulta OpenSearch PPL e OpenSearch SQL, além do Logs Insights QL, uma linguagem de consulta criada especificamente para analisar registros. CloudWatch

- OpenSearch SQL: O OpenSearch SQL fornece uma opção familiar se você estiver acostumado a trabalhar com bancos de dados relacionais. OpenSearch O SQL oferece um subconjunto da funcionalidade SQL, o que o torna uma boa opção para realizar consultas ad-hoc e tarefas de análise de dados. Com o OpenSearch SQL, você pode usar comandos como SELECT, FROM, WHERE, GROUP BY, HAVING e vários outros comandos e funções SQL disponíveis no SQL. Você pode executar JOINs várias tabelas (ou grupos de registros), correlacionar dados entre tabelas (ou grupos de registros) usando subconsultas e usar o rico conjunto de funções JSON, matemáticas, de seqüência de caracteres, condicionais e outras funções SQL para realizar análises poderosas em dados de log e de segurança.
- OpenSearch PPL (Piped Processing Language): com o OpenSearch PPL, você pode recuperar, consultar e analisar dados usando comandos agrupados, facilitando a compreensão e a composição de consultas complexas. Sua sintaxe é baseada em pipes Unix e permite o encadeamento de comandos para transformar e processar dados. Com o PPL, você pode filtrar e agregar dados e usar comandos como subconsultas JOINs, LOOKUP e um rico conjunto de funções matemáticas, de seqüências de caracteres, de data, condicionais e outras para análise.

Embora a maioria dos comandos nas linguagens de consulta OpenSearch PPL e OpenSearch SQL sejam comuns em todos CloudWatch os Logs OpenSearch, há algumas diferenças nos conjuntos de comandos e funções compatíveis com cada um desses serviços. Para obter mais detalhes, consulte as tabelas nas próximas páginas.

- the section called “Comandos SQL compatíveis”
  - the section called “CloudWatch Informações sobre registros”
  - the section called “Restrições gerais do SQL”
- the section called “Comandos PPL compatíveis”
  - the section called “Informações adicionais para usuários do CloudWatch Logs Insights que usam OpenSearch PPL”

Comandos e funções OpenSearch SQL compatíveis

As tabelas de referência a seguir mostram quais comandos SQL são compatíveis com o OpenSearch Discover para consultar dados no Amazon S3, Security Lake CloudWatch ou Logs, e quais comandos SQL são compatíveis com CloudWatch o Logs Insights. A sintaxe SQL compatível com o CloudWatch Logs Insights e a compatível com o OpenSearch Discover para consultar CloudWatch registros são as mesmas e são referenciadas como CloudWatch registros nas tabelas a seguir.

> **Note**
>
> OpenSearch também tem suporte a SQL para consultar dados que são ingeridos OpenSearch e armazenados em índices. Esse dialeto SQL é diferente do SQL usado na consulta direta e é chamado de OpenSearch SQL nos índices.

Tópicos

- Comandos
- Funções
- Restrições gerais do SQL
- Informações adicionais para usuários do CloudWatch Logs Insights que usam OpenSearch SQL

Comandos

> **Note**
>
> No exemplo de coluna de comandos, substitua *<tableName/logGroup>* conforme necessário, dependendo da fonte de dados que você for consultar.
>
> - Exemplo de comando: `SELECT Body , Operation FROM` **`<tableName/logGroup>`**
> - Se for consultar o Amazon S3 ou o Security Lake, use: `SELECT Body , Operation FROM` **`table_name`**
> - Se você estiver consultando CloudWatch registros, use: `SELECT Body , Operation FROM` **`` `LogGroupA` ``**

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Cláusula SELECT”	Exibe os valores projetado s.	compatív l	compatív l	compatív l	`SELECT method, status FROM <tableName/logGroup>`
the section called “Cláusula WHERE”	Filtra eventos de log com base nos critérios de campo fornecido s.	compatív l	compatív l	compatív l	`SELECT * FROM <tableName/logGroup> WHERE status = 100`
the section called “Cláusula GROUP BY”	Agrupa eventos de log com base na categoria e encontra a média com base nas	compatív l	compatív l	compatív l	`SELECT method, status, COUNT(*) AS request_count, SUM(bytes) AS total_bytes FROM <tableName/logGroup> GROUP BY method, status`

Command	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
	estatísticas.				
the section called "Cláusula HAVING"	Filtra os resultados com base em condições de agrupamento.	compatível	compatível	compatível	`SELECT method, status, COUNT(*) AS request_count, SUM(bytes) AS total_bytes FROM <tableName/logGroup> GROUP BY method, status HAVING COUNT(*) > 5`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Cláusula ORDER BY”	Ordena os resultado s com base em campos da cláusula de classific ação. É possível classific ar em ordem crescent ou decresce te.	compatív l	compatív l	compatív l	`SELECT * FROM <tableName/logGroup> ORDER BY status DESC`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Cláusula JOIN” (INNER | CROSS | LEFT OUTER)	Une os resultado s de duas tabelas com base em campos comuns.	Compatí l com (é necessá o usar as palavras chave Inner e Left Outer para união. Apenas uma operaçã JOIN é aceita em uma instruçã SELECT	Compatí l com (é necessá o usar as palavras chave Inner, Left Outer e Cross para unir)	Compatí l com (é necessá o usar as palavras chave Inner, Left Outer e Cross para unir)	`SELECT A.Body, B.Timestamp FROM <tableNameA/logGroupA> AS A INNER JOIN <tableNameB/logGroupB> AS B ON A.`requestId` = B.`requestId``
the section called “Cláusula LIMIT”	Restring os resultado s às primeira N linhas.	compatív l	compatív l	compatív l	`SELECT * FROM <tableName/logGroup> LIMIT 10`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Cláusula CASE”	Avalia as condiçõe e retorna um valor quando a primeira condiçã é atendida	compatív l	compatív l	compatív l	SELECT method, status, CASE WHEN status BETWEEN 100 AND 199 THEN 'Informational' WHEN status BETWEEN 200 AND 299 THEN 'Success' WHEN status BETWEEN 300 AND 399 THEN 'Redirection' WHEN status BETWEEN 400 AND 499 THEN 'Client Error' WHEN status BETWEEN 500 AND 599 THEN 'Server Error' ELSE 'Unknown Status' END AS status_category, CASE method WHEN 'GET' THEN 'Read Operation' WHEN 'POST' THEN 'Create Operation' WHEN 'PUT' THEN 'Update Operation' WHEN 'PATCH' THEN 'Partial Update Operation' WHEN 'DELETE' THEN 'Delete Operation' ELSE 'Other Operation' END AS operation_type, bytes, datetime FROM *<tableName/logGroup>*

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Expressã o de tabela comum”	Cria um conjunto de resultado s temporá o nomeado em uma instruçã SELECT INSERT UPDATE DELETE ou MERGE.	não compatív l	compatív l	compatív l	`WITH RequestStats AS ( SELECT method, status, bytes, COUNT(*) AS request_count FROM tableName GROUP BY method, status, bytes ) SELECT method, status, bytes, request_count FROM RequestStats WHERE bytes > 1000`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “EXPLAIN	Exibe o plano de execuçã de uma instruçã SQL sem executá-la de fato.	não compatív l	compatív l	compatív l	`EXPLAIN SELECT k, SUM(v) FROM VALUES (1, 2), (1, 3) AS t(k, v) GROUP BY k`
the section called “Cláusula LATERAL SUBQUEF	Permite que uma subcons ta na cláusula FROM referenci e colunas de itens precede es na mesma cláusula FROM.	não compatív l	compatív l	compatív l	`SELECT * FROM tableName LATERAL ( SELECT * FROM t2 WHERE t1.c1 = t2.c1 )`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Cláusula LATERAL VIEW”	Gera uma tabela virtual aplicand uma função geradora de tabela a cada linha de uma tabela básica.	não compatív l	compatív l	compatív l	`SELECT * FROM tableName LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age`
the section called “Predicad o LIKE”	Compara uma string com um padrão usando caractere s curinga.	compatív l	compatív l	compatív l	`SELECT method, status, request, host FROM <tableName/logGroup> WHERE method LIKE 'D%'`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “OFFSET”	Especific a o número de linhas a serem ignorada antes de começar a retornar as linhas da consulta	compatív l quando usado em conjunto com uma cláusula LIMIT em uma consulta Por exemplo • Suport : SELEC * FROM Table LIMIT 100 OFFSE 10 • Não suport : SELEC *	compatív l	compatív l	SELECT method, status, bytes, datetime FROM <tableName/logGroup> ORDER BY datetime OFFSET 10

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
		FROM Table OFFSE 10			
the section called “Cláusula PIVOT”	Transfor a linhas em colunas, girando os dados de um formato baseado em linhas para um formato baseado em colunas.	não compatív l	compatív l	compatív l	SELECT * FROM (SELECT method, status, bytes FROM *<tableName/logGroup>*) AS SourceTable PIVOT (SUM(bytes) FOR method IN ('GET', 'POST', 'PATCH', 'PUT', 'DELETE')) AS PivotTable

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “Configur ar operadore s”	Combina os resultado s de duas ou mais instruçõe s SELECT (por exemplo UNION, INTERSI , EXCEPT	compatív l	compatív l	compatív l	`SELECT method, status, bytes FROM <tableName/logGroup> WHERE status = '416' UNION SELECT method, status, bytes FROM <tableName/logGroup> WHERE bytes > 20000`
the section called “Cláusula SORT BY”	Especific a a ordem em que os resultado s da consulta devem ser retornad s.	compatív l	compatív l	compatív l	`SELECT method, status, bytes FROM <tableName/logGroup> SORT BY bytes DESC`

Command	Descripti on	CloudWa h Registros	Amazon S3	Lago de Seguran	Exemplo de comando da
the section called “UNPIVOT	Transfor a colunas em linhas, girando dados de um formato baseado em colunas para um formato baseado em linhas.	não compatív l	compatív l	compatív l	SELECT status, REPLACE(method, '_bytes', '') AS request_method, bytes, datetime FROM PivotedData UNPIVOT (bytes FOR method IN (GET_bytes, POST_bytes, PATCH_bytes, PUT_bytes, DELETE_bytes)) AS UnpivotedData

Funções

> **Note**
>
> No exemplo de coluna de comandos, substitua *<tableName/logGroup>* conforme necessário, dependendo da fonte de dados que você for consultar.
>
> - Exemplo de comando: `SELECT Body , Operation FROM` **`<tableName/logGroup>`**
> - Se for consultar o Amazon S3 ou o Security Lake, use: `SELECT Body , Operation FROM` **`table_name`**
> - Se você estiver consultando CloudWatch registros, use: `SELECT Body , Operation FROM` **`` `LogGroupA` ``**

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ de string”	Funções integrada s SQL que podem manipular e transform ar dados em string e texto em consultas SQL. Por exemplo, converter maiúsculas e minúscula s, combinar strings, extrair partes e apagar texto.	compat l	compat l	compat l	`SELECT UPPER(method) AS upper_method, LOWER(host) AS lower_hos t FROM <tableName/logGroup>`
the section called “Perfis de data e hora”	Funções integradas para tratar e transform ar dados de data e timestamp em consultas	compat l	compat l	compat l	`SELECT TO_TIMESTAMP(datetime) AS timestamp, TIMESTAMP_SECONDS( UNIX_TIMESTAMP(datetime)) AS from_seconds, UNIX_TIMESTAMP(datetime) AS to_unix, FROM_UTC_TIMESTAMP (datetime, 'PST') AS to_pst,`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
	. Por exemplo, date_add, date_form at, datediff e current_d ate.				TO_UTC_TIMESTAMP(d atetime, 'EST') AS from_est FROM <tableName/logGroup>
the section called “Funçõ agrega ” -	Funções integradas que realizam cálculos em várias linhas para produzir um único valor resumido. Por exemplo: sum, count, avg, max e min.	S	compa l	compa l	SELECT COUNT(*) AS total_rec ords, COUNT(DISTINCT method) AS unique_methods, SUM(bytes) AS total_byt es, AVG(bytes) AS avg_bytes, MIN(bytes) AS min_bytes, MAX(bytes) AS max_bytes FROM <tableName/logGroup>

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ condici ais”	Funções integradas que realizam ações com base em condições especific adas ou que avaliam expressõe s condicion almente. Por exemplo: CASE e IF.	compat l	compat l	compat l	`SELECT CASE WHEN method = 'GET' AND bytes < 1000 THEN 'Small Read' WHEN method = 'POST' AND bytes > 10000 THEN 'Large Write' WHEN status >= 400 OR bytes = 0 THEN 'Problem' ELSE 'Normal' END AS request_type FROM <tableName/logGroup>`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ JSON”	Funções integrada s para analisar, extrair, modificar e consultar dados formatado s em JSON em consultas SQL (por exemplo, from_json , to_json, get_json_ object, json_tuple), permitindo a manipulação de estrutura s JSON em conjuntos de dados.	compat l	compat l	compat l	`SELECT FROM_JSON( @message, 'STRUCT< host: STRING, user-identifier: STRING, datetime: STRING, method: STRING, status: INT, bytes: INT >' ) AS parsed_json FROM <tableName/logGroup>`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da	
the section called “Funçõ de array”	Funções integrada s para trabalhar com colunas do tipo array em consultas SQL, permitindo operações como acessar, modificar e analisar dados matriciai s (por exemplo, size, explode, array_con tains).	compat l	compat l	compat l	`SELECT scores, size(scores) AS length, array_contains(scores, 90) AS has_90 FROM <tableName/logGroup>`	

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ de janela”	Funções integradas que realizam cálculos em um conjunto específico de linhas relacionadas à linha atual (janela), permitindo operações como classific ação, totalização e médias móveis (por exemplo, ROW_NUMBE R, RANK, LAG, LEAD)	compa l	compa l	compa l	`SELECT field1, field2, RANK() OVER (ORDER BY field2 DESC) AS field2Rank FROM <tableName/logGroup>`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da	
the section called “Funçõ de conver ” -	Funções integrada s para converter dados de um tipo para outro em consultas SQL, permitindo transform ações de tipos de dados e conversõe s de formato (por exemplo, CAST, TO_DATE, TO_TIMEST AMP, BINARY)	compat l	compat l	compat l	`SELECT CAST('123' AS INT) AS converted_number, CAST(123 AS STRING) AS converted_string FROM <tableName/logGroup>`	

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ de predica ” -	Funções integradas que avaliam condições e retornam valores booleanos (verdadei ro/falso) com base em critérios ou padrões especific ados (por exemplo, IN, LIKE, BETWEEN, IS NULL, EXISTS)	compa l	compa l	compa l	`SELECT * FROM <tableName/logGroup> WHERE id BETWEEN 50000 AND 75000`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ de mapa”	Aplica uma função específic a a cada elemento de um conjunto, transform ando os dados em um novo conjunto de valores.	não compat l	compa l	compa l	`SELECT MAP_FILTER( MAP( 'method', method, 'status', CAST(status AS STRING), 'bytes', CAST(byte s AS STRING) ), (k, v) -> k IN ('method', 'status') AND v ! = 'null' ) AS filtered_map FROM <tableName/logGroup> WHERE status = 100`
the section called “Funçõ matem as”	Executa operações matemátic as em dados numéricos , como calcular médias, somas ou valores trigonomé tricos.	compa l	compa l	compa l	`SELECT bytes, bytes + 1000 AS added, bytes - 1000 AS subtracte d, bytes * 2 AS doubled, bytes / 1024 AS kilobytes , bytes % 1000 AS remainder FROM <tableName/logGroup>`

Gramá SQL disponi l	Description	CloudV h Registr	Amazo S3	Lago de Segura	Exemplo de comando da
the section called “Funçõ de vários grupos de logs”	Permite que os usuários especifiq uem vários grupos de logs em uma instrução SQL SELECT	compat l	Não aplicáv	Não aplicáv	`SELECT lg1.Column1, lg1.Column2 FROM `logGroups(logGrou pIdentifier: ['LogGroup1', 'LogGroup2'])` AS lg1 WHERE lg1.Column3 = "Success"`
the section called “Funçõ gerado ”	Cria um objeto iterador que gera uma sequência de valores, permitindo o uso eficiente da memória em grandes conjuntos de dados.	não compat l	compat l	compat l	`SELECT explode(array(10, 20))`

Restrições gerais do SQL

As restrições a seguir se aplicam ao usar OpenSearch SQL com CloudWatch Logs, Amazon S3 e Security Lake.

1. Você pode usar apenas um único JOIN em uma instrução SELECT.
2. Somente um nível de sub-consultas aninhadas é suportado.
3. Consultas com várias instruções separadas por ponto e vírgula não são compatíveis.

4. Consultas contendo nomes de campo idênticos, mas que diferem somente em maiúsculas e minúsculas (como field1 e FIELD1) não são suportadas.

   Por exemplo, as seguintes consultas não são compatíveis:

   ```
   Select AWSAccountId, awsaccountid from LogGroup
   ```

   No entanto, a seguinte consulta é compatível porque o nome do campo (@logStream) é idêntico em ambos os grupos de logs:

   ```
   Select a.`@logStream`, b.`@logStream` from Table A INNER Join Table B on a.id = b.id
   ```

5. Funções e expressões devem operar em nomes de campo e fazer parte de uma instrução SELECT com um grupo de logs especificado na cláusula FROM.

   Por exemplo, esta consulta não é compatível:

   ```
   SELECT cos(10) FROM LogGroup
   ```

   Esta consulta é compatível:

   ```
   SELECT cos(field1) FROM LogGroup
   ```

Informações adicionais para usuários do CloudWatch Logs Insights que usam OpenSearch SQL

CloudWatch O Logs oferece suporte a consultas OpenSearch SQL no console, na API e na CLI do Logs Insights. Ele é compatível com a maioria dos comandos, incluindo SELECT, FROM, WHERE, GROUP BY, HAVING, JOINS e consultas aninhadas, além de funções JSON, matemáticas, string e condicionais. No entanto, o CloudWatch Logs suporta somente operações de leitura, portanto, não permite instruções DDL ou DML. Consulte as tabelas das seções anteriores para obter uma lista completa dos comandos e funções compatíveis.

Funções de vários grupos de logs

CloudWatch O Logs Insights oferece suporte à capacidade de consultar vários grupos de registros. Para lidar com esse caso de uso no SQL, você pode usar o comando `logGroups`. Esse comando é específico para consultar dados no CloudWatch Logs Insights envolvendo um ou mais grupos de

registros. Use essa sintaxe para consultar vários grupos de logs especificando-os no comando, em vez de escrever uma consulta para cada grupo de logs e combiná-los com um comando `UNION`.

Sintaxe:

```
`logGroups(
    logGroupIdentifier: ['LogGroup1','LogGroup2', ...'LogGroupn']
)
```

Nessa sintaxe, é possível especificar até 50 grupos de logs no parâmetro `logGroupIndentifier`. Para referenciar grupos de registros em uma conta de monitoramento, use ARNs em vez de `LogGroup` nomes.

Consulta de exemplo:

```
SELECT LG1.Column1, LG1.Column2 from `logGroups(
    logGroupIdentifier: ['LogGroup1', 'LogGroup2']
)` as LG1
WHERE LG1.Column1 = 'ABC'
```

A sintaxe a seguir, envolvendo vários grupos de registros após a `FROM` declaração, não é compatível com a consulta CloudWatch de registros:

```
SELECT Column1, Column2 FROM 'LogGroup1', 'LogGroup2', ...'LogGroupn'
WHERE Column1 = 'ABC'
```

Restrições

Ao usar comandos SQL ou PPL, coloque certos campos entre acentos graves para consultá-los. Acentos graves são obrigatórios em campos com caracteres especiais (não alfabéticos e não numéricos). Por exemplo, coloque `@message`, `Operation.Export,` e `Test::Field` entre acentos graves. Não é necessário colocar entre acentos graves campos com nomes puramente alfabéticos.

Exemplo de consulta com campos simples:

```
SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
LIMIT 1000;
```

Mesma consulta com acentos graves acrescentados:

```
SELECT `SessionToken`, `Operation`, `StartTime`  FROM `LogGroup-A`
LIMIT 1000;
```

Para ver outras restrições gerais que não são específicas CloudWatch dos registros, consultethe section called “Restrições gerais do SQL”.

Exemplos de consultas e cotas

> **Note**
>
> O seguinte se aplica tanto aos usuários do CloudWatch Logs Insights quanto OpenSearch aos usuários que consultam CloudWatch dados.

Para exemplos de consultas SQL que você pode usar em CloudWatch Logs, consulte Consultas salvas e amostras no console Amazon CloudWatch Logs Insights para ver exemplos.

Para obter informações sobre os limites que se aplicam ao consultar CloudWatch registros do OpenSearch serviço, consulte Cotas de CloudWatch registros no Guia do usuário do Amazon CloudWatch Logs. Os limites envolvem o número de grupos de CloudWatch registros que você pode consultar, o máximo de consultas simultâneas que você pode executar, o tempo máximo de execução da consulta e o número máximo de linhas retornadas nos resultados. Os limites são os mesmos, independentemente da linguagem usada para consultar CloudWatch os registros (ou seja, OpenSearch PPL, SQL e Logs Insights).

Comandos SQL

Tópicos

- Funções de string
- Perfis de data e hora
- Funções agregadas
- Funções condicionais
- Funções JSON
- Funções de array
- Funções de janela

- Funções de conversão
- Funções de predicado
- Funções de mapa
- Funções matemáticas
- Funções geradoras
- Cláusula SELECT
- Cláusula WHERE
- Cláusula GROUP BY
- Cláusula HAVING
- Cláusula ORDER BY
- Cláusula JOIN
- Cláusula LIMIT
- Cláusula CASE
- Expressão de tabela comum
- EXPLAIN
- Cláusula LATERAL SUBQUERY
- Cláusula LATERAL VIEW
- Predicado LIKE
- OFFSET
- Cláusula PIVOT
- Configurar operadores
- Cláusula SORT BY
- UNPIVOT

Funções de string

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
ascii(str)	Retorna o valor numérico do primeiro caractere de `str`.
base64(bin)	Converte o argumento de um binário `bin` em uma string de base 64.
bit_length(expr)	Retorna o comprimento em bits dos dados de string ou o número de bits de dados binários.
btrim(str)	Remove os caracteres de espaço antes e depois de `str`.
btrim(str, trimStr)	Remove os caracteres `trimStr` antes e depois de `str`.
char(expr)	Retorna o caractere ASCII com o binário equivalente a. `expr` Se n for maior do que 256, o resultado será equivalente a chr(n % 256)
char_length(expr)	Retorna o comprimento em caracteres dos dados de string ou o número de bits de dados binários. O comprimento dos dados de string inclui os espaços no fim. O comprimento dos dados binários inclui os zeros binários.
character_length(expr)	Retorna o comprimento em caracteres dos dados de string ou o número de bits de dados binários. O comprimento dos dados de string inclui os espaços no fim. O comprimento dos dados binários inclui os zeros binários.
chr(expr)	Retorna o caractere ASCII com o binário equivalente a. `expr` Se n for maior do que 256, o resultado será equivalente a chr(n % 256)

Função	Description
concat_ws(sep[, str | array(str)]+)	Retorna a concatenação das strings separadas por `sep`, ignorando os valores nulos.
contains(left, right)	Retorna um booleano. O valor será Verdadeir o se direita estiver dentro de esquerda. Retornará NULL se uma das expressões de entrada for NULL. Caso contrário, retorna Falso. Tanto esquerda quanto direita devem ser do tipo STRING ou BINARY.
decode(bin, charset)	Decodifica o primeiro argumento usando o conjunto de caracteres do segundo argumento.
decode(expr, search, result [, search, result] ... [, default])	Compara expr a cada valor de pesquisa em ordem. Se expr for igual a um valor de pesquisa, decode retornará o resultado correspondente. Se nenhuma correspon dência for encontrada, retornará o padrão. Se o padrão for omitido, retornará nulo.
elt(n, input1, input2, ...)	Retorna a enésima entrada, por exemplo, retorna `input2` quando `n` é 2.
encode(str, charset)	Codifica o primeiro argumento usando o conjunto de caracteres do segundo argumento.
endswith(left, right)	Retorna um booleano. O valor será verdadeir o se esquerda terminar com direita. Retornará NULL se uma das expressões de entrada for NULL. Caso contrário, retorna Falso. Tanto esquerda quanto direita devem ser do tipo STRING ou BINARY.

Função	Description
find_in_set(str, str_array)	Retorna o índice (na base 1) da string fornecida (`str`) na lista delimitada por vírgula (`str_array`). Retornará 0, se a string não foi encontrada ou se a string fornecida (`str`) contiver uma vírgula.
format_number(expr1, expr2)	Formata o número `expr1` como “#, ###, ###.##”, arredondado para `expr2` casas decimais. Se `expr2` for 0, o resultado não terá ponto decimal ou parte fracionária. `expr2` também aceitam um formato especificado pelo usuário. Deve funcionar como o FORMAT do MySQL.
format_string(strfmt, obj, ...)	Retorna uma string formatada a partir das strings com formato printf-style.
initcap(str)	Retorna `str` com a primeira letra de cada palavra em maiúscula. Todas as outras letras são em minúsculas. As palavras são delimitad as por espaço em branco.
instr(str, substr)	Retorna o índice (na base 1) da primeira ocorrência de `substr` em `str`.
lcase(str)	Retorna `str` com todos os caracteres convertidos em minúsculas.
left(str, len)	Retorna os caracteres `len` (`len` pode ser do tipo string) mais à esquerda da string `str`; se `len` for menor ou igual a 0, o resultado será uma string vazia.

Função	Description
len(expr)	Retorna o comprimento em caracteres dos dados de string ou o número de bits de dados binários. O comprimento dos dados de string inclui os espaços no fim. O comprimento dos dados binários inclui os zeros binários.
length(expr)	Retorna o comprimento em caracteres dos dados de string ou o número de bits de dados binários. O comprimento dos dados de string inclui os espaços no fim. O comprimento dos dados binários inclui os zeros binários.
levenshtein(str1, str2[, threshold])	Retorna a distância de Levenshtein entre as duas strings fornecidas. Se o limite for definido e a distância for maior que ele, retornará -1.
locate(substr, str[, pos])	Retorna a posição da primeira ocorrência de `substr` em `str` depois da posição `pos`. O `pos` fornecido e o valor retornado são na base 1.
lower(str)	Retorna `str` com todos os caracteres convertidos em minúsculas.
lpad(str, len[, pad])	Retorna `str`, preenchido à esquerda com `pad` até um comprimento de `len`. Se `str` tiver mais que `len`, o valor retornado será encurtado para `len` caracteres ou bytes. Se `pad` não for especificado, `str` será preenchido à esquerda com caracteres de espaço se for uma string de caracteres e com zeros se for uma sequência de bytes.
ltrim(str)	Remove os caracteres de espaço antes de `str`.

Função	Description
luhn_check(str)	Verifica se uma string e dígitos é válida de acordo com o algoritmo de Luhn. Essa função de soma de verificação é amplamente aplicada em números de cartão de crédito e números de identificação governamental para distingui r números válidos de números com erros de digitação e incorretos.
mask(input[, upperChar, lowerChar, digitChar, otherChar])	mascara o valor de string fornecido. A função substitui caracteres por “X” ou “x” e números por “n”. Isso pode ser útil para criar copiar tabelas com informações sensíveis removidas.
octet_length(expr)	Retorna o comprimento em bytes dos dados de string ou o número de bits de dados binários.
overlay(input, replace, pos[, len])	Substitui `input` por `replace` que começa em `pos` e tem o comprimento `len`.
position(substr, str[, pos])	Retorna a posição da primeira ocorrência de `substr` em `str` depois da posição `pos`. O `pos` fornecido e o valor retornado são na base 1.
printf(strfmt, obj, ...)	Retorna uma string formatada a partir das strings com formato printf-style.
regexp_count(str, regexp)	Retorna quantas vezes que o padrão de expressão regular `regexp` ocorre na string `str`.
regexp_extract (str, regexp [, idx])	Extrai a primeira string na `str` que correspon de à expressão `regexp` e ao índice do grupo regex.

Função	Description
regexp_extract_all(str, regexp[, idx])	Extrai todas as strings na `str` que correspondem à expressão `regexp` e ao índice do grupo regex.
regexp_instr(str, regexp)	Pesquisa uma string para uma expressão regular e retorna um inteiro que indica a posição inicial da string correspondente. As posições são baseadas em 1, não em 0. Se nenhuma correspondência for encontrada, retornará 0.
regexp_replace(str, regexp, rep[, position])	Substitui todas as substrings `str` que correspondem a `regexp` por `rep`.
regexp_substr(str, regexp)	Retorna a substring que corresponde à expressão regular `regexp` na string `str`. Se a expressão regular não for encontrada, o resultado será nulo.
repeat(str, n)	Retorna a string que repete o valor da string fornecido n vezes.
replace(str, search[, replace])	Substitui todas as ocorrências de `search` por `replace`.
right(str, len)	Retorna os caracteres `len` (`len` pode ser do tipo string) mais à direita da string `str`; se `len` for menor ou igual a 0, o resultado será uma string vazia.

Função	Description
rpad(str, len[, pad])	Retorna `str`, preenchido à direita com `pad` até um comprimento de `len`. Se `str` tiver mais que `len`, o valor retornado será encurtado para `len` caracteres. Se `pad` não for especific ado, `str` será preenchido à direita com caracteres de espaço se for uma string de caracteres e com zeros se for uma string binária.
rtrim(str)	Remove os caracteres de espaço depois de `str`.
sentences(str[, lang, country])	Divide `str` em uma matriz de palavras.
soundex(str)	Retorna o código Soundex da string.
space(n)	Retorna uma string que consiste em n espaços.
split(str, regex, limit)	Divide `str` em torno de ocorrências que coincidem com `regex` e retorna uma matriz com um comprimento máximo de `limit`
split_part(str, delimiter, partNum)	Divide `str` por delimitador e retorna a parte da divisão solicitada (com base 1). Se alguma entrada for nula, retornará nulo. Se `partNum` estiver fora do intervalo de partes divididas, retornará uma string vazia. Se `partNum` for 0, gerará erro. Se `partNum` for negativo, as partes serão contadas de trás para a frente a partir do fim da string. Se `delimiter` for uma string vazia, `str` não será dividida.

Função	Description
startswith(left, right)	Retorna um booleano. O valor será verdadeiro se esquerda começar com a direita. Retornará NULL se uma das expressões de entrada for NULL. Caso contrário, retorna Falso. Tanto esquerda quanto direita devem ser do tipo STRING ou BINARY.
substr(str, pos[, len])	Retorna a substring de `str` que começa em `pos` e tem comprimento de `len`, ou a fatia da matriz de bytes que começa em `pos` e tem o comprimento de `len`.
substr(str FROM pos[FOR len]])	Retorna a substring de `str` que começa em `pos` e tem comprimento de `len`, ou a fatia da matriz de bytes que começa em `pos` e tem o comprimento de `len`.
substring(str, pos[, len])	Retorna a substring de `str` que começa em `pos` e tem comprimento de `len`, ou a fatia da matriz de bytes que começa em `pos` e tem o comprimento de `len`.
substring(str FROM pos[FOR len]])	Retorna a substring de `str` que começa em `pos` e tem comprimento de `len`, ou a fatia da matriz de bytes que começa em `pos` e tem o comprimento de `len`.

Função	Description
substring_index(str, delim, count)	Retorna a substring de `str` antes de `count` ocorrências do delimitador `delim`. Se `count` for positivo, tudo à esquerda do delimitad or final (contado a partir da esquerda) será retornado. Se `count` for negativo, tudo à esquerda do delimitador final (contado a partir da esquerda) será retornado. A função substring_index executa uma correspondência com distinção entre maiúsculas e minúsculas ao pesquisar `delim`.
to_binary(str[, fmt])	Converte a entrada `str` em um valor binário com base no `fmt` dado. `fmt` pode ser uma string literal sem distinção entre maiúscula s e minúsculas de “hex”, “utf-8”, “utf8” ou “base64”. Por padrão, o formato binário para conversão será “hex” se `fmt` for omitido. A função retornará NULL se pelo menos um dos parâmetros de entrada for NULL.

Função	Description
to_char(numberExpr, formatExpr)	Converte `numberExpr` em uma string baseada em `formatExpr`. Gerará uma exceção se houver falha na conversão. O formato pode consistir nos seguintes caracteres, sem distinção entre maiúsculas e minúsculas: “0” ou “9”: especifica um dígito esperado entre 0 e 9. Uma sequência de 0 ou 9 na string de formato corresponde a uma sequência de dígitos no valor da entrada, gerando uma string de resultado com o mesmo comprimento da sequência correspondente na string de formato. A string do resultado será preenchida à esquerda com zeros se a string 0/9 incluir mais dígitos do que a parte correspondente do valor decimal, começar com 0 e vier antes do ponto decimal. Caso contrário, será preenchida com espaços. “.” ou “D”: especifica a posição do ponto decimal (opcional, permitido apenas uma vez). “,” ou “G”: especifica a posição do separador de agrupamento (milhares) (,). Deve haver um 0 ou 9 à esquerda e à direita de cada separador de agrupamento. '

Função	Description
to_number(expr, fmt)	Converte a string “expr” em um número com base no formato de string “fmt”. Gerará uma exceção se houver falha na conversão. O formato pode consistir nos seguintes caractere s, sem distinção entre maiúsculas e minúscula s: “0” ou “9”: especifica um dígito esperado entre 0 e 9. Uma sequência de 0 ou 9 na string de formato corresponde a uma sequência de dígitos na string de entrada. Se a sequência 0/9 começar com 0 e vier antes do ponto decimal, ela só poderá corresponder a uma sequência de dígitos do mesmo tamanho. Caso contrário, se a sequência começar com 9 ou vier depois do ponto decimal, ela poderá corresponder a uma sequência de dígitos do mesmo tamanho ou menor. “.” ou “D”: especifica a posição do ponto decimal (opcional, permitido apenas uma vez). “,” ou “G”: especifica a posição do separador de agrupamento (milhares) (,). Deve haver um 0 ou 9 à esquerda e à direita de cada separador de agrupamento. “expr” deve corresponder ao separador de agrupamento relevante para o tamanho do número. '

Função	Description
to_varchar(numberExpr, formatExpr)	Converte `numberExpr` em uma string baseada em `formatExpr`. Gerará uma exceção se houver falha na conversão. O formato pode consistir nos seguintes caracteres, sem distinção entre maiúsculas e minúsculas: “0” ou “9”: especifica um dígito esperado entre 0 e 9. Uma sequência de 0 ou 9 na string de formato corresponde a uma sequência de dígitos no valor da entrada, gerando uma string de resultado com o mesmo comprimento da sequência correspondente na string de formato. A string do resultado será preenchida à esquerda com zeros se a string 0/9 incluir mais dígitos do que a parte correspondente do valor decimal, começar com 0 e vier antes do ponto decimal. Caso contrário, será preenchida com espaços. “.” ou “D”: especifica a posição do ponto decimal (opcional, permitido apenas uma vez). “,” ou “G”: especifica a posição do separador de agrupamento (milhares) (,). Deve haver um 0 ou 9 à esquerda e à direita de cada separador de agrupamento. '
translate(input, from, to)	Traduz a string `input` substituindo os caracteres encontrados na string `from` pelos caracteres correspondentes da string `to`.
trim(str)	Remove os caracteres de espaço antes e depois de `str`.
trim(BOTH FROM str)	Remove os caracteres de espaço antes e depois de `str`.
trim(LEADING FROM str)	Remove os caracteres de espaço antes de `str`.

Função	Description
trim(TRAILING FROM str)	Remove os caracteres de espaço depois de `str`.
trim(trimStr FROM str)	Remove os caracteres `trimStr` antes e depois de `str`.
trim(BOTH trimStr FROM str)	Remove os caracteres `trimStr` antes e depois de `str`.
trim(LEADING trimStr FROM str)	Remove os caracteres `trimStr` iniciais de `str`.
trim(TRAILING trimStr FROM str)	Remove os caracteres `trimStr` depois de `str`.
try_to_binary(str[, fmt])	Essa é uma versão especial de `to_binary` que realiza a mesma operação, mas retorna um valor NULL em vez de gerar um erro quando a conversão não pode ser executada.
try_to_number(expr, fmt)	Converte a string “expr” em um número com base no formato de string `fmt`. Retornará NULL se a string “expr” não corresponder ao formato esperado. O formato segue a mesma semântica que a função to_number.
ucase(str)	Retorna `str` com todos os caracteres convertidos em maiúsculas.
unbase64(str)	Converte o argumento de uma string de base 64 `str` em um binário.
upper(str)	Retorna `str` com todos os caracteres convertidos em maiúsculas.

Exemplos

```
-- ascii
SELECT ascii('222');
+----------+
|ascii(222)|
+----------+
|        50|
+----------+
SELECT ascii(2);
+--------+
|ascii(2)|
+--------+
|      50|
+--------+
-- base64
SELECT base64('Feathers');
+-----------------+
|base64(Feathers)|
+-----------------+
|     RmVhdGhlcnM=|
+-----------------+
SELECT base64(x'537061726b2053514c');
+-----------------------------+
|base64(X'537061726B2053514C')|
+-----------------------------+
|                 U3BhcmsgU1FM|
+-----------------------------+
-- bit_length
SELECT bit_length('Feathers');
+---------------------+
|bit_length(Feathers)|
+---------------------+
|                   64|
+---------------------+
SELECT bit_length(x'537061726b2053514c');
+---------------------------------+
|bit_length(X'537061726B2053514C')|
+---------------------------------+
|                               72|
+---------------------------------+
-- btrim
SELECT btrim('    Feathers   ');
+---------------------+
|btrim(    Feathers   )|
```

```
+----------------------+
|              Feathers|
+----------------------+
SELECT btrim(encode('    Feathers   ', 'utf-8'));
+-----------------------------------+
|btrim(encode(    Feathers   , utf-8))|
+-----------------------------------+
|                           Feathers|
+-----------------------------------+
SELECT btrim('Feathers', 'Fe');
+--------------------+
|btrim(Alphabet, Al)|
+--------------------+
|               athers|
+--------------------+
SELECT btrim(encode('Feathers', 'utf-8'), encode('Al', 'utf-8'));
+-----------------------------------------------------+
|btrim(encode(Feathers, utf-8), encode(Al, utf-8))|
+-----------------------------------------------------+
|                                               athers|
+-----------------------------------------------------+
-- char
SELECT char(65);
+--------+
|char(65)|
+--------+
|       A|
+--------+
-- char_length
SELECT char_length('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9 |
+-----------------------+
SELECT char_length(x'537061726b2053514c');
+---------------------------------+
|char_length(X'537061726B2053514C')|
+---------------------------------+
|                                9|
+---------------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
```

```
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- character_length
SELECT character_length('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
SELECT character_length(x'537061726b2053514c');
+--------------------------------------+
|character_length(X'537061726B2053514C')|
+--------------------------------------+
|                                     9|
+--------------------------------------+
SELECT CHAR_LENGTH('Feathers ');
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- chr
SELECT chr(65);
+-------+
|chr(65)|
+-------+
|      A|
+-------+
-- concat_ws
SELECT concat_ws(' ', 'Fea', 'thers');
+----------------------+
```

```
|concat_ws( , Fea, thers)|
+-------------------------+
|               Feathers|
+-------------------------+
SELECT concat_ws('s');
+------------+
|concat_ws(s)|
+------------+
|            |
+------------+
SELECT concat_ws('/', 'foo', null, 'bar');
+----------------------------+
|concat_ws(/, foo, NULL, bar)|
+----------------------------+
|                     foo/bar|
+----------------------------+
SELECT concat_ws(null, 'Fea', 'thers');
+--------------------------+
|concat_ws(NULL, Fea, thers)|
+--------------------------+
|                      NULL|
+--------------------------+
-- contains
SELECT contains('Feathers', 'Fea');
+-------------------------+
|contains(Feathers, Fea)|
+-------------------------+
|                     true|
+-------------------------+
SELECT contains('Feathers', 'SQL');
+-------------------------+
|contains(Feathers, SQL)|
+-------------------------+
|                     false|
+-------------------------+
SELECT contains('Feathers', null);
+-------------------------+
|contains(Feathers, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT contains(x'537061726b2053514c', x'537061726b');
+-----------------------------------------------+
|contains(X'537061726B2053514C', X'537061726B')|
```

```
+-----------------------------------------------+
|                                           true|
+-----------------------------------------------+
-- decode
SELECT decode(encode('abc', 'utf-8'), 'utf-8');
+---------------------------------+
|decode(encode(abc, utf-8), utf-8)|
+---------------------------------+
|                              abc|
+---------------------------------+
SELECT decode(2, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------------+
|decode(2, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------------+
|                                                                        San Francisco|
+-------------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------------+
|                                                                         Non domestic|
+-------------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle');
+-----------------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle)|
+-----------------------------------------------------------------------+
|                                                                   NULL|
+-----------------------------------------------------------------------+
SELECT decode(null, 6, 'Fea', NULL, 'thers', 4, 'rock');
+-----------------------------------------+
|decode(NULL, 6, Fea, NULL, thers, 4, rock)|
+-----------------------------------------+
|                                    thers|
+-----------------------------------------+
-- elt
SELECT elt(1, 'scala', 'java');
+------------------+
|elt(1, scala, java)|
+------------------+
|             scala|
+------------------+
SELECT elt(2, 'a', 1);
```

```
+------------+
|elt(2, a, 1)|
+------------+
|           1|
+------------+
-- encode
SELECT encode('abc', 'utf-8');
+------------------+
|encode(abc, utf-8)|
+------------------+
|        [61 62 63]|
+------------------+
-- endswith
SELECT endswith('Feathers', 'ers');
+------------------------+
|endswith(Feathers, ers)|
+------------------------+
|                    true|
+------------------------+
SELECT endswith('Feathers', 'SQL');
+--------------------------+
|endswith(Feathers, SQL)|
+--------------------------+
|                     false|
+--------------------------+
SELECT endswith('Feathers', null);
+-------------------------+
|endswith(Feathers, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT endswith(x'537061726b2053514c', x'537061726b');
+-----------------------------------------------+
|endswith(X'537061726B2053514C', X'537061726B')|
+-----------------------------------------------+
|                                          false|
+-----------------------------------------------+
SELECT endswith(x'537061726b2053514c', x'53514c');
+---------------------------------------+
|endswith(X'537061726B2053514C', X'53514C')|
+---------------------------------------+
|                                   true|
+---------------------------------------+
-- find_in_set
```

```
SELECT find_in_set('ab','abc,b,ab,c,def');
+-------------------------------+
|find_in_set(ab, abc,b,ab,c,def)|
+-------------------------------+
|                              3|
+-------------------------------+
-- format_number
SELECT format_number(12332.123456, 4);
+------------------------------+
|format_number(12332.123456, 4)|
+------------------------------+
|                  12,332.1235|
+------------------------------+
SELECT format_number(12332.123456, '##################.###');
+---------------------------------------------------+
|format_number(12332.123456, ##################.###)|
+---------------------------------------------------+
|                                           12332.123|
+---------------------------------------------------+
-- format_string
SELECT format_string("Hello World %d %s", 100, "days");
+-------------------------------------------+
|format_string(Hello World %d %s, 100, days)|
+-------------------------------------------+
|                       Hello World 100 days|
+-------------------------------------------+
-- initcap
SELECT initcap('Feathers');
+------------------+
|initcap(Feathers)|
+------------------+
|         Feathers|
+------------------+
-- instr
SELECT instr('Feathers', 'ers');
+--------------------+
|instr(Feathers, ers)|
+--------------------+
|                   6|
+--------------------+
-- lcase
SELECT lcase('Feathers');
+---------------+
|lcase(Feathers)|
```

```
+---------------+
|       feathers|
+---------------+
-- left
SELECT left('Feathers', 3);
+------------------+
|left(Feathers, 3)|
+------------------+
|               Fea|
+------------------+
SELECT left(encode('Feathers', 'utf-8'), 3);
+---------------------------------+
|left(encode(Feathers, utf-8), 3)|
+---------------------------------+
|                       [RmVh]|
+---------------------------------+
-- len
SELECT len('Feathers ');
+---------------+
|len(Feathers )|
+---------------+
|             9|
+---------------+
SELECT len(x'537061726b2053514c');
+--------------------------+
|len(X'537061726B2053514C')|
+--------------------------+
|                         9|
+--------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9|
+-----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+----------------------------+
|character_length(Feathers )|
+----------------------------+
|                           9|
+----------------------------+
-- length
SELECT length('Feathers ');
+------------------+
```

```
|length(Feathers )|
+-----------------+
|                9|
+-----------------+
SELECT length(x'537061726b2053514c');
+----------------------------+
|length(X'537061726B2053514C')|
+----------------------------+
|                           9|
+----------------------------+
SELECT CHAR_LENGTH('Feathers ');
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- levenshtein
SELECT levenshtein('kitten', 'sitting');
+--------------------------+
|levenshtein(kitten, sitting)|
+--------------------------+
|                           3|
+--------------------------+
SELECT levenshtein('kitten', 'sitting', 2);
+-------------------------------+
|levenshtein(kitten, sitting, 2)|
+-------------------------------+
|                             -1|
+-------------------------------+
-- locate
SELECT locate('bar', 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
SELECT locate('bar', 'foobarbar', 5);
+-------------------------+
```

```
|locate(bar, foobarbar, 5)|
+-------------------------+
|                        7|
+-------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- lower
SELECT lower('Feathers');
+---------------+
|lower(Feathers)|
+---------------+
|       feathers|
+---------------+
-- lpad
SELECT lpad('hi', 5, '??');
+---------------+
|lpad(hi, 5, ??)|
+---------------+
|          ???hi|
+---------------+
SELECT lpad('hi', 1, '??');
+---------------+
|lpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT lpad('hi', 5);
+--------------+
|lpad(hi, 5,  )|
+--------------+
|            hi|
+--------------+
SELECT hex(lpad(unhex('aabb'), 5));
+---------------------------------+
|hex(lpad(unhex(aabb), 5, X'00'))|
+---------------------------------+
|                     000000AABB|
+---------------------------------+
SELECT hex(lpad(unhex('aabb'), 5, unhex('1122')));
+-----------------------------------+
```

```
|hex(lpad(unhex(aabb), 5, unhex(1122)))|
+--------------------------------------+
|                           112211AABB|
+--------------------------------------+
-- ltrim
SELECT ltrim('    Feathers   ');
+----------------------+
|ltrim(    Feathers   )|
+----------------------+
|           Feathers   |
+----------------------+
-- luhn_check
SELECT luhn_check('8112189876');
+----------------------+
|luhn_check(8112189876)|
+----------------------+
|                  true|
+----------------------+
SELECT luhn_check('79927398713');
+-----------------------+
|luhn_check(79927398713)|
+-----------------------+
|                   true|
+-----------------------+
SELECT luhn_check('79927398714');
+-----------------------+
|luhn_check(79927398714)|
+-----------------------+
|                  false|
+-----------------------+
-- mask
SELECT mask('abcd-EFGH-8765-4321');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, X, x, n, NULL)|
+---------------------------------------+
|                     xxxx-XXXX-nnnn-nnnn|
+---------------------------------------+
SELECT mask('abcd-EFGH-8765-4321', 'Q');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, Q, x, n, NULL)|
+---------------------------------------+
|                     xxxx-QQQQ-nnnn-nnnn|
+---------------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
```

```
+--------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+--------------------------------+
|                     QqQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#');
+--------------------------------+
|mask(AbCD123-@$#, X, x, n, NULL)|
+--------------------------------+
|                     XxXXnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q');
+--------------------------------+
|mask(AbCD123-@$#, Q, x, n, NULL)|
+--------------------------------+
|                     QxQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
+--------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+--------------------------------+
|                     QqQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd');
+--------------------------------+
|mask(AbCD123-@$#, Q, q, d, NULL)|
+--------------------------------+
|                     QqQQddd-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd', 'o');
+-----------------------------+
|mask(AbCD123-@$#, Q, q, d, o)|
+-----------------------------+
|                  QqQQdddoooo|
+-----------------------------+
SELECT mask('AbCD123-@$#', NULL, 'q', 'd', 'o');
+--------------------------------+
|mask(AbCD123-@$#, NULL, q, d, o)|
+--------------------------------+
|                     AqCDdddoooo|
+--------------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, 'd', 'o');
+---------------------------------+
|mask(AbCD123-@$#, NULL, NULL, d, o)|
```

```
+----------------------------------+
|                       AbCDdddoooo|
+----------------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, 'o');
+-------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, o)|
+-------------------------------------+
|                           AbCD123oooo|
+-------------------------------------+
SELECT mask(NULL, NULL, NULL, NULL, 'o');
+------------------------------+
|mask(NULL, NULL, NULL, NULL, o)|
+------------------------------+
|                           NULL|
+------------------------------+
SELECT mask(NULL);
+-------------------------+
|mask(NULL, X, x, n, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, NULL);
+----------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, NULL)|
+----------------------------------------+
|                             AbCD123-@$#|
+----------------------------------------+
-- octet_length
SELECT octet_length('Feathers');
+----------------------+
|octet_length(Feathers)|
+----------------------+
|                     8|
+----------------------+
SELECT octet_length(x'537061726b2053514c');
+----------------------------------+
|octet_length(X'537061726B2053514C')|
+----------------------------------+
|                                 9|
+----------------------------------+
-- overlay
SELECT overlay('Feathers' PLACING '_' FROM 6);
+---------------------------+
|overlay(Feathers, _, 6, -1)|
```

```
+---------------------------+
|                   Feathe_ers|
+---------------------------+
SELECT overlay('Feathers' PLACING 'ures' FROM 5);
+-------------------------------+
|overlay(Feathers, ures, 5, -1)|
+-------------------------------+
|                     Features  |
+-------------------------------+
-- position
SELECT position('bar', 'foobarbar');
+---------------------------+
|position(bar, foobarbar, 1)|
+---------------------------+
|                          4|
+---------------------------+
SELECT position('bar', 'foobarbar', 5);
+---------------------------+
|position(bar, foobarbar, 5)|
+---------------------------+
|                          7|
+---------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- printf
SELECT printf("Hello World %d %s", 100, "days");
+-----------------------------------+
|printf(Hello World %d %s, 100, days)|
+-----------------------------------+
|                Hello World 100 days|
+-----------------------------------+
-- regexp_count
SELECT regexp_count('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-----------------------------------------------------------------------------+
|regexp_count(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-----------------------------------------------------------------------------+
|                                                                            2|
+-----------------------------------------------------------------------------+
SELECT regexp_count('abcdefghijklmnopqrstuvwxyz', '[a-z]{3}');
```

```
+-------------------------------------------------+
|regexp_count(abcdefghijklmnopqrstuvwxyz, [a-z]{3})|
+-------------------------------------------------+
|                                                8|
+-------------------------------------------------+
-- regexp_extract
SELECT regexp_extract('100-200', '(\\d+)-(\\d+)', 1);
+--------------------------------------+
|regexp_extract(100-200, (\d+)-(\d+), 1)|
+--------------------------------------+
|                                   100|
+--------------------------------------+
-- regexp_extract_all
SELECT regexp_extract_all('100-200, 300-400', '(\\d+)-(\\d+)', 1);
+-----------------------------------------------------+
|regexp_extract_all(100-200, 300-400, (\d+)-(\d+), 1)|
+-----------------------------------------------------+
|                                           [100, 300]|
+-----------------------------------------------------+
-- regexp_instr
SELECT regexp_instr('user@opensearch.org', '@[^.]*');
+-----------------------------------------------+
|regexp_instr(user@opensearch.org, @[^.]*, 0)|
+-----------------------------------------------+
|                                              5|
+-----------------------------------------------+
-- regexp_replace
SELECT regexp_replace('100-200', '(\\d+)', 'num');
+--------------------------------------+
|regexp_replace(100-200, (\d+), num, 1)|
+--------------------------------------+
|                               num-num|
+--------------------------------------+
-- regexp_substr
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-----------------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-----------------------------------------------------------------------------+
|                                                                       Steven|
+-----------------------------------------------------------------------------+
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Jeck');
+-------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Jeck)|
```

```
+--------------------------------------------------------------------------+
|                                                                      NULL|
+--------------------------------------------------------------------------+
-- repeat
SELECT repeat('123', 2);
+--------------+
|repeat(123, 2)|
+--------------+
|        123123|
+--------------+
-- replace
SELECT replace('ABCabc', 'abc', 'DEF');
+-------------------------+
|replace(ABCabc, abc, DEF)|
+-------------------------+
|                   ABCDEF|
+-------------------------+
-- right
SELECT right('Feathers', 3);
+-------------------+
|right(Feathers, 3)|
+-------------------+
|                ers|
+-------------------+
-- rpad
SELECT rpad('hi', 5, '??');
+---------------+
|rpad(hi, 5, ??)|
+---------------+
|          hi???|
+---------------+
SELECT rpad('hi', 1, '??');
+---------------+
|rpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT rpad('hi', 5);
+--------------+
|rpad(hi, 5,  )|
+--------------+
|         hi   |
+--------------+
SELECT hex(rpad(unhex('aabb'), 5));
```

```
+--------------------------------+
|hex(rpad(unhex(aabb), 5, X'00'))|
+--------------------------------+
|                      AABB000000|
+--------------------------------+
SELECT hex(rpad(unhex('aabb'), 5, unhex('1122')));
+-------------------------------------+
|hex(rpad(unhex(aabb), 5, unhex(1122)))|
+-------------------------------------+
|                           AABB112211|
+-------------------------------------+
-- rtrim
SELECT rtrim('    Feathers   ');
+----------------------+
|rtrim(    Feathers   )|
+----------------------+
|              Feathers|
+----------------------+
-- sentences
SELECT sentences('Hi there! Good morning.');
+-------------------------------------+
|sentences(Hi there! Good morning., , )|
+-------------------------------------+
|                  [[Hi, there], [Go...|
+-------------------------------------+
-- soundex
SELECT soundex('Miller');
+---------------+
|soundex(Miller)|
+---------------+
|           M460|
+---------------+
-- space
SELECT concat(space(2), '1');
+-------------------+
|concat(space(2), 1)|
+-------------------+
|                  1|
+-------------------+
-- split
SELECT split('oneAtwoBthreeC', '[ABC]');
+--------------------------------+
|split(oneAtwoBthreeC, [ABC], -1)|
+--------------------------------+
```

```
|             [one, two, three, ]|
+--------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', -1);
+--------------------------------+
|split(oneAtwoBthreeC, [ABC], -1)|
+--------------------------------+
|             [one, two, three, ]|
+--------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', 2);
+-------------------------------+
|split(oneAtwoBthreeC, [ABC], 2)|
+-------------------------------+
|              [one, twoBthreeC]|
+-------------------------------+
-- split_part
SELECT split_part('11.12.13', '.', 3);
+--------------------------+
|split_part(11.12.13, ., 3)|
+--------------------------+
|                        13|
+--------------------------+
-- startswith
SELECT startswith('Feathers', 'Fea');
+----------------------------+
|startswith(Feathers, Fea)|
+----------------------------+
|                        true|
+----------------------------+
SELECT startswith('Feathers', 'SQL');
+-------------------------+
|startswith(Feathers, SQL)|
+-------------------------+
|                     false|
+-------------------------+
SELECT startswith('Feathers', null);
+-------------------------+
|startswith(Feathers, NULL)|
+-------------------------+
|                      NULL|
+-------------------------+
SELECT startswith(x'537061726b2053514c', x'537061726b');
+------------------------------------------------+
|startswith(X'537061726B2053514C', X'537061726B')|
+------------------------------------------------+
```

```
|                                              true|
+-----------------------------------------------+
SELECT startswith(x'537061726b2053514c', x'53514c');
+-----------------------------------------+
|startswith(X'537061726B2053514C', X'53514C')|
+-----------------------------------------+
|                                    false|
+-----------------------------------------+
-- substr
SELECT substr('Feathers', 5);
+-------------------------------+
|substr(Feathers, 5, 2147483647)|
+-------------------------------+
|                          hers |
+-------------------------------+
SELECT substr('Feathers', -3);
+--------------------------------+
|substr(Feathers, -3, 2147483647)|
+--------------------------------+
|                             ers|
+--------------------------------+
SELECT substr('Feathers', 5, 1);
+-----------------------+
|substr(Feathers, 5, 1)|
+-----------------------+
|                      h|
+-----------------------+
SELECT substr('Feathers' FROM 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                             hers |
+---------------------------------+
SELECT substr('Feathers' FROM -3);
+----------------------------------+
|substring(Feathers, -3, 2147483647)|
+----------------------------------+
|                               ers|
+----------------------------------+
SELECT substr('Feathers' FROM 5 FOR 1);
+-------------------------+
|substring(Feathers, 5, 1)|
+-------------------------+
|                        h|
```

```
+-------------------------+
-- substring
SELECT substring('Feathers', 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                            hers |
+---------------------------------+
SELECT substring('Feathers', -3);
+---------------------------------+
|substring(Feathers, -3, 2147483647)|
+---------------------------------+
|                               ers|
+---------------------------------+
SELECT substring('Feathers', 5, 1);
+-------------------------+
|substring(Feathers, 5, 1)|
+-------------------------+
|                        h|
+-------------------------+
SELECT substring('Feathers' FROM 5);
+---------------------------------+
|substring(Feathers, 5, 2147483647)|
+---------------------------------+
|                            hers |
+---------------------------------+
SELECT substring('Feathers' FROM -3);
+---------------------------------+
|substring(Feathers, -3, 2147483647)|
+---------------------------------+
|                               ers|
+---------------------------------+
SELECT substring('Feathers' FROM 5 FOR 1);
+-------------------------+
|substring(Feathers, 5, 1)|
+-------------------------+
|                        h|
+-------------------------+
-- substring_index
SELECT substring_index('www.apache.org', '.', 2);
+-----------------------------------+
|substring_index(www.apache.org, ., 2)|
+-----------------------------------+
|                         www.apache|
```

```
+-------------------------------------+
-- to_binary
SELECT to_binary('abc', 'utf-8');
+---------------------+
|to_binary(abc, utf-8)|
+---------------------+
|           [61 62 63]|
+---------------------+
-- to_char
SELECT to_char(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_char(454.00, '000D00');
+------------------------+
|to_char(454.00, 000D00)|
+------------------------+
|                 454.00|
+------------------------+
SELECT to_char(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
+----------------------+
|                12,454|
+----------------------+
SELECT to_char(78.12, '$99.99');
+----------------------+
|to_char(78.12, $99.99)|
+----------------------+
|                $78.12|
+----------------------+
SELECT to_char(-12454.8, '99G999D9S');
+----------------------------+
|to_char(-12454.8, 99G999D9S)|
+----------------------------+
|                   12,454.8-|
+----------------------------+
-- to_number
SELECT to_number('454', '999');
+-------------------+
|to_number(454, 999)|
+-------------------+
```

```
|                454|
+-------------------+
SELECT to_number('454.00', '000.00');
+-------------------------+
|to_number(454.00, 000.00)|
+-------------------------+
|                   454.00|
+-------------------------+
SELECT to_number('12,454', '99,999');
+-------------------------+
|to_number(12,454, 99,999)|
+-------------------------+
|                    12454|
+-------------------------+
SELECT to_number('$78.12', '$99.99');
+-------------------------+
|to_number($78.12, $99.99)|
+-------------------------+
|                    78.12|
+-------------------------+
SELECT to_number('12,454.8-', '99,999.9S');
+-------------------------------+
|to_number(12,454.8-, 99,999.9S)|
+-------------------------------+
|                       -12454.8|
+-------------------------------+
-- to_varchar
SELECT to_varchar(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_varchar(454.00, '000D00');
+-----------------------+
|to_char(454.00, 000D00)|
+-----------------------+
|                 454.00|
+-----------------------+
SELECT to_varchar(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
+----------------------+
|                12,454|
```

```
+----------------------+
SELECT to_varchar(78.12, '$99.99');
+----------------------+
|to_char(78.12, $99.99)|
+----------------------+
|               $78.12|
+----------------------+
SELECT to_varchar(-12454.8, '99G999D9S');
+---------------------------+
|to_char(-12454.8, 99G999D9S)|
+---------------------------+
|                  12,454.8-|
+---------------------------+
-- translate
SELECT translate('AaBbCc', 'abc', '123');
+---------------------------+
|translate(AaBbCc, abc, 123)|
+---------------------------+
|                     A1B2C3|
+---------------------------+
-- try_to_binary
SELECT try_to_binary('abc', 'utf-8');
+-------------------------+
|try_to_binary(abc, utf-8)|
+-------------------------+
|              [61 62 63]|
+-------------------------+
select try_to_binary('a!', 'base64');
+-------------------------+
|try_to_binary(a!, base64)|
+-------------------------+
|                     NULL|
+-------------------------+
select try_to_binary('abc', 'invalidFormat');
+---------------------------------+
|try_to_binary(abc, invalidFormat)|
+---------------------------------+
|                             NULL|
+---------------------------------+
-- try_to_number
SELECT try_to_number('454', '999');
+-----------------------+
|try_to_number(454, 999)|
+-----------------------+
```

```
|                    454|
+-----------------------+
SELECT try_to_number('454.00', '000.00');
+----------------------------+
|try_to_number(454.00, 000.00)|
+----------------------------+
|                      454.00|
+----------------------------+
SELECT try_to_number('12,454', '99,999');
+----------------------------+
|try_to_number(12,454, 99,999)|
+----------------------------+
|                       12454|
+----------------------------+
SELECT try_to_number('$78.12', '$99.99');
+----------------------------+
|try_to_number($78.12, $99.99)|
+----------------------------+
|                       78.12|
+----------------------------+
SELECT try_to_number('12,454.8-', '99,999.9S');
+-----------------------------------+
|try_to_number(12,454.8-, 99,999.9S)|
+-----------------------------------+
|                          -12454.8|
+-----------------------------------+
-- ucase
SELECT ucase('Feathers');
+---------------+
|ucase(Feathers)|
+---------------+
|       FEATHERS|
+---------------+
-- unbase64
SELECT unbase64('U3BhcmsgU1FM');
+----------------------+
|unbase64(U3BhcmsgU1FM)|
+----------------------+
|  [53 70 61 72 6B 2...|
+----------------------+
-- upper
SELECT upper('Feathers');
+---------------+
|upper(Feathers)|
```

```
+---------------+
|       FEATHERS|
+---------------+
```

Perfis de data e hora

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
add_months(start_date, num_months)	Retorna a data que é `num_months` depois de `start_date` .
convert_timezone([sourceTz,]targetTz, sourceTs)	Converte o timestamp sem fuso horário `sourceTs` do fuso horário `sourceTz` para o fuso horário `targetTz`.
curdate()	Retorna a data atual no início da avaliação da consulta. Todas as chamadas de curdate na mesma consulta retornam o mesmo valor.
current_date()	Retorna a data atual no início da avaliação da consulta. Todas as chamadas de current_date na mesma consulta retornam o mesmo valor.
current_date	Retorna a data atual no início da avaliação da consulta.
current_timestamp()	Retorna o timestamp atual no início da avaliação da consulta. Todas as chamadas de current_timestamp na mesma consulta retornam o mesmo valor.
current_timestamp	Retorna o timestamp atual no início da avaliação da consulta.

Função	Description
current_timezone()	Retorna o fuso horário local da sessão atual.
date_add(start_date, num_days)	Retorna a data que é `num_days` depois de `start_date` .
date_diff(endDate, startDate)	Retorna o número de dias de `startDate` a `endDate`.
date_format(timestamp, fmt)	Converte `timestamp` em um valor de string no formato especificado pelo formato de data `fmt`.
date_from_unix_date(days)	Cria uma data a partir do número de dias desde 1/01/1970.
date_part(field, source)	Extrai uma parte da fonte date/timestamp ou do intervalo.
date_sub(start_date, num_days)	Retorna a data que é `num_days` antes de `start_date` .
date_trunc(fmt, ts)	Retorna o timestamp `ts` truncado para a unidade especificada pelo modelo de formato `fmt`.
dateadd(start_date, num_days)	Retorna a data que é `num_days` depois de `start_date` .
datediff(endDate, startDate)	Retorna o número de dias de `startDate` a `endDate`.
datepart(field, source)	Extrai uma parte da fonte date/timestamp ou do intervalo.
day(date)	Retorna o dia do mês do date/timestamp.
dayofmonth(date)	Retorna o dia do mês do date/timestamp.

Função	Description
dayofweek(date)	Retorna o dia da semana para date/time stamp (1 = domingo, 2 = segunda-feira,..., 7 = sábado).
dayofyear(date)	Retorna o dia do ano do date/timestamp.
extract(field FROM source)	Extrai uma parte da fonte date/timestamp ou do intervalo.
from_unixtime(unix_time[, fmt])	Retorna `unix_time` no `fmt` especificado.
from_utc_timestamp(timestamp, timezone)	Com um timestamp como “2017-07-14 02:40:00.0”, é interpretado como uma hora em UTC e processa essa hora como um timestamp no fuso horário fornecido. Por exemplo, “GMT+1” produziria “2017-07-14 03:40:00.0”.
hour(timestamp)	Retorna o componente de hora da string/ti mestamp.
last_day(date)	Retorna o último dia do mês ao qual a data pertence.
localtimestamp()	Retorna o timestamp atual sem o fuso horário no início da avaliação da consulta. Todas as chamadas de localtimestamp na mesma consulta retornam o mesmo valor.
localtimestamp	Retorna a data/hora atual no fuso horário da sessão no início da avaliação da consulta.
make_date(year, month, day)	Cria a data a partir dos campos de ano, mês e dia.
make_dt_interval([days[, hours[, mins[, secs]]]])	Faça a DayTimeIntervalType duração de dias, horas, minutos e segundos.

Função	Description
make_interval([years[, months[, weeks[, days[, hours[, mins[, secs]]]]]]])	Gera o intervalo a partir de anos, menos, semanas, dias, horas minutos e segundos.
make_timestamp(year, month, day, hour, min, sec[, timezone])	Cria um timestamp a partir dos campos de ano, mês, dia, hora, minuto, segundo e fuso horário.
make_timestamp_ltz(year, month, day, hour, min, sec[, timezone])	Cria o timestamp atual com o fuso horário local a partir dos campos de ano, mês, dia, hora, minuto, segundo e fuso horário.
make_timestamp_ntz(year, month, day, hour, min, sec)	Cria a data/hora local a partir dos campos de ano, mês, dia, hora, minuto, segundo.
make_ym_interval([years[, months]])	Gera um intervalo ano-mês a partir de anos, meses.
minute(timestamp)	Retorna o componente de minuto da string/timestamp.
month(date)	Retorna o componente de mês da date/timestamp.
months_between(timestamp1, timestamp2[, roundOff])	Se `timestamp1` for posterior a `timestamp2`, o resultado será positivo. Se `timestamp1` e `timestamp2` forem no mesmo dia do mês, ou se ambos forem o último dia do mês, a hora do dia será ignorada. Caso contrário, a diferença será calculada com base em 31 dias por mês e arredondada para 8 dígitos, a menos que roundOff=false.
next_day(start_date, day_of_week)	Retorna a primeira data posterior a `start_date` e nomeada como indicado. A função retornará NULL se pelo menos um dos parâmetros de entrada for NULL.

Função	Description
now()	Retorna o timestamp atual no início da avaliação da consulta.
quarter(date)	Retorna o trimestre do ano para a data, no intervalo de 1 a 4.
second(timestamp)	Retorna o componente de segundo da string/timestamp.
session_window(time_column, gap_duration)	Gera uma janela de sessão com um timestamp especificando a coluna e a duração do intervalo. Consulte “Types of time windows” no documento Structured Streaming guide para obter explicações e exemplos detalhados.
timestamp_micros(microseconds)	Cria um timestamp a partir do número de microssegundos desde a UTC epoch.
timestamp_millis(milliseconds)	Cria um timestamp a partir do número de milissegundos desde a UTC epoch.
timestamp_seconds(seconds)	Cria um timestamp a partir do número de segundos (pode ser fracionário) desde UTC epoch.
to_date(date_str[, fmt])	Analisa a expressão `date_str` com a expressão `fmt` para uma data. Retorna nulo com entrada inválida. Por padrão, segue as regras de conversão em data, se `fmt` for omitido.
to_timestamp(timestamp_str[, fmt])	Analisa a expressão `timestamp_str` com a expressão `fmt` para um timestamp. Retorna nulo com entrada inválida. Por padrão, seguirá as regras de conversão em timestamp, se `fmt` for omitido.

Função	Description
to_timestamp_ltz(timestamp_str[, fmt])	Analisa a expressão `timestamp_str` com a expressão `fmt` para um timestamp com o fuso horário local. Retorna nulo com entrada inválida. Por padrão, seguirá as regras de conversão em timestamp, se `fmt` for omitido.
to_timestamp_ntz(timestamp_str[, fmt])	Analisa a expressão `timestamp_str` com a expressão `fmt` para um timestamp sem fuso horário local. Retorna nulo com entrada inválida. Por padrão, seguirá as regras de conversão em timestamp, se `fmt` for omitido.
to_unix_timestamp(timeExp[, fmt])	Retorna o timestamp UNIX da hora fornecida.
to_utc_timestamp(timestamp, timezone)	Fornecido um timestamp como “2017-07-14 02:40:00.0 ”, interpreta-o como uma hora em UTC e processa essa hora como um timestamp em UTC. Por exemplo, “GMT+1” produziria “2017-07-14 01:40:00.0”.
trunc(date, fmt)	Retorna `date` com a parte de hora do dia truncada para a unidade especificada pelo modelo de formato `fmt`.
try_to_timestamp(timestamp_str[, fmt])	Analisa a expressão `timestamp_str` com a expressão `fmt` para um timestamp.
unix_date(date)	Retorna o número de dias desde 1/01/1970.
unix_micros(timestamp)	Retorna o número de microssegundos desde 1/01/1970 00:00:00 UTC.
unix_millis(timestamp)	Retorna o número de milissegundos desde 1/01/1970 00:00:00 UTC. Trunca níveis mais altos de precisão.

Função	Description
unix_seconds(timestamp)	Retorna o número de segundos desde 1/01/1970 00:00:00 UTC. Trunca níveis mais altos de precisão.
unix_timestamp([timeExp[, fmt]])	Retorna o timestamp UNIX da hora atual ou da hora especificada.
weekday(date)	Retorna o dia da semana para date/timestamp (0 = segunda-feira, 1 = terça-feira,..., 6 = domingo).
weekofyear(date)	Retorna a semana do ano da data fornecida . Considera-se que uma semana começa na segunda-feira e a semana 1 é a primeira semana com mais de 3 dias.
window(time_column, window_duration[, slide_duration[, start_time]])	Fornecido um timestamp, agrupa linhas em uma ou mais janelas de tempo. O início da janela é inclusivo, mas o fim da janelas é exclusivo, por exemplo, 12:05 está na janela [12:05,12:10), mas não em [12:00,12:05). As janelas são compatíveis com uma precisão de microssegundos. Janelas da ordem de meses não são compatíveis. Consulte “Window Operations on Event Time” no documento Structured Streaming guide para obter explicações e exemplos detalhados.

Função	Description
window_time(window_column)	Extraia o valor da hora da coluna da time/session janela que pode ser usada para o valor da hora do evento da janela. A hora extraída é (window.end - 1), o que reflete o fato de que as janelas de agregação têm um limite superior exclusivo - [start, end). Consulte “Window Operations on Event Time” no documento Structured Streaming guide para obter explicações e exemplos detalhados.
year(date)	Retorna o componente de ano da date/time stamp.

Exemplos

```
-- add_months
SELECT add_months('2016-08-31', 1);
+-------------------------+
|add_months(2016-08-31, 1)|
+-------------------------+
|               2016-09-30|
+-------------------------+
-- convert_timezone
SELECT convert_timezone('Europe/Brussels', 'America/Los_Angeles',
 timestamp_ntz'2021-12-06 00:00:00');
+-----------------------------------------------------------------------------------------------
+
|convert_timezone(Europe/Brussels, America/Los_Angeles, TIMESTAMP_NTZ '2021-12-06
 00:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                                2021-12-05
 15:00:00|
+-----------------------------------------------------------------------------------------------
+
SELECT convert_timezone('Europe/Brussels', timestamp_ntz'2021-12-05 15:00:00');
+-----------------------------------------------------------------------------------------------
+
```

```
|convert_timezone(current_timezone(), Europe/Brussels, TIMESTAMP_NTZ '2021-12-05
 15:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                                2021-12-05
 07:00:00|
+-----------------------------------------------------------------------------------------------
+
-- curdate
SELECT curdate();
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_date
SELECT current_date();
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
SELECT current_date;
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_timestamp
SELECT current_timestamp();
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
SELECT current_timestamp;
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- current_timezone
SELECT current_timezone();
+------------------+
```

```
|current_timezone()|
+------------------+
|        Asia/Seoul|
+------------------+
-- date_add
SELECT date_add('2016-07-30', 1);
+-----------------------+
|date_add(2016-07-30, 1)|
+-----------------------+
|             2016-07-31|
+-----------------------+
-- date_diff
SELECT date_diff('2009-07-31', '2009-07-30');
+---------------------------------+
|date_diff(2009-07-31, 2009-07-30)|
+---------------------------------+
|                                1|
+---------------------------------+
SELECT date_diff('2009-07-30', '2009-07-31');
+---------------------------------+
|date_diff(2009-07-30, 2009-07-31)|
+---------------------------------+
|                               -1|
+---------------------------------+
-- date_format
SELECT date_format('2016-04-08', 'y');
+--------------------------+
|date_format(2016-04-08, y)|
+--------------------------+
|                      2016|
+--------------------------+
-- date_from_unix_date
SELECT date_from_unix_date(1);
+----------------------+
|date_from_unix_date(1)|
+----------------------+
|            1970-01-02|
+----------------------+
-- date_part
SELECT date_part('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-------------------------------------------------------+
|date_part(YEAR, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-------------------------------------------------------+
|                                                   2019|
```

```
+-----------------------------------------------------+
SELECT date_part('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------+
|date_part(week, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------+
|                                                   33|
+-----------------------------------------------------+
SELECT date_part('doy', DATE'2019-08-12');
+--------------------------------+
|date_part(doy, DATE '2019-08-12')|
+--------------------------------+
|                             224|
+--------------------------------+
SELECT date_part('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+------------------------------------------------------------+
|date_part(SECONDS, TIMESTAMP '2019-10-01 00:00:01.000001')|
+------------------------------------------------------------+
|                                                    1.000001|
+------------------------------------------------------------+
SELECT date_part('days', interval 5 days 3 hours 7 minutes);
+-------------------------------------------------+
|date_part(days, INTERVAL '5 03:07' DAY TO MINUTE)|
+-------------------------------------------------+
|                                                5|
+-------------------------------------------------+
SELECT date_part('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-------------------------------------------------------------+
|date_part(seconds, INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-------------------------------------------------------------+
|                                                    30.001001|
+-------------------------------------------------------------+
SELECT date_part('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+-------------------------------------------------+
|date_part(MONTH, INTERVAL '2021-11' YEAR TO MONTH)|
+-------------------------------------------------+
|                                               11|
+-------------------------------------------------+
SELECT date_part('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+---------------------------------------------------------------+
|date_part(MINUTE, INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+---------------------------------------------------------------+
|                                                             55|
+---------------------------------------------------------------+
-- date_sub
```

```
SELECT date_sub('2016-07-30', 1);
+-----------------------+
|date_sub(2016-07-30, 1)|
+-----------------------+
|             2016-07-29|
+-----------------------+
-- date_trunc
SELECT date_trunc('YEAR', '2015-03-05T09:32:05.359');
+----------------------------------------+
|date_trunc(YEAR, 2015-03-05T09:32:05.359)|
+----------------------------------------+
|                     2015-01-01 00:00:00|
+----------------------------------------+
SELECT date_trunc('MM', '2015-03-05T09:32:05.359');
+---------------------------------------+
|date_trunc(MM, 2015-03-05T09:32:05.359)|
+---------------------------------------+
|                    2015-03-01 00:00:00|
+---------------------------------------+
SELECT date_trunc('DD', '2015-03-05T09:32:05.359');
+---------------------------------------+
|date_trunc(DD, 2015-03-05T09:32:05.359)|
+---------------------------------------+
|                    2015-03-05 00:00:00|
+---------------------------------------+
SELECT date_trunc('HOUR', '2015-03-05T09:32:05.359');
+----------------------------------------+
|date_trunc(HOUR, 2015-03-05T09:32:05.359)|
+----------------------------------------+
|                     2015-03-05 09:00:00|
+----------------------------------------+
SELECT date_trunc('MILLISECOND', '2015-03-05T09:32:05.123456');
+-----------------------------------------------------+
|date_trunc(MILLISECOND, 2015-03-05T09:32:05.123456)|
+-----------------------------------------------------+
|                              2015-03-05 09:32:...|
+-----------------------------------------------------+
-- dateadd
SELECT dateadd('2016-07-30', 1);
+-----------------------+
|date_add(2016-07-30, 1)|
+-----------------------+
|             2016-07-31|
+-----------------------+
```

```
-- datediff
SELECT datediff('2009-07-31', '2009-07-30');
+--------------------------------+
|datediff(2009-07-31, 2009-07-30)|
+--------------------------------+
|                               1|
+--------------------------------+
SELECT datediff('2009-07-30', '2009-07-31');
+--------------------------------+
|datediff(2009-07-30, 2009-07-31)|
+--------------------------------+
|                              -1|
+--------------------------------+
-- datepart
SELECT datepart('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|datepart(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT datepart('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|datepart(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT datepart('doy', DATE'2019-08-12');
+-----------------------------------+
|datepart(doy FROM DATE '2019-08-12')|
+-----------------------------------+
|                                224|
+-----------------------------------+
SELECT datepart('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|datepart(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT datepart('days', interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|datepart(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
```

```
SELECT datepart('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-------------------------------------------------------------------+
|datepart(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-------------------------------------------------------------------+
|                                                          30.001001|
+-------------------------------------------------------------------+
SELECT datepart('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+-----------------------------------------------------+
|datepart(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+-----------------------------------------------------+
|                                                   11|
+-----------------------------------------------------+
SELECT datepart('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|datepart(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- day
SELECT day('2009-07-30');
+---------------+
|day(2009-07-30)|
+---------------+
|             30|
+---------------+
-- dayofmonth
SELECT dayofmonth('2009-07-30');
+----------------------+
|dayofmonth(2009-07-30)|
+----------------------+
|                    30|
+----------------------+
-- dayofweek
SELECT dayofweek('2009-07-30');
+---------------------+
|dayofweek(2009-07-30)|
+---------------------+
|                    5|
+---------------------+
-- dayofyear
SELECT dayofyear('2016-04-09');
+---------------------+
|dayofyear(2016-04-09)|
+---------------------+
```

```
|                  100|
+---------------------+
-- extract
SELECT extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT extract(week FROM timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT extract(doy FROM DATE'2019-08-12');
+----------------------------------+
|extract(doy FROM DATE '2019-08-12')|
+----------------------------------+
|                               224|
+----------------------------------+
SELECT extract(SECONDS FROM timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|extract(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT extract(days FROM interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|extract(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
SELECT extract(seconds FROM interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-----------------------------------------------------------------+
|extract(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-----------------------------------------------------------------+
|                                                        30.001001|
+-----------------------------------------------------------------+
SELECT extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH);
+---------------------------------------------------+
|extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+---------------------------------------------------+
|                                                  11|
```

```
+-----------------------------------------------------+
SELECT extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- from_unixtime
SELECT from_unixtime(0, 'yyyy-MM-dd HH:mm:ss');
+-------------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-------------------------------------+
|                  1970-01-01 09:00:00|
+-------------------------------------+
SELECT from_unixtime(0);
+-------------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-------------------------------------+
|                  1970-01-01 09:00:00|
+-------------------------------------+
-- from_utc_timestamp
SELECT from_utc_timestamp('2016-08-31', 'Asia/Seoul');
+-----------------------------------------+
|from_utc_timestamp(2016-08-31, Asia/Seoul)|
+-----------------------------------------+
|                      2016-08-31 09:00:00|
+-----------------------------------------+
-- hour
SELECT hour('2009-07-30 12:58:59');
+-------------------------+
|hour(2009-07-30 12:58:59)|
+-------------------------+
|                       12|
+-------------------------+
-- last_day
SELECT last_day('2009-01-12');
+--------------------+
|last_day(2009-01-12)|
+--------------------+
|          2009-01-31|
+--------------------+
-- localtimestamp
SELECT localtimestamp();
+--------------------+
```

```
|    localtimestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- make_date
SELECT make_date(2013, 7, 15);
+----------------------+
|make_date(2013, 7, 15)|
+----------------------+
|            2013-07-15|
+----------------------+
SELECT make_date(2019, 7, NULL);
+------------------------+
|make_date(2019, 7, NULL)|
+------------------------+
|                    NULL|
+------------------------+
-- make_dt_interval
SELECT make_dt_interval(1, 12, 30, 01.001001);
+-------------------------------------+
|make_dt_interval(1, 12, 30, 1.001001)|
+-------------------------------------+
|                 INTERVAL '1 12:30...|
+-------------------------------------+
SELECT make_dt_interval(2);
+---------------------------------+
|make_dt_interval(2, 0, 0, 0.000000)|
+---------------------------------+
|               INTERVAL '2 00:00...|
+---------------------------------+
SELECT make_dt_interval(100, null, 3);
+--------------------------------------+
|make_dt_interval(100, NULL, 3, 0.000000)|
+--------------------------------------+
|                                  NULL|
+--------------------------------------+
-- make_interval
SELECT make_interval(100, 11, 1, 1, 12, 30, 01.001001);
+-----------------------------------------------+
|make_interval(100, 11, 1, 1, 12, 30, 1.001001)|
+-----------------------------------------------+
|                         100 years 11 mont...|
+-----------------------------------------------+
SELECT make_interval(100, null, 3);
```

```
+----------------------------------------------+
|make_interval(100, NULL, 3, 0, 0, 0, 0.000000)|
+----------------------------------------------+
|                                          NULL|
+----------------------------------------------+
SELECT make_interval(0, 1, 0, 1, 0, 0, 100.000001);
+-----------------------------------------+
|make_interval(0, 1, 0, 1, 0, 0, 100.000001)|
+-----------------------------------------+
|                      1 months 1 days 1...|
+-----------------------------------------+
-- make_timestamp
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------+
|make_timestamp(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------+
|                      2014-12-28 06:30:...|
+-----------------------------------------+
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887, 'CET');
+-------------------------------------------------+
|make_timestamp(2014, 12, 28, 6, 30, 45.887, CET)|
+-------------------------------------------------+
|                           2014-12-28 14:30:...|
+-------------------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 60);
+--------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 60)|
+--------------------------------------+
|                    2019-07-01 00:00:00|
+--------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 1);
+--------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 1)|
+--------------------------------------+
|                   2019-06-30 23:59:01|
+--------------------------------------+
SELECT make_timestamp(null, 7, 22, 15, 30, 0);
+-------------------------------------+
|make_timestamp(NULL, 7, 22, 15, 30, 0)|
+-------------------------------------+
|                                 NULL|
+-------------------------------------+
-- make_timestamp_ltz
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887);
```

```
+-----------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                           2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, 'CET');
+-------------------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, CET)|
+-------------------------------------------------------+
|                                2014-12-28 14:30:...|
+-------------------------------------------------------+
SELECT make_timestamp_ltz(2019, 6, 30, 23, 59, 60);
+-----------------------------------------+
|make_timestamp_ltz(2019, 6, 30, 23, 59, 60)|
+-----------------------------------------+
|                      2019-07-01 00:00:00|
+-----------------------------------------+
SELECT make_timestamp_ltz(null, 7, 22, 15, 30, 0);
+-----------------------------------------+
|make_timestamp_ltz(NULL, 7, 22, 15, 30, 0)|
+-----------------------------------------+
|                                     NULL|
+-----------------------------------------+
-- make_timestamp_ntz
SELECT make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------------+
|make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                           2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ntz(2019, 6, 30, 23, 59, 60);
+-----------------------------------------+
|make_timestamp_ntz(2019, 6, 30, 23, 59, 60)|
+-----------------------------------------+
|                      2019-07-01 00:00:00|
+-----------------------------------------+
SELECT make_timestamp_ntz(null, 7, 22, 15, 30, 0);
+-----------------------------------------+
|make_timestamp_ntz(NULL, 7, 22, 15, 30, 0)|
+-----------------------------------------+
|                                     NULL|
+-----------------------------------------+
-- make_ym_interval
SELECT make_ym_interval(1, 2);
```

```
+----------------------+
|make_ym_interval(1, 2)|
+----------------------+
|  INTERVAL '1-2' YE...|
+----------------------+
SELECT make_ym_interval(1, 0);
+----------------------+
|make_ym_interval(1, 0)|
+----------------------+
|  INTERVAL '1-0' YE...|
+----------------------+
SELECT make_ym_interval(-1, 1);
+----------------------+
|make_ym_interval(-1, 1)|
+----------------------+
|   INTERVAL '-0-11' ...|
+----------------------+
SELECT make_ym_interval(2);
+----------------------+
|make_ym_interval(2, 0)|
+----------------------+
|  INTERVAL '2-0' YE...|
+----------------------+
-- minute
SELECT minute('2009-07-30 12:58:59');
+---------------------------+
|minute(2009-07-30 12:58:59)|
+---------------------------+
|                         58|
+---------------------------+
-- month
SELECT month('2016-07-30');
+-----------------+
|month(2016-07-30)|
+-----------------+
|                7|
+-----------------+
-- months_between
SELECT months_between('1997-02-28 10:30:00', '1996-10-30');
+-----------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, true)|
+-----------------------------------------------------+
|                                           3.94959677|
+-----------------------------------------------------+
```

```
SELECT months_between('1997-02-28 10:30:00', '1996-10-30', false);
+-------------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, false)|
+-------------------------------------------------------+
|                                    3.9495967741935485|
+-------------------------------------------------------+
-- next_day
SELECT next_day('2015-01-14', 'TU');
+------------------------+
|next_day(2015-01-14, TU)|
+------------------------+
|              2015-01-20|
+------------------------+
-- now
SELECT now();
+--------------------+
|               now()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- quarter
SELECT quarter('2016-08-31');
+-------------------+
|quarter(2016-08-31)|
+-------------------+
|                  3|
+-------------------+
-- second
SELECT second('2009-07-30 12:58:59');
+---------------------------+
|second(2009-07-30 12:58:59)|
+---------------------------+
|                         59|
+---------------------------+
-- session_window
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, session_window(b, '5 minutes')
 ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
```

```
| A2|2021-01-01 00:01:00|2021-01-01 00:06:00|  1|
+---+-------------------+-------------------+---+
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00'), ('A2', '2021-01-01 00:04:30') AS tab(a, b) GROUP by a,
 session_window(b, CASE WHEN a = 'A1' THEN '5 minutes' WHEN a = 'A2' THEN '1 minute'
 ELSE '10 minutes' END) ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
| A2|2021-01-01 00:01:00|2021-01-01 00:02:00|  1|
| A2|2021-01-01 00:04:30|2021-01-01 00:05:30|  1|
+---+-------------------+-------------------+---+
-- timestamp_micros
SELECT timestamp_micros(1230219000123123);
+----------------------------------+
|timestamp_micros(1230219000123123)|
+----------------------------------+
|              2008-12-26 00:30:...|
+----------------------------------+
-- timestamp_millis
SELECT timestamp_millis(1230219000123);
+-------------------------------+
|timestamp_millis(1230219000123)|
+-------------------------------+
|           2008-12-26 00:30:...|
+-------------------------------+
-- timestamp_seconds
SELECT timestamp_seconds(1230219000);
+-----------------------------+
|timestamp_seconds(1230219000)|
+-----------------------------+
|          2008-12-26 00:30:00|
+-----------------------------+
SELECT timestamp_seconds(1230219000.123);
+---------------------------------+
|timestamp_seconds(1230219000.123)|
+---------------------------------+
|             2008-12-26 00:30:...|
+---------------------------------+
-- to_date
SELECT to_date('2009-07-30 04:17:52');
```

```
+---------------------------+
|to_date(2009-07-30 04:17:52)|
+---------------------------+
|                 2009-07-30|
+---------------------------+
SELECT to_date('2016-12-31', 'yyyy-MM-dd');
+-------------------------------+
|to_date(2016-12-31, yyyy-MM-dd)|
+-------------------------------+
|                     2016-12-31|
+-------------------------------+
-- to_timestamp
SELECT to_timestamp('2016-12-31 00:12:00');
+---------------------------------+
|to_timestamp(2016-12-31 00:12:00)|
+---------------------------------+
|              2016-12-31 00:12:00|
+---------------------------------+
SELECT to_timestamp('2016-12-31', 'yyyy-MM-dd');
+-----------------------------------+
|to_timestamp(2016-12-31, yyyy-MM-dd)|
+-----------------------------------+
|                2016-12-31 00:00:00|
+-----------------------------------+
-- to_timestamp_ltz
SELECT to_timestamp_ltz('2016-12-31 00:12:00');
+-----------------------------------+
|to_timestamp_ltz(2016-12-31 00:12:00)|
+-----------------------------------+
|                 2016-12-31 00:12:00|
+-----------------------------------+
SELECT to_timestamp_ltz('2016-12-31', 'yyyy-MM-dd');
+---------------------------------------+
|to_timestamp_ltz(2016-12-31, yyyy-MM-dd)|
+---------------------------------------+
|                     2016-12-31 00:00:00|
+---------------------------------------+
-- to_timestamp_ntz
SELECT to_timestamp_ntz('2016-12-31 00:12:00');
+-----------------------------------+
|to_timestamp_ntz(2016-12-31 00:12:00)|
+-----------------------------------+
|                 2016-12-31 00:12:00|
+-----------------------------------+
```

```
SELECT to_timestamp_ntz('2016-12-31', 'yyyy-MM-dd');
+----------------------------------------+
|to_timestamp_ntz(2016-12-31, yyyy-MM-dd)|
+----------------------------------------+
|                     2016-12-31 00:00:00|
+----------------------------------------+
-- to_unix_timestamp
SELECT to_unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+----------------------------------------+
|to_unix_timestamp(2016-04-08, yyyy-MM-dd)|
+----------------------------------------+
|                                1460041200|
+----------------------------------------+
-- to_utc_timestamp
SELECT to_utc_timestamp('2016-08-31', 'Asia/Seoul');
+----------------------------------------+
|to_utc_timestamp(2016-08-31, Asia/Seoul)|
+----------------------------------------+
|                     2016-08-30 15:00:00|
+----------------------------------------+
-- trunc
SELECT trunc('2019-08-04', 'week');
+-----------------------+
|trunc(2019-08-04, week)|
+-----------------------+
|             2019-07-29|
+-----------------------+
SELECT trunc('2019-08-04', 'quarter');
+--------------------------+
|trunc(2019-08-04, quarter)|
+--------------------------+
|                2019-07-01|
+--------------------------+
SELECT trunc('2009-02-12', 'MM');
+---------------------+
|trunc(2009-02-12, MM)|
+---------------------+
|           2009-02-01|
+---------------------+
SELECT trunc('2015-10-27', 'YEAR');
+-----------------------+
|trunc(2015-10-27, YEAR)|
+-----------------------+
|             2015-01-01|
```

```
+-----------------------+
-- try_to_timestamp
SELECT try_to_timestamp('2016-12-31 00:12:00');
+-------------------------------------+
|try_to_timestamp(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT try_to_timestamp('2016-12-31', 'yyyy-MM-dd');
+---------------------------------------+
|try_to_timestamp(2016-12-31, yyyy-MM-dd)|
+---------------------------------------+
|                     2016-12-31 00:00:00|
+---------------------------------------+
SELECT try_to_timestamp('foo', 'yyyy-MM-dd');
+---------------------------------+
|try_to_timestamp(foo, yyyy-MM-dd)|
+---------------------------------+
|                             NULL|
+---------------------------------+
-- unix_date
SELECT unix_date(DATE("1970-01-02"));
+---------------------+
|unix_date(1970-01-02)|
+---------------------+
|                    1|
+---------------------+
-- unix_micros
SELECT unix_micros(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
|unix_micros(1970-01-01 00:00:01Z)|
+--------------------------------+
|                         1000000|
+--------------------------------+
-- unix_millis
SELECT unix_millis(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
|unix_millis(1970-01-01 00:00:01Z)|
+--------------------------------+
|                            1000|
+--------------------------------+
-- unix_seconds
SELECT unix_seconds(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
```

```
|unix_seconds(1970-01-01 00:00:01Z)|
+----------------------------------+
|                                 1|
+----------------------------------+
-- unix_timestamp
SELECT unix_timestamp();
+-----------------------------------------------------------+
|unix_timestamp(current_timestamp(), yyyy-MM-dd HH:mm:ss)|
+-----------------------------------------------------------+
|                                                 1708760216|
+-----------------------------------------------------------+
SELECT unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+--------------------------------------+
|unix_timestamp(2016-04-08, yyyy-MM-dd)|
+--------------------------------------+
|                            1460041200|
+--------------------------------------+
-- weekday
SELECT weekday('2009-07-30');
+-------------------+
|weekday(2009-07-30)|
+-------------------+
|                  3|
+-------------------+
-- weekofyear
SELECT weekofyear('2008-02-20');
+----------------------+
|weekofyear(2008-02-20)|
+----------------------+
|                     8|
+----------------------+
-- window
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a,
 start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|  1|
+---+-------------------+-------------------+---+
```

```
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '10 minutes', '5 minutes')
 ORDER BY a, start;
+---+------------------+------------------+---+
|  a|             start|               end|cnt|
+---+------------------+------------------+---+
| A1|2020-12-31 23:55:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:00:00|2021-01-01 00:10:00|  3|
| A1|2021-01-01 00:05:00|2021-01-01 00:15:00|  1|
| A2|2020-12-31 23:55:00|2021-01-01 00:05:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:10:00|  1|
+---+------------------+------------------+---+
-- window_time
SELECT a, window.start as start, window.end as end, window_time(window), cnt FROM
 (SELECT a, window, count(*) as cnt FROM VALUES ('A1', '2021-01-01 00:00:00'), ('A1',
 '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2', '2021-01-01 00:01:00')
 AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a, window.start);
+---+------------------+------------------+--------------------+---+
|  a|             start|               end| window_time(window)|cnt|
+---+------------------+------------------+--------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|2021-01-01 00:09:...|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  1|
+---+------------------+------------------+--------------------+---+
-- year
SELECT year('2016-07-30');
+----------------+
|year(2016-07-30)|
+----------------+
|            2016|
+----------------+
```

Funções agregadas

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

As funções agregadas operam em valores em linhas para realizar cálculos matemáticos, como soma, média, contagem, minimum/maximum valores, desvio padrão e estimativa, bem como algumas operações não matemáticas.

Sintaxe

```
aggregate_function(input1 [, input2, ...]) FILTER (WHERE boolean_expression)
```

Parâmetros

- `boolean_expression`: especifica qualquer expressão que é avaliada como um resultado do tipo booleano. Duas ou mais expressões podem ser combinadas usando os operadores lógicos (AND, OR).

Funções de agregação em conjunto ordenado

Essas funções de agregação usam uma sintaxe diferente das outras funções de agregação para especificar uma expressão (normalmente um nome de coluna) segundo a qual os valores serão ordenados.

Sintaxe

```
{ PERCENTILE_CONT | PERCENTILE_DISC }(percentile) WITHIN GROUP (ORDER BY
 { order_by_expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] }) FILTER
 (WHERE boolean_expression)
```

Parâmetros

- `percentile`: o percentil do valor que você deseja encontrar. O percentil deve ser uma constante entre 0,0 e 1,0.
- `order_by_expression`: a expressão (normalmente um nome de coluna) segundo a qual os valores serão ordenados antes de serem agregados.
- `boolean_expression`: especifica qualquer expressão que é avaliada como um resultado do tipo booleano. Duas ou mais expressões podem ser combinadas usando os operadores lógicos (AND, OR).

Exemplos

```
CREATE OR REPLACE TEMPORARY VIEW basic_pays AS SELECT * FROM VALUES
```

```
('Jane Doe','Accounting',8435),
('Akua Mansa','Accounting',9998),
('John Doe','Accounting',8992),
('Juan Li','Accounting',8870),
('Carlos Salazar','Accounting',11472),
('Arnav Desai','Accounting',6627),
('Saanvi Sarkar','IT',8113),
('Shirley Rodriguez','IT',5186),
('Nikki Wolf','Sales',9181),
('Alejandro Rosalez','Sales',9441),
('Nikhil Jayashankar','Sales',6660),
('Richard Roe','Sales',10563),
('Pat Candella','SCM',10449),
('Gerard Hernandez','SCM',6949),
('Pamela Castillo','SCM',11303),
('Paulo Santos','SCM',11798),
('Jorge Souza','SCM',10586)
AS basic_pays(employee_name, department, salary);
SELECT * FROM basic_pays;
+------------------+----------+------+
|    employee_name  |department|salary|
+------------------+----------+------+
| Arnav Desai       |Accounting|  6627|
| Jorge Souza       |       SCM| 10586|
| Jane Doe          |Accounting|  8435|
| Nikhil Jayashankar|     Sales|  6660|
| Diego Vanauf      |     Sales| 10563|
| Carlos Salazar    |Accounting| 11472|
| Gerard Hernandez  |       SCM|  6949|
| John Doe          |Accounting|  8992|
| Nikki Wolf        |     Sales|  9181|
| Paulo Santos      |       SCM| 11798|
| Saanvi Sarkar     |        IT|  8113|
| Shirley Rodriguez |        IT|  5186|
| Pat Candella      |       SCM| 10449|
| Akua Mansa        |Accounting|  9998|
| Pamela Castillo   |       SCM| 11303|
| Alejandro Rosalez |     Sales|  9441|
| Juan Li           |Accounting|  8870|
+------------------+----------+------+
SELECT
department,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) AS pc1,
```

```
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pc2,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pc3,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pc4,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) AS pd1,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pd2,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pd3,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pd4
FROM basic_pays
GROUP BY department
ORDER BY department;
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|department|    pc1|     pc2|    pc3|     pc4|  pd1|  pd2|  pd3|  pd4|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|Accounting|8543.75| 7838.25| 9746.5|10260.75| 8435| 6627| 9998|11472|
|        IT|5917.75|    NULL|7381.25|    NULL| 5186| NULL| 8113| NULL|
|     Sales|8550.75|    NULL| 9721.5|    NULL| 6660| NULL|10563| NULL|
|       SCM|10449.0|10786.25|11303.0|11460.75|10449|10449|11303|11798|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
```

Funções condicionais

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
coalesce(expr1, expr2, ...)	Retornará o primeiro argumento não nulo, se existir. Caso contrário, nulo.
if(expr1, expr2, expr3)	Se `expr1` for avaliado como verdadeiro, retornará `expr2`; caso contrário, retornará `expr3`.

Função	Description
ifnulo(expr1, expr2)	Retornará `expr2` se `expr1` for nulo, caso contrário, retornará `expr1`.
nanvl(expr1, expr2)	Retornará `expr1` se for NaN, caso contrário, retornará `expr2`.
nuloif(expr1, expr2)	Retornará nulo se `expr1` for igual a `expr2`, caso contrário retornará `expr1`.
nvl(expr1, expr2)	Retornará `expr2` se `expr1` for nulo, caso contrário, retornará `expr1`.
nvl2(expr1, expr2, expr3)	Retornará `expr2` se `expr1` não for nulo, caso contrário, retornará `expr3`.
CASE WHEN expr1 THEN expr2 [WHEN expr3 THEN expr4]* [ELSE expr5] END	Quando `expr1` = verdadeiro, retorna `expr2`; senão, quando `expr3` = verdadeiro , retorna `expr4`; caso contrário, retorna `expr5`.

Exemplos

```
-- coalesce
SELECT coalesce(NULL, 1, NULL);
+----------------------+
|coalesce(NULL, 1, NULL)|
+----------------------+
|                     1|
+----------------------+
-- if
SELECT if(1 < 2, 'a', 'b');
+------------------+
|(IF((1 < 2), a, b))|
+------------------+
|                 a|
+------------------+
-- ifnull
SELECT ifnull(NULL, array('2'));
+---------------------+
```

```
|ifnull(NULL, array(2))|
+----------------------+
|                   [2]|
+----------------------+
-- nanvl
SELECT nanvl(cast('NaN' as double), 123);
+------------------------------+
|nanvl(CAST(NaN AS DOUBLE), 123)|
+------------------------------+
|                         123.0|
+------------------------------+
-- nullif
SELECT nullif(2, 2);
+------------+
|nullif(2, 2)|
+------------+
|        NULL|
+------------+
-- nvl
SELECT nvl(NULL, array('2'));
+------------------+
|nvl(NULL, array(2))|
+------------------+
|               [2]|
+------------------+
-- nvl2
SELECT nvl2(NULL, 2, 1);
+----------------+
|nvl2(NULL, 2, 1)|
+----------------+
|               1|
+----------------+
-- when
SELECT CASE WHEN 1 > 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 > 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          1.0|
+-------------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          2.0|
```

```
+-----------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 < 0 THEN 2.0 END;
+-------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 < 0) THEN 2.0 END|
+-------------------------------------------------+
|                                             NULL|
+-------------------------------------------------+
```

Funções JSON

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
from_json (jsonStr, schema[, options])	Retorna um valor de estrutura com o `jsonStr`e o `schema` fornecidos.
get_json_ object(js on_txt, path)	Extrai um objeto json de `path`.
json_arra y_length(jsonArray)	Retorna o número de elementos na matriz JSON mais externa.
json_obje ct_keys(j son_object)	Retorna todas as chaves do objeto JSON mais externo como uma matriz.
json_tupl e(jsonStr, p1, p2, ..., pn)	Retorna uma tupla como a função get_json_object, mas aceita vários nomes. Os tipos de todos os parâmetros de entrada e colunas de saída são strings.

Função	Description
schema_of _json(json[, options])	Retorna esquema no formato DDL de string JSON.
to_json(expr[, options])	Retorna uma string JSON com um valor de estrutura fornecido

Exemplos

```
-- from_json
SELECT from_json('{"a":1, "b":0.8}', 'a INT, b DOUBLE');
+--------------------------+
| from_json({"a":1, "b":0.8}) |
+--------------------------+
| {1, 0.8}                 |
+--------------------------+

SELECT from_json('{"time":"26/08/2015"}', 'time Timestamp', map('timestampFormat', 'dd/
MM/yyyy'));
+--------------------------------+
| from_json({"time":"26/08/2015"}) |
+--------------------------------+
| {2015-08-26 00:00...           |
+--------------------------------+

SELECT from_json('{"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}', 'STRUCT<teacher: STRING, student: ARRAY<STRUCT<name: STRING,
 rank: INT>>>');
+-----------------------------------------------------------------------------------------------
+
| from_json({"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}) |
+-----------------------------------------------------------------------------------------------
+
| {Alice, [{Bob, 1}...
                  |
+-----------------------------------------------------------------------------------------------
+
```

```
-- get_json_object
SELECT get_json_object('{"a":"b"}', '$.a');
+-------------------------------+
| get_json_object({"a":"b"}, $.a) |
+-------------------------------+
| b                             |
+-------------------------------+

-- json_array_length
SELECT json_array_length('[1,2,3,4]');
+----------------------------+
| json_array_length([1,2,3,4]) |
+----------------------------+
| 4                          |
+----------------------------+

SELECT json_array_length('[1,2,3,{"f1":1,"f2":[5,6]},4]');
+-------------------------------------------------+
| json_array_length([1,2,3,{"f1":1,"f2":[5,6]},4]) |
+-------------------------------------------------+
| 5                                               |
+-------------------------------------------------+

SELECT json_array_length('[1,2');
+----------------------+
| json_array_length([1,2) |
+----------------------+
| NULL                 |
+----------------------+

-- json_object_keys
SELECT json_object_keys('{}');
+--------------------+
| json_object_keys({}) |
+--------------------+
| []                 |
+--------------------+

SELECT json_object_keys('{"key": "value"}');
+---------------------------------+
| json_object_keys({"key": "value"}) |
+---------------------------------+
| [key]                           |
+---------------------------------+
```

```
SELECT json_object_keys('{"f1":"abc","f2":{"f3":"a", "f4":"b"}}');
+---------------------------------------------------------+
| json_object_keys({"f1":"abc","f2":{"f3":"a", "f4":"b"}}) |
+---------------------------------------------------------+
| [f1, f2]                                                |
+---------------------------------------------------------+

-- json_tuple
SELECT json_tuple('{"a":1, "b":2}', 'a', 'b');
+---+---+
| c0| c1|
+---+---+
|  1|  2|
+---+---+

-- schema_of_json
SELECT schema_of_json('[{"col":0}]');
+---------------------------+
| schema_of_json([{"col":0}]) |
+---------------------------+
| ARRAY<STRUCT<col:...      |
+---------------------------+

SELECT schema_of_json('[{"col":01}]', map('allowNumericLeadingZeros', 'true'));
+----------------------------+
| schema_of_json([{"col":01}]) |
+----------------------------+
| ARRAY<STRUCT<col:...       |
+----------------------------+

-- to_json
SELECT to_json(named_struct('a', 1, 'b', 2));
+---------------------------------+
| to_json(named_struct(a, 1, b, 2)) |
+---------------------------------+
| {"a":1,"b":2}                   |
+---------------------------------+

SELECT to_json(named_struct('time', to_timestamp('2015-08-26', 'yyyy-MM-dd')),
 map('timestampFormat', 'dd/MM/yyyy'));
+-----------------------------------------------------------------+
| to_json(named_struct(time, to_timestamp(2015-08-26, yyyy-MM-dd))) |
+-----------------------------------------------------------------+
```

```
| {"time":"26/08/20...                                                  |
+-----------------------------------------------------------------------+

SELECT to_json(array(named_struct('a', 1, 'b', 2)));
+----------------------------------------+
| to_json(array(named_struct(a, 1, b, 2))) |
+----------------------------------------+
| [{"a":1,"b":2}]                        |
+----------------------------------------+

SELECT to_json(map('a', named_struct('b', 1)));
+-----------------------------------+
| to_json(map(a, named_struct(b, 1))) |
+-----------------------------------+
| {"a":{"b":1}}                     |
+-----------------------------------+

SELECT to_json(map(named_struct('a', 1),named_struct('b', 2)));
+-------------------------------------------------------+
| to_json(map(named_struct(a, 1), named_struct(b, 2))) |
+-------------------------------------------------------+
| {"[1]":{"b":2}}                                       |
+-------------------------------------------------------+

SELECT to_json(map('a', 1));
+------------------+
| to_json(map(a, 1)) |
+------------------+
| {"a":1}          |
+------------------+

SELECT to_json(array(map('a', 1)));
+--------------------------+
| to_json(array(map(a, 1))) |
+--------------------------+
| [{"a":1}]                |
+--------------------------+
```

Funções de array

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
array(expr, ...)	Retorna uma matriz com os elementos fornecidos.
array_append(array, element)	Adiciona o elemento no final da matriz passada como primeiro argumento. O tipo de elemento deve ser semelhante ao tipo dos elementos da matriz. O elemento nulo também é acrescentado à matriz. Mas se a matriz for passada for NULL, a saída será NULL
array_compact(array)	Remove valores nulos da matriz.
array_contains(array, value)	Retornará verdadeiro se a matriz contiver o valor.
array_distinct(array)	Remove valores duplicados da matriz.
array_except(array1, array2)	Retorna uma matriz dos elementos em array1 mas não em array2 sem duplicações.
array_insert(x, pos, val)	Insere val no índice pos da matriz x. Os índices da matriz começam em 1. O índice negativo máximo é -1 para o qual a função insere um novo elemento após o último elemento atual. O índice acima do tamanho da matriz ou acrescenta elementos “nulos” ao início da matriz, ou ao fim da matriz se o índice for negativo.

Função	Description
array_intersect(array1, array2)	Retorna uma matriz dos elementos na interseção de array1 e array2, sem duplicaçõ es.
array_join(array, delimiter[, nuloReplacement])	Concatena os elementos da matriz fornecida usando o delimitador e uma string opcional para substituir nulos. Se nenhum valor for definido para nullReplacement, todo valor nulo será filtrado.
array_max(array)	Retorna o valor máximo na matriz. NaN é maior do que qualquer elemento não NaN para o double/float tipo. Os elementos NULL são ignorados.
array_min(array)	Retorna o valor mínimo na matriz. NaN é maior do que qualquer elemento não NaN para o double/float tipo. Os elementos NULL são ignorados.
array_position(array, element)	Retorna o índice (de base 1) do primeiro elemento correspondente da matriz do mesmo tamanho ou 0 se nenhuma correspondência for encontrada.
array_prepend(array, element)	Adiciona o elemento no início da matriz passada como primeiro argumento. O tipo de elemento deve ser do mesmo tipo dos elementos da matriz. O elemento nulo também é incluído no início da matriz. Mas se a matriz for passada for NULL, a saída será NULL
array_remove(array, element)	Remove todos os elementos iguais ao elemento da matriz.

Função	Description
array_repeat(element, count)	Retorna a matriz contendo o número de vezes que o elemento ocorre.
array_union(array1, array2)	Retorna uma matriz dos elementos na união de array1 e array2, sem duplicações.
arrays_overlap(a1, a2)	Retornará verdadeiro se a1 contiver pelo menos um elemento não nulo presente também em a2. Se as matrizes não tiverem nenhum elemento em comum e ambas não forem vazias e uma delas contiver um elemento nulo, nulo será retornado, caso contrário, false será retornado.
arrays_zip(a1, a2, ...)	Retorna uma matriz mesclada de estruturas na qual a enésima estrutura contém todos os enésimos valores das matrizes inseridas.
achatar () arrayOfArrays	Transforma uma matriz de arrays em um único array.
get(array, index)	Retorna elemento de matriz no índice (baseado em 0) fornecido. Se o índice apontar para fora dos limites da matriz, essa função retornará NULL.

Função	Description
sequence(start, stop, step)	Gera uma matriz de elementos de iniciar até parar (inclusive), em incrementos. O tipo dos elementos retornados é igual ao tipo das expressões de argumento. Os tipos compatíveis são: byte, short, integer, long, date, timestamp. As expressões de iniciar e parar devem ser resolvidas para o mesmo tipo. Se as expressões de início e fim forem resolvidas para o tipo “date” ou “timestamp”, a expressão de etapa deverá ser resolvida para o tipo “interval”, “year-month interval” ou “day-time interval”, caso contrário, para o mesmo tipo das expressões de início e fim.
shuffle(array)	Retorna uma permutação aleatória da matriz fornecida.
slice(x, start, length)	Subdefine a matriz x começando do início do índice (os índices de matriz começam em 1 ou do fim se o início for negativo) com o comprimento especificado.
sort_array(array[, ascendingOrder])	Classifica a matriz de entrada em ordem crescente ou decrescente de acordo com a ordem natural dos elementos da matriz. NaN é maior do que qualquer elemento não NaN para o double/float tipo. Os elementos nulos serão colocados no início da matriz retornada em ordem crescente ou no fim da matriz retornada em ordem decrescente.

Exemplos

```
-- array
SELECT array(1, 2, 3);
```

```
+--------------+
|array(1, 2, 3)|
+--------------+
|     [1, 2, 3]|
+--------------+
-- array_append
SELECT array_append(array('b', 'd', 'c', 'a'), 'd');
+---------------------------------+
|array_append(array(b, d, c, a), d)|
+---------------------------------+
|                   [b, d, c, a, d]|
+---------------------------------+
SELECT array_append(array(1, 2, 3, null), null);
+--------------------------------------+
|array_append(array(1, 2, 3, NULL), NULL)|
+--------------------------------------+
|                    [1, 2, 3, NULL, N...|
+--------------------------------------+
SELECT array_append(CAST(null as Array<Int>), 2);
+--------------------+
|array_append(NULL, 2)|
+--------------------+
|                NULL|
+--------------------+
-- array_compact
SELECT array_compact(array(1, 2, 3, null));
+---------------------------------+
|array_compact(array(1, 2, 3, NULL))|
+---------------------------------+
|                          [1, 2, 3]|
+---------------------------------+
SELECT array_compact(array("a", "b", "c"));
+----------------------------+
|array_compact(array(a, b, c))|
+----------------------------+
|                    [a, b, c]|
+----------------------------+
-- array_contains
SELECT array_contains(array(1, 2, 3), 2);
+---------------------------------+
|array_contains(array(1, 2, 3), 2)|
+---------------------------------+
|                             true|
+---------------------------------+
```

```
-- array_distinct
SELECT array_distinct(array(1, 2, 3, null, 3));
+--------------------------------------+
|array_distinct(array(1, 2, 3, NULL, 3))|
+--------------------------------------+
|                        [1, 2, 3, NULL]|
+--------------------------------------+
-- array_except
SELECT array_except(array(1, 2, 3), array(1, 3, 5));
+---------------------------------------------+
|array_except(array(1, 2, 3), array(1, 3, 5))|
+---------------------------------------------+
|                                          [2]|
+---------------------------------------------+
-- array_insert
SELECT array_insert(array(1, 2, 3, 4), 5, 5);
+-------------------------------------+
|array_insert(array(1, 2, 3, 4), 5, 5)|
+-------------------------------------+
|                     [1, 2, 3, 4, 5]|
+-------------------------------------+
SELECT array_insert(array(5, 4, 3, 2), -1, 1);
+-------------------------------------+
|array_insert(array(5, 4, 3, 2), -1, 1)|
+-------------------------------------+
|                      [5, 4, 3, 2, 1]|
+-------------------------------------+
SELECT array_insert(array(5, 3, 2, 1), -4, 4);
+-------------------------------------+
|array_insert(array(5, 3, 2, 1), -4, 4)|
+-------------------------------------+
|                      [5, 4, 3, 2, 1]|
+-------------------------------------+
-- array_intersect
SELECT array_intersect(array(1, 2, 3), array(1, 3, 5));
+-----------------------------------------------+
|array_intersect(array(1, 2, 3), array(1, 3, 5))|
+-----------------------------------------------+
|                                         [1, 3]|
+-----------------------------------------------+
-- array_join
SELECT array_join(array('hello', 'world'), ' ');
+---------------------------------+
|array_join(array(hello, world),  )|
```

```
+---------------------------------+
|                       hello world|
+---------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ');
+-----------------------------------------+
|array_join(array(hello, NULL, world),  )|
+-----------------------------------------+
|                             hello world|
+-----------------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ', ',');
+-------------------------------------------+
|array_join(array(hello, NULL, world),  , ,)|
+-------------------------------------------+
|                              hello , world|
+-------------------------------------------+
-- array_max
SELECT array_max(array(1, 20, null, 3));
+--------------------------------+
|array_max(array(1, 20, NULL, 3))|
+--------------------------------+
|                              20|
+--------------------------------+
-- array_min
SELECT array_min(array(1, 20, null, 3));
+--------------------------------+
|array_min(array(1, 20, NULL, 3))|
+--------------------------------+
|                               1|
+--------------------------------+
-- array_position
SELECT array_position(array(312, 773, 708, 708), 708);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 708)|
+-----------------------------------------------+
|                                              3|
+-----------------------------------------------+
SELECT array_position(array(312, 773, 708, 708), 414);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 414)|
+-----------------------------------------------+
|                                              0|
+-----------------------------------------------+
-- array_prepend
SELECT array_prepend(array('b', 'd', 'c', 'a'), 'd');
```

```
+----------------------------------+
|array_prepend(array(b, d, c, a), d)|
+----------------------------------+
|                    [d, b, d, c, a]|
+----------------------------------+
SELECT array_prepend(array(1, 2, 3, null), null);
+---------------------------------------+
|array_prepend(array(1, 2, 3, NULL), NULL)|
+---------------------------------------+
|                     [NULL, 1, 2, 3, N...|
+---------------------------------------+
SELECT array_prepend(CAST(null as Array<Int>), 2);
+----------------------+
|array_prepend(NULL, 2)|
+----------------------+
|                  NULL|
+----------------------+
-- array_remove
SELECT array_remove(array(1, 2, 3, null, 3), 3);
+--------------------------------------+
|array_remove(array(1, 2, 3, NULL, 3), 3)|
+--------------------------------------+
|                           [1, 2, NULL]|
+--------------------------------------+
-- array_repeat
SELECT array_repeat('123', 2);
+--------------------+
|array_repeat(123, 2)|
+--------------------+
|          [123, 123]|
+--------------------+
-- array_union
SELECT array_union(array(1, 2, 3), array(1, 3, 5));
+-----------------------------------------+
|array_union(array(1, 2, 3), array(1, 3, 5))|
+-----------------------------------------+
|                              [1, 2, 3, 5]|
+-----------------------------------------+
-- arrays_overlap
SELECT arrays_overlap(array(1, 2, 3), array(3, 4, 5));
+-------------------------------------------+
|arrays_overlap(array(1, 2, 3), array(3, 4, 5))|
+-------------------------------------------+
|                                       true|
```

```
+-----------------------------------------------+
-- arrays_zip
SELECT arrays_zip(array(1, 2, 3), array(2, 3, 4));
+----------------------------------------+
|arrays_zip(array(1, 2, 3), array(2, 3, 4))|
+----------------------------------------+
|                      [{1, 2}, {2, 3}, ...|
+----------------------------------------+
SELECT arrays_zip(array(1, 2), array(2, 3), array(3, 4));
+-------------------------------------------------+
|arrays_zip(array(1, 2), array(2, 3), array(3, 4))|
+-------------------------------------------------+
|                             [{1, 2, 3}, {2, 3...|
+-------------------------------------------------+
-- flatten
SELECT flatten(array(array(1, 2), array(3, 4)));
+---------------------------------------+
|flatten(array(array(1, 2), array(3, 4)))|
+---------------------------------------+
|                           [1, 2, 3, 4]|
+---------------------------------------+
-- get
SELECT get(array(1, 2, 3), 0);
+----------------------+
|get(array(1, 2, 3), 0)|
+----------------------+
|                     1|
+----------------------+
SELECT get(array(1, 2, 3), 3);
+----------------------+
|get(array(1, 2, 3), 3)|
+----------------------+
|                  NULL|
+----------------------+
SELECT get(array(1, 2, 3), -1);
+-----------------------+
|get(array(1, 2, 3), -1)|
+-----------------------+
|                   NULL|
+-----------------------+
-- sequence
SELECT sequence(1, 5);
+---------------+
| sequence(1, 5)|
```

```
+---------------+
|[1, 2, 3, 4, 5]|
+---------------+
SELECT sequence(5, 1);
+---------------+
| sequence(5, 1)|
+---------------+
|[5, 4, 3, 2, 1]|
+---------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval 1 month);
+------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '1' MONTH)|
+------------------------------------------------------------------------+
|                                                    [2018-01-01, 2018...|
+------------------------------------------------------------------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval '0-1' year to
 month);
+-----------------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '0-1' YEAR TO MONTH)|
+-----------------------------------------------------------------------------------+
|                                                                 [2018-01-01, 2018...|
+-----------------------------------------------------------------------------------+
-- shuffle
SELECT shuffle(array(1, 20, 3, 5));
+---------------------------+
|shuffle(array(1, 20, 3, 5))|
+---------------------------+
|              [5, 1, 20, 3]|
+---------------------------+
SELECT shuffle(array(1, 20, null, 3));
+-----------------------------+
|shuffle(array(1, 20, NULL, 3))|
+-----------------------------+
|              [1, NULL, 20, 3]|
+-----------------------------+
-- slice
SELECT slice(array(1, 2, 3, 4), 2, 2);
+-----------------------------+
|slice(array(1, 2, 3, 4), 2, 2)|
+-----------------------------+
|                        [2, 3]|
+-----------------------------+
SELECT slice(array(1, 2, 3, 4), -2, 2);
+-----------------------------+
```

```
|slice(array(1, 2, 3, 4), -2, 2)|
+------------------------------+
|                        [3, 4]|
+------------------------------+
-- sort_array
SELECT sort_array(array('b', 'd', null, 'c', 'a'), true);
+---------------------------------------+
|sort_array(array(b, d, NULL, c, a), true)|
+---------------------------------------+
|                      [NULL, a, b, c, d]|
+---------------------------------------+
```

Funções de janela

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

As funções de janelas operam em um grupo de linhas, chamado de janela, e calculam um valor de retorno para cada linha com base no grupo de linhas. As funções de janela são úteis para processar tarefas, como calcular uma média móvel, computar uma estatística cumulativa ou acessar o valor das linhas com base na posição relativa da linha atual.

Sintaxe

```
window_function [ nulls_option ] OVER ( [ { PARTITION | DISTRIBUTE } BY
 partition_col_name = partition_col_val ( [ , ... ] ) ] { ORDER | SORT } BY expression
 [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] [ window_frame ] )
```

Parâmetros

- Funções de classificação

  Sintaxe: `RANK | DENSE_RANK | PERCENT_RANK | NTILE | ROW_NUMBER`

  Funções analíticas

  Sintaxe: `CUME_DIST | LAG | LEAD | NTH_VALUE | FIRST_VALUE | LAST_VALUE`

  Funções agregadas

Sintaxe: `MAX | MIN | COUNT | SUM | AVG | ...`

- `nulls_option`: especifica se os valores nulos devem ou não ser ignorados ao avaliar a função de janela. RESPECT NULLS significa não pular valores nulos, enquanto IGNORE NULLS significa ignorá-los. Se não especificado, o padrão será RESPECT NULLS.

  Sintaxe: `{ IGNORE | RESPECT } NULLS`

  Observação: `Only LAG | LEAD | NTH_VALUE | FIRST_VALUE | LAST_VALUE` podem ser usadas com `IGNORE NULLS`.

- `window_frame`: especifica em qual linha começar a janela e onde terminá-la.

  Sintaxe: `{ RANGE | ROWS } { frame_start | BETWEEN frame_start AND frame_end }`

  frame_start e frame_end têm a seguinte sintaxe:

  Sintaxe: `UNBOUNDED PRECEDING | offset PRECEDING | CURRENT ROW | offset FOLLOWING | UNBOUNDED FOLLOWING`

  offset: especifica o deslocamento da posição da linha atual.

  Observação: se frame_end for omitido, o padrão CURRENT ROW será usado.

Exemplos

```
CREATE TABLE employees (name STRING, dept STRING, salary INT, age INT);
INSERT INTO employees VALUES ("Lisa", "Sales", 10000, 35);
INSERT INTO employees VALUES ("Evan", "Sales", 32000, 38);
INSERT INTO employees VALUES ("Fred", "Engineering", 21000, 28);
INSERT INTO employees VALUES ("Alex", "Sales", 30000, 33);
INSERT INTO employees VALUES ("Tom", "Engineering", 23000, 33);
INSERT INTO employees VALUES ("Jane", "Marketing", 29000, 28);
INSERT INTO employees VALUES ("Jeff", "Marketing", 35000, 38);
INSERT INTO employees VALUES ("Paul", "Engineering", 29000, 23);
INSERT INTO employees VALUES ("Chloe", "Engineering", 23000, 25);
SELECT * FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  age|
+-----+-----------+------+-----+
|Chloe|Engineering| 23000|   25|
```

```
| Fred|Engineering| 21000|   28|
| Paul|Engineering| 29000|   23|
|Helen|  Marketing| 29000|   40|
|  Tom|Engineering| 23000|   33|
| Jane|  Marketing| 29000|   28|
| Jeff|  Marketing| 35000|   38|
| Evan|      Sales| 32000|   38|
| Lisa|      Sales| 10000|   35|
| Alex|      Sales| 30000|   33|
+-----+-----------+------+-----+
SELECT name, dept, salary, RANK() OVER (PARTITION BY dept ORDER BY salary) AS rank FROM
 employees;
+-----+-----------+------+----+
| name|       dept|salary|rank|
+-----+-----------+------+----+
| Lisa|      Sales| 10000|   1|
| Alex|      Sales| 30000|   2|
| Evan|      Sales| 32000|   3|
| Fred|Engineering| 21000|   1|
|  Tom|Engineering| 23000|   2|
|Chloe|Engineering| 23000|   2|
| Paul|Engineering| 29000|   4|
|Helen|  Marketing| 29000|   1|
| Jane|  Marketing| 29000|   1|
| Jeff|  Marketing| 35000|   3|
+-----+-----------+------+----+
SELECT name, dept, salary, DENSE_RANK() OVER (PARTITION BY dept ORDER BY salary ROWS
 BETWEEN
UNBOUNDED PRECEDING AND CURRENT ROW) AS dense_rank FROM employees;
+-----+-----------+------+----------+
| name|       dept|salary|dense_rank|
+-----+-----------+------+----------+
| Lisa|      Sales| 10000|         1|
| Alex|      Sales| 30000|         2|
| Evan|      Sales| 32000|         3|
| Fred|Engineering| 21000|         1|
|  Tom|Engineering| 23000|         2|
|Chloe|Engineering| 23000|         2|
| Paul|Engineering| 29000|         3|
|Helen|  Marketing| 29000|         1|
| Jane|  Marketing| 29000|         1|
| Jeff|  Marketing| 35000|         2|
+-----+-----------+------+----------+
SELECT name, dept, age, CUME_DIST() OVER (PARTITION BY dept ORDER BY age
```

```
RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS cume_dist FROM employees;
+-----+-----------+------+------------------+
| name|       dept|age   |         cume_dist|
+-----+-----------+------+------------------+
| Alex|      Sales|    33|0.3333333333333333|
| Lisa|      Sales|    35|0.6666666666666666|
| Evan|      Sales|    38|               1.0|
| Paul|Engineering|    23|              0.25|
|Chloe|Engineering|    25|              0.75|
| Fred|Engineering|    28|              0.25|
|  Tom|Engineering|    33|               1.0|
| Jane|  Marketing|    28|0.3333333333333333|
| Jeff|  Marketing|    38|0.6666666666666666|
|Helen|  Marketing|    40|               1.0|
+-----+-----------+------+------------------+
SELECT name, dept, salary, MIN(salary) OVER (PARTITION BY dept ORDER BY salary) AS min
FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  min|
+-----+-----------+------+-----+
| Lisa|      Sales| 10000|10000|
| Alex|      Sales| 30000|10000|
| Evan|      Sales| 32000|10000|
|Helen|  Marketing| 29000|29000|
| Jane|  Marketing| 29000|29000|
| Jeff|  Marketing| 35000|29000|
| Fred|Engineering| 21000|21000|
|  Tom|Engineering| 23000|21000|
|Chloe|Engineering| 23000|21000|
| Paul|Engineering| 29000|21000|
+-----+-----------+------+-----+
SELECT name, salary,
LAG(salary) OVER (PARTITION BY dept ORDER BY salary) AS lag,
LEAD(salary, 1, 0) OVER (PARTITION BY dept ORDER BY salary) AS lead
FROM employees;
+-----+-----------+------+-----+-----+
| name|       dept|salary|  lag| lead|
+-----+-----------+------+-----+-----+
| Lisa|      Sales| 10000|NULL |30000|
| Alex|      Sales| 30000|10000|32000|
| Evan|      Sales| 32000|30000|    0|
| Fred|Engineering| 21000| NULL|23000|
|Chloe|Engineering| 23000|21000|23000|
|  Tom|Engineering| 23000|23000|29000|
```

```
| Paul|Engineering| 29000|23000|    0|
|Helen|  Marketing| 29000| NULL|29000|
| Jane|  Marketing| 29000|29000|35000|
| Jeff|  Marketing| 35000|29000|    0|
+-----+-----------+------+-----+-----+
SELECT id, v,
LEAD(v, 0) IGNORE NULLS OVER w lead,
LAG(v, 0) IGNORE NULLS OVER w lag,
NTH_VALUE(v, 2) IGNORE NULLS OVER w nth_value,
FIRST_VALUE(v) IGNORE NULLS OVER w first_value,
LAST_VALUE(v) IGNORE NULLS OVER w last_value
FROM test_ignore_null
WINDOW w AS (ORDER BY id)
ORDER BY id;
+--+----+----+----+---------+-----------+----------+
|id|   v|lead| lag|nth_value|first_value|last_value|
+--+----+----+----+---------+-----------+----------+
| 0|NULL|NULL|NULL|     NULL|       NULL|      NULL|
| 1|   x|   x|   x|     NULL|          x|         x|
| 2|NULL|NULL|NULL|     NULL|          x|         x|
| 3|NULL|NULL|NULL|     NULL|          x|         x|
| 4|   y|   y|   y|        y|          x|         y|
| 5|NULL|NULL|NULL|        y|          x|         y|
| 6|   z|   z|   z|        y|          x|         z|
| 7|   v|   v|   v|        y|          x|         v|
| 8|NULL|NULL|NULL|        y|          x|         v|
+--+----+----+----+---------+-----------+----------+
```

Funções de conversão

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
bigint(expr)	Converte o valor `expr` no tipo de dados de destino `bigint`.
binary(expr)	Converte o valor `expr` no tipo de dados de destino `binary`.

Função	Description
boolean(expr)	Converte o valor `expr` no tipo de dados de destino `boolean`.
cast(expr AS type)	Converte o valor `expr` no tipo de dados de destino `type`.
date(expr)	Converte o valor `expr` no tipo de dados de destino `date`.
decimal(expr)	Converte o valor `expr` no tipo de dados de destino `decimal`.
double(expr)	Converte o valor `expr` no tipo de dados de destino `double`.
float(expr)	Converte o valor `expr` no tipo de dados de destino `float`.
int(expr)	Converte o valor `expr` no tipo de dados de destino `int`.
smallint(expr)	Converte o valor `expr` no tipo de dados de destino `smallint`.
string(expr)	Converte o valor `expr` no tipo de dados de destino `string`.
timestamp(expr)	Converte o valor `expr` no tipo de dados de destino `timestamp`.
tinyint(expr)	Converte o valor `expr` no tipo de dados de destino `tinyint`.

Exemplos

```
-- cast
SELECT cast(field as int);
+---------------+
|CAST(field AS INT)|
+---------------+
|             10|
+---------------+
```

Funções de predicado

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
! expr	Lógico que não.
expr1 < expr2	Retornará verdadeiro se `expr1` for menor que `expr2`.
expr1 <= expr2	Retornará verdadeiro se `expr1` for menor que ou igual a `expr2`.
expr1 <=> expr2	Retorna o mesmo resultado que o operador EQUAL (=) para operandos não nulos, mas retornará verdadeiro, se ambos forem nulos, e falso se um deles for nulo.
expr1 = expr2	Retornará verdadeiro se `expr1` for igual a `expr2`, caso contrário, retornará falso.
expr1 == expr2	Retornará verdadeiro se `expr1` for igual a `expr2`, caso contrário, retornará falso.
expr1 > expr2	Retornará verdadeiro se `expr1` for maior que `expr2`.
expr1 >= expr2	Retornará verdadeiro se `expr1` for maior ou igual a `expr2`.
expr1 e expr2	AND lógico.
str ilike pattern[ESCAPE escape]	Retornará verdadeiro se str corresponder a `pattern` com `escape` sem distinção entre maiúsculas e minúsculas, nulo se algum argumento for nulo, caso contrário, falso.
expr1 in(expr2, expr3, ...)	Retornará verdadeiro se `expr` for igual a qualquer valN.
isnan(expr)	Retornará verdadeiro se `expr` for NaN, caso contrário, retornará falso.
isnotnulo(expr)	Retornará verdadeiro se `expr` não for nulo, caso contrário, retornará falso.
isnulo(expr)	Retornará verdadeiro se `expr` for nulo, caso contrário, retornará falso.

Função	Description
str like pattern[ESCAPE escape]	Retornará verdadeiro se str corresponder a `pattern` com `escape`, nulo se algum argumento for nulo, caso contrário, retornará falso.
not expr	Lógico que não.
expr1 ou expr2	OR lógico.
regexp(str, regexp)	Retornará verdadeiro se `str` corresponder a `regexp`, caso contrário, retornará falso.
regexp_like(str, regexp)	Retornará verdadeiro se `str` corresponder a `regexp`, caso contrário, retornará falso.
rlike(str, regexp)	Retornará verdadeiro se `str` corresponder a `regexp`, caso contrário, retornará falso.

Exemplos

```
-- !
SELECT ! true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT ! false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT ! NULL;
+----------+
|(NOT NULL)|
+----------+
|      NULL|
+----------+
-- <
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-07-30 04:17:52');
```

```
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                         true|
+-------------------------------------------------------------+
SELECT 1 < NULL;
+----------+
|(1 < NULL)|
+----------+
|      NULL|
+----------+
-- <=
SELECT 2 <= 2;
+--------+
|(2 <= 2)|
+--------+
|    true|
+--------+
SELECT 1.0 <= '1';
+----------+
|(1.0 <= 1)|
+----------+
|      true|
+----------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                         true|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                         true|
+-------------------------------------------------------------+
SELECT 1 <= NULL;
+-----------+
```

```
|(1 <= NULL)|
+-----------+
|       NULL|
+-----------+
-- <=>
SELECT 2 <=> 2;
+---------+
|(2 <=> 2)|
+---------+
|     true|
+---------+
SELECT 1 <=> '1';
+---------+
|(1 <=> 1)|
+---------+
|     true|
+---------+
SELECT true <=> NULL;
+---------------+
|(true <=> NULL)|
+---------------+
|          false|
+---------------+
SELECT NULL <=> NULL;
+---------------+
|(NULL <=> NULL)|
+---------------+
|           true|
+---------------+
-- =
SELECT 2 = 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 = '1';
+-------+
|(1 = 1)|
+-------+
|   true|
+-------+
SELECT true = NULL;
+-------------+
```

```
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL = NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- ==
SELECT 2 == 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 == '1';
+-------+
|(1 = 1)|
+-------+
|   true|
+-------+
SELECT true == NULL;
+-------------+
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL == NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- >
SELECT 2 > 1;
+-------+
|(2 > 1)|
+-------+
|   true|
+-------+
SELECT 2 > 1.1;
+-------+
```

```
|(2 > 1)|
+-------+
|   true|
+-------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT 1 > NULL;
+----------+
|(1 > NULL)|
+----------+
|      NULL|
+----------+
-- >=
SELECT 2 >= 1;
+--------+
|(2 >= 1)|
+--------+
|    true|
+--------+
SELECT 2.0 >= '2.1';
+------------+
|(2.0 >= 2.1)|
+------------+
|       false|
+------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        true|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-08-01 04:17:52))|
```

```
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT 1 >= NULL;
+-----------+
|(1 >= NULL)|
+-----------+
|       NULL|
+-----------+
-- and
SELECT true and true;
+---------------+
|(true AND true)|
+---------------+
|           true|
+---------------+
SELECT true and false;
+----------------+
|(true AND false)|
+----------------+
|           false|
+----------------+
SELECT true and NULL;
+---------------+
|(true AND NULL)|
+---------------+
|           NULL|
+---------------+
SELECT false and NULL;
+----------------+
|(false AND NULL)|
+----------------+
|           false|
+----------------+
-- ilike
SELECT ilike('Wagon', '_Agon');
+-------------------+
|ilike(Wagon, _Agon)|
+-------------------+
|               true|
+-------------------+
SELECT '%SystemDrive%\Users\John' ilike '\%SystemDrive\%\\users%';
+-----------------------------------------------------------+
|ilike(%SystemDrive%\Users\John, \%SystemDrive\%\\users%)|
```

```
+-------------------------------------------------------+
|                                                   true|
+-------------------------------------------------------+
SELECT '%SystemDrive%\\USERS\\John' ilike '\%SystemDrive\%\\\\Users%';
+-------------------------------------------------------+
|ilike(%SystemDrive%\USERS\John, \%SystemDrive\%\\Users%)|
+-------------------------------------------------------+
|                                                   true|
+-------------------------------------------------------+
SELECT '%SystemDrive%/Users/John' ilike '/%SYSTEMDrive/%//Users%' ESCAPE '/';
+-------------------------------------------------------+
|ilike(%SystemDrive%/Users/John, /%SYSTEMDrive/%//Users%)|
+-------------------------------------------------------+
|                                                   true|
+-------------------------------------------------------+
-- in
SELECT 1 in(1, 2, 3);
+----------------+
|(1 IN (1, 2, 3))|
+----------------+
|            true|
+----------------+
SELECT 1 in(2, 3, 4);
+----------------+
|(1 IN (2, 3, 4))|
+----------------+
|           false|
+----------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 1), named_struct('a',
 1, 'b', 3));
+-----------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 1), named_struct(a, 1, b, 3)))|
+-----------------------------------------------------------------------------------+
|                                                                              false|
+-----------------------------------------------------------------------------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 2), named_struct('a',
 1, 'b', 3));
+-----------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 2), named_struct(a, 1, b, 3)))|
+-----------------------------------------------------------------------------------+
|                                                                               true|
+-----------------------------------------------------------------------------------+
-- isnan
SELECT isnan(cast('NaN' as double));
```

```
+--------------------------+
|isnan(CAST(NaN AS DOUBLE))|
+--------------------------+
|                      true|
+--------------------------+
-- isnotnull
SELECT isnotnull(1);
+---------------+
|(1 IS NOT NULL)|
+---------------+
|           true|
+---------------+
-- isnull
SELECT isnull(1);
+-----------+
|(1 IS NULL)|
+-----------+
|      false|
+-----------+
-- like
SELECT like('Wagon', '_Agon');
+----------------+
|Wagon LIKE _Agon|
+----------------+
|            true|
+----------------+
-- not
SELECT not true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT not false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT not NULL;
+----------+
|(NOT NULL)|
+----------+
|      NULL|
```

```
+----------+
-- or
SELECT true or false;
+--------------+
|(true OR false)|
+--------------+
|           true|
+--------------+
SELECT false or false;
+----------------+
|(false OR false)|
+----------------+
|           false|
+----------------+
SELECT true or NULL;
+-------------+
|(true OR NULL)|
+-------------+
|          true|
+-------------+
SELECT false or NULL;
+---------------+
|(false OR NULL)|
+---------------+
|           NULL|
+---------------+
```

Funções de mapa

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
element_at(array, index)	Retorna elemento de matriz no índice (baseado em 1) fornecido.

Função	Description
element_at(map, key)	Retorna o valor da chave fornecida. A função retornará NULL se a chave não estiver contida no mapa.
map(key0, value0, key1, value1, ...)	Cria um mapa com os key/value pares fornecidos.
map_concat(map, ...)	Retorna a união de todos os mapas fornecidos
map_contains_key(map, key)	Retornará verdadeiro se o mapa contiver a chave.
map_entries(map)	Retorna uma matriz não ordenada de todas as entradas no mapa fornecido.
map_from_arrays(keys, values)	Cria um mapa com um par das key/value matrizes fornecidas. Todos os elementos nas chaves não podem ser nulos
map_from_entries () arrayOfEntries	Retorna um mapa criado a partir de uma matriz de entradas fornecida.
map_keys(map)	Retorna uma matriz não ordenada contendo as chaves do mapa.
map_values(map)	Retorna uma matriz não ordenada contendo os valores do mapa.
str_to_map (texto [, pairDelim [,]]) keyValueD elim	Cria um mapa depois de dividir o texto em pares de chave/valor usando delimitadores. Os delimitadores padrão são ',' para `pairDeli m` e ':' para ``. keyValueDelim Tanto `PairDeli m` quanto `keyValueDelim` são tratados como expressões regulares.

Função	Description
try_element_at(array, index)	Retorna elemento de matriz no índice (baseado em 1) fornecido. Se o Índice for 0, o sistema gerará um erro. Se o índice for < 0, acessará os elementos do último para o primeiro. A função sempre retornará NULL se o índice exceder o comprimento da matriz.
try_element_at(map, key)	Retorna o valor da chave fornecida. A função sempre retornará NULL se a chave não estiver contida no mapa.

Exemplos

```
-- element_at
SELECT element_at(array(1, 2, 3), 2);
+-----------------------------+
|element_at(array(1, 2, 3), 2)|
+-----------------------------+
|                            2|
+-----------------------------+
SELECT element_at(map(1, 'a', 2, 'b'), 2);
+-----------------------------+
|element_at(map(1, a, 2, b), 2)|
+-----------------------------+
|                             b|
+-----------------------------+
-- map
SELECT map(1.0, '2', 3.0, '4');
+--------------------+
| map(1.0, 2, 3.0, 4)|
+--------------------+
|{1.0 -> 2, 3.0 -> 4}|
+--------------------+
-- map_concat
SELECT map_concat(map(1, 'a', 2, 'b'), map(3, 'c'));
+--------------------------------------+
|map_concat(map(1, a, 2, b), map(3, c))|
+--------------------------------------+
```

```
|                  {1 -> a, 2 -> b, ...|
+--------------------------------------+
-- map_contains_key
SELECT map_contains_key(map(1, 'a', 2, 'b'), 1);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 1)|
+-----------------------------------+
|                               true|
+-----------------------------------+
SELECT map_contains_key(map(1, 'a', 2, 'b'), 3);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 3)|
+-----------------------------------+
|                              false|
+-----------------------------------+
-- map_entries
SELECT map_entries(map(1, 'a', 2, 'b'));
+---------------------------+
|map_entries(map(1, a, 2, b))|
+---------------------------+
|            [{1, a}, {2, b}]|
+---------------------------+
-- map_from_arrays
SELECT map_from_arrays(array(1.0, 3.0), array('2', '4'));
+-----------------------------------------------+
|map_from_arrays(array(1.0, 3.0), array(2, 4))|
+-----------------------------------------------+
|                         {1.0 -> 2, 3.0 -> 4}|
+-----------------------------------------------+
-- map_from_entries
SELECT map_from_entries(array(struct(1, 'a'), struct(2, 'b')));
+---------------------------------------------------+
|map_from_entries(array(struct(1, a), struct(2, b)))|
+---------------------------------------------------+
|                                 {1 -> a, 2 -> b}|
+---------------------------------------------------+
-- map_keys
SELECT map_keys(map(1, 'a', 2, 'b'));
+-------------------------+
|map_keys(map(1, a, 2, b))|
+-------------------------+
|                   [1, 2]|
+-------------------------+
-- map_values
```

```
SELECT map_values(map(1, 'a', 2, 'b'));
+--------------------------+
|map_values(map(1, a, 2, b))|
+--------------------------+
|                    [a, b]|
+--------------------------+
-- str_to_map
SELECT str_to_map('a:1,b:2,c:3', ',', ':');
+----------------------------+
|str_to_map(a:1,b:2,c:3, ,, :)|
+----------------------------+
|         {a -> 1, b -> 2, ...|
+----------------------------+
SELECT str_to_map('a');
+-------------------+
|str_to_map(a, ,, :)|
+-------------------+
|        {a -> NULL}|
+-------------------+
-- try_element_at
SELECT try_element_at(array(1, 2, 3), 2);
+---------------------------------+
|try_element_at(array(1, 2, 3), 2)|
+---------------------------------+
|                                2|
+---------------------------------+
SELECT try_element_at(map(1, 'a', 2, 'b'), 2);
+---------------------------------+
|try_element_at(map(1, a, 2, b), 2)|
+---------------------------------+
|                                 b|
+---------------------------------+
```

Funções matemáticas

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
expr1 % expr2	Retorna o resto após `expr1`/`expr2`.
expr1 * expr2	Retorna `expr1`*`expr2`.
expr1 + expr2	Retorna `expr1`+`expr2`.
expr1 - expr2	Retorna `expr1`-`expr2`.
expr1 / expr2	Retorna expr1 / expr2. Sempre realiza divisão de ponto flutuante.
abs(expr)	Retorna o valor absoluto do valor numérico ou do valor do intervalo.
acos(expr)	Retorna o cosseno inverso (também conhecido como arco cosseno) de `expr`, como se computado por `java.lang.Math.acos`.
acosh(expr)	Retorna o cosseno hiperbólico inverso de `expr`.
asin(expr)	Retorna o seno inverso (também conhecido como arco seno) de `expr`, como se computado por `java.lang.Math.asin`.
asinh(expr)	Retorna o seno hiperbólico inverso de `expr`.
atan(expr)	Retorna a tangente inversa (também conhecida como arco tangente) de `expr`, como se computada por `java.lang.Math.atan`
atan2(exprY, exprX)	Retorna o ângulo em radianos entre o eixo x positivo de um plano e o ponto dado pelas coordenadas (`exprX`, `exprY`), como se computado por `java.lang.Math.atan2`.

Função	Description
atanh(expr)	Retorna a tangente hiperbólica inversa de `expr`.
bin(expr)	Retorna a representação em string do valor longo `expr` representado em binário.
bround(expr, d)	Retorna `expr` arredondado para `d` casas decimais usando o modo de arredondamento HALF_EVEN.
cbrt(expr)	Retorna a raiz cúbica de `expr`.
ceil(expr[, scale])	Retorna o menor número após o arredonda mento que não seja menor que `expr`. Um parâmetro opcional `scale` pode ser especific ado para controlar o comportamento do arredondamento.
ceiling(expr[, scale])	Retorna o menor número após o arredonda mento que não seja menor que `expr`. Um parâmetro opcional `scale` pode ser especific ado para controlar o comportamento do arredondamento.
conv(num, from_base, to_base)	Converte `num` de `from_base` para `to_base`.
cos(expr)	Retorna o cosseno de `expr`, como se computado por `java.lang.Math.cos`.
cosh(expr)	Retorna o cosseno hiperbólico de `expr`, como se computado por `java.lang.math.cosh`.
cot(expr)	Retorna a cotangente de `expr`, como se computada por `1/java.lang.Math.tan`.
csc(expr)	Retorna a cossecante de `expr`,como se computada por `1/java.lang.Math.sin`.

Função	Description
degrees(expr)	Converte radianos em graus.
expr1 div expr2	Divide `expr1` por `expr2`. Retornará NULL se um operando for NULL ou `expr2` se for 0. O resultado é convertido em longo.
e()	Retorna o número de Euler, e.
exp(expr)	Retorna e à potência de `expr`.
expm1(expr): returna exp(`expr`)	1
factorial(expr)	Retorna o fatorial de `expr`. `expr` é [0..20]. Caso contrário, nulo.
floor(expr[, scale])	Retorna o maior número após o arredonda mento para baixo que não seja maior que `expr`. Um parâmetro opcional `scale` pode ser especificado para controlar o comportam ento do arredondamento.
greatest(expr, ...)	Retorna o maior valor de todos os parâmetros, ignorando valores nulos.
hex(expr)	Converte `expr` em hexadecimal.
hypot(expr1, expr2)	Retorna sqrt(`expr1`**2 + `expr2`**2).
least(expr, ...)	Retorna o menor valor de todos os parâmetro s, ignorando valores nulos.
ln(expr)	Retorna o logaritmo natural (base e) de `expr`.
log(base, expr)	Retorna o logaritmo de `expr` com `base`.
log10(expr)	Retorna o logaritmo de `expr` com base 10.
log1p(expr)	Retorna log(1 + `expr`).

Função	Description
log2(expr)	Retorna o logaritmo de `expr` com base 2.
expr1 mod expr2	Retorna o resto após `expr1`/`expr2`.
negative(expr)	Retorna o valor negado de `expr`.
pi()	Retorna pi.
pmod(expr1, expr2)	Retorna o valor positivo de `expr1` mod `expr2`.
positive(expr)	Retorna o valor de `expr`.
pow(expr1, expr2)	Eleva `expr1` à potência de `expr2`.
power(expr1, expr2)	Eleva `expr1` à potência de `expr2`.
radians(expr)	Converte graus em radianos.
rand([seed])	Retorna um valor aleatório com valores independentes e identicamente distribuídos (i.i.d.) uniformemente distribuídos em [0, 1).
randn([seed])	Retorna um valor aleatório com valores independentes e distribuídos de forma idêntica (i.i.d.) extraídos da distribuição normal padrão.
random([seed])	Retorna um valor aleatório com valores independentes e identicamente distribuídos (i.i.d.) uniformemente distribuídos em [0, 1).
rint(expr)	Retorna o valor duplo mais próximo em valor ao argumento e igual a um inteiro matemático.
round(expr, d)	Retorna `expr` arredondado para `d` casas decimais usando o modo de arredondamento HALF_UP.

Função	Description
sec(expr)	Retorna a secante de `expr`, como se computada por `1/java.lang.Math.cos`.
shiftleft(base, expr)	Deslocamento à esquerda em nível de bits.
sign(expr)	Retorna -1,0, 0,0 ou 1,0 quando `expr` é negativa, 0 ou positiva.
signum(expr)	Retorna -1,0, 0,0 ou 1,0 quando `expr` é negativa, 0 ou positiva.
sin(expr)	Retorna o seno de `expr`, como se computado por `java.lang.Math.sin`.
sinh(expr)	Retorna o seno hiperbólico de `expr`, como se computado por `java.lang.Math.sinh`.
sqrt(expr)	Retorna a raiz quadrada de `expr`.
tan(expr)	Retorna a tangente de `expr`, como se computada por `java.lang.Math.tan`.
tanh(expr)	Retorna a tangente hiperbólica de `expr`, como se computada por `java.lang.Math.tanh`.
try_add(expr1, expr2)	Retorna a soma de `expr1`e `expr2`, e o resultado é nulo em caso de estouro. Os tipos de entrada aceitáveis são os mesmos com o operador `+`.
try_divide(dividend, divisor)	Retorna `dividend`/`divisor`. Sempre realiza divisão de ponto flutuante. Seu resultado será sempre nulo se `expr2` for 0. `dividendo` deve ser numérico ou um intervalo. `divisor` deve ser numérico.

Função	Description
try_multiply(expr1, expr2)	Retorna `expr1`*`expr2` e o resultado é nulo em caso de estouro. Os tipos de entrada aceitáveis são os mesmos com o operador `*`.
try_subtract(expr1, expr2)	Retorna `expr1`-`expr2` e o resultado é nulo em caso de estouro. Os tipos de entrada aceitáveis são os mesmos com o operador `-`.
unhex(expr)	Converte `expr` hexadecimal em binário.
width_bucket(value, min_value, max_value, num_bucket)	Retorna o número do bucketao qual `value` seria atribuído em um histograma de equilargu ra com `num_bucket` buckets, no intervalo de `min_value` a `max_value`.”

Exemplos

```
-- %
SELECT 2 % 1.8;
+---------+
|(2 % 1.8)|
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
+-----------+
-- *
SELECT 2 * 3;
+-------+
|(2 * 3)|
+-------+
|      6|
+-------+
-- +
```

```
SELECT 1 + 2;
+-------+
|(1 + 2)|
+-------+
|      3|
+-------+
-- -
SELECT 2 - 1;
+-------+
|(2 - 1)|
+-------+
|      1|
+-------+
-- /
SELECT 3 / 2;
+-------+
|(3 / 2)|
+-------+
|    1.5|
+-------+
SELECT 2L / 2L;
+-------+
|(2 / 2)|
+-------+
|    1.0|
+-------+
-- abs
SELECT abs(-1);
+-------+
|abs(-1)|
+-------+
|      1|
+-------+
SELECT abs(INTERVAL -'1-1' YEAR TO MONTH);
+----------------------------------+
|abs(INTERVAL '-1-1' YEAR TO MONTH)|
+----------------------------------+
|              INTERVAL '1-1' YE...|
+----------------------------------+
-- acos
SELECT acos(1);
+-------+
|ACOS(1)|
+-------+
```

```
|    0.0|
+-------+
SELECT acos(2);
+-------+
|ACOS(2)|
+-------+
|    NaN|
+-------+
-- acosh
SELECT acosh(1);
+--------+
|ACOSH(1)|
+--------+
|     0.0|
+--------+
SELECT acosh(0);
+--------+
|ACOSH(0)|
+--------+
|     NaN|
+--------+
-- asin
SELECT asin(0);
+-------+
|ASIN(0)|
+-------+
|    0.0|
+-------+
SELECT asin(2);
+-------+
|ASIN(2)|
+-------+
|    NaN|
+-------+
-- asinh
SELECT asinh(0);
+--------+
|ASINH(0)|
+--------+
|     0.0|
+--------+
-- atan
SELECT atan(0);
+-------+
```

```
|ATAN(0)|
+-------+
|    0.0|
+-------+
-- atan2
SELECT atan2(0, 0);
+-----------+
|ATAN2(0, 0)|
+-----------+
|        0.0|
+-----------+
-- atanh
SELECT atanh(0);
+--------+
|ATANH(0)|
+--------+
|     0.0|
+--------+
SELECT atanh(2);
+--------+
|ATANH(2)|
+--------+
|     NaN|
+--------+
-- bin
SELECT bin(13);
+-------+
|bin(13)|
+-------+
|   1101|
+-------+
SELECT bin(-13);
+--------------------+
|            bin(-13)|
+--------------------+
|1111111111111111...|
+--------------------+
SELECT bin(13.3);
+---------+
|bin(13.3)|
+---------+
|     1101|
+---------+
-- bround
```

```
SELECT bround(2.5, 0);
+--------------+
|bround(2.5, 0)|
+--------------+
|             2|
+--------------+
SELECT bround(25, -1);
+--------------+
|bround(25, -1)|
+--------------+
|            20|
+--------------+
-- cbrt
SELECT cbrt(27.0);
+----------+
|CBRT(27.0)|
+----------+
|       3.0|
+----------+
-- ceil
SELECT ceil(-0.1);
+----------+
|CEIL(-0.1)|
+----------+
|         0|
+----------+
SELECT ceil(5);
+-------+
|CEIL(5)|
+-------+
|      5|
+-------+
SELECT ceil(3.1411, 3);
+---------------+
|ceil(3.1411, 3)|
+---------------+
|          3.142|
+---------------+
SELECT ceil(3.1411, -3);
+----------------+
|ceil(3.1411, -3)|
+----------------+
|            1000|
+----------------+
```

```
-- ceiling
SELECT ceiling(-0.1);
+-------------+
|ceiling(-0.1)|
+-------------+
|            0|
+-------------+
SELECT ceiling(5);
+----------+
|ceiling(5)|
+----------+
|         5|
+----------+
SELECT ceiling(3.1411, 3);
+------------------+
|ceiling(3.1411, 3)|
+------------------+
|             3.142|
+------------------+
SELECT ceiling(3.1411, -3);
+-------------------+
|ceiling(3.1411, -3)|
+-------------------+
|               1000|
+-------------------+
-- conv
SELECT conv('100', 2, 10);
+----------------+
|conv(100, 2, 10)|
+----------------+
|               4|
+----------------+
SELECT conv(-10, 16, -10);
+------------------+
|conv(-10, 16, -10)|
+------------------+
|               -16|
+------------------+
-- cos
SELECT cos(0);
+------+
|COS(0)|
+------+
|   1.0|
```

```
+------+
-- cosh
SELECT cosh(0);
+-------+
|COSH(0)|
+-------+
|    1.0|
+-------+
-- cot
SELECT cot(1);
+------------------+
|            COT(1)|
+------------------+
|0.6420926159343306|
+------------------+
-- csc
SELECT csc(1);
+------------------+
|            CSC(1)|
+------------------+
|1.1883951057781212|
+------------------+
-- degrees
SELECT degrees(3.141592653589793);
+--------------------------+
|DEGREES(3.141592653589793)|
+--------------------------+
|                     180.0|
+--------------------------+
-- div
SELECT 3 div 2;
+---------+
|(3 div 2)|
+---------+
|        1|
+---------+
SELECT INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH;
+-----------------------------------------------------------+
|(INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH)|
+-----------------------------------------------------------+
|                                                        -13|
+-----------------------------------------------------------+
-- e
SELECT e();
```

```
+-----------------+
|             E()|
+-----------------+
|2.718281828459045|
+-----------------+
-- exp
SELECT exp(0);
+------+
|EXP(0)|
+------+
|   1.0|
+------+
-- expm1
SELECT expm1(0);
+--------+
|EXPM1(0)|
+--------+
|     0.0|
+--------+
-- factorial
SELECT factorial(5);
+------------+
|factorial(5)|
+------------+
|         120|
+------------+
-- floor
SELECT floor(-0.1);
+-----------+
|FLOOR(-0.1)|
+-----------+
|         -1|
+-----------+
SELECT floor(5);
+--------+
|FLOOR(5)|
+--------+
|       5|
+--------+
SELECT floor(3.1411, 3);
+----------------+
|floor(3.1411, 3)|
+----------------+
|           3.141|
```

```
+----------------+
SELECT floor(3.1411, -3);
+-----------------+
|floor(3.1411, -3)|
+-----------------+
|                0|
+-----------------+
-- greatest
SELECT greatest(10, 9, 2, 4, 3);
+------------------------+
|greatest(10, 9, 2, 4, 3)|
+------------------------+
|                      10|
+------------------------+
-- hex
SELECT hex(17);
+-------+
|hex(17)|
+-------+
|     11|
+-------+
SELECT hex('SQL');
+------------------+
|    hex(SQL)|
+------------------+
|53514C|
+------------------+
-- hypot
SELECT hypot(3, 4);
+-----------+
|HYPOT(3, 4)|
+-----------+
|        5.0|
+-----------+
-- least
SELECT least(10, 9, 2, 4, 3);
+---------------------+
|least(10, 9, 2, 4, 3)|
+---------------------+
|                    2|
+---------------------+
-- ln
SELECT ln(1);
+-----+
```

```
|ln(1)|
+-----+
|  0.0|
+-----+
-- log
SELECT log(10, 100);
+------------+
|LOG(10, 100)|
+------------+
|         2.0|
+------------+
-- log10
SELECT log10(10);
+---------+
|LOG10(10)|
+---------+
|      1.0|
+---------+
-- log1p
SELECT log1p(0);
+--------+
|LOG1P(0)|
+--------+
|     0.0|
+--------+
-- log2
SELECT log2(2);
+-------+
|LOG2(2)|
+-------+
|    1.0|
+-------+
-- mod
SELECT 2 % 1.8;
+---------+
|(2 % 1.8)|
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
```

```
+-----------+
-- negative
SELECT negative(1);
+-----------+
|negative(1)|
+-----------+
|         -1|
+-----------+
-- pi
SELECT pi();
+-----------------+
|             PI()|
+-----------------+
|3.141592653589793|
+-----------------+
-- pmod
SELECT pmod(10, 3);
+-----------+
|pmod(10, 3)|
+-----------+
|          1|
+-----------+
SELECT pmod(-10, 3);
+------------+
|pmod(-10, 3)|
+------------+
|           2|
+------------+
-- positive
SELECT positive(1);
+-----+
|(+ 1)|
+-----+
|    1|
+-----+
-- pow
SELECT pow(2, 3);
+---------+
|pow(2, 3)|
+---------+
|      8.0|
+---------+
-- power
SELECT power(2, 3);
```

```
+-----------+
|POWER(2, 3)|
+-----------+
|        8.0|
+-----------+
-- radians
SELECT radians(180);
+-----------------+
|     RADIANS(180)|
+-----------------+
|3.141592653589793|
+-----------------+
-- rand
SELECT rand();
+------------------+
|            rand()|
+------------------+
|0.7211420708112387|
+------------------+
SELECT rand(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT rand(null);
+------------------+
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- randn
SELECT randn();
+-------------------+
|            randn()|
+-------------------+
|-0.8175603217732732|
+-------------------+
SELECT randn(0);
+------------------+
|          randn(0)|
+------------------+
|1.6034991609278433|
+------------------+
```

```
SELECT randn(null);
+------------------+
|       randn(NULL)|
+------------------+
|1.6034991609278433|
+------------------+
-- random
SELECT random();
+-----------------+
|           rand()|
+-----------------+
|0.394205008255365|
+-----------------+
SELECT random(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT random(null);
+------------------+
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- rint
SELECT rint(12.3456);
+-------------+
|rint(12.3456)|
+-------------+
|         12.0|
+-------------+
-- round
SELECT round(2.5, 0);
+-------------+
|round(2.5, 0)|
+-------------+
|            3|
+-------------+
-- sec
SELECT sec(0);
+------+
|SEC(0)|
+------+
```

```
|   1.0|
+------+
-- shiftleft
SELECT shiftleft(2, 1);
+---------------+
|shiftleft(2, 1)|
+---------------+
|              4|
+---------------+
-- sign
SELECT sign(40);
+--------+
|sign(40)|
+--------+
|     1.0|
+--------+
SELECT sign(INTERVAL -'100' YEAR);
+--------------------------+
|sign(INTERVAL '-100' YEAR)|
+--------------------------+
|                      -1.0|
+--------------------------+
-- signum
SELECT signum(40);
+----------+
|SIGNUM(40)|
+----------+
|       1.0|
+----------+
SELECT signum(INTERVAL -'100' YEAR);
+----------------------------+
|SIGNUM(INTERVAL '-100' YEAR)|
+----------------------------+
|                        -1.0|
+----------------------------+
-- sin
SELECT sin(0);
+------+
|SIN(0)|
+------+
|   0.0|
+------+
-- sinh
SELECT sinh(0);
```

```
+-------+
|SINH(0)|
+-------+
|    0.0|
+-------+
-- sqrt
SELECT sqrt(4);
+-------+
|SQRT(4)|
+-------+
|    2.0|
+-------+
-- tan
SELECT tan(0);
+------+
|TAN(0)|
+------+
|   0.0|
+------+
-- tanh
SELECT tanh(0);
+-------+
|TANH(0)|
+-------+
|    0.0|
+-------+
-- try_add
SELECT try_add(1, 2);
+-------------+
|try_add(1, 2)|
+-------------+
|            3|
+-------------+
SELECT try_add(2147483647, 1);
+----------------------+
|try_add(2147483647, 1)|
+----------------------+
|                  NULL|
+----------------------+
SELECT try_add(date'2021-01-01', 1);
+-----------------------------+
|try_add(DATE '2021-01-01', 1)|
+-----------------------------+
|                   2021-01-02|
```

```
+----------------------------+
SELECT try_add(date'2021-01-01', interval 1 year);
+--------------------------------------------+
|try_add(DATE '2021-01-01', INTERVAL '1' YEAR)|
+--------------------------------------------+
|                                  2022-01-01|
+--------------------------------------------+
SELECT try_add(timestamp'2021-01-01 00:00:00', interval 1 day);
+------------------------------------------------------------+
|try_add(TIMESTAMP '2021-01-01 00:00:00', INTERVAL '1' DAY)|
+------------------------------------------------------------+
|                                     2021-01-02 00:00:00|
+------------------------------------------------------------+
SELECT try_add(interval 1 year, interval 2 year);
+-------------------------------------------+
|try_add(INTERVAL '1' YEAR, INTERVAL '2' YEAR)|
+-------------------------------------------+
|                           INTERVAL '3' YEAR|
+-------------------------------------------+
-- try_divide
SELECT try_divide(3, 2);
+----------------+
|try_divide(3, 2)|
+----------------+
|             1.5|
+----------------+
SELECT try_divide(2L, 2L);
+----------------+
|try_divide(2, 2)|
+----------------+
|             1.0|
+----------------+
SELECT try_divide(1, 0);
+----------------+
|try_divide(1, 0)|
+----------------+
|            NULL|
+----------------+
SELECT try_divide(interval 2 month, 2);
+---------------------------------+
|try_divide(INTERVAL '2' MONTH, 2)|
+---------------------------------+
|             INTERVAL '0-1' YE...|
+---------------------------------+
```

```
SELECT try_divide(interval 2 month, 0);
+--------------------------------+
|try_divide(INTERVAL '2' MONTH, 0)|
+--------------------------------+
|                            NULL|
+--------------------------------+
-- try_multiply
SELECT try_multiply(2, 3);
+------------------+
|try_multiply(2, 3)|
+------------------+
|                 6|
+------------------+
SELECT try_multiply(-2147483648, 10);
+----------------------------+
|try_multiply(-2147483648, 10)|
+----------------------------+
|                        NULL|
+----------------------------+
SELECT try_multiply(interval 2 year, 3);
+---------------------------------+
|try_multiply(INTERVAL '2' YEAR, 3)|
+---------------------------------+
|              INTERVAL '6-0' YE...|
+---------------------------------+
-- try_subtract
SELECT try_subtract(2, 1);
+------------------+
|try_subtract(2, 1)|
+------------------+
|                 1|
+------------------+
SELECT try_subtract(-2147483648, 1);
+----------------------------+
|try_subtract(-2147483648, 1)|
+----------------------------+
|                        NULL|
+----------------------------+
SELECT try_subtract(date'2021-01-02', 1);
+---------------------------------+
|try_subtract(DATE '2021-01-02', 1)|
+---------------------------------+
|                       2021-01-01|
+---------------------------------+
```

```
SELECT try_subtract(date'2021-01-01', interval 1 year);
+-----------------------------------------------+
|try_subtract(DATE '2021-01-01', INTERVAL '1' YEAR)|
+-----------------------------------------------+
|                                     2020-01-01|
+-----------------------------------------------+
SELECT try_subtract(timestamp'2021-01-02 00:00:00', interval 1 day);
+-------------------------------------------------------------+
|try_subtract(TIMESTAMP '2021-01-02 00:00:00', INTERVAL '1' DAY)|
+-------------------------------------------------------------+
|                                          2021-01-01 00:00:00|
+-------------------------------------------------------------+
SELECT try_subtract(interval 2 year, interval 1 year);
+-----------------------------------------------+
|try_subtract(INTERVAL '2' YEAR, INTERVAL '1' YEAR)|
+-----------------------------------------------+
|                               INTERVAL '1' YEAR|
+-----------------------------------------------+
-- unhex
SELECT decode(unhex('53514C'), 'UTF-8');
+---------------------------------------+
|decode(unhex(53514C), UTF-8)|
+---------------------------------------+
|                                    SQL|
+---------------------------------------+
-- width_bucket
SELECT width_bucket(5.3, 0.2, 10.6, 5);
+-------------------------------+
|width_bucket(5.3, 0.2, 10.6, 5)|
+-------------------------------+
|                              3|
+-------------------------------+
SELECT width_bucket(-2.1, 1.3, 3.4, 3);
+-------------------------------+
|width_bucket(-2.1, 1.3, 3.4, 3)|
+-------------------------------+
|                              0|
+-------------------------------+
SELECT width_bucket(8.1, 0.0, 5.7, 4);
+-------------------------------+
|width_bucket(8.1, 0.0, 5.7, 4)|
+-------------------------------+
|                             5|
+-------------------------------+
```

```
SELECT width_bucket(-0.9, 5.2, 0.5, 2);
+------------------------------+
|width_bucket(-0.9, 5.2, 0.5, 2)|
+------------------------------+
|                             3|
+------------------------------+
SELECT width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            1|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            2|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+-------------------------------------------------------------------------+
|width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
+-------------------------------------------------------------------------+
|                                                                        1|
+-------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+-------------------------------------------------------------------------+
|width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
+-------------------------------------------------------------------------+
|                                                                        2|
+-------------------------------------------------------------------------+
```

Funções geradoras

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essas funções SQL, consultethe section called “Comandos SQL compatíveis”.

Função	Description
explode(expr)	Separa os elementos da matriz `expr` em várias linhas ou os elementos do mapa `expr em várias linhas e colunas. A menos que especificado de outra forma, usa o nome de coluna padrão `col` para elementos da matriz ou `key` e `value` para os elementos do mapa.
explode_outer(expr)	Separa os elementos da matriz `expr` em várias linhas ou os elementos do mapa `expr em várias linhas e colunas. A menos que especificado de outra forma, usa o nome de coluna padrão `col` para elementos da matriz ou `key` e `value` para os elementos do mapa.
inline(expr)	Explode uma matriz de estruturas em uma tabela. Usa os nomes das colunas col1, col2, etc. por padrão, a menos que especificado de outra forma.
inline_outer(expr)	Explode uma matriz de estruturas em uma tabela. Usa os nomes das colunas col1, col2, etc. por padrão, a menos que especificado de outra forma.
posexplode(expr)	Separa os elementos da matriz `expr` em várias linhas com posições ou os elementos do mapa `expr` em várias linhas e colunas com posições. A menos que especificado de outra forma, usa o nome de coluna padrão `pos` para posição, `col` para os elementos da matriz ou `key` e `value` para os elementos do mapa.
posexplode_outer(expr)	Separa os elementos da matriz `expr` em várias linhas com posições ou os elementos do mapa `expr` em várias linhas e colunas com posições. A menos que especificado de outra forma, usa o nome de coluna padrão `pos` para posição, `col` para os elementos da matriz ou `key` e `value` para os elementos do mapa.
stack(n, expr1, ..., exprk)	Separa `expr1`, ..., `exprk` em `n` linhas. Usa os nomes das colunas col0, col1 etc. por padrão, a menos que especificado de outra forma.

Exemplos

```
-- explode
```

```
SELECT explode(array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

-- explode_outer
SELECT explode_outer(array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT explode_outer(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode_outer(collection => array(10, 20));
+---+
|col|
```

```
+---+
| 10|
| 20|
+---+

-- inline
SELECT inline(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- inline_outer
SELECT inline_outer(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- posexplode
SELECT posexplode(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

SELECT * FROM posexplode(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- posexplode_outer
SELECT posexplode_outer(array(10,20));
+---+---+
|pos|col|
```

```
+---+---+
|  0| 10|
|  1| 20|
+---+---+

SELECT * FROM posexplode_outer(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- stack
SELECT stack(2, 1, 2, 3);
+----+----+
|col0|col1|
+----+----+
|   1|   2|
|   3|NULL|
+----+----+
```

Cláusula SELECT

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

OpenSearch O SQL suporta uma `SELECT` instrução usada para recuperar conjuntos de resultados de uma ou mais tabelas. A seção a seguir descreve a sintaxe de consulta em geral e as diferentes construções de uma consulta.

Sintaxe

```
select_statement
[ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select_statement, ... ]
[ ORDER BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
```

```
]
[ SORT BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
]
[ WINDOW { named_window [ , WINDOW named_window, ... ] } ]
[ LIMIT { ALL | expression } ]
```

Enquanto `select_statement` é definido como:

```
SELECT [ ALL | DISTINCT ] { [ [ named_expression ] [ , ... ] ] }
FROM { from_item [ , ... ] }
[ PIVOT clause ]
[ UNPIVOT clause ]
[ LATERAL VIEW clause ] [ ... ]
[ WHERE boolean_expression ]
[ GROUP BY expression [ , ... ] ]
[ HAVING boolean_expression ]
```

Parâmetros

- TUDO

  Seleciona todas as linhas correspondentes da relação e é habilitada por padrão.
- DISTINTO

  Seleciona todas as linhas correspondentes da relação após remover as duplicações dos resultados.
- named_expression

  Uma expressão com um nome atribuído. Em geral, denota uma expressão de coluna.

  Sintaxe: `expression [[AS] alias]`
- from_item

  Relação de tabela

  Relação de união

  Relação de pivô

Relação de despivô

Funções de valor de tabela

Tabela em linha

```
[ LATERAL ] ( Subquery )
```

- PIVÔ

  A cláusula `PIVOT` é usada para perspectiva de dados. Você pode obter os valores agregados com base no valor específico da coluna.

- UNPIVOT

  A cláusula `UNPIVOT` transforma colunas em linhas. É o inverso de `PIVOT`, exceto pela agregação de valores.

- LATERAL VIEW

  A cláusula `LATERAL VIEW` é usada em conjunto com funções geradoras, como `EXPLODE`, que gerará uma tabela virtual contendo uma ou mais linhas.

  `LATERAL VIEW` aplicará as linhas a cada linha de saída original.

- WHERE

  Filtra o resultado da cláusula `FROM` com base nos predicados fornecidos.

- AGRUPAR POR

  Especifica as expressões usadas para agrupar as linhas.

  Isso é usado em conjunto com funções de agregação (`MIN`, `MAX`, `COUNT`, `SUM`, `AVG` e assim por diante) para agrupar linhas com base nas expressões de agrupamento e agregar os valores de cada grupo.

  Quando uma cláusula `FILTER` é anexada a uma função de agregação, somente as linhas correspondentes são passadas para essa função.

- TENDO

  Especifica os predicados pelos quais as linhas produzidas por `GROUP BY` são filtradas.

  A cláusula `HAVING` é usada para filtrar linhas após o agrupamento ser realizado.

Se HAVING for especificada sem GROUP BY, indica uma GROUP BY sem expressões de agrupamento (agregado global).

- ORDER BY

  Especifica a ordem das linhas de todo o conjunto de resultados da consulta.

  As linhas de saída são ordenadas entre as partições.

  Esse parâmetro é mutuamente exclusivo com SORT BY e DISTRIBUTE BY, e não podem ser especificados juntos.

- SORT BY

  Especifica a ordem em que as linhas são ordenadas em cada partição.

  Esse parâmetro é mutuamente exclusivo com ORDER BY, não podem ser especificados juntos.

- LIMIT

  Especifica o número máximo de linhas que pode ser retornado por uma instrução ou subconsulta.

  Essa cláusula é usada principalmente em conjunto com ORDER BY para produzir um resultado determinístico.

- boolean_expression

  Especifica qualquer expressão que é avaliada como um resultado do tipo booleano.

  Duas ou mais expressões podem ser combinadas usando os operadores lógicos (AND, OR).

- expressão

  Especifica uma combinação de um ou mais valores, operadores e funções SQL que são avaliadas como um valor.

- named_window

  Especifica aliases para uma ou mais especificações da janela de origem.

  As especificações da janela de origem podem ser referenciadas nas definições da janela na consulta.

Cláusula WHERE

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `WHERE` é usada para limitar os resultados da cláusula `FROM` de uma consulta ou subconsulta com base na condição especificada.

Sintaxe

```
WHERE boolean_expression
```

Parâmetros

- boolean_expression

  Especifica qualquer expressão que é avaliada como um resultado do tipo booleano.

  Duas ou mais expressões podem ser combinadas usando os operadores lógicos (`AND`, `OR`).

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan',  50);

-- Comparison operator in `WHERE` clause.
SELECT * FROM person WHERE id > 200 ORDER BY id;
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
|400| Dan| 50|
+---+----+---+
```

```
-- Comparison and logical operators in `WHERE` clause.
SELECT * FROM person WHERE id = 200 OR id = 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- IS NULL expression in `WHERE` clause.
SELECT * FROM person WHERE id > 300 OR age IS NULL ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|400| Dan|  50|
+---+----+----+

-- Function expression in `WHERE` clause.
SELECT * FROM person WHERE length(name) > 3 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- `BETWEEN` expression in `WHERE` clause.
SELECT * FROM person WHERE id BETWEEN 200 AND 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- Scalar Subquery in `WHERE` clause.
SELECT * FROM person WHERE age > (SELECT avg(age) FROM person);
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
+---+----+---+
```

```
-- Correlated Subquery in `WHERE` clause.
SELECT id FROM person
WHERE exists (SELECT id FROM person where id = 200);
+---+----+----+
|id |name|age |
+---+----+----+
|200|Mary|null|
+---+----+----+
```

Cláusula GROUP BY

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `GROUP BY` é usada para agrupar as linhas com base em um conjunto de expressões de agrupamento especificadas e computar agregações no grupo de linhas com base em uma ou mais funções de agregação especificadas.

O sistema também faz várias agregações para o mesmo registro de entrada definido pelas cláusulas `GROUPING SETS`, `CUBE` e `ROLLUP`. As expressões de agrupamento e as agregações avançadas podem ser combinadas na cláusula `GROUP BY` e aninhadas em uma cláusula `GROUPING SETS` . Veja mais detalhes na seção `Mixed/Nested Grouping Analytics` .

Quando uma cláusula `FILTER` é anexada a uma função de agregação, somente as linhas correspondentes são passadas para essa função.

Sintaxe

```
GROUP BY group_expression [ , group_expression [ , ... ] ] [ WITH { ROLLUP | CUBE } ]
GROUP BY { group_expression | { ROLLUP | CUBE | GROUPING SETS } (grouping_set
 [ , ...]) } [ , ... ]
```

Embora as funções de agregação sejam definidas como:

```
aggregate_name ( [ DISTINCT ] expression [ , ... ] ) [ FILTER ( WHERE
 boolean_expression ) ]
```

Parâmetros

- group_expression

  Especifica os critérios com base nos quais as linhas são agrupadas. O agrupamento de linhas é realizado com base nos valores dos resultados das expressões de agrupamento.

  Uma expressão de agrupamento pode ser um nome de coluna, como `GROUP BY a`, uma posição de coluna como `GROUP BY 0` ou uma expressão como `GROUP BY a + b`.

- grouping_set

  Um conjunto de agrupamento é especificado por zero ou mais expressões separadas por vírgula entre parênteses. Quando o conjunto de agrupamento tem somente um elemento, os parênteses podem ser omitidos.

  Por exemplo, `GROUPING SETS ((a), (b))` é o mesmo que `GROUPING SETS (a, b)`.

  Sintaxe: `{ ( [ expression [ , ... ] ] ) | expression }`

- GROUPING SETS

  Agrupa as linhas para cada conjunto de agrupamento especificado depois de `GROUPING SETS`.

  Por exemplo, `GROUP BY GROUPING SETS ((warehouse), (product))` é semanticamente equivalente à união dos resultados de `GROUP BY warehouse` e `GROUP BY product`. Essa cláusula é uma abreviação de UNION ALL em que cada etapa do operador `UNION ALL` realiza a agregação de cada conjunto de agrupamento especificado na cláusula `GROUPING SETS`.

  Da mesma forma, `GROUP BY GROUPING SETS ((warehouse, product), (product), ())` é semanticamente equivalente à união dos resultados de `GROUP BY warehouse, product`, `GROUP BY product` e uma agregação global.

- ROLLUP

  Especifica vários níveis de agregações em uma única instrução. Essa cláusula é usada para computar agregações com base em vários conjuntos de agrupamentos. `ROLLUP` é uma abreviatura de `GROUPING SETS`.

  Por exemplo, `GROUP BY warehouse, product WITH ROLLUP` or `GROUP BY ROLLUP(warehouse, product)` equivale a `GROUP BY GROUPING SETS((warehouse, product), (warehouse), ())`.

`GROUP BY ROLLUP(warehouse, product, (warehouse, location))` é equivalente a `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse), ())`.

Os N elementos de uma especificação ROLLUP resultam em N+1 GROUPING SETS.

- CUBE

  A cláusula CUBE é usada para realizar agregações com base na combinação de colunas de agrupamento especificadas na cláusula GROUP BY. CUBE é uma abreviatura de GROUPING SETS.

  Por exemplo, `GROUP BY warehouse, product WITH CUBE or GROUP BY CUBE(warehouse, product)` equivale a `GROUP BY GROUPING SETS((warehouse, product), (warehouse), (product), ())`.

  `GROUP BY CUBE(warehouse, product, (warehouse, location))` é equivalente a `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse, location), (product, warehouse, location), (warehouse), (product), (warehouse, product), ())`. Os N elementos de uma especificação de `CUBE` resultam em 2^N `GROUPING SETS`.

- Analytics de agrupamento misto/aninhado

  Uma cláusula `GROUP BY` pode incluir várias group_expressions e vários `CUBE|ROLLUP|GROUPING SETS`. `GROUPING SETS` também podem ter cláusulas `CUBE|ROLLUP|GROUPING SETS` aninhadas, como, `GROUPING SETS(ROLLUP(warehouse, location), CUBE(warehouse, location))`. `GROUPING SETS(warehouse, GROUPING SETS(location, GROUPING SETS(ROLLUP(warehouse, location), CUBE(warehouse, location))))`.

  `CUBE|ROLLUP` é apenas um açúcar sintático para `GROUPING SETS`. Consulte as seções acima para saber como traduzir `CUBE|ROLLUP` para `GROUPING SETS`. Pode-se tratar `group_expression` como `GROUPING SETS` de grupo único nesse contexto.

  Para vários `GROUPING SETS` na cláusula `GROUP BY`, geramos um único `GROUPING SETS` fazendo um produto cartesiano do original `GROUPING SETS`. Para `GROUPING SETS` aninhado na cláusula `GROUPING SETS`, simplesmente pegamos seus conjuntos de agrupamentos e os removemos.

Por exemplo, `GROUP BY warehouse, GROUPING SETS((product), ()), GROUPING SETS((location, size), (location), (size), ()) and GROUP BY warehouse, ROLLUP(product), CUBE(location, size)` equivale a `GROUP BY GROUPING SETS( (warehouse, product, location, size), (warehouse, product, location), (warehouse, product, size), (warehouse, product), (warehouse, location, size), (warehouse, location), (warehouse, size), (warehouse))`.

`GROUP BY GROUPING SETS(GROUPING SETS(warehouse), GROUPING SETS((warehouse, product)))` é equivalente a `GROUP BY GROUPING SETS((warehouse), (warehouse, product))`.

- aggregate_name

  Especifica um nome de função de agregação (`MIN`, `MAX`, `COUNT`, `SUM`, `AVG` e assim por diante).

- DISTINTO

  Remove duplicações nas linhas de entrada antes que elas sejam passadas para funções de agregação.

- FILTRO

  Filtra as linhas de entrada para as quais a cláusula `boolean_expression` na cláusula `WHERE` é avaliada como verdadeiro são passadas para a função de agregação; outras linhas são descartadas.

Exemplos

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- Sum of quantity per dealership. Group by `id`.
SELECT id, sum(quantity) FROM dealer GROUP BY id ORDER BY id;
+---+-------------+
```

```
| id|sum(quantity)|
+---+-------------+
|100|           32|
|200|           33|
|300|           13|
+---+-------------+

-- Use column position in GROUP by clause.
SELECT id, sum(quantity) FROM dealer GROUP BY 1 ORDER BY 1;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           32|
|200|           33|
|300|           13|
+---+-------------+

-- Multiple aggregations.
-- 1. Sum of quantity per dealership.
-- 2. Max quantity per dealership.
SELECT id, sum(quantity) AS sum, max(quantity) AS max FROM dealer GROUP BY id ORDER BY
 id;
+---+---+---+
| id|sum|max|
+---+---+---+
|100| 32| 15|
|200| 33| 20|
|300| 13|  8|
+---+---+---+

-- Count the number of distinct dealer cities per car_model.
SELECT car_model, count(DISTINCT city) AS count FROM dealer GROUP BY car_model;
+------------+-----+
|   car_model|count|
+------------+-----+
| Honda Civic|    3|
|   Honda CRV|    2|
|Honda Accord|    3|
+------------+-----+

-- Sum of only 'Honda Civic' and 'Honda CRV' quantities per dealership.
SELECT id, sum(quantity) FILTER (
WHERE car_model IN ('Honda Civic', 'Honda CRV')
) AS `sum(quantity)` FROM dealer
```

```
GROUP BY id ORDER BY id;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           17|
|200|           23|
|300|            5|
+---+-------------+

-- Aggregations using multiple sets of grouping columns in a single statement.
-- Following performs aggregations based on four sets of grouping columns.
-- 1. city, car_model
-- 2. city
-- 3. car_model
-- 4. Empty grouping set. Returns quantities for all city and car models.
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
ORDER BY city;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `ROLLUP` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH ROLLUP
ORDER BY city, car_model;
+---------+------------+---+
```

```
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `CUBE` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH CUBE
ORDER BY city, car_model;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

--Prepare data for ignore nulls example
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
```

```
(100, 'Mary', NULL),
(200, 'John', 30),
(300, 'Mike', 80),
(400, 'Dan', 50);

--Select the first row in column age
SELECT FIRST(age) FROM person;
+--------------------+
| first(age, false)  |
+--------------------+
| NULL               |
+--------------------+

--Get the first row in column `age` ignore nulls,last row in column `id` and sum of
 column `id`.
SELECT FIRST(age IGNORE NULLS), LAST(id), SUM(id) FROM person;
+-------------------+------------------+----------+
| first(age, true)  | last(id, false)  | sum(id)  |
+-------------------+------------------+----------+
| 30                | 400              | 1000     |
+-------------------+------------------+----------+
```

Cláusula HAVING

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `HAVING` é usada para filtrar os resultados produzidos por `GROUP BY` com base na condição especificada. É frequentemente usada em conjunto com uma cláusula `GROUP BY`.

Sintaxe

```
HAVING boolean_expression
```

Parâmetros

- boolean_expression

Especifica qualquer expressão que é avaliada como um resultado do tipo booleano. Duas ou mais expressões podem ser combinadas usando os operadores lógicos (AND, OR).

Observação As expressões especificadas na cláusula HAVING apenas podem se referir a:

1. Constantes
2. Expressões que aparecem em GROUP BY
3. Funções agregadas

Exemplos

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- `HAVING` clause referring to column in `GROUP BY`.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING city = 'Fremont';
+-------+---+
|   city|sum|
+-------+---+
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum(quantity) > 15;
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function by its alias.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum > 15;
```

```
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to a different aggregate function than what is present in
-- `SELECT` list.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING max(quantity) > 15;
+------+---+
|  city|sum|
+------+---+
|Dublin| 33|
+------+---+

-- `HAVING` clause referring to constant expression.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING 1 > 0 ORDER BY city;
+--------+---+
|    city|sum|
+--------+---+
|  Dublin| 33|
| Fremont| 32|
|San Jose| 13|
+--------+---+

-- `HAVING` clause without a `GROUP BY` clause.
SELECT sum(quantity) AS sum FROM dealer HAVING sum(quantity) > 10;
+---+
|sum|
+---+
| 78|
+---+
```

Cláusula ORDER BY

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `ORDER BY` é usada para retornar as linhas de resultados classificados na ordem especificada pelo usuário. Diferentemente da cláusula SORT BY, essa cláusula garante uma ordem total na saída.

Sintaxe

```
ORDER BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Parâmetros

- ORDER BY

  Especifica uma lista de expressões separadas por vírgula junto com parâmetros opcionais `sort_direction` e `nulls_sort_order` que são usados para classificar as linhas.

- sort_direction

  Opcionalmente, especifica se as linhas devem ser classificadas em ordem crescente ou decrescente.

  Os valores válidos para a direção de classificação são `ASC` para crescente e `DESC` para decrescente.

  Se a direção da classificação não for especificada explicitamente, as linhas serão classificadas em ordem crescente por padrão.

  Sintaxe: `[ ASC | DESC ]`

- nulos_sort_order

  Opcionalmente, especifica se `NULL` os valores são retornados valores before/after não NULL.

  Se nulo_sort_order não for especificada, `NULLs` serão classificados primeiro se a ordem de classificação for `ASC` e NULLS serão classificados por último se a ordem de classificação for `DESC`.

  1. Se `NULLS FIRST` for especificado, os valores NULL serão retornados primeiro, independentemente da ordem de classificação.

  2. Se `NULLS LAST` for especificado, os valores NULL serão retornados por último, independentemente da ordem de classificação.

  Sintaxe: `[ NULLS { FIRST | LAST } ]`

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Jerry', NULL),
(500, 'Dan',  50);

-- Sort rows by age. By default rows are sorted in ascending manner with NULL FIRST.
SELECT name, age FROM person ORDER BY age;
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| John|  30|
|  Dan|  50|
| Mike|  80|
+-----+----+

-- Sort rows in ascending manner keeping null values to be last.
SELECT name, age FROM person ORDER BY age NULLS LAST;
+-----+----+
| name| age|
+-----+----+
| John|  30|
|  Dan|  50|
| Mike|  80|
| Mary|null|
|Jerry|null|
+-----+----+

-- Sort rows by age in descending manner, which defaults to NULL LAST.
SELECT name, age FROM person ORDER BY age DESC;
+-----+----+
| name| age|
+-----+----+
| Mike|  80|
|  Dan|  50|
| John|  30|
|Jerry|null|
| Mary|null|
```

```
+-----+----+

-- Sort rows in ascending manner keeping null values to be first.
SELECT name, age FROM person ORDER BY age DESC NULLS FIRST;
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| Mike|  80|
|  Dan|  50|
| John|  30|
+-----+----+

-- Sort rows based on more than one column with each column having different
-- sort direction.
SELECT * FROM person ORDER BY name ASC, age DESC;
+---+-----+----+
| id| name| age|
+---+-----+----+
|500|  Dan|  50|
|400|Jerry|null|
|100| John|  30|
|200| Mary|null|
|300| Mike|  80|
+---+-----+----+
```

Cláusula JOIN

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Uma união SQL é usada para combinar linhas de duas relações com base nos critérios de união. A seção a seguir descreve a sintaxe de união em geral e os diferentes tipos de uniões, com exemplos.

Sintaxe

```
relation INNER JOIN relation [ join_criteria ]
```

Parâmetros

- relation

  Especifica a relação a ser unida.

- join_type

  Especifica o tipo de união.

  Sintaxe: `INNER | CROSS | LEFT OUTER`

- join_criteria

  Especifica como as linhas de uma relação serão combinadas com as linhas de outra relação.

  Sintaxe: `ON boolean_expression | USING ( column_name [ , ... ] )`

- boolean_expression

  Especifica uma expressão com um retorno do tipo booleano.

Tipos de união

- União interna

  A união interna precisa ser especificada explicitamente. Ele seleciona linhas que têm valores correspondentes em ambas as relações.

  Sintaxe: `relation INNER JOIN relation [ join_criteria ]`

- União à esquerda

  Uma união à esquerda retorna todos os valores da relação à esquerda e os valores correspondentes da relação à direita, ou acrescentará NULL se não houver nenhuma correspondência. Também é chamada de união externa à esquerda.

  Sintaxe: `relation LEFT OUTER JOIN relation [ join_criteria ]`

- União cruzada

  Uma união cruzada retorna o produto cartesiano de duas relações.

  Sintaxe: `relation CROSS JOIN relation [ join_criteria ]`

Exemplos

```
-- Use employee and department tables to demonstrate different type of joins.
SELECT * FROM employee;
+---+-----+------+
| id| name|deptno|
+---+-----+------+
|105|Chloe|     5|
|103| Paul|     3|
|101| John|     1|
|102| Lisa|     2|
|104| Evan|     4|
|106|  Amy|     6|
+---+-----+------+
SELECT * FROM department;
+------+-----------+
|deptno|   deptname|
+------+-----------+
|     3|Engineering|
|     2|      Sales|
|     1|  Marketing|
+------+-----------+

-- Use employee and department tables to demonstrate inner join.
SELECT id, name, employee.deptno, deptname
FROM employee INNER JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate left join.
SELECT id, name, employee.deptno, deptname
FROM employee LEFT JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|       NULL|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
```

```
|104| Evan|     4|       NULL|
|106|  Amy|     6|       NULL|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate cross join.
SELECT id, name, employee.deptno, deptname FROM employee CROSS JOIN department;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|Engineering|
|105|Chloe|     5|  Marketing|
|105|Chloe|     5|      Sales|
|103| Paul|     3|Engineering|
|103| Paul|     3|  Marketing|
|103| Paul|     3|      Sales|
|101| John|     1|Engineering|
|101| John|     1|  Marketing|
|101| John|     1|      Sales|
|102| Lisa|     2|Engineering|
|102| Lisa|     2|  Marketing|
|102| Lisa|     2|      Sales|
|104| Evan|     4|Engineering|
|104| Evan|     4|  Marketing|
|104| Evan|     4|      Sales|
|106|  Amy|     4|Engineering|
|106|  Amy|     4|  Marketing|
|106|  Amy|     4|      Sales|
+---+-----+------+-----------|
```

Cláusula LIMIT

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `LIMIT` é usada para restringir o número de linhas retornado pela instrução `SELECT`. Em geral, essa cláusula é usada em conjunto com `ORDER BY` para garantir que os resultados sejam determinísticos.

Sintaxe

```
LIMIT { ALL | integer_expression }
```

Parâmetros

- TUDO

  Se especificada, a consulta retornará todas as linhas. Em outras palavras, nenhum limite será aplicado se essa opção for especificada.

- integer_expression

  Especifica uma expressão redutível que retorna um número inteiro.

Exemplos

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('John D', 25),
('Juan L', 18),
('Jorge S', 16);

-- Select the first two rows.
SELECT name, age FROM person ORDER BY name LIMIT 2;
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
|Jorge S| 16|
+------+---+

-- Specifying ALL option on LIMIT returns all the rows.
SELECT name, age FROM person ORDER BY name LIMIT ALL;
+--------+---+
|    name|age|
+--------+---+
|   Pat C| 18|
| Jorge S| 16|
|  Juan L| 18|
|  John D| 25|
```

```
| Nikki W| 16|
|Jane Doe| 25|
+--------+---+

-- A function expression as an input to LIMIT.
SELECT name, age FROM person ORDER BY name LIMIT length('OPENSEARCH');
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
|Jorge S| 16|
| Juan L| 18|
| John D| 25|
|Nikki W| 16|
+-------+---+
```

Cláusula CASE

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `CASE` usa uma regra para retornar um resultado específico com base na condição especificada, semelhante às instruções if/else em outras linguagens de programação.

Sintaxe

```
CASE [ expression ] { WHEN boolean_expression THEN then_expression } [ ... ]
[ ELSE else_expression ]
END
```

Parâmetros

- boolean_expression

  Especifica qualquer expressão que é avaliada como um resultado do tipo booleano.

  Duas ou mais expressões podem ser combinadas usando os operadores lógicos (`AND`, `OR`).
- then_expression

Especifica a expressão then com base na condição de boolean_expression.

`then_expression` e `else_expression` devem ser do mesmo tipo ou coercíveis a um tipo comum.

- else_expression

  Especifica a expressão padrão.

  `then_expression` e `else_expression` devem ser do mesmo tipo ou coercíveis a um tipo comum.

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan', 50);
SELECT id, CASE WHEN id > 200 THEN 'bigger' ELSE 'small' END FROM person;
+------+-----------------------------------------------------+
|  id  | CASE WHEN (id > 200) THEN bigger ELSE small END  |
+------+-----------------------------------------------------+
| 100  | small                                               |
| 200  | small                                               |
| 300  | bigger                                              |
| 400  | bigger                                              |
+------+-----------------------------------------------------+
SELECT id, CASE id WHEN 100 then 'bigger' WHEN  id > 300 THEN '300' ELSE 'small' END
 FROM person;
+------
+-------------------------------------------------------------------------------------------------------
+
|  id  | CASE WHEN (id = 100) THEN bigger WHEN (id = CAST((id > 300) AS INT)) THEN 300
 ELSE small END  |
+------
+-------------------------------------------------------------------------------------------------------
+
| 100  | bigger
                |
```

```
| 200  | small
                |
| 300  | small
                |
| 400  | small
                |
+------
+-------------------------------------------------------------------------------------------------------------
+
```

Expressão de tabela comum

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Uma expressão de tabela comum (CTE) define um conjunto de resultados temporários que um usuário pode referenciar várias vezes dentro do escopo de uma instrução SQL. Uma CTE é usada principalmente em um instrução `SELECT`.

Sintaxe

```
WITH common_table_expression [ , ... ]
```

Enquanto `common_table_expression` é definido como:

```
Syntexpression_name [ ( column_name [ , ... ] ) ] [ AS ] ( query )
```

Parâmetros

- expression_name

  Especifica um nome para a expressão de tabela comum.
- query

  Uma instrução `SELECT`.

Exemplos

```
-- CTE with multiple column aliases
WITH t(x, y) AS (SELECT 1, 2)
SELECT * FROM t WHERE x = 1 AND y = 2;
+---+---+
|  x|  y|
+---+---+
|  1|  2|
+---+---+

-- CTE in CTE definition
WITH t AS (
WITH t2 AS (SELECT 1)
SELECT * FROM t2
)
SELECT * FROM t;
+---+
|  1|
+---+
|  1|
+---+

-- CTE in subquery
SELECT max(c) FROM (
WITH t(c) AS (SELECT 1)
SELECT * FROM t
);
+------+
|max(c)|
+------+
|     1|
+------+

-- CTE in subquery expression
SELECT (
WITH t AS (SELECT 1)
SELECT * FROM t
);
+----------------+
|scalarsubquery()|
+----------------+
|               1|
+----------------+
```

```
-- CTE in CREATE VIEW statement
CREATE VIEW v AS
WITH t(a, b, c, d) AS (SELECT 1, 2, 3, 4)
SELECT * FROM t;
SELECT * FROM v;
+---+---+---+---+
|  a|  b|  c|  d|
+---+---+---+---+
|  1|  2|  3|  4|
+---+---+---+---+
```

EXPLAIN

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A instrução `EXPLAIN` é usada para fornecer planos lógicos/físicos para uma instrução de entrada. Por padrão, essa cláusula fornece informações somente sobre um plano físico.

Sintaxe

```
EXPLAIN [ EXTENDED | CODEGEN | COST | FORMATTED ] statement
```

Parâmetros

- EXTENDED

  Gera um plano lógico analisado, um plano lógico analisado sintaticamente, um plano lógico otimizado e um plano físico.

  O plano lógico analisado sintaticamente é um plano não resolvido que foi extraído da consulta.

  Os planos lógicos analisados transformam o que é traduzido como `unresolvedAttribute` e `unresolvedRelation` em objetos totalmente tipificados.

  O plano lógico otimizado faz a transformação por meio de um conjunto de regras de otimização, resultando no plano físico.
- CODEGEN

Gera código para a instrução, se existir, e um plano físico.

- COST

  Se as estatísticas do nó do plano estiverem disponíveis, gera um plano lógico e as estatísticas.

- FORMATTED

  Gera duas seções: um esboço do plano físico e detalhes de nó.

- instrução

  Especifica uma instrução SQL a ser explicada.

Exemplos

```
-- Default Output
EXPLAIN select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+----------------------------------------------------+
|                                                plan|
+----------------------------------------------------+
| == Physical Plan ==
*(2) HashAggregate(keys=[k#33], functions=[sum(cast(v#34 as bigint))])
+- Exchange hashpartitioning(k#33, 200), true, [id=#59]
+- *(1) HashAggregate(keys=[k#33], functions=[partial_sum(cast(v#34 as bigint))])
+- *(1) LocalTableScan [k#33, v#34]
|
+----------------------------------------------------

-- Using Extended
EXPLAIN EXTENDED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+----------------------------------------------------+
|                                                plan|
+----------------------------------------------------+
| == Parsed Logical Plan ==
'Aggregate ['k], ['k, unresolvedalias('sum('v), None)]
 +- 'SubqueryAlias `t`
+- 'UnresolvedInlineTable [k, v], [List(1, 2), List(1, 3)]

 == Analyzed Logical Plan ==
 k: int, sum(v): bigint
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- SubqueryAlias `t`
    +- LocalRelation [k#47, v#48]
```

```
 == Optimized Logical Plan ==
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- LocalRelation [k#47, v#48]

 == Physical Plan ==
 *(2) HashAggregate(keys=[k#47], functions=[sum(cast(v#48 as bigint))], output=[k#47,
 sum(v)#50L])
+- Exchange hashpartitioning(k#47, 200), true, [id=#79]
   +- *(1) HashAggregate(keys=[k#47], functions=[partial_sum(cast(v#48 as bigint))],
 output=[k#47, sum#52L])
    +- *(1) LocalTableScan [k#47, v#48]
|
+------------------------------------------------------+

-- Using Formatted
EXPLAIN FORMATTED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+------------------------------------------------------+
|                                                  plan|
+------------------------------------------------------+
| == Physical Plan ==
 * HashAggregate (4)
 +- Exchange (3)
    +- * HashAggregate (2)
       +- * LocalTableScan (1)


 (1) LocalTableScan [codegen id : 1]
 Output: [k#19, v#20]

 (2) HashAggregate [codegen id : 1]
 Input: [k#19, v#20]

 (3) Exchange
 Input: [k#19, sum#24L]

 (4) HashAggregate [codegen id : 2]
 Input: [k#19, sum#24L]
|
+------------------------------------------------------+
```

Cláusula LATERAL SUBQUERY

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

`LATERAL SUBQUERY` é uma subconsulta que é precedida pela palavra-chave `LATERAL`. Fornece uma maneira de referenciar colunas na cláusula `FROM` precendente. Sem a palavra-chave `LATERAL`, as subconsultas só podem referenciar colunas na consulta externa, mas não na cláusula `FROM`. `LATERAL SUBQUERY` torna as consultas complicadas mais simples e eficientes.

Sintaxe

```
[ LATERAL ] primary_relation [ join_relation ]
```

Parâmetros

- primary_relation

  Especifica a relação primária. Uma das seguintes opções é possível:

  1. Relação de tabela
  2. Consulta com alias

     Sintaxe: `( query ) [ [ AS ] alias ]`
  3. Relação com alias

     `Syntax: ( relation ) [ [ AS ] alias ]`

Exemplos

```
CREATE TABLE t1 (c1 INT, c2 INT);
INSERT INTO t1 VALUES (0, 1), (1, 2);
CREATE TABLE t2 (c1 INT, c2 INT);
INSERT INTO t2 VALUES (0, 2), (0, 3);
SELECT * FROM t1,
LATERAL (SELECT * FROM t2 WHERE t1.c1 = t2.c1);
+--------+-------+--------+-------+
|  t1.c1 | t1.c2 |  t2.c1 | t2.c2 |
```

```
+-------+--------+--------+-------+
|    0  |   1    |    0   |   3   |
|    0  |   1    |    0   |   2   |
+-------+--------+--------+-------+
SELECT a, b, c FROM t1,
LATERAL (SELECT c1 + c2 AS a),
LATERAL (SELECT c1 - c2 AS b),
LATERAL (SELECT a * b AS c);
+--------+-------+--------+
|    a   |   b   |    c   |
+-------+--------+--------+
|    3   |  -1   |   -3   |
|    1   |  -1   |   -1   |
+-------+--------+--------+
```

Cláusula LATERAL VIEW

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `LATERAL VIEW` é usada em conjunto com funções geradoras, como `EXPLODE`, o que gerará uma tabela virtual contendo uma ou mais linhas. `LATERAL VIEW` aplicará as linhas a cada linha de saída original.

Sintaxe

```
LATERAL VIEW [ OUTER ] generator_function ( expression [ , ... ] ) [ table_alias ] AS
 column_alias [ , ... ]
```

Parâmetros

- EXTERNO

  Se `OUTER` especificado, retornará null se uma entrada array/map estiver vazia ou nula.
- generator_function

  Especifica uma função geradora (`EXPLODE`, `INLINE` e assim por diante.).
- table_alias

O alias para `generator_function`, que é opcional.

- column_alias

  Lista os aliases da coluna da `generator_function`, que podem ser usados nas linhas da saída.

  Você pode ter vários aliases se `generator_function` tiver várias colunas de saída.

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age;
+------+-------+-------+--------+-----------+--------+--------+
|  id  | name  |  age  | class  |  address  | c_age  | d_age  |
+------+-------+-------+--------+-----------+--------+--------+
| 100  | John  | 30    | 1      | Street 1  | 30     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 30     | 80     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 80     |
+------+-------+-------+--------+-----------+--------+--------+
SELECT c_age, COUNT(1) FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age
GROUP BY c_age;
```

```
+--------+-----------+
| c_age  | count(1)  |
+--------+-----------+
| 60     | 8         |
| 30     | 8         |
+--------+-----------+
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY()) tableName AS c_age;
+-----+-------+------+--------+----------+--------+
| id  | name  | age  | class  | address  | c_age  |
+-----+-------+------+--------+----------+--------+
+-----+-------+------+--------+----------+--------+
SELECT * FROM person
LATERAL VIEW OUTER EXPLODE(ARRAY()) tableName AS c_age;
+------+-------+-------+--------+-----------+--------+
|  id  | name  |  age  | class  |  address  | c_age  |
+------+-------+-------+--------+-----------+--------+
| 100  | John  | 30    | 1      | Street 1  | NULL   |
| 200  | Mary  | NULL  | 1      | Street 2  | NULL   |
| 300  | Mike  | 80    | 3      | Street 3  | NULL   |
| 400  | Dan   | 50    | 4      | Street 4  | NULL   |
+------+-------+-------+--------+-----------+--------+
```

Predicado LIKE

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Um predicado `LIKE` é usado para pesquisar um padrão específico. Esse predicado também é compatível com vários padrões com quantificadores que incluem `ANY`, `SOME` e `ALL`.

Sintaxe

```
[ NOT ] { LIKE search_pattern [ ESCAPE esc_char ] | [ RLIKE | REGEXP ] regex_pattern }
[ NOT ] { LIKE quantifiers ( search_pattern [ , ... ]) }
```

Parâmetros

- search_pattern

Especifica um padrão de string a ser pesquisado pela cláusula LIKE. Pode conter caracteres especiais correspondentes a um padrão:

- `%` corresponde a 0 ou mais caracteres.
- `_` corresponde a exatamente um caractere.

- esc_char

  Especifica o caractere de escape. O caractere de escape padrão é `\`.

- regex_pattern

  Especifica um padrão de pesquisa de expressão regular a ser pesquisado pela cláusula `RLIKE` ou `REGEXP`.

- quantifiers

  Especifica que os quantificadores de predicado que incluem `ANY SOME` e `ALL`.

  `ANY` ou `SOME` significa que se um dos padrões corresponder à entrada, verdadeiro será retornado.

  `ALL` significa que se todos os padrões corresponderem à entrada, verdadeiro será retornado.

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan',  50),
(500, 'Evan_w', 16);
SELECT * FROM person WHERE name LIKE 'M%';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE 'M_ry';
+---+----+----+
| id|name| age|
+---+----+----+
```

```
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE 'M_ry';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
|300|  Mike| 80|
|100|  John| 30|
|400|   Dan| 50|
+---+------+---+
SELECT * FROM person WHERE name RLIKE 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name REGEXP 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE '%\_%';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE '%$_%' ESCAPE '$';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE ALL ('%an%', '%an');
+---+----+----+
| id|name| age|
+---+----+----+
|400| Dan|  50|
+---+----+----+
SELECT * FROM person WHERE name LIKE ANY ('%an%', '%an');
```

```
+---+------+---+
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE SOME ('%an%', '%an');
+---+------+---+
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name NOT LIKE ALL ('%an%', '%an');
+---+----+----+
| id|name| age|
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE ANY ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
SELECT * FROM person WHERE name NOT LIKE SOME ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
```

OFFSET

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `OFFSET` é usada para especificar o número de linhas a serem ignoradas antes de começar a retornar as linhas retornadas pela instrução `SELECT`. Em geral, essa cláusula é usada em conjunto com `ORDER BY` para garantir que os resultados sejam determinísticos.

Sintaxe

```
OFFSET integer_expression
```

Parâmetros

- integer_expression

  Especifica uma expressão redutível que retorna um número inteiro.

Exemplos

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('Juan L', 25),
('John D', 18),
('Jorge S', 16);

-- Skip the first two rows.
SELECT name, age FROM person ORDER BY name OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
| Juan L| 25|
|Nikki W| 16|
```

```
|Jane Doe| 25|
+-------+---+

-- Skip the first two rows and returns the next three rows.
SELECT name, age FROM person ORDER BY name LIMIT 3 OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
| Juan L| 25|
|Nikki W| 16|
+-------+---+

-- A function expression as an input to OFFSET.
SELECT name, age FROM person ORDER BY name OFFSET length('WAGON');
+-------+---+
|   name|age|
+-------+---+
|Jane Doe| 25|
+-------+---+
```

Cláusula PIVOT

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `PIVOT` é usada para perspectiva de dados. Podemos obter os valores agregados com base em valores de colunas específicos, que serão transformados em várias colunas usadas na cláusula `SELECT`. A cláusula `PIVOT` pode ser especificada depois do nome da tabela ou da subconsulta.

Sintaxe

```
PIVOT ( { aggregate_expression [ AS aggregate_expression_alias ] } [ , ... ] FOR
 column_list IN ( expression_list ) )
```

Parâmetros

- aggregate_expression

Especifica uma expressão de agregação (`SUM(a)`, `COUNT(DISTINCT b)` etc.).

- aggregate_expression_alias

  Especifica um alias para a expressão de agregação.

- column_list

  Contém colunas na cláusula `FROM`, que especifica as colunas que você deseja substituir por novas colunas. Você pode colocar as colunas entre colchetes, como `(c1, c2)`.

- expression_list

  Especifica novas colunas, que são usadas para verificar a correspondência com valores em `column_list` como condição de agregação. Você também pode definir aliases para elas.

Exemplos

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR name IN ('John' AS john, 'Mike' AS mike)
);
+------+-----------+---------+---------+---------+---------+
|  id  |  address  | john_a  | john_c  | mike_a  | mike_c  |
+------+-----------+---------+---------+---------+---------+
| 200  | Street 2  | NULL    | NULL    | NULL    | NULL    |
| 100  | Street 1  | 30      | 1.0     | NULL    | NULL    |
| 300  | Street 3  | NULL    | NULL    | 80      | 3.0     |
| 400  | Street 4  | NULL    | NULL    | NULL    | NULL    |
+------+-----------+---------+---------+---------+---------+
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR (name, age) IN (('John', 30) AS c1, ('Mike', 40) AS c2)
);
+------+-----------+-------+-------+-------+-------+
```

```
|  id  |  address  | c1_a  | c1_c  | c2_a  | c2_c  |
+------+-----------+-------+-------+-------+-------+
| 200  | Street 2  | NULL  | NULL  | NULL  | NULL  |
| 100  | Street 1  | 30    | 1.0   | NULL  | NULL  |
| 300  | Street 3  | NULL  | NULL  | NULL  | NULL  |
| 400  | Street 4  | NULL  | NULL  | NULL  | NULL  |
+------+-----------+-------+-------+-------+-------+
```

Configurar operadores

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

Os operadores de conjunto são usados para combinar duas relações de entrada em uma. OpenSearch O SQL oferece suporte a três tipos de operadores de conjunto:

- `EXCEPT` ou `MINUS`
- `INTERSECT`
- `UNION`

As relações de entrada devem ter o mesmo número de colunas e tipos de dados compatíveis com as respectivas colunas.

EXCETO

`EXCEPT` e `EXCEPT ALL` retornam as linhas encontradas em uma relação, mas não na outra. `EXCEPT` (como alternativa, `EXCEPT DISTINCT`) usa somente linhas distintas, enquanto `EXCEPT ALL` não remove duplicações das linhas de resultado. Observe que `MINUS` é um alias de `EXCEPT`.

Sintaxe

```
[ ( ] relation [ ) ] EXCEPT | MINUS [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemplos

```
-- Use table1 and table2 tables to demonstrate set operators in this page.
SELECT * FROM table1;
```

```
+---+
|  c|
+---+
|  3|
|  1|
|  2|
|  2|
|  3|
|  4|
+---+
SELECT * FROM table2;
+---+
|  c|
+---+
|  5|
|  1|
|  2|
|  2|
+---+
SELECT c FROM table1 EXCEPT SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 EXCEPT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS ALL (SELECT c FROM table2);
+---+
|  c|
```

```
+---+
|  3|
|  3|
|  4|
+---+
```

CRUZAR

`INTERSECT` e `INTERSECT ALL` retornam as linhas encontradas em ambas as relações. `INTERSECT` (como alternativa, `INTERSECT DISTINCT`) usa somente linhas distintas, enquanto `INTERSECT ALL` não remove duplicações das linhas de resultado.

Sintaxe

```
[ ( ] relation [ ) ] INTERSECT [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemplos

```
(SELECT c FROM table1) INTERSECT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
|  2|
+---+
```

UNIÃO

UNION e UNION ALL retornam as linhas encontradas em qualquer das duas relações. UNION (como alternativa, UNION DISTINCT) usa somente linhas distintas, enquanto UNION ALL não remove duplicações das linhas de resultado.

Sintaxe

```
[ ( ] relation [ ) ] UNION [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemplos

```
(SELECT c FROM table1) UNION (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
|  5|
|  4|
|  2|
+---+
(SELECT c FROM table1) UNION DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
|  5|
|  4|
|  2|
+---+
SELECT c FROM table1 UNION ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  1|
|  2|
|  2|
|  3|
|  4|
|  5|
|  1|
|  2|
```

```
|  2|
+---+
```

Cláusula SORT BY

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `SORT BY` é usada para retornar as linhas de resultado classificadas em cada partição na ordem especificada pelo usuário. Quando há mais de uma partição, `SORT BY` pode retornar um resultado parcialmente ordenado. Isso é diferente da cláusula `ORDER BY` que garante a ordem total da saída.

Sintaxe

```
SORT BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Parâmetros

- SORT BY

  Especifica uma lista de expressões separadas por vírgula junto com parâmetros opcionais sort_direction e nulos_sort_order que são usados para classificar as linhas em cada partição.

- sort_direction

  Opcionalmente, especifica se as linhas devem ser classificadas em ordem crescente ou decrescente.

  Os valores válidos para a direção de classificação são `ASC` para crescente e `DESC` para decrescente.

  Se a direção da classificação não for especificada explicitamente, as linhas serão classificadas em ordem crescente por padrão.

  Sintaxe: `[ ASC | DESC ]`

- nulos_sort_order

Opcionalmente, especifica se os valores NULL são retornados antes/depois dos valores não NULL.

Se não `null_sort_order` for especificado, NULLs classifique primeiro se a ordem de classificação for `ASC` e NULLS classificará por último se a ordem de classificação for. `DESC`

1. Se `NULLS FIRST` for especificado, os valores NULL serão retornados primeiro, independentemente da ordem de classificação.

2. Se `NULLS LAST` for especificado, os valores NULL serão retornados por último, independentemente da ordem de classificação.

Sintaxe: `[ NULLS { FIRST | LAST } ]`

Exemplos

```
CREATE TABLE person (zip_code INT, name STRING, age INT);
INSERT INTO person VALUES
(94588, 'Shirley Rodriguez', 50),
(94588, 'Juan Li', 18),
(94588, 'Anil K', 27),
(94588, 'John D', NULL),
(94511, 'David K', 42),
(94511, 'Aryan B.', 18),
(94511, 'Lalit B.', NULL);
-- Sort rows by `name` within each partition in ascending manner
SELECT name, age, zip_code FROM person SORT BY name;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
-- Sort rows within each partition using column position.
SELECT name, age, zip_code FROM person SORT BY 1;
+---------------+----+----------+
```

```
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+

-- Sort rows within partition in ascending manner keeping null values to be last.
SELECT age, name, zip_code FROM person SORT BY age NULLS LAST;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|  18|           Juan Li|   94588|
|  27|            Anil K|   94588|
|  50| Shirley Rodriguez|   94588|
|null|            John D|   94588|
|  18|          Aryan B.|   94511|
|  42|           David K|   94511|
|null|          Lalit B.|   94511|
+--------------+--------+--------+

-- Sort rows by age within each partition in descending manner, which defaults to NULL
 LAST.
SELECT age, name, zip_code FROM person SORT BY age DESC;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|  50|          Shirley Rodriguez|   94588|
|  27|            Anil K|   94588|
|  18|           Juan Li|   94588|
|null|            John D|   94588|
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
|null|          Lalit B.|   94511|
+----+------------------+--------+

-- Sort rows by age within each partition in descending manner keeping null values to
 be first.
SELECT age, name, zip_code FROM person SORT BY age DESC NULLS FIRST;
+----+------------------+--------+
```

```
| age|              name|zip_code|
+----+------------------+--------+
|null|            John D|   94588|
|  50| Shirley Rodriguez|   94588|
|  27|             Anil K|   94588|
|  18|           Juan Li|   94588|
|null|          Lalit B.|   94511|
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
+--------------+--------+--------+

-- Sort rows within each partition based on more than one column with each column
 having
-- different sort direction.
SELECT name, age, zip_code FROM person
SORT BY name ASC, age DESC;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|             Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
```

UNPIVOT

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando SQL, consultethe section called “Comandos SQL compatíveis”.

A cláusula `UNPIVOT` transforma várias colunas em várias linhas usadas na cláusula `SELECT`. A cláusula `UNPIVOT` pode ser especificada depois do nome da tabela ou da subconsulta.

Sintaxe

```
UNPIVOT [ { INCLUDE | EXCLUDE } NULLS ] (
```

```
    { single_value_column_unpivot | multi_value_column_unpivot }
) [[AS] alias]

single_value_column_unpivot:
    values_column
    FOR name_column
    IN (unpivot_column [[AS] alias] [, ...])

multi_value_column_unpivot:
    (values_column [, ...])
    FOR name_column
    IN ((unpivot_column [, ...]) [[AS] alias] [, ...])
```

Parâmetros

- unpivot_column

  Contém colunas na cláusula FROM, o que especifica as colunas que você deseja despivotar.

- name_column

  O nome da coluna que contém os nomes das colunas despivotadas.

- values_column

  O nome da coluna que contém os valores das colunas despivotadas.

Exemplos

```
CREATE TABLE sales_quarterly (year INT, q1 INT, q2 INT, q3 INT, q4 INT);
INSERT INTO sales_quarterly VALUES
(2020, null, 1000, 2000, 2500),
(2021, 2250, 3200, 4200, 5900),
(2022, 4200, 3100, null, null);
-- column names are used as unpivot columns
SELECT * FROM sales_quarterly
UNPIVOT (
sales FOR quarter IN (q1, q2, q3, q4)
);
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | q2      | 1000  |
| 2020 | q3      | 2000  |
```

```
| 2020 | q4      | 2500  |
| 2021 | q1      | 2250  |
| 2021 | q2      | 3200  |
| 2021 | q3      | 4200  |
| 2021 | q4      | 5900  |
| 2022 | q1      | 4200  |
| 2022 | q2      | 3100  |
+------+---------+-------+
-- NULL values are excluded by default, they can be included
-- unpivot columns can be alias
-- unpivot result can be referenced via its alias
SELECT up.* FROM sales_quarterly
UNPIVOT INCLUDE NULLS (
sales FOR quarter IN (q1 AS Q1, q2 AS Q2, q3 AS Q3, q4 AS Q4)
) AS up;
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | Q1      | NULL  |
| 2020 | Q2      | 1000  |
| 2020 | Q3      | 2000  |
| 2020 | Q4      | 2500  |
| 2021 | Q1      | 2250  |
| 2021 | Q2      | 3200  |
| 2021 | Q3      | 4200  |
| 2021 | Q4      | 5900  |
| 2022 | Q1      | 4200  |
| 2022 | Q2      | 3100  |
| 2022 | Q3      | NULL  |
| 2022 | Q4      | NULL  |
+------+---------+-------+
-- multiple value columns can be unpivoted per row
SELECT * FROM sales_quarterly
UNPIVOT EXCLUDE NULLS (
(first_quarter, second_quarter)
FOR half_of_the_year IN (
(q1, q2) AS H1,
(q3, q4) AS H2
)
);
+------+------------------+---------------+----------------+
|  id  | half_of_the_year | first_quarter | second_quarter |
+------+------------------+---------------+----------------+
| 2020 | H1               | NULL          | 1000           |
```

```
| 2020 | H2                | 2000          | 2500           |
| 2021 | H1                | 2250          | 3200           |
| 2021 | H2                | 4200          | 5900           |
| 2022 | H1                | 4200          | 3100           |
+------+-------------------+---------------+----------------+
```

Comandos PPL compatíveis

As tabelas a seguir mostram quais comandos PPL o OpenSearch Dashboards suporta para consultar CloudWatch Logs, Amazon S3 ou Security Lake, e quais comandos o Logs Insights suporta. CloudWatch CloudWatch O Logs Insights usa a mesma sintaxe PPL dos OpenSearch painéis ao consultar CloudWatch registros, e as tabelas se referem a ambos como registros. CloudWatch

> **Note**
>
> Quando você analisa dados fora do OpenSearch Serviço, os comandos podem ser executados de forma diferente do que nos OpenSearch índices.

Tópicos

- Comandos
- Funções
- Informações adicionais para usuários do CloudWatch Logs Insights que usam OpenSearch PPL

Comandos

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “fields”	Exibe um conjunto de campos que precisa de projeção.	compatível	compatível	compatível	`fields field1, field2`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “para onde”	Filtra os dados com base nas condições que você especific a.	compatível	compatível	compatível	`where field1="success" | where field2 != "i-023fe0a90929d8822" | fields field3, col4, col5, col6 | head 1000`
the section called “stats”	Executa agregações e cálculos.	compatível	compatível	compatível	`stats count(), count(`field1`), min(`field1`), max(`field1`), avg(`field1`) by field2 | head 1000`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “parse”	Extrai um padrão de expressão regular (regex) de uma string e exibe o padrão extraído. O padrão extraído pode ser usado posterior mente para criar novos campos ou filtrar dados.	compatível	compatível	compatível	`parse `field1` ".*/(?<field2>[^/]+$)" | where field2 = "requestId" | fields field2, `field2` | head 1000`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “Padrões”	Extrai padrões de log de um campo de texto e acrescenta os resultados ao resultado da pesquisa. Agrupar os logs por seus padrões facilita a agregação de estatísticas de grandes volumes de dados de log para análise e solução de problemas.	não compatível	compatível	compatível	`patterns new_field='no_numbers' pattern='[0-9]' message | fields message, no_numbers`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “sort”	Classifique os resultados exibidos por um nome de campo. Use classificar - FieldName para classificar em ordem decrescente.	compatível	compatível	compatível	`stats count(), count(`field1`), min(`field1`) as field1Alias, max(`field1`), avg(`field1`) by field2 | sort - field1Alias | head 1000`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “eval”	Modifica ou processa o valor de um campo e o armazena em outro campo. Esse comando é útil para modificar matematic amente uma coluna, aplicar funções de string a uma coluna ou aplicar funções de data a uma coluna.	compatível	compatível	compatível	`eval field2 = `field1` * 2 | fields field1, field2 | head 20`
the section called “rename”	Renomeia um ou mais campos no resultado da pesquisa.	compatível	compatível	compatível	`rename field2 as field1 | fields field1`
the section called “head”	Limita os resultados da consulta exibidos às primeiras N linhas.	compatível	compatível	compatível	`fields `@message` | head 20`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “grok”	Analisa um campo de texto com um padrão grok baseado em expressão regular e acrescenta os resultados ao resultado da pesquisa.	compatível	compatível	compatível	`grok email '.+@%{HOSTNAME:host}' | fields email`
the section called “top”	Encontra os valores mais frequentes para um campo.	compatível	compatível	compatível	`top 2 Field1 by Field2`
the section called “dedup”	Remove entradas duplicadas com base nos campos que você especifica.	compatível	compatível	compatível	`dedup field1 | fields field1, field2, field3`
the section called “ingressar”	Une dois conjuntos de dados.	compatível	compatível	compatível	`source=customer | join ON c_custkey = o_custkey orders | head 10`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “lookup”	Enriquece seus dados de pesquisa adicionando ou substituindo dados de um índice de pesquisa (tabela de dimensões). Você pode estender os campos de um índice com valores de uma tabela de dimensões, acrescentar ou substituir valores quando a condição de pesquisa é atendida	não compatível	compatível	compatível	`where orderType = 'Cancelled' | lookup account_list, mkt_id AS mkt_code replace amount, account_name as name | stats count(mkt_code), avg(amount) by name`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “subconsulta”	Realiza consultas aninhadas complexas nas instruções Piped Processing Language (PPL).	compatível	compatível	compatível	`where id in [ subquery source=users | where user in [ subquery source=actions | where action="login" | fields user ] | fields uid ]`
the section called “rare”	Encontra os valores menos frequentes de todos os campos na lista de campos.	compatível	compatível	compatível	`rare Field1 by Field2`
the section called “trendline”	Calcula as médias móveis dos campos.	compatível	compatível	compatível	`trendline sma(2, field1) as field1Alias`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “eventstats”	Enriquece os dados de evento com estatísticas resumidas calculadas. Esse comando analisa campos especific ados em seus eventos, calcula várias medidas estatísticas e, em seguida, anexa esses resultados a cada evento original como novos campos.	Compatível com (exceto `count()`)	compatível	compatível	`eventstats sum(field1) by field2`
the section called “nivelamento”	Nivela um campo. O campo deve ser deste tipo: `struct<?,?> or array<struct<?,?>>`	compatível	compatível	compatível	`source=table | flatten field1`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “fieldsummary”	Calcula estatísticas básicas para cada campo (contagem, contagem distinta, mín, máx, méd., stddev e mediana).	Compatível com (um campo por consulta)	compatível	compatível	`where field1 != 200 | fieldsummary includefields=field1 nulls=true`
the section called “fillnull”	Preenche campos nulo com o valor que você fornece. Pode ser usado em um ou mais campos.	compatível	compatível	compatível	`fields field1 | eval field2=field1 | fillnull value=0 field1`
the section called “Ampliar”	Divide um campo contendo vários valores em linhas separadas, criando uma nova linha para cada valor no campo especificado.	compatível	compatível	compatível	`expand employee | stats max(salary) as max by state, company`

Comando PPL	Description	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “describe”	Obtém informaçõ es detalhada s sobre a estrutura e os metadados de tabelas, esquemas e catálogos	não compatível	compatível	compatível	`describe schema.table`

Funções

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “String” (`CONCAT`, `CONCAT_WS` , `LENGTH`, `LOWER`, `LTRIM`, `POSITION`, `REVERSE`, `RIGHT`, `RTRIM`, `SUBSTRING` , `TRIM`, `UPPER`)	Funções integrada s na PPL que podem manipular e transform ar dados de string e texto em consultas PPL. Por exemplo, converter maiúscula s e	compatível	compatíve l	compatíve l	`eval col1Len = LENGTH(co l1) | fields col1Len`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
	minúscula s, combinar strings, extrair partes e apagar texto.				

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “Data e hora” (DAY, DAYOFMONTH , DAY_OF_MONTH ,DAYOFWEEK , DAY_OF_WEEK , DAYOFYEAR , DAY_OF_YEAR , DAYNAME, FROM_UNIXTIME , HOUR, HOUR_OF_DAY , LAST_DAY, LOCALTIMESTAMP , LOCALTIME , MAKE_DATE , MINUTE, MINUTE_OF_HOUR , MONTH, MONTHNAME , MONTH_OF_YEAR , NOW, QUARTER, SECOND, SECOND_OF_MINUTE , SUBDATE, SYSDATE, TIMESTAMP , UNIX_TIMESTAMP , WEEK, WEEKDAY, WEEK_OF_YEAR , DATE_ADD,	Funções integradas para manipular e transformar dados de data e carimbo de data/hora em consultas PPL. Por exemplo, date_add, date_format, datediff e current_date.	compatível	compatível	compatível	`eval newDate = ADDDATE(DATE('2020-08-26'), 1) | fields newDate`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
DATE_SUB, TIMESTAMP ADD , TIMESTAMP DIFF , UTC_TIMES TAMP , CURRENT_T IMEZONE)					
the section called “Condição” (EXISTS, IF, IFNULL, ISNOTNULL , ISNULL, NULLIF)	Funções integrada s que realizam cálculos em várias linhas para produzir um único valor resumido. Por exemplo: sum, count, avg, max e min.	compatível	compatíve l	compatíve l	`eval field2 = isnull(co l1) | fields field2, col1, field3`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “Matemática” (ABS, ACOS, ASIN, ATAN, ATAN2, CEIL, CEILING, CONV, COS, COT, CRC32, DEGREES, E, EXP, FLOOR, LN, LOG, LOG2, LOG10, MOD, PI. POW, POWER, RADIANS, RAND, ROUND, SIGN, SIN, SQRT, CBRT)	Funções integrada s para realizar cálculos matemátic os e transform ações em consultas PPL. Por exemplo: abs (valor absoluto) , round (arredond a números), sqrt (raiz quadrada) , pow (cálculo de potência) e ceil (arredond a para o inteiro mais próximo).	compatível	compatíve l	compatíve l	`eval field2 = ACOS(col1) | fields col1`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “Expressões” (Operadores aritméticos (`+`, `-`, `*`), operadores de predicados (`>. <`, `IN`))	Funções integrada s para expressõe s, especialm ente expressõe s de valor, retornam um valor escalar. As expressõe s têm diferente s tipos e formatos.	compatível	compatíve l	compatíve l	`where age > (25 + 5) | fields age`
the section called “IP address (endereço de IP)” (`CIDRMATCH `)	Funções integrada s para lidar com endereços IP, como CIDR.	compatível	compatíve l	compatíve l	`where cidrmatch (ip, '******** ***/24') | fields ip`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “JSON” (ARRAY_LENGTH , ARRAY_LENGTH , JSON, JSON_ARRAY , JSON_EXTRACT , JSON_KEYS , JSON_OBJECT , JSON_VALID , TO_JSON_STRING)	Funções integradas para lidar com JSON, incluindo matrizes, extração e validação.	compatível	compatível	compatível	`eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}', '$a')`
the section called “Lambda” (EXISTS, FILTER, REDUCE, TRANSFORM)	Funções integradas para lidar com JSON, incluindo matrizes, extração e validação.	não compatível	compatível	compatível	`eval array = json_array(1, -1, 2), result = filter(array, x -> x > 0) | fields result`

Função PPL	Descripti on	CloudWatch Registros	Amazon S3	Lago de Segurança	Exemplo de comando da
the section called “Criptográfico” (MD5, SHA1, SHA2)	Funções integrada s que permitem gerar impressõe s digitais exclusiva s de dados, que podem ser usadas para verificaç ão, comparaçã o ou como parte de protocolo s de segurança mais complexos .	compatível	compatíve l	compatíve l	`eval `MD5('hel lo')` = MD5('hell o') | fields `MD5('hel lo')`

Informações adicionais para usuários do CloudWatch Logs Insights que usam OpenSearch PPL

Embora o CloudWatch Logs Insights ofereça suporte à maioria dos comandos e funções do OpenSearch PPL, alguns comandos e funções não são compatíveis atualmente. Por exemplo,

atualmente ele não é compatível com a comandos de pesquisa em PPL. A partir de 2 de junho de 2025, o CloudWatch Logs Insights agora oferece suporte às funções JOIN, subqueries, Flatten, Fillnull, Expand, Cidrmatch e JSON no PPL. Para obter uma lista completa dos comandos e funções de consulta compatíveis, consulte as colunas Amazon CloudWatch Logs nas tabelas acima.

Exemplos de consultas e cotas

O seguinte se aplica tanto aos usuários do CloudWatch Logs Insights quanto OpenSearch aos usuários que consultam CloudWatch dados.

Para obter informações sobre os limites que se aplicam ao consultar CloudWatch registros do OpenSearch serviço, consulte Cotas de CloudWatch registros no Guia do usuário do Amazon CloudWatch Logs. Os limites envolvem o número de grupos de CloudWatch registros que você pode consultar, o máximo de consultas simultâneas que você pode executar, o tempo máximo de execução da consulta e o número máximo de linhas retornadas nos resultados. Os limites são os mesmos, independentemente da linguagem usada para consultar CloudWatch os registros (ou seja, OpenSearch PPL, SQL e Logs Insights QL).

Comandos PPL

Tópicos

- comment
- Comando correlation
- Comando dedup
- Comando describe
- Comando eval
- Comando eventstats
- Comando expand
- Comando explain
- Comando fillnull
- Comando fields
- Comando flatten
- Comando grok
- Comando head
- Comando join
- Comando lookup

- comando parse
- Comando patterns
- Comando rare
- Comando rename
- Comando search
- Comando sort
- Comando stats
- Comando subquery
- Comando top
- Comando trendline
- Comando where
- field summary
- Comando expand
- Funções PPL

comment

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O PPL é compatível com comentários em linha e comentários em bloco. O sistema não avalia texto de comentário.

Comentários de linha

Os comentários de linha começam com duas barras // e terminam com uma nova linha.

Exemplo:

```
os> source=accounts | top gender // finds most common gender of all the accounts
fetched rows / total rows = 2/2
+----------+
| gender   |
```

```
|----------|
| M        |
| F        |
+----------+
```

Comentários de bloco

Os comentários de bloco começam com uma barra invertida seguida por um asterisco * e terminam com um asterisco seguido por uma barra */.

Exemplo:

```
os> source=accounts | dedup 2 gender /* dedup the document with gender field keep 2
 duplication */ | fields account_number, gender
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

Comando correlation

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Você pode correlacionar diferentes fontes de dados de acordo com dimensões e prazos em comum.

Essa correlação é crucial quando você lida com grandes quantidades de dados de vários setores que compartilham os mesmos períodos, mas não estão sincronizados formalmente.

Ao correlacionar essas diferentes fontes de dados com base em prazos e dimensões semelhantes, você pode enriquecer seus dados e revelar informações valiosas.

Exemplo

O domínio de observabilidade tem três fontes de dados distintas:

- Logs
- Metrics
- Rastreamentos

Essas fontes de dados podem compartilhar dimensões comuns. Para a transição de uma fonte de dados para outra, você precisa correlacioná-las corretamente. Usando convenções de nomenclatura semântica, você pode identificar os elementos compartilhados entre logs, rastreamentos e métricas.

Exemplo:

```
{
  "@timestamp": "2018-07-02T22:23:00.186Z",
  "aws": {
    "elb": {
      "backend": {
        "http": {
          "response": {
            "status_code": 500
          }
        },
        "ip": "********",
        "port": "80"
      },
      ...
     "target_port": [
        "10.0.0.1:80"
      ],
      "target_status_code": [
        "500"
      ],
      "traceId": "Root=1-58337262-36d228ad5d99923122bbe354",
      "type": "http"
    }
  },
  "cloud": {
    "provider": "aws"
  },
  "http": {
    "request": {
    ...
  },
  "communication": {
```

```
    "source": {
      "address": "**************",
      "ip": "**************",
      "port": 2817
    }
  },
  "traceId": "Root=1-58337262-36d228ad5d99923122bbe354"
}
```

Este exemplo mostra um log do AWS ELB chegando de um serviço residente em. AWS Ele mostra uma resposta HTTP de backend com um código de status de 500, indicando um erro. Isso pode acionar um alerta ou fazer parte do seu processo regular de monitoramento. Sua próxima etapa é reunir dados relevantes sobre esse evento para uma investigação minuciosa.

Embora você possa ficar tentado a consultar todos os dados relacionados ao período, essa abordagem pode ser esmagadora. Você pode acabar com informações demais, gastando mais tempo filtrando dados irrelevantes do que identificando a causa raiz.

Em vez disso, você pode usar uma abordagem mais objetiva correlacionando dados de diferentes fontes. Você pode usar essas dimensões para correlação:

- IP: `"ip": "10.0.0.1" | "ip": "**************"`
- Porto - `"port": 2817 | "target_port": "10.0.0.1:80"`

Partindo do pressuposto de que você tem acesso a rastreamentos e índices de métricas adicionais e está familiarizado com a estrutura do esquema, pode criar uma consulta de correlação mais precisa.

Aqui está um exemplo de um documento de índice de rastreamento contendo informações HTTP que você talvez queira correlacionar:

```
{
  "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
  "spanId": "55a698828fe06a42",
  "traceState": [],
  "parentSpanId": "",
  "name": "mysql",
  "kind": "CLIENT",
  "@timestamp": "2021-11-13T20:20:39+00:00",
  "events": [
    {
      "@timestamp": "2021-03-25T17:21:03+00:00",
```

```
      ...
    }
  ],
  "links": [
    {
      "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
      "spanId": "55a698828fe06a42w2",
      },
      "droppedAttributesCount": 0
    }
  ],
  "resource": {
    "service@name": "database",
    "telemetry@sdk@name": "opentelemetry",
    "host@hostname": "ip-172-31-10-8.us-west-2.compute.internal"
  },
  "status": {
    ...
  },
  "attributes": {
    "http": {
      "user_agent": {
        "original": "Mozilla/5.0"
      },
      "network": {
         ...
        }
      },
      "request": {
         ...
        }
      },
      "response": {
        "status_code": "200",
        "body": {
          "size": 500
        }
      },
      "client": {
        "server": {
          "socket": {
            "address": "***********",
            "domain": "example.com",
            "port": 80
```

```
          },
          "address": "***********",
          "port": 80
        },
        "resend_count": 0,
        "url": {
          "full": "http://example.com"
        }
      },
      "server": {
        "route": "/index",
        "address": "***********",
        "port": 8080,
        "socket": {
         ...
        },
        "client": {
         ...
         }
        },
        "url": {
         ...
        }
      }
    }
  }
}
```

Nessa abordagem, você pode ver os http `traceId` e os http client/server `ip` que podem ser correlacionados com os registros do elb para entender melhor o comportamento e a condição do sistema.

Novo comando de consulta de correlação

Aqui está o novo comando que permitiria esse tipo de investigação:

```
source alb_logs, traces | where alb_logs.ip="10.0.0.1" AND
 alb_logs.cloud.provider="aws"|
correlate exact fields(traceId, ip) scope(@timestamp, 1D) mapping(alb_logs.ip =
 traces.attributes.http.server.address, alb_logs.traceId = traces.traceId )
```

Aqui está o que cada faz parte do comando:

1. `source alb_logs, traces`: seleciona as fontes de dados que você deseja correlacionar.
2. `where ip="10.0.0.1" AND cloud.provider="aws"`: restringe o escopo da pesquisa.
3. `correlate exact fields(traceId, ip)`: faz com que o sistema correlacione os dados com base em correspondências exatas dos seguintes campos:
   - O campo `ip` tem uma condição de filtro explícita, assim, será usado na correlação de todas as fontes de dados.
   - O campo `traceId` não tem filtro explícito, assim, fará a correspondência dos mesmos traceIDs em todas as fontes de dados.

Os nomes de campo indicam o significado lógico da função no comando de correlação. A condição real de união depende da instrução de mapeamento fornecida.

O termo `exact` significa que as instruções de correlação exigirão que todos os campos correspondam para atender à instrução de consulta.

O termo `approximate` tentará estabelecer a correspondência no melhor cenário possível e não rejeitará linhas com correspondências parciais.

Lidar com diferentes mapeamentos de campos

Nos casos em que o mesmo campo lógico (como `ip`) tem nomes diferentes nas diversas fontes de dados, você precisará informar o mapeamento explícito dos campos de caminho. Para resolver isso, você pode ampliar as condições de correlação para estabelecer a correspondência com nomes de campos diferentes com significados lógicos semelhantes. Veja como isso pode ser feito:

```
alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId
```

Para cada campo que participa da união de correlação, você deve fornecer uma instrução de mapeamento relevante que inclua todas as tabelas a serem unidas por esse comando de correlação.

Exemplo

Neste exemplo, existem duas fontes: `alb_logs, traces`

Existem dois campos: `traceId, ip`

Existem duas instruções de mapeamento: `alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId`

Definir o escopo dos prazos de correlação

Para simplificar o trabalho realizado pelo mecanismo de execução (driver), você pode adicionar a instrução scope. Isso instrui explicitamente a consulta de união sobre o tempo a que ela deve alocar a essa pesquisa.

`scope(@timestamp, 1D)` i

Neste exemplo, o foco do escopo da pesquisa é diário, portanto as correlações que aparecem no mesmo dia são agrupadas. Esse mecanismo de definição de escopo simplifica e permite um melhor controle sobre os resultados, permitindo uma resolução incremental de pesquisa de acordo com suas necessidades.

Drivers compatíveis

O novo comando de correlação é, na verdade, um comando de união “oculto”. Portanto, somente os drivers PPL a seguir são compatíveis com esse comando. Nesses drivers, o comando de correlação será traduzido diretamente para o plano lógico Catalyst Join apropriado.

Exemplo

```
source alb_logs, traces, metrics | where ip="10.0.0.1" AND
cloud.provider="aws"| correlate exact on (ip, port) scope(@timestamp,
2018-07-02T22:23:00, 1 D)
```

Plano lógico:

```
'Project [*]
+- 'Join Inner, ('ip && 'port)
   :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
      +- 'UnresolvedRelation [alb_logs]
   +- 'Join Inner, ('ip & 'port)
      :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
         +- 'UnresolvedRelation [traces]
      +- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
         +- 'UnresolvedRelation [metrics]
```

O mecanismo catalisador otimiza essa consulta de acordo com a ordem de união mais eficiente.

Comando dedup

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando dedup para remover documentos idênticos dos resultados da pesquisa com base nos campos especificados.

Sintaxe

Use a seguinte sintaxe:

```
dedup [int] <field-list> [keepempty=<bool>] [consecutive=<bool>]
```

`int`

- Opcional.
- O comando dedup retém vários eventos para cada combinação quando você especifica <int>. O número para <int>. deve ser maior que 0. Se você não especificar um número, somente o primeiro evento ocorrido será retido. Todas as outras duplicações são removidas dos resultados.
- Padrão: 1

`keepempty`

- Opcional.
- Se verdadeiro, mantém documentos nos quais algum campo na field-list tenha um valor NULL ou esteja AUSENTE.
- Padrão: Falso

`consecutive`

- Opcional.
- Se verdadeiro, remove somente eventos com combinações de valores consecutivas duplicadas.

- Padrão: Falso

`field-list`

- Obrigatório.
- Uma lista de campos delimitada por vírgulas. Pelo menos um campo é obrigatório.

Exemplo 1: Dedup por um campo

Este exemplo mostra como remover duplicações de documentos usando o campo de gênero.

Consulta PPL:

```
os> source=accounts | dedup gender | fields account_number, gender;
fetched rows / total rows = 2/2
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Exemplo 2: mantenha 2 documentos duplicados

O exemplo mostra como remover duplicações de documentos com o campo de gênero, mantendo duas duplicações.

Consulta PPL:

```
os> source=accounts | dedup 2 gender | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

Exemplo 3: manter ou ignorar o campo vazio por padrão

O exemplo mostra como remover duplicações de documentos mantendo o campo de valor nulo.

Consulta PPL:

```
os> source=accounts | dedup email keepempty=true | fields account_number, email;
fetched rows / total rows = 4/4
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 13               | null                  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

O exemplo mostra como remover duplicações de documentos igmorando o campo de valor vazio.

Consulta PPL:

```
os> source=accounts | dedup email | fields account_number, email;
fetched rows / total rows = 3/3
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

Exemplo 4: remover duplicações em documentos consecutivos

O exemplo mostra como remover duplicações em documentos consecutivos.

Consulta PPL:

```
os> source=accounts | dedup gender consecutive=true | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
+------------------+----------+
| 1                | M        |
| 13               | F        |
```

```
| 18               | M        |
+------------------+----------+
```

Exemplos adicionais

- `source = table | dedup a | fields a,b,c`
- `source = table | dedup a,b | fields a,b,c`
- `source = table | dedup a keepempty=true | fields a,b,c`
- `source = table | dedup a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a | fields a,b,c`
- `source = table | dedup 1 a,b | fields a,b,c`
- `source = table | dedup 1 a keepempty=true | fields a,b,c`
- `source = table | dedup 1 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 2 a | fields a,b,c`
- `source = table | dedup 2 a,b | fields a,b,c`
- `source = table | dedup 2 a keepempty=true | fields a,b,c`
- `source = table | dedup 2 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a consecutive=true| fields a,b,c` (a remoção de duplicação consecutiva não é compatível)

Limitação

- Para `| dedup 2 a, b keepempty=false`

```
DataFrameDropColumns('_row_number_)
+- Filter ('_row_number_ <= 2) // allowed duplication = 2
   +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
      +- Filter (isnotnull('a) AND isnotnull('b)) // keepempty=false
         +- Project
            +- UnresolvedRelation
```

- Para `| dedup 2 a, b keepempty=true`

```
Union
:- DataFrameDropColumns('_row_number_)
```

```
:  +- Filter ('_row_number_ <= 2)
:     +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
:        +- Filter (isnotnull('a) AND isnotnull('b))
:           +- Project
:              +- UnresolvedRelation
+- Filter (isnull('a) OR isnull('b))
   +- Project
      +- UnresolvedRelation
```

Comando describe

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `describe` para obter informações detalhadas sobre a estrutura e os metadados de tabelas, esquemas e catálogos. Aqui estão vários exemplos e casos de uso do comando `describe`.

Descrever

- `describe table` Esse comando é igual ao comando SQL `DESCRIBE EXTENDED table`
- `describe schema.table`
- `describe schema.\`table\``
- `describe catalog.schema.table`
- `describe catalog.schema.\`table\``
- `describe \`catalog\`.\`schema\`.\`table\``

Comando eval

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando `eval` avalia a expressão e acrescenta o resultado ao resultado da pesquisa.

Sintaxe

Use a seguinte sintaxe:

```
eval <field>=<expression> ["," <field>=<expression> ]...
```

- `field`: obrigatório. Se o nome do campo não existir, um novo campo será adicionado. Se o nome do campo já existir, ele será substituído.
- `expression`: obrigatório. Qualquer expressão compatível com o sistema.

Exemplo 1: criar o novo campo

Este exemplo mostra como criar um novo campo `doubleAge` para cada documento. O novo `doubleAge` é o resultado da avaliação da idade multiplicado por 2.

Consulta PPL:

```
os> source=accounts | eval doubleAge = age * 2 | fields age, doubleAge ;
fetched rows / total rows = 4/4
+-------+-------------+
| age   | doubleAge   |
|-------+-------------|
| 32    | 64          |
| 36    | 72          |
| 28    | 56          |
| 33    | 66          |
+-------+-------------+
```

Exemplo 2: substituir o campo existente

Este exemplo mostra como substituir o campo de idade existente por idade mais 1.

Consulta PPL:

```
os> source=accounts | eval age = age + 1 | fields age ;
fetched rows / total rows = 4/4
+-------+
| age   |
|-------|
```

```
| 33    |
| 37    |
| 29    |
| 34    |
+-------+
```

Exemplo 3: criar o novo campo com o campo definido em eval

Este exemplo mostra como criar um novo campo `ddAge` com um campo definido no comando eval. O novo campo `ddAge` é o resultado da avaliação de `doubleAge` multiplicado por 2, no qual `doubleAge` é definido no comando eval.

Consulta PPL:

```
os> source=accounts | eval doubleAge = age * 2, ddAge = doubleAge * 2 | fields age,
 doubleAge, ddAge ;
fetched rows / total rows = 4/4
+-------+-------------+---------+
| age   | doubleAge   | ddAge   |
|-------+-------------+---------|
| 32    | 64          | 128     |
| 36    | 72          | 144     |
| 28    | 56          | 112     |
| 33    | 66          | 132     |
+-------+-------------+---------+
```

Pressupostos: a, b, c são campos existentes em `table`

Exemplos adicionais

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1` (campos de saída a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`

- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Avaliação com exemplo de caso:

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Avaliação com outro exemplo de caso:

Pressupostos: a, b, c são campos existentes em `table`

Exemplos adicionais

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1` (campos de saída a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`

- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Avaliação com exemplo de caso:

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Avaliação com outro exemplo de caso:

```
source = table |  where ispresent(a) |
eval status_category =
 case(a >= 200 AND a < 300, 'Success',
  a >= 300 AND a < 400, 'Redirection',
  a >= 400 AND a < 500, 'Client Error',
  a >= 500, 'Server Error'
  else 'Incorrect HTTP status code'
 )
 | stats count() by status_category
```

Limitações

- Não é possível substituir campos existentes. As consultas que tentarem fazer isso gerarão exceções com a mensagem “A referência ‘a’ é ambígua”.

```
- `source = table | eval a = 10 | fields a,b,c`
- `source = table | eval a = a * 2 | stats avg(a)`
- `source = table | eval a = abs(a) | where a > 0`
- `source = table | eval a = signum(a) | where a < 0`
```

Comando eventstats

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `eventstats` para enriquecer os dados do evento com estatísticas resumidas calculadas. Ele opera analisando os campos especificados nos eventos, calculando várias medições estatísticas e acrescentando esses resultados a cada evento original como novos campos.

Principais aspectos de eventstats

1. Ele realiza cálculos em todo o conjunto de resultados ou em todos os grupos definidos.
2. Os eventos originais permanecem intactos, com novos campos adicionados para conter os resultados estatísticos.
3. O comando é particularmente útil para análise comparativa, identificação de valores discrepantes ou fornecimento de contexto adicional a eventos individuais.

Diferença entre stats e eventstats

Os comandos `stats` e `eventstats` são usados para calcular estatísticas, mas eles têm algumas diferenças importantes em como operam e no que produzem.

Formato de saída

- `stats`: produz uma tabela resumida somente com as estatísticas calculadas.

- `eventstats`: adiciona as estatísticas calculadas como novos campos aos eventos existentes, preservando os dados originais.

Retention de eventos

- `stats`: reduz o conjunto de resultados a apenas o resumo estatístico, descartando eventos individuais.
- `eventstats`: retém todos os eventos originais e adiciona novos campos com as estatísticas calculadas.

Casos de uso

- `stats`: Ideal para criar relatórios resumidos ou painéis. Geralmente usado como um comando final para resumir os resultados.
- `eventstats`: útil quando você precisa enriquecer eventos com contexto estatístico para análise ou filtragem adicional. Pode ser usado no meio da pesquisa para adicionar estatísticas que podem ser usadas em comandos subsequentes.

Sintaxe

Use a seguinte sintaxe:

```
eventstats <aggregation>... [by-clause]
```

agregação

- Obrigatório.
- Uma função de agregação.
- O argumento da agregação deve ser um campo.

by-clause

- Opcional.
- Sintaxe: `by [span-expression,] [field,]...`
- A cláusula by pode incluir campos e expressões, como funções escalares e funções de agregação. Você também pode usar a cláusula span para dividir um campo específico em buckets de

intervalos iguais. Em seguida, o comando eventstats realiza a agregação com base nesses buckets de span.

- Padrão: se você não especificar um by clause, o comando eventstats agregará todo o conjunto de resultados.

span-expression

- Opcional, no máximo um.
- Sintaxe: `span(field_expr, interval_expr)`
- A unidade da expressão de intervalo é a unidade natural por padrão. No entanto, para campos do tipo data e hora, é necessário especificar a unidade na expressão de intervalo ao usar unidades de data/hora.

  Por exemplo, para dividir o campo `age` em buckets por 10 anos, use `span(age, 10)`. Para campos baseados em hora, você pode dividir um campo `timestamp` em intervalos de hora usando `span(timestamp, 1h)`.

Unidades de tempo disponíveis

Unidades de intervalo de span
millisegundos (ms)
segundo (s)
minuto (m, diferencia maiúsculas de minúsculas)
hora (h)
dia (d)
semana (w)
mês (M, diferencia maiúsculas de minúsculas)
trimestre (q)
ano (y)

Funções de agregação

COUNT

COUNT retorna uma contagem do número de expr nas linhas recuperadas por uma instrução SELECT.

Para consultas de uso de CloudWatch registros, não COUNT é compatível.

Exemplo:

```
os> source=accounts | eventstats count();
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| account_number | balance  | firstname | lastname | age | gender | address
 | employer   | email                    | city   | state | count() |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
 | AnyCorp    | janedoe@anycorp.com      | Brogan | IL    | 4       |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
 | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 4       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
 | AnyOrg     |                          | Nogal  | VA    | 4       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
 |            | juanli@exampleorg.com    | Orick  | MD    | 4       |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
```

SUM

SUM(expr) retorna a soma de expr.

Exemplo:

```
os> source=accounts | eventstats sum(age) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer   | email                    | city   | state | sum(age) by gender |
```

```
+---------------+----------+-----------+----------+-----+--------
+----------------------+------------+-------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp    | janedoe@anycorp.com      | Brogan | IL    | 101               |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 101               |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg     |                          | Nogal  | VA    | 28                |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |            | juanli@exampleorg.com    | Orick  | MD    | 101               |
+---------------+----------+-----------+----------+-----+--------
+----------------------+------------+-------------------------+--------+-------
+--------------------+
```

AVG

AVG(expr) retorna o valor médio de expr.

Exemplo:

```
os> source=accounts | eventstats avg(age) by gender;
fetched rows / total rows = 4/4
+---------------+----------+-----------+----------+-----+--------
+----------------------+------------+-------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                   | city   | state | avg(age) by gender |
+---------------+----------+-----------+----------+-----+--------
+----------------------+------------+-------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.67             |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.67             |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28.00             |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 33.67             |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+-------------------------+--------+-------
+--------------------+
```

MAX

`MAX(expr)` retorna o valor máximo de expr.

Exemplo

```
os> source=accounts | eventstats max(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                     | city   | state | max(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com       | Brogan | IL    | 36        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                           | Nogal  | VA    | 36        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com     | Orick  | MD    | 36        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

MIN

`MIN(expr)` retorna o valor mínimo de expr.

Exemplo

```
os> source=accounts | eventstats min(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                     | city   | state | min(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

```
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 28        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 28        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 28        |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+--------------------------+--------+-------
+-----------+
```

STDDEV_SAMP

`STDDEV_SAMP(expr)` retorna o desvio padrão da amostra de expr.

Exemplo

```
os> source=accounts | eventstats stddev_samp(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | stddev_samp(age)       |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3.304037933599835      |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3.304037933599835      |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 3.304037933599835      |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3.304037933599835      |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

STDDEV_POP

`STDDEV_POP(expr)` retorna o desvio padrão da população de expr.

Exemplo

```
os> source=accounts | eventstats stddev_pop(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                     | city   | state | stddev_pop(age)        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com       | Brogan | IL    | 2.****************     |
| 6              | 5686     | Mary      | Major    | 36  | M      | *** Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 2.****************     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                           | Nogal  | VA    | 2.****************     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com     | Orick  | MD    | 2.****************     |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

PERCENTILE ou PERCENTILE_APPROX

`PERCENTILE(expr, percent)` ou `PERCENTILE_APPROX(expr, percent)` retorna o percentil aproximado de expr na porcentagem especificada.

percentual

- O número deve ser uma constante entre 0 e 100.

Exemplo

```
os> source=accounts | eventstats percentile(age, 90) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
```

```
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | percentile(age, 90) by
 gender  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 36
       |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36
       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28
       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 36
       |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
```

Exemplo 1: calcular a média, a soma e a contagem de um campo por grupo

O exemplo mostra como calcular a idade média, a soma da idade e a contagem de eventos de todas as contas agrupadas por sexo.

```
os> source=accounts | eventstats avg(age) as avg_age, sum(age) as sum_age, count() as
 count by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+-------+
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | avg_age   | sum_age   |
 count |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+-------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
    | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.666667 | 101       |
 3     |
```

```
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
    | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.666667 | 101       |
 3     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
    | AnyOrg      |                          | Nogal  | VA    | 28.000000 | 28        |
 1     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
    |             | juanli@exampleorg.com    | Orick  | MD    | 33.666667 | 101       |
 3     |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+--------+
```

Exemplo 2: calcular a contagem por um span

O exemplo obtém a contagem de idade pelo intervalo de 10 anos.

```
os> source=accounts | eventstats count(age) by span(age, 10) as age_span
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | age_span |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 1        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

Exemplo 3: calcular a contagem por um sexo e span

O exemplo obtém a contagem de idade pelo intervalo de 5 anos e agrupa por sexo.

```
os> source=accounts | eventstats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+---------------------------+--------+-------
+-----+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                     | city   | state | cnt |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+---------------------------+--------+-------
+-----+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com       | Brogan | IL    | 2   |
| 6              | 5686     | Mary      | Majo     | 36  | M      | 671 Example Street
   | Any Company | hattiebond@anycompany.com | Dante  | TN    | 1   |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                           | Nogal  | VA    | 1   |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com     | Orick  | MD    | 2   |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+-------------+---------------------------+--------+-------
+-----+
```

Usage

- `source = table | eventstats avg(a)`
- `source = table | where a < 50 | eventstats avg(c)`
- `source = table | eventstats max(c) by b`
- `source = table | eventstats count(c) by b | head 5`
- `source = table | eventstats distinct_count(c)`
- `source = table | eventstats stddev_samp(c)`
- `source = table | eventstats stddev_pop(c)`
- `source = table | eventstats percentile(c, 90)`
- `source = table | eventstats percentile_approx(c, 99)`

Agregações com span

- `source = table | eventstats count(a) by span(a, 10) as a_span`

- `source = table | eventstats sum(age) by span(age, 5) as age_span | head 2`
- `source = table | eventstats avg(age) by span(age, 20) as age_span, country | sort - age_span | head 2`

Agregações com span de janela de tempo (função de janela tumble)

- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1d) as age_date | sort age_date`
- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1w) as age_date, productId`

Grupo de agregações por vários níveis

- `source = table | eventstats avg(age) as avg_state_age by country, state | eventstats avg(avg_state_age) as avg_country_age by country`
- `source = table | eventstats avg(age) as avg_city_age by country, state, city | eval new_avg_city_age = avg_city_age - 1 | eventstats avg(new_avg_city_age) as avg_state_age by country, state | where avg_state_age > 18 | eventstats avg(avg_state_age) as avg_adult_country_age by country`

Comando expand

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `expand` para nivelar um campo do tipo:

- `Array<Any>`
- `Map<Any>`

Sintaxe

Use a seguinte sintaxe:

```
expand <field> [As alias]
```

Campo

- O campo a ser expandido (explodido). Deve ser de um tipo compatível.

alias

- Opcional. O nome a ser usado em vez do nome original do campo.

Usage

O comando `expand` produz uma linha para cada elemento na matriz ou no campo de mapa especificado, onde:

- Os elementos da matriz se tornam linhas individuais.
- Os pares de chave-valor do mapa são divididos em linhas separadas, com cada par chave-valor representado como uma linha.

- Quando um alias é fornecido, os valores explodidos são representados sob o alias em vez do nome original do campo.
- Isso pode ser usado em combinação com outros comandos, como `stats`, `eval` e `parse` para manipular ou extrair dados após a expansão.

Exemplos

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`

- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

Comando explain

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando `explain` ajuda a entender os planos de execução de consultas, permitindo que você analise e otimize as consultas para melhorar a performance. Esta introdução fornece uma visão geral concisa da finalidade do comando explain e de sua importância na otimização de consultas.

Comment

- `source=accounts | top gender // finds most common gender of all the accounts` (comentário de linha)
- `source=accounts | dedup 2 gender /* dedup the document with gender field keep 2 duplication */ | fields account_number, gender` (comentários de bloco)

Descrever

- `describe table` Esse comando é igual ao comando SQL `DESCRIBE EXTENDED table`
- `describe schema.table`
- `describe schema.`table``
- `describe catalog.schema.table`
- `describe catalog.schema.`table``
- `describe `catalog`.`schema`.`table``

Explicar

- `explain simple | source = table | where a = 1 | fields a,b,c`

- `explain extended | source = table`
- `explain codegen | source = table | dedup a | fields a,b,c`
- `explain cost | source = table | sort a | fields a,b,c`
- `explain formatted | source = table | fields - a`
- `explain simple | describe table`

Campos

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Resumo de campo

- `source = t | fieldsummary includefields=status_code nulls=false`
- `source = t | fieldsummary includefields= id, status_code, request_path nulls=true`
- `source = t | where status_code != 200 | fieldsummary includefields= status_code nulls=true`

Campo aninhado

- `source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
- `source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
- `source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`

Filtros

- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`
- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a not in (1, 2, 3) | fields a,b,c`
- `source = table | where a between 1 and 4`: nota: isso retorna um >= 1 e um <= 4, ou seja, [1, 4]
- `source = table | where b not between '2024-09-10' and '2025-09-10'`: nota: isso retorna b >= "**********" e b <= "2025-09-10"
- `source = table | where cidrmatch(ip, '**********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`
- `source = table | trendline sma(2, temperature) as temp_trend`

Consultas relacionadas a IP

- `source = table | where cidrmatch(ip, '***************')`
- `source = table | where isV6 = false and isValid = true and cidrmatch(ipAddress, '***************')`
- `source = table | where isV6 = true | eval inRange = case(cidrmatch(ipAddress, '2003:***::/32'), 'in' else 'out') | fields ip, inRange`

Filtros complexos

```
source = table | eval status_category =
```

```
case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code')
| where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code'
) = 'Incorrect HTTP status code'
```

```
source = table
| eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
| where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor = 8,
 'even' else 'odd') = 'even'
| stats count() by factor
```

Filtros com condições lógicas

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`
- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`

Eval

Pressupostos: a, b, c são campos existentes em `table`

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1` (campos de saída a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`

- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval digest = md5(fieldName) | fields digest`
- `source = table | eval digest = sha1(fieldName) | fields digest`
- `source = table | eval digest = sha2(fieldName,256) | fields digest`
- `source = table | eval digest = sha2(fieldName,512) | fields digest`

Comando fillnull

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Description

Use o comando `fillnull` para substituir valores nulo por um valor especificado em um ou mais campos dos resultados da pesquisa.

Sintaxe

Use a seguinte sintaxe:

```
fillnull [with <null-replacement> in <nullable-field>["," <nullable-field>]] | [using
 <source-field> = <null-replacement> ["," <source-field> = <null-replacement>]]
```

- null-replacement: obrigatório. O valor usado para substituir valores nulo.

- nullable-field: obrigatório. Referência do campo. Os valores nulo nesse campo serão substituídos pelo valor especificado em nulo-replacement.

Exemplo 1: fillnull em um campo

O exemplo mostra como usar fillnull em um único campo:

```
os> source=logs | fields status_code | eval input=status_code | fillnull with 0 in
 status_code;
| input | status_code |
|-------|-------------|
| 403   | 403         |
| 403   | 403         |
| NULL  | 0           |
| NULL  | 0           |
| 200   | 200         |
| 404   | 404         |
| 500   | 500         |
| NULL  | 0           |
| 500   | 500         |
| 404   | 404         |
| 200   | 200         |
| 500   | 500         |
| NULL  | 0           |
| NULL  | 0           |
| 404   | 404         |
```

Exemplo 2: fillnull aplicado a vários campos

O exemplo mostra fillnull aplicado a vários campos.

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull with '???' in
 request_path, timestamp;
| input_request_path | input_timestamp       | request_path | timestamp             |
|--------------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | ???                   |
| /home              | NULL                  | /home        | ???                   |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00   |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00   |
| NULL               | 2023-10-01 10:20:00   | ???          | 2023-10-01 10:20:00   |
| NULL               | 2023-10-01 11:05:00   | ???          | 2023-10-01 11:05:00   |
```

```
| /about              | NULL                  | /about       | ???                   |
| /home               | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00   |
| /contact            | NULL                  | /contact     | ???                   |
| NULL                | 2023-10-01 10:05:00   | ???          | 2023-10-01 10:05:00   |
| NULL                | 2023-10-01 10:50:00   | ???          | 2023-10-01 10:50:00   |
| /services           | NULL                  | /services    | ???                   |
| /home               | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00   |
| /services           | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00   |
| NULL                | 2023-10-01 10:35:00   | ???          | 2023-10-01 10:35:00   |
```

Exemplo 3: fillnull aplicado a vários campos com vários valores de substituição de nulo.

O exemplo mostra fillnull com vários valores usados para substituir nulos.

- `/error` no campo `request_path`
- `1970-01-01 00:00:00` no campo `timestamp`

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull using
 request_path = '/error', timestamp='1970-01-01 00:00:00';

| input_request_path | input_timestamp       | request_path | timestamp             |
|--------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00   |
| /home              | NULL                  | /home        | 1970-01-01 00:00:00   |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00   |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00   |
| NULL               | 2023-10-01 10:20:00   | /error       | 2023-10-01 10:20:00   |
| NULL               | 2023-10-01 11:05:00   | /error       | 2023-10-01 11:05:00   |
| /about             | NULL                  | /about       | 1970-01-01 00:00:00   |
| /home              | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00   |
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00   |
| NULL               | 2023-10-01 10:05:00   | /error       | 2023-10-01 10:05:00   |
| NULL               | 2023-10-01 10:50:00   | /error       | 2023-10-01 10:50:00   |
| /services          | NULL                  | /services    | 1970-01-01 00:00:00   |
| /home              | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00   |
| /services          | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00   |
| NULL               | 2023-10-01 10:35:00   | /error       | 2023-10-01 10:35:00   |
```

Comando fields

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `fields` para manter ou remover campos do resultado da pesquisa.

Sintaxe

Use a seguinte sintaxe:

```
field [+|-] <field-list>
```

- `index`: opcional.

  Se o sinal de adição (+) for usado, somente os campos especificados na lista de campos serão mantidos.

  Se o sinal de subtração (+) for usado, somente os campos especificados na lista de campos serão mantidos.

  Padrão: +
- `field list`: obrigatório. Uma lista delimitada por vírgulas dos campos a manter ou remover.

Exemplo 1: selecione campos especificados no resultado

Este exemplo mostra como buscar os campos `account_number`, `firstname` e `lastname` dos resultados da pesquisa.

Consulta PPL:

```
os> source=accounts | fields account_number, firstname, lastname;
fetched rows / total rows = 4/4
+------------------+-------------+------------+
| account_number   | firstname   | lastname   |
|------------------+-------------+------------|
```

```
| 1                | Jane        | Doe        |
| 6                | John        | Doe        |
| 13               | Jorge       | Souza      |
| 18               | Juan        | Li         |
+------------------+-------------+------------+
```

Exemplo 2: remover campos especificados do resultado

Este exemplo mostra como remover o campo `account_number` dos resultados da pesquisa.

Consulta PPL:

```
os> source=accounts | fields account_number, firstname, lastname | fields -
 account_number ;
fetched rows / total rows = 4/4
+-------------+------------+
| firstname   | lastname   |
|-------------+------------|
| Jane        | Doe        |
| John        | Doe        |
| Jorge       | Souza      |
| Juan        | Li         |
+-------------+------------+
```

Exemplos adicionais

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Exemplo de campos aninhados:

```
`source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
`source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col |
 fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
`source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort
 int_col | fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
```

Comando flatten

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando flatten para expandir campos dos seguintes tipos:

- `struct<?,?>`
- `array<struct<?,?>>`

Sintaxe

Use a seguinte sintaxe:

```
flatten <field>
```

- campo: o campo a ser nivelado. O campo deve ser de um tipo compatível.

Esquema

col_name	data_type
_time	string
pontes	array<struct<length:bigint,name:string>>
cidade	string
coor	struct<alt:bigint,lat:double,long:double>
country	string

Dados

_time	pontes	cidade	coor	país
2024-09-13T12:00:00	[{801, Tower Bridge}, {928, London Bridge}]	Londres	{35, 51,5074, -0,1278}	Inglaterra
2024-09-13T12:00:00	[{232, Pont Neuf}, {160, Pont Alexandre III}]	Paris	{35, 48,8566, 2,3522}	França
2024-09-13T12:00:00	[{48, Ponte Rialto}, {11, Ponte dos Suspiros}]	Veneza	{2, 45,4408, 12,3155}	Itália
2024-09-13T12:00:00	[{***, Ponte Carlos}, {343, Ponte da Legião}]	Praga	{200, 50,0755, 14,4378}	República Tcheca
2024-09-13T12:00:00	[{375, Ponte Chain}, {333, Ponte da Liberdade}]	Budapeste	{96, 47,4979, 19,0402}	Hungria
1990-09-13T12:00:00	NULL	Varsóvia	NULL	Polônia

Exemplo 1: nivelar campo struct

Este exemplo mostra como nivelar um campo struct.

Consulta PPL:

```
source=table | flatten coor
```

_time	pontes	cidade	país	alt	lat	longo
2024-09-13T12:00:00	[{801, Tower Bridge}, {928, London Bridge}]	Londres	Inglaterra	35	51.5074	-0,1278
2024-09-13T12:00:00	[{232, Pont Neuf}, {160, Pont Alexandre III}]	Paris	França	35	48.856	2.352
2024-09-13T12:00:00	[{48, Ponte Rialto}, {11, Ponte dos Suspiros}]	Veneza	Itália	2	45.4408	12.315
2024-09-13T12:00:00	[{516, Ponte Carlos}, {343, Ponte da Legião}]	Praga	República Tcheca	200	50.075	14.4378

_time	pontes	cidade	país	alt	lat	longo
2024-09-13T12:00:00	[{375, Ponte Chain}, {333, Ponte da Liberdade}]	Budapeste	Hungria	96	47.4979	19.04.02
1990-09-13T 12:00:00	NULL	Varsóvia	Polônia	NULL	NULL	NULL

Exemplo 2: nivelar matriz

O exemplo mostra como nivelar uma matriz de campos struct.

Consulta PPL:

```
source=table | flatten bridges
```

_time	cidade	coor	país	length	name
2024-09-13T12:00:00	Londres	{35, 51,5074, -0,1278}	Inglaterra	801	Ponte da Torre
2024-09-13T12:00:00	Londres	{35, 51,5074, -0,1278}	Inglaterra	928	Ponte de Londres
2024-09-13T12:00:00	Paris	{35, 48,8566, 2,3522}	França	232	Pont Neuf
2024-09-13T12:00:00	Paris	{35, 48,8566, 2,3522}	França	160	Ponte Alexandre III

_time	cidade	coor	país	length	name
2024-09-13T12:00:00	Veneza	{2, 45,4408, 12,3155}	Itália	48	Ponte de Rialto
2024-09-13T12:00:00	Veneza	{2, 45,4408, 12,3155}	Itália	11	Ponte dos Suspiros
2024-09-13T12:00:00	Praga	{200, 50,0755, 14,4378}	República Tcheca	516	Ponte Carlos
2024-09-13T12:00:00	Praga	{200, 50,0755, 14,4378}	República Tcheca	343	Ponte da Legião
2024-09-13T12:00:00	Budapeste	{96, 47,4979, 19,0402}	Hungria	375	Ponte das Correntes
2024-09-13T12:00:00	Budapeste	{96, 47,4979, 19,0402}	Hungria	333	Ponte da Liberdade
1990-09-13T12:00:00	Varsóvia	NULL	Polônia	NULL	NULL

Exemplo 3: nivelar matriz e struct

Este exemplo mostra como nivelar vários campos.

Consulta PPL:

```
source=table | flatten bridges | flatten coor
```

_time	cidade	país	length	name	alt	lat	longo
2024-09-13T12:00:00	Londres	Inglaterra	801	Ponte da Torre	35	51.5074	-0,1278

_time	cidade	país	length	name	alt	lat	longo
2024-09-13T12:00:00	Londres	Inglaterra	928	Ponte de Londres	35	51.5074	-0,1278
2024-09-13T12:00:00	Paris	França	232	Pont Neuf	35	48.856	2.352
2024-09-13T12:00:00	Paris	França	160	Ponte Alexandre III	35	48.856	2.352
2024-09-13T12:00:00	Veneza	Itália	48	Ponte de Rialto	2	45.4408	12.315
2024-09-13T12:00:00	Veneza	Itália	11	Ponte dos Suspiros	2	45.4408	12.315
2024-09-13T12:00:00	Praga	República Tcheca	516	Ponte Carlos	200	50.075	14.4378
2024-09-13T12:00:00	Praga	República Tcheca	343	Ponte da Legião	200	50.075	14.4378
2024-09-13T12:00:00	Budape	Hungria	375	Ponte das Correntes	96	47.4979	19.04.02
2024-09-13T12:00:00	Budape	Hungria	333	Ponte da Liberdade	96	47.4979	19.04.02

_time	cidade	país	length	name	alt	lat	longo
1990-09-13T 12:00:00	Varsóvi	Polônia	NULL	NULL	NULL	NULL	NULL

Comando grok

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando `grok` analisa um campo de texto com um padrão grok e acrescenta os resultados ao resultado da pesquisa.

Sintaxe

Use a seguinte sintaxe:

```
grok <field> <pattern>
```

Campo

- Obrigatório.
- O campo deve ser um campo de texto.

pattern

- Obrigatório.
- O padrão grok usado para extrair novos campos do campo de texto fornecido.
- Se um novo nome de campo já existir, ele substituirá o campo original.

Padrão grok

O padrão grok é usado para combinar o campo de texto de cada documento para extrair novos campos.

Exemplo 1: criar o novo campo

Este exemplo mostra como criar um novo campo `host` para cada documento. `host` será o nome do host depois de `@` no campo `email`. A análise de um campo nulo retornará uma string vazia.

```
os> source=accounts | grok email '.+@%{HOSTNAME:host}' | fields email, host ;
fetched rows / total rows = 4/4
+-------------------------+-------------+
| email                   | host        |
|-------------------------+-------------|
| jane_doe@example.com    | example.com |
| arnav_desai@example.net | example.net |
| null                    |             |
| juan_li@example.org     | example.org |
+-------------------------+-------------+
```

Exemplo 2: substituir o campo existente

Este exemplo mostra como substituir o campo `address` existente pelo número da rua removido.

```
os> source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address ;
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Any Street       |
| Main Street      |
| Example Court    |
+------------------+
```

Exemplo 3: usar grok para analisar logs

Este exemplo mostra como usar grok para analisar logs brutos.

```
os> source=apache | grok message '%{COMMONAPACHELOG}' | fields COMMONAPACHELOG,
 timestamp, response, bytes ;
fetched rows / total rows = 4/4
```

```
+-----------------------------------------------------------------------------------------------------------
+----------------------------+-----------+---------+
| COMMONAPACHELOG
                                          | timestamp                  | response   |
 bytes   |
|-----------------------------------------------------------------------------------------------------------
+----------------------------+-----------+---------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                       | 28/Sep/2022:10:15:57 -0700 | 404        |
 19927   |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | 28/Sep/2022:10:15:57 -0700 | 100
    | 28722   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | 28/Sep/2022:10:15:57 -0700 | 401        |
 27439   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                         | 28/Sep/2022:10:15:57 -0700 | 301        | 9481
    |
+-----------------------------------------------------------------------------------------------------------
+----------------------------+-----------+---------+
```

Limitações

O comando grok tem as mesmas limitações que o comando parse.

Comando head

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `head` para retornar o primeiro N número de resultados especificados após um deslocamento opcional na ordem de pesquisa.

Sintaxe

Use a seguinte sintaxe:

```
head [<size>] [from <offset>]
```

<size>

- Opcional inteiro.
- o número máximo de resultados a serem retornados.
- Padrão: 10

<offset>

- Inteiro após `from` opcional.
- O número de resultados a serem ignoradps.
- Padrão: 0

Exemplo 1: obter os 10 primeiros resultados

Este exemplo mostra como recuperar no máximo 10 resultados do índice de contas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head;
fetched rows / total rows = 4/4
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

Exemplo 2: obter os N primeiros resultados

O exemplo mostra os primeiros N resultados do índice de contas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head 3;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
```

```
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
+-------------+-------+
```

Exemplo 3: obter os primeiros N resultados após o deslocamento M

Este exemplo mostra como recuperar os primeiros N resultados depois de ignorar M resultados do índice de contas.

Consulta PPL:

```
os> source=accounts | fields firstname, age | head 3 from 1;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

Comando join

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando join permite combinar dados de várias fontes com base em campos comuns, permitindo que você realize análises complexas e obtenha insights mais profundos de seus conjuntos de dados distribuídos

Schema

Existem pelo menos dois índices, `otel-v1-apm-span-*` (grande) e `otel-v1-apm-service-map` (pequeno).

Campos relevantes de índices:

`otel-v1-apm-span-*`

- traceId: um identificador exclusivo para um rastreamento. Todos os spans do mesmo rastreamento compartilham o mesmo traceid.
- spanId: um identificador exclusivo para um span dentro de um rastreamento, atribuído quando o span é criado.
- parentSpanId - O SpanID do intervalo principal desse período. Se for um span raiz, esse campo deverá ficar vazio.
- durationInNanos - A diferença em nanossegundos entre StartTime e EndTime. (isso está `latency` na interface do usuário)
- serviceName: o recurso do qual o span se origina.
- traceGroup: o nome do span raiz do rastreamento.

`otel-v1-apm-service-map`

- serviceName: o nome do serviço que emitiu o span.
- destination.domain: o serviceName do serviço chamado por esse cliente.
- destination.resource: o nome do span (API, operação etc.) chamado por esse cliente.
- target.domain: o serviceName do serviço sendo chamado por um cliente.
- target.resource: o nome do span (API, operação etc.) chamado por esse cliente.
- traceGroupName - O nome do intervalo de nível superior que iniciou a cadeia de solicitações.

Requisito

Compatibilidade com join para calcular o seguinte:

Para cada serviço, junte o índice de span no índice de mapa do serviço para calcular métricas em diferentes tipos de filtro.

Esse exemplo de consulta calcula a latência quando filtrada pelo grupo de rastreamento `client_cancel_order` para o serviço `order`.

```
SELECT avg(durationInNanos)
FROM `otel-v1-apm-span-000001` t1
WHERE t1.serviceName = `order`
  AND ((t1.name in
```

```
          (SELECT target.resource
           FROM `otel-v1-apm-service-map`
           WHERE serviceName = `order`
             AND traceGroupName = `client_cancel_order`)
        AND t1.parentSpanId != NULL)
       OR (t1.parentSpanId = NULL
           AND t1.name = `client_cancel_order`))
  AND t1.traceId in
    (SELECT traceId
     FROM `otel-v1-apm-span-000001`
     WHERE serviceName = `order`)
```

Migrar para PPL

Sintaxe do comando join

```
SEARCH source=<left-table>
| <other piped command>
| [joinType] JOIN
    [leftAlias]
    ON joinCriteria
    <right-table>
| <other piped command>
```

Reescrever

```
SEARCH source=otel-v1-apm-span-000001
| WHERE serviceName = 'order'
| JOIN left=t1 right=t2
    ON t1.traceId = t2.traceId AND t2.serviceName = 'order'
    otel-v1-apm-span-000001 -- self inner join
| EVAL s_name = t1.name -- rename to avoid ambiguous
| EVAL s_parentSpanId = t1.parentSpanId -- RENAME command would be better when it is
 supported
| EVAL s_durationInNanos = t1.durationInNanos
| FIELDS s_name, s_parentSpanId, s_durationInNanos -- reduce colunms in join
| LEFT JOIN left=s1 right=t3
    ON s_name = t3.target.resource AND t3.serviceName = 'order' AND t3.traceGroupName =
 'client_cancel_order'
    otel-v1-apm-service-map
| WHERE (s_parentSpanId IS NOT NULL OR (s_parentSpanId IS NULL AND s_name =
 'client_cancel_order'))
```

```
| STATS avg(s_durationInNanos) -- no need to add alias if there is no ambiguous
```

joinType

- Sintaxe: `INNER | LEFT OUTER | CROSS`
- Opcional
- O tipo de união a ser realizada. Se não especificada, o padrão será `INNER`.

leftAlias

- Sintaxe: `left = <leftAlias>`
- Opcional
- O alias da subconsulta a ser usado com o left join side, para evitar nomenclaturas ambíguas.

joinCriteria

- Sintaxe: `<expression>`
- Obrigatório
- A sintaxe começa com `ON`. Pode ser qualquer expressão de comparação. Geralmente, o critério de união é assim: `<leftAlias>.<leftField>=<rightAlias>.<rightField>`.

  Por exemplo: `l.id = r.id`. Se o critério de união contiver várias condições, você poderá especificar o operador `AND` e `OR` entre cada expressão de comparação. Por exemplo, `.l.id = r.id AND l.email = r.email AND (r.age > 65 OR r.age < 18)`

Mais exemplos de

Migração de consulta SQL (TPC-H Q13):

```
SELECT c_count, COUNT(*) AS custdist
FROM
  ( SELECT c_custkey, COUNT(o_orderkey) c_count
    FROM customer LEFT OUTER JOIN orders ON c_custkey = o_custkey
        AND o_comment NOT LIKE '%unusual%packages%'
    GROUP BY c_custkey
  ) AS c_orders
GROUP BY c_count
```

```
ORDER BY custdist DESC, c_count DESC;
```

Reescrito pela consulta PPL join:

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN
    ON c_custkey = o_custkey AND o_comment NOT LIKE '%unusual%packages%'
    orders
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

Limitação: join right side não permite subpesquisas.

Se forem permitidas subpesquisas, você poderá reescrever a consulta PPL acima da seguinte forma:

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN
   ON c_custkey = o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

Comando lookup

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `lookup` para enriquecer os dados de pesquisa adicionando ou substituindo dados de um índice de pesquisa (tabela de dimensões). Esse comando permite estender campos de um índice com os valores de uma tabela de dimensões. Você também pode usá-lo para acrescentar ou

substituir valores quando as condições de pesquisa forem atendidas. O comando `lookup` é mais adequado do que o comando `Join` para enriquecer os dados da fonte com um conjunto de dados estático.

Sintaxe

Use a seguinte sintaxe:

```
SEARCH source=<sourceIndex>
| <other piped command>
| LOOKUP <lookupIndex> (<lookupMappingField> [AS <sourceMappingField>])...
    [(REPLACE | APPEND) (<inputField> [AS <outputField>])...]
| <other piped command>
```

lookupIndex

- Obrigatório.
- O nome do índice de pesquisa (tabela de dimensões).

lookupMappingField

- Obrigatório.
- Uma chave de mapeamento no índice de pesquisa, análoga a uma chave de união da tabela da direita. É possível especificar vários campos separados por vírgulas.

sourceMappingField

- Opcional.
- Padrão: < lookupMappingField >.
- Uma chave de mapeamento da consulta de origem, análoga a uma chave de união do lado esquerdo.

inputField

- Opcional.
- Padrão: todos os campos do índice de pesquisa nos quais valores correspondentes são encontrados.

- Um campo no índice de pesquisa nos quais valores correspondentes são aplicados à saída do resultado. É possível especificar vários campos separados por vírgulas.

outputField

- Opcional.
- Padrão: `<inputField>`.
- Um campo na saída. Você pode especificar vários campos de saída. Se você especificar um nome de campo existente na consulta de origem, seus valores serão substituídos ou acrescidos dos valores correspondentes de inputField. Se você especificar um novo nome de campo, ele será adicionado aos resultados.

REPLACE | APPEND

- Opcional.
- Padrão: REPLACE
- Especifica como lidar com valores correspondentes. Se você especificar REPLACE, os valores correspondentes no campo **<lookupIndex>** substituirão os valores no resultado. Se você especificar `APPEND`, os valores correspondentes no campo **<lookupIndex>** serão acrescentados apenas aos valores que faltam no resultado.

Usage

- LOOKUP **<lookupIndex>** id AS cid REPLACE mail AS email
- LOOKUP **<lookupIndex>** name REPLACE mail AS email
- LOOKUP **<lookupIndex>** id AS cid, name APPEND address, mail AS email
- LOOKUP **<lookupIndex>** id

Exemplo

Veja os exemplos de a seguir.

```
SEARCH source=<sourceIndex>
| WHERE orderType = 'Cancelled'
| LOOKUP account_list, mkt_id AS mkt_code REPLACE amount, account_name AS name
```

```
| STATS count(mkt_code), avg(amount) BY name
```

```
SEARCH source=<sourceIndex>
| DEDUP market_id
| EVAL category=replace(category, "-", ".")
| EVAL category=ltrim(category, "dvp.")
| LOOKUP bounce_category category AS category APPEND classification
```

```
SEARCH source=<sourceIndex>
| LOOKUP bounce_category category
```

comando parse

O comando `parse` analisa um campo de texto com uma expressão regular e acrescenta o resultado ao resultado da pesquisa.

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Sintaxe

Use a seguinte sintaxe:

```
parse <field> <pattern>
```

`field`

- Obrigatório.
- O campo deve ser um campo de texto.

`pattern`

- String obrigatória.
- Esse é um padrão de expressão regular usado para extrair novos campos do campo de texto fornecido.

- Se um novo nome de campo já existir, ele substituirá o campo original.

Expressão regular

O padrão de expressão regular é usado para combinar todo o campo de texto de cada documento com o mecanismo Java regex. Cada grupo de captura nomeado na expressão se tornará um novo campo `STRING`.

Exemplo 1: criar um novo campo

O exemplo mostra como criar um novo campo `host` para cada documento. `host` será o nome do host depois de `@` no campo `email`. A análise de um campo nulo retornará uma string vazia.

Consulta PPL:

```
os> source=accounts | parse email '.+@(?<host>.+)' | fields email, host ;
fetched rows / total rows = 4/4
+----------------------+-------------+
| email                | host        |
|----------------------+-------------|
| jane_doe@example.com  | example.com |
| john_doe@example.net  | example.net |
| null                 |             |
| juan_li@example.org   | example.org |
+----------------------+-------------+
```

Exemplo 2: substituir um campo existente

O exemplo mostra como substituir o campo `address` existente pelo número da rua removido.

Consulta PPL:

```
os> source=accounts | parse address '\d+ (?<address>.+)' | fields address ;
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Example Street   |
| Example Avenue   |
| Example Court    |
+------------------+
```

Exemplo 3: filtrar e classificar por campo analisado convertido

O exemplo mostra como classificar números de ruas acima de 500 no campo `address`.

Consulta PPL:

```
os> source=accounts | parse address '(?<streetNumber>\d+) (?<street>.+)' | where
 cast(streetNumber as int) > 500 | sort num(streetNumber) | fields streetNumber,
 street ;
fetched rows / total rows = 3/3
+----------------+----------------+
| streetNumber   | street         |
|----------------+----------------|
| ***            | Example Street |
| ***            | Example Avenue |
| 880            | Example Lane   |
+----------------+----------------+
```

Limitações

Há algumas limitações com o comando parse:

- Campos definidos por parse não podem ser analisados novamente.

  O seguinte comando não funcionará:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | parse street '\w+ (?<road>\w+)'
  ```

- Os campos definidos por parse não podem ser substituídos por outros comandos.

  `where` não corresponderá a nenhum documento, pois `street` não pode ser substituído:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval street='1' | where
   street='1' ;
  ```

- O campo de texto usado por parse não pode ser substituído.

  `street` não será analisado com sucesso, pois foi `address` substituído:

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval address='1' ;
  ```

- Os campos definidos por parse não podem ser filtrados nem classificados após serem usados no comando `stats`.

`where` no seguinte comando não funcionará:

```
source=accounts | parse email '.+@(?<host>.+)' | stats avg(age) by host | where
 host=pyrami.com ;
```

Comando patterns

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando `patterns` extrai padrões de log de um campo de texto e acrescenta os resultados ao resultado da pesquisa. Agrupar os logs por seus padrões facilita a agregação de estatísticas de grandes volumes de dados de log para análise e solução de problemas.

Sintaxe

Use a seguinte sintaxe:

```
patterns [new_field=<new-field-name>] [pattern=<pattern>] <field>
```

new-field-name

- String opcional.
- Esse é o nome do novo campo para padrões extraídos.
- O padrão é `patterns_field`.
- Se o nome já existir, ele substituirá o campo original.

pattern

- String opcional.
- Esse é o padrão regex de caracteres que devem ser filtrados para serem excluídos do campo de texto.
- Se ausente, o padrão são os caracteres alfanuméricos (`[a-zA-Z\d]`).

Campo

- Obrigatório.
- O campo deve ser um campo de texto.

Exemplo 1: criar o novo campo

O exemplo mostra como usar pontuações de extração em `email` para cada documento. A análise de um campo nulo retornará uma string vazia.

Consulta PPL:

```
os> source=accounts | patterns email | fields email, patterns_field ;
fetched rows / total rows = 4/4
+----------------------+------------------+
| email                | patterns_field   |
|----------------------+------------------|
| jane_doe@example.com | @.               |
| john_doe@example.net | @.               |
| null                 |                  |
| juan_li@example.org  | @.               |
+----------------------+------------------+
```

Exemplo 2: extrair padrões de log

O exemplo mostra como extrair pontuações de um campo de log bruto usando os padrões padrão.

Consulta PPL:

```
os> source=apache | patterns message | fields message, patterns_field ;
fetched rows / total rows = 4/4
+-------------------------------------------------------------------------------------------------------
+---------------------------------+
| message
                                       | patterns_field                |
|-------------------------------------------------------------------------------------------------------
+---------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                         | ... -  [//::: -] " /-/ /."      |
| ************ - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... -  [//::: -] " //// /."     |
```

```
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | ... - - [//::: -] " //--- /."   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                        | ... - - [//::: -] " / /."       |
+-----------------------------------------------------------------------------------------
+--------------------------------+
```

Exemplo 3: extrair padrões de log com padrão regex personalizado

O exemplo mostra como extrair pontuações de um campo de log bruto usando os padrões definidos pelo usuário.

Consulta PPL:

```
os> source=apache | patterns new_field='no_numbers' pattern='[0-9]' message | fields
 message, no_numbers ;
fetched rows / total rows = 4/4
+-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------
+
| message
                                        | no_numbers
                                        |
|-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                         | ... - upton [/Sep/::: -] "HEAD /e-
business/mindshare HTTP/."                           |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... - pouros [/Sep/::: -] "GET /
architectures/convergence/niches/mindshare HTTP/."   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | ... - - [/Sep/::: -] "PATCH /strategize/
out-of-the-box HTTP/."                         |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                        | ... - - [/Sep/::: -] "POST /users HTTP/."
                                        |
+-----------------------------------------------------------------------------------------
+-------------------------------------------------------------------------------
+
```

Limitação

O comando patterns tem as mesmas limitações que o comando parse.

Comando rare

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `rare` para encontrar a tupla de valores menos comum de todos os campos na lista de campos.

> **Note**
>
> Um máximo de 10 resultados é retornado para cada tupla distinta de valores dos campos group-by.

Sintaxe

Use a seguinte sintaxe:

```
rare [N] <field-list> [by-clause] rare_approx [N] <field-list> [by-clause]
```

field-list

- Obrigatório.
- Uma lista de nomes de campos delimitada por vírgulas.

by-clause

- Opcional.
- Um ou mais campos pelos quais agrupar os resultados.

N

- o número máximo de resultados a serem retornados.

- Padrão: 10

rare_approx

- A contagem aproximada dos campos raros (n) usando a cardinalidade estimada pelo HyperLogLog algoritmo ++.

Exemplo 1: encontre os valores menos comuns em um campo

O exemplo encontra o sexo menos comum de todas as contas.

Consulta PPL:

```
os> source=accounts | rare gender;
os> source=accounts | rare_approx 10 gender;
os> source=accounts | rare_approx gender;
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| F        |
| M        |
+----------+
```

Exemplo 2: encontre os valores menos comuns organizados por sexo

O exemplo encontra o sexo menos comum de todas as contas agrupadas por sexo.

Consulta PPL:

```
os> source=accounts | rare 5 age by gender;
os> source=accounts | rare_approx 5 age by gender;
fetched rows / total rows = 4/4
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
| M        | 33    |
| M        | 36    |
+----------+-------+
```

Comando rename

Use o comando `rename` para alterar os nomes de um ou mais campos no resultado da pesquisa.

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Sintaxe

Use a seguinte sintaxe:

```
rename <source-field> AS <target-field>["," <source-field> AS <target-field>]...
```

source-field

- Obrigatório.
- Esse é o nome do campo que você deseja renomear.

target-field

- Obrigatório.
- Esse é o nome com o qual você deseja renomear o campo.

Exemplo 1: renomear um campo

Este exemplo mostra como renomear um único campo.

Consulta PPL:

```
os> source=accounts | rename account_number as an | fields an;
fetched rows / total rows = 4/4
+------+
| an   |
|------|
| 1    |
| 6    |
```

```
| 13   |
| 18   |
+------+
```

Exemplo 2: renomear vários campos

Este exemplo mostra como renomear vários campos.

Consulta PPL:

```
os> source=accounts | rename account_number as an, employer as emp | fields an, emp;
fetched rows / total rows = 4/4
+------+---------+
| an   | emp     |
|------+---------|
| 1    | Pyrami  |
| 6    | Netagy  |
| 13   | Quility |
| 18   | null    |
+------+---------+
```

Limitações

- Não é possível substituir campo existente:

```
source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address
```

Comando search

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `search` para recuperar documentos de um índice. O comando `search` apenas pode ser usado como o primeiro comando em uma consulta PPL.

Sintaxe

Use a seguinte sintaxe:

```
search source=[<remote-cluster>:]<index> [boolean-expression]
```

pesquisar

- Opcional.
- Palavras-chave de pesquisa, que podem ser omitidas.

índice

- Obrigatório.
- O comando search deve especificar em qual índice consultar.
- O nome do índice pode ser prefixado por `<cluster name>:` para pesquisas entre clusters.

bool-expression

- Opcional.
- Qualquer expressão que seja avaliada como um valor booleano.

Exemplo 1: buscar todos os dados

O exemplo mostra a busca de todo o documento do índice de contas.

Consulta PPL:

```
os> source=accounts;
+------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------+
| account_number   | firstname   | address              | balance   | gender   | city
 | employer       | state   | age   | email                 | lastname   |
|------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------|
| 1                | Jorge       | *** Any Lane         | 39225     | M        | Brogan
 | ExampleCorp    | IL      | 32    | jane_doe@example.com  | Souza      |
| 6                | John        | *** Example Street   | 5686      | M        | Dante
 | AnyCorp        | TN      | 36    | john_doe@example.com  | Doe        |
| 13               | Jane        | *** Any Street       | *****     | F        | Nogal
 | ExampleCompany | VA      | 28    | null                  | Doe        |
```

```
| 18               | Juan        | *** Example Court    | 4180      | M        | Orick
 | null           | MD      | 33    | juan_li@example.org   | Li         |
+------------------+-------------+----------------------+-----------+----------
+--------+---------------+---------+-------+----------------------+------------+
```

Exemplo 2: buscar dados com condição

O exemplo mostra a busca de todo o documento do índice de contas com uma condição.

Consulta PPL:

```
os> SEARCH source=accounts account_number=1 or gender="F";
+------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------+
| account_number   | firstname   | address            | balance   | gender   | city   |
 employer       | state   | age   | email                 -  | lastname   |
|------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------|
| 1                | Jorge       | *** Any Lane       | *****     | M        | Brogan |
 ExampleCorp    | IL      | 32    | jorge_souza@example.com | Souza      |
| 13               | Jane        | *** Any Street     | *****     | F        | Nogal  |
 ExampleCompany | VA      | 28    | null                    | Doe        |
+------------------+-------------+--------------------+-----------+----------+--------
+-----------------+---------+-------+-------------------------+------------+
```

Comando sort

Use o comando `sort` para classificar o resultado da pesquisa por campos especificados.

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Sintaxe

Use a seguinte sintaxe:

```
sort <[+|-] sort-field>...
```

[+|-]

- Opcional.
- O sinal de mais [+] significa ordem crescente com NULL/MISSING os valores primeiro.
- O sinal de menos [-] representa a ordem decrescente com NULL/MISSING os valores por último.
- Padrão: ordem crescente com NULL/MISSING os valores primeiro.

sort-field

- Obrigatório.
- O campo usado para classificação.

Exemplo 1: classificar por um campo

O exemplo mostra como classificar o documento com o campo de idade em ordem crescente.

Consulta PPL:

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Exemplo 2: classificar por um único campo e retornar todos os resultados

O exemplo mostra como classificar o documento com o campo de idade em ordem crescente.

Consulta PPL:

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
```

```
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Exemplo 3: classificar por um único campo em ordem decrescente

O exemplo mostra como classificar o documento com o campo de idade em ordem decrescente.

Consulta PPL:

```
os> source=accounts | sort - age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 6                | 36    |
| 18               | 33    |
| 1                | 32    |
| 13               | 28    |
+------------------+-------+
```

Exemplo 4: classificar por vários campos

O exemplo mostra como classificar o documento com o campo de sexo em ordem crescente e o campo de idade em ordem decrescente.

Consulta PPL:

```
os> source=accounts | sort + gender, - age | fields account_number, gender, age;
fetched rows / total rows = 4/4
+------------------+----------+-------+
| account_number   | gender   | age   |
|------------------+----------+-------|
| 13               | F        | 28    |
| 6                | M        | 36    |
| 18               | M        | 33    |
| 1                | M        | 32    |
+------------------+----------+-------+
```

Exemplo 5: classificar por campo e incluir valor nulo

O exemplo mostra como classificar o campo do empregador pela opção padrão (ordem crescente e nulo primeiro). O resultado mostra que o valor nulo está na primeira linha.

Consulta PPL:

```
os> source=accounts | sort employer | fields employer;
fetched rows / total rows = 4/4
+------------+
| employer   |
|------------|
| null       |
| AnyCompany |
| AnyCorp    |
| AnyOrgty   |
+------------+
```

Comando stats

Use o comando `stats` para calcular a agregação a partir do resultado da pesquisa.

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Manipulação de valores NULL/MISSING

Manipulação de valores NULL/MISSING

Função	NULL	MISSING (AUSENTE)
CONTAGEM	Não contado	Não contado
SUM	Ignorar	Ignorar
AVG	Ignorar	Ignorar
MAX	Ignorar	Ignorar

Função	NULL	MISSING (AUSENTE)
MIN	Ignorar	Ignorar

Sintaxe

Use a seguinte sintaxe:

```
stats <aggregation>... [by-clause]
```

agregação

- Obrigatório.
- Uma função de agregação aplicada a um campo.

by-clause

- Opcional.
- Sintaxe: `by [span-expression,] [field,]...`
- Especifica campos e expressões para agrupar os resultados da agregação. A by-clause permite agrupar os resultados da agregação usando campos e expressões. Você pode usar funções escalares, funções de agregação e até mesmo expressões de span para dividir campos específicos em buckets de intervalos iguais.
- Padrão: se `<by-clause>` não for especificado, o comando stats retornará uma única linha representando a agregação para todo o conjunto de resultados.

span-expression

- Opcional, no máximo um.
- Sintaxe: `span(field_expr, interval_expr)`
- A unidade da expressão de intervalo é a unidade natural por padrão. Se o campo for do tipo data e hora e o intervalo estiver em date/time unidades, você especifica a unidade na expressão do intervalo.
- Por exemplo, a divisão do campo `age` em buckets por 10 anos, fica assim `span(age, 10)`. Para dividir um campo de timestamp em intervalos de uma hora, use `span(timestamp, 1h)`.

Unidades de tempo disponíveis

Unidades de intervalo de span
millisegundos (ms)
segundo (s)
minuto (m, diferencia maiúsculas de minúsculas)
hora (h)
dia (d)
semana (w)
mês (M, diferencia maiúsculas de minúsculas)
trimestre (q)
ano (y)

Funções de agregação

COUNT

Retorna uma contagem do número de expr nas linhas recuperadas por uma instrução SELECT.

Exemplo:

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
| count()   |
|-----------|
| 4         |
+-----------+
```

SUM

Use `SUM(expr)` para retornar a soma de expr.

Exemplo

```
os> source=accounts | stats sum(age) by gender;
fetched rows / total rows = 2/2
+------------+----------+
| sum(age)   | gender   |
|------------+----------|
| 28         | F        |
| 101        | M        |
+------------+----------+
```

AVG

Use `AVG(expr)` para retornar o valor médio de expr.

Exemplo

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

MAX

Use `MAX(expr)` para retornar o valor máximo de expr.

Exemplo

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
+------------+
```

MIN

Use `MIN(expr)` para retornar o valor mínimo de expr.

Exemplo

```
os> source=accounts | stats min(age);
fetched rows / total rows = 1/1
+------------+
| min(age)   |
|------------|
| 28         |
+------------+
```

STDDEV_SAMP

Use `STDDEV_SAMP(expr)` para retornar o desvio padrão da amostra de expr.

Exemplo:

```
os> source=accounts | stats stddev_samp(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_samp(age)   |
|--------------------|
| 3.304037933599835  |
+--------------------+
```

STDDEV_POP

Use `STDDEV_POP(expr)` para retornar o desvio padrão da população de expr.

Exemplo:

```
os> source=accounts | stats stddev_pop(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_pop(age)    |
|--------------------|
| 2.**************** |
+--------------------+
```

TAKE

Use `TAKE(field [, size])` para retornar os valores originais de um campo. Não garante a ordem dos valores.

Campo

- Obrigatório.
- O campo deve ser um campo de texto.

size

- Opcional inteiro.
- O número de valores deve ser retornado.
- O padrão é 10.

Exemplo

```
os> source=accounts | stats take(firstname);
fetched rows / total rows = 1/1
+-----------------------------+
| take(firstname)             |
|-----------------------------|
| [Jane, Mary, Nikki, Juan    |
+-----------------------------+
```

PERCENTILE ou PERCENTILE_APPROX

Use `PERCENTILE(expr, percent)` ou `PERCENTILE_APPROX(expr, percent)` para retornar o percentil aproximado de expr na porcentagem especificada.

percentual

- O número deve ser uma constante entre 0 e 100.

Exemplo

```
os> source=accounts | stats percentile(age, 90) by gender;
```

```
fetched rows / total rows = 2/2
+-----------------------+----------+
| percentile(age, 90)   | gender   |
|-----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+-----------------------+----------+
```

Exemplo 1: calcular a contagem de eventos

O exemplo mostra como calcular a contagem de eventos nas contas.

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
| count()   |
|-----------|
| 4         |
+-----------+
```

Exemplo 2: calcular a média de um campo

O exemplo mostra como calcular a idade média de todas as contas.

```
os> source=accounts | stats avg(age);
fetched rows / total rows = 1/1
+------------+
| avg(age)   |
|------------|
| 32.25      |
+------------+
```

Exemplo 3: calcular a média de um campo por grupo

O exemplo mostra como calcular a idade média de todas as contas, agrupadas por sexo.

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
```

```
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

Exemplo 4: calcular a média, a soma e a contagem de um campo por grupo

O exemplo mostra como calcular a idade média, a soma da idade e a contagem de eventos de todas as contas agrupadas por sexo.

```
os> source=accounts | stats avg(age), sum(age), count() by gender;
fetched rows / total rows = 2/2
+--------------------+------------+-----------+----------+
| avg(age)           | sum(age)   | count()   | gender   |
|--------------------+------------+-----------+----------|
| 28.0               | 28         | 1         | F        |
| 33.666666666666664 | 101        | 3         | M        |
+--------------------+------------+-----------+----------+
```

Exemplo 5: calcular o máximo de um campo

O exemplo calcula a idade máxima para todas as contas.

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
+------------+
```

Exemplo 6: calcular o máximo e o mínimo de um campo por grupo

O exemplo calcula os valores de idade máxima e mínima para todas as contas, agrupados por sexo.

```
os> source=accounts | stats max(age), min(age) by gender;
fetched rows / total rows = 2/2
+------------+------------+----------+
| max(age)   | min(age)   | gender   |
|------------+------------+----------|
| 28         | 28         | F        |
| 36         | 32         | M        |
```

```
+------------+-----------+----------+
```

Exemplo 7: calcular a contagem distinta de um campo

Para obter a contagem de valores distintos de um campo, você pode usar a função `DISTINCT_COUNT` (ou `DC`) em vez de `COUNT`. O exemplo calcula a contagem e a contagem distinta do campo sexode todas as contas.

```
os> source=accounts | stats count(gender), distinct_count(gender);
fetched rows / total rows = 1/1
+-----------------+--------------------------+
| count(gender)   | distinct_count(gender)   |
|-----------------+--------------------------|
| 4               | 2                        |
+-----------------+--------------------------+
```

Exemplo 8: calcular a contagem por um span

O exemplo obtém a contagem de idade pelo intervalo de 10 anos.

```
os> source=accounts | stats count(age) by span(age, 10) as age_span
fetched rows / total rows = 2/2
+--------------+------------+
| count(age)   | age_span   |
|--------------+------------|
| 1            | 20         |
| 3            | 30         |
+--------------+------------+
```

Exemplo 9: calcular a contagem por um sexo e span

Este exemplo conta registros agrupados por sexo e spans de idade de 5 anos.

```
os> source=accounts | stats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 3/3
+-------+------------+----------+
| cnt   | age_span   | gender   |
|-------+------------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
```

```
+-------+-----------+----------+
```

A expressão span sempre aparece como a primeira chave de agrupamento, independentemente da ordem especificada no comando.

```
os> source=accounts | stats count() as cnt by gender, span(age, 5) as age_span
fetched rows / total rows = 3/3
+-------+-----------+----------+
| cnt   | age_span   | gender   |
|-------+-----------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
+-------+-----------+----------+
```

Exemplo 10: calcular a contagem e receber lista de e-mail por um sexo e span

O exemplo obtém a contagem da idade pelo intervalo de 10 anos e agrupa por sexo. Além disso, para cada linha, obtém uma lista de até 5 e-mails.

```
os> source=accounts | stats count() as cnt, take(email, 5) by span(age, 5) as age_span,
 gender
fetched rows / total rows = 3/3
+-------+-----------------------------------------------------+-----------+----------+
| cnt   | take(email, 5)                                      | age_span   | gender   |
|-------+-----------------------------------------------------+-----------+----------|
| 1     | []                                                  | 25         | F        |
| 2     | [janedoe@anycompany.com,juanli@examplecompany.org] | 30         | M        |
| 1     | [marymajor@examplecorp.com]                         | 35         | M        |
+-------+-----------------------------------------------------+-----------+----------+
```

Exemplo 11: calcular o percentil de um campo

O exemplo mostra como calcular a idade do percentil 90 de todas as contas.

```
os> source=accounts | stats percentile(age, 90);
fetched rows / total rows = 1/1
+----------------------+
| percentile(age, 90)   |
|----------------------|
| 36                    |
```

```
+-----------------------+
```

Exemplo 12: calcular o percentil de um campo por grupo

O exemplo mostra como calcular a idade do percentil 90 de todas as contas por sexo.

```
os> source=accounts | stats percentile(age, 90) by gender;
fetched rows / total rows = 2/2
+-----------------------+----------+
| percentile(age, 90)   | gender   |
|-----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+-----------------------+----------+
```

Exemplo 13: calcular o percentil por um sexo e span

O exemplo obtém a idade do percentil 90 pelo intervalo de 10 anos e agrupa por gênero.

```
os> source=accounts | stats percentile(age, 90) as p90 by span(age, 10) as age_span,
 gender
fetched rows / total rows = 2/2
+-------+------------+----------+
| p90   | age_span   | gender   |
|-------+------------+----------|
| 28    | 20         | F        |
| 36    | 30         | M        |
+-------+------------+----------+
```

```
- `source = table | stats avg(a) `
- `source = table | where a < 50 | stats avg(c) `
- `source = table | stats max(c) by b`
- `source = table | stats count(c) by b | head 5`
- `source = table | stats distinct_count(c)`
- `source = table | stats stddev_samp(c)`
- `source = table | stats stddev_pop(c)`
- `source = table | stats percentile(c, 90)`
- `source = table | stats percentile_approx(c, 99)`
```

Agregações com span

```
- `source = table  | stats count(a) by span(a, 10) as a_span`
- `source = table  | stats sum(age) by span(age, 5) as age_span | head 2`
- `source = table  | stats avg(age) by span(age, 20) as age_span, country  | sort -
 age_span |  head 2`
```

Agregações com span de janela de tempo (função de janela tumble)

```
- `source = table | stats sum(productsAmount) by span(transactionDate, 1d) as age_date
 | sort age_date`
- `source = table | stats sum(productsAmount) by span(transactionDate, 1w) as age_date,
 productId`
```

Grupo de agregações por vários níveis

```
- `source = table | stats avg(age) as avg_state_age by country, state | stats
 avg(avg_state_age) as avg_country_age by country`
- `source = table | stats avg(age) as avg_city_age by country, state, city | eval
 new_avg_city_age = avg_city_age - 1 | stats avg(new_avg_city_age) as avg_state_age
 by country, state | where avg_state_age > 18 | stats avg(avg_state_age) as
 avg_adult_country_age by country`
```

Comando subquery

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `subquery` para realizar consultas complexas e aninhadas em instruções Piped Processing Language (PPL).

```
source=logs | where field in [ subquery source=events | where condition | fields
 field ]
```

Neste exemplo, a pesquisa primária (`source=logs`) é filtrada pelos resultados da subconsulta (`source=events`).

O comando subquery é compatível com a vários níveis de aninhamento para análise de dados complexos.

Exemplo de subconsulta aninhada

```
source=logs | where id in [ subquery source=users | where user in [ subquery
 source=actions | where action="login" | fields user] | fields uid ]
```

InSubquery Uso

- `source = outer | where a in [ source = inner | fields b ]`
- `source = outer | where (a) in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) in [ source = inner | fields d,e,f ]`
- `source = outer | where a not in [ source = inner | fields b ]`
- `source = outer | where (a) not in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) not in [ source = inner | fields d,e,f ]`
- `source = outer a in [ source = inner | fields b ]` (filtragem de pesquisa com subconsulta)
- `source = outer a not in [ source = inner | fields b ]` (filtragem de pesquisa com subconsulta)
- `source = outer | where a in [ source = inner1 | where b not in [ source = inner2 | fields c ] | fields b ]` (aninhada)
- `source = table1 | inner join left = l right = r on l.a = r.a AND r.a in [ source = inner | fields d ] | fields l.a, r.a, b, c` (como filtro join)

Exemplos de migração de SQL com PPL IN-Subquery

TPC-H Q4 (in-subquery com agregação)

```
select
  o_orderpriority,
  count(*) as order_count
from
  orders
where
  o_orderdate >= date '1993-07-01'
  and o_orderdate < date '1993-07-01' + interval '3' month
```

```
  and o_orderkey in (
    select
      l_orderkey
    from
      lineitem
    where l_commitdate < l_receiptdate
  )
group by
  o_orderpriority
order by
  o_orderpriority
```

Reescrito pela consulta PPL: InSubquery

```
source = orders
| where o_orderdate >= "1993-07-01" and o_orderdate < "1993-10-01" and o_orderkey IN
  [ source = lineitem
    | where l_commitdate < l_receiptdate
    | fields l_orderkey
  ]
| stats count(1) as order_count by o_orderpriority
| sort o_orderpriority
| fields o_orderpriority, order_count
```

TPC-H Q20 (aninhado em subconsulta)

```
select
  s_name,
  s_address
from
  supplier,
  nation
where
  s_suppkey in (
    select
      ps_suppkey
    from
      partsupp
    where
      ps_partkey in (
        select
          p_partkey
        from
```

```
          part
        where
          p_name like 'forest%'
      )
  )
  and s_nationkey = n_nationkey
  and n_name = 'CANADA'
order by
  s_name
```

Reescrito pela consulta PPL: InSubquery

```
source = supplier
| where s_suppkey IN [
    source = partsupp
    | where ps_partkey IN [
        source = part
        | where like(p_name, "forest%")
        | fields p_partkey
      ]
    | fields ps_suppkey
  ]
| inner join left=l right=r on s_nationkey = n_nationkey and n_name = 'CANADA'
  nation
| sort s_name
```

ExistsSubquery uso

Pressupostos: a, b são campos de tabela externa; c, d são campos de tabela interna e e, f são campos da tabela interna2.

- `source = outer | where exists [ source = inner | where a = c ]`
- `source = outer | where not exists [ source = inner | where a = c ]`
- `source = outer | where exists [ source = inner | where a = c and b = d ]`
- `source = outer | where not exists [ source = inner | where a = c and b = d ]`
- `source = outer exists [ source = inner | where a = c ]` (filtragem de pesquisa com subconsulta)
- `source = outer not exists [ source = inner | where a = c ]` (filtragem de pesquisa com subconsulta)

- `source = table as t1 exists [ source = table as t2 | where t1.a = t2.a ]` (alias de tabela é útil em subconsulta exists)
- `source = outer | where exists [ source = inner1 | where a = c and exists [ source = inner2 | where c = e ] ]` (aninhada)
- `source = outer | where exists [ source = inner1 | where a = c | where exists [ source = inner2 | where c = e ] ]` (aninhada)
- `source = outer | where exists [ source = inner | where c > 10 ]` (existe não correlacionado)
- `source = outer | where not exists [ source = inner | where c > 10 ]` (existe não correlacionado)
- `source = outer | where exists [ source = inner ] | eval l = "nonEmpty" | fields l` (existe não correlacionado especial)

ScalarSubquery uso

Pressupostos: `a`, `b` são campos de tabela externa; `c`, `d` são campos de tabela interna e `e`, `f` são campos da tabela aninhada

Subconsulta escalar não correlacionada

Em Select:

- `source = outer | eval m = [ source = inner | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | stats max(c) ] + b | fields m, a`

Em Where:

- `source = outer | where a > [ source = inner | stats min(c) ] | fields a`

Em filtro Search:

- `source = outer a > [ source = inner | stats min(c) ] | fields a`

Subconsulta escalar correlacionada

Em Select:

- `source = outer | eval m = [ source = inner | where outer.b = inner.d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where b = d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where outer.b > inner.d | stats max(c) ] | fields m, a`

Em Where:

- `source = outer | where a = [ source = inner | where outer.b = inner.d | stats max(c) ]`
- `source = outer | where a = [ source = inner | where b = d | stats max(c) ]`
- `source = outer | where [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Em filtro Search:

- `source = outer a = [ source = inner | where b = d | stats max(c) ]`
- `source = outer [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Subconsulta escalar aninhada

- `source = outer | where a = [ source = inner | stats max(c) | sort c ] OR b = [ source = inner | where c = 1 | stats min(d) | sort d ]`
- `source = outer | where a = [ source = inner | where c = [ source = nested | stats max(e) by f | sort f ] | stats max(d) by c | sort c | head 1 ]`

(Relação) Subconsulta

`InSubquery`, `ExistsSubquery` e `ScalarSubquery` são expressões de subconsulta. Mas `RelationSubquery` não é uma expressão de subconsulta, é um plano de subconsulta que é comumente usado na cláusula Join ou From.

- `source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ]` (subconsulta em join right side)
- `source = [ source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ] | stats count(a) by b ] as outer | head 1`

Contexto adicional

`InSubquery`, `ExistsSubquery` e `ScalarSubquery` são expressões de subconsulta usadas frequentemente em cláusulas e filtros de pesquisa `where`.

Comando Where:

```
| where <boolean expression> | ...
```

Filtro de pesquisa:

```
search source=* <boolean expression> | ...
```

Uma expressão de subconsulta pode ser usada em uma expressão booleana:

```
| where orders.order_id in [ source=returns | where return_reason="damaged" | field
 order_id ]
```

O `orders.order_id in [ source=... ]` é uma `<boolean expression>`.

Em geral, chamamos esse tipo de cláusula de subconsulta é a expressão `InSubquery`. É uma `<boolean expression>`.

Subconsulta com diferentes tipos join

Exemplo usando um `ScalarSubquery`:

```
source=employees
| join source=sales on employees.employee_id = sales.employee_id
| where sales.sale_amount > [ source=targets | where target_met="true" | fields
 target_value ]
```

Ao contrário de InSubquery ExistsSubquery,, e ScalarSubquery, a não RelationSubquery é uma expressão de subconsulta. É um plano de subconsulta.

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN left = c, right = o ON c.c_custkey = o.o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS ...
```

Comando top

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `top` para encontrar a tupla de valores mais comum de todos os campos na lista de campos.

Sintaxe

Use a seguinte sintaxe:

```
top [N] <field-list> [by-clause] top_approx [N] <field-list> [by-clause]
```

N

- o número máximo de resultados a serem retornados.
- Padrão: 10

field-list

- Obrigatório.
- Uma lista de nomes de campos delimitada por vírgulas.

by-clause

- Opcional.
- Um ou mais campos pelos quais agrupar os resultados.

top_approx

- Uma contagem aproximada dos (n) principais campos usando a cardinalidade estimada pelo HyperLogLog algoritmo ++.

Exemplo 1: encontrar os valores mais comuns em um campo

O exemplo encontra o sexo mais comum para todas as contas.

Consulta PPL:

```
os> source=accounts | top gender;
os> source=accounts | top_approx gender;
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| M        |
| F        |
+----------+
```

Exemplo 2: encontrar os valores mais comuns em um campo (limitado a 1)

O exemplo encontra o sexo mais comum de todos para todas as contas.

Consulta PPL:

```
os> source=accounts | top_approx 1 gender;
fetched rows / total rows = 1/1
+----------+
| gender   |
|----------|
| M        |
+----------+
```

Exemplo 3: encontrar os valores mais comuns, agrupados por gênero

O exemplo encontra a idade mais comum para todas as contas, agrupadas por gênero.

Consulta PPL:

```
os> source=accounts | top 1 age by gender;
os> source=accounts | top_approx 1 age by gender;
fetched rows / total rows = 2/2
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
+----------+-------+
```

Comando trendline

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `trendline` para calcular as médias móveis dos campos.

Sintaxe

Use a sintaxe a seguir

```
TRENDLINE [sort <[+|-] sort-field>] SMA(number-of-datapoints, field) [AS alias]
 [SMA(number-of-datapoints, field) [AS alias]]...
```

[+|-]

- Opcional.
- O sinal de mais [+] significa ordem crescente com NULL/MISSING os valores primeiro.
- O sinal de menos [-] representa a ordem decrescente com NULL/MISSING os valores por último.
- Padrão: ordem crescente com NULL/MISSING os valores primeiro.

sort-field

- Obrigatório quando a classificação é usada.
- O campo usado para classificação.

number-of-datapoints

- Obrigatório.
- O número de pontos de dados que calculam a média móvel.
- Deve ser maior do que zero.

Campo

- Obrigatório.
- O nome do campo para o qual a média móvel deve ser calculada.

alias

- Opcional.
- O nome da coluna resultante que contém a média móvel.

Somente o tipo Simple Moving Average (SMA) é compatível. É calculado assim:

```
f[i]: The value of field 'f' in the i-th data-point
n: The number of data-points in the moving window (period)
t: The current time index

SMA(t) = (1/n) * Σ(f[i]), where i = t-n+1 to t
```

Exemplo 1: calcular a média móvel simples para uma série temporal de temperaturas

O exemplo calcula a média móvel simples de temperaturas usando dois pontos de dados.

Consulta PPL:

```
os> source=t | trendline sma(2, temperature) as temp_trend;
fetched rows / total rows = 5/5
```

```
+-----------+---------+--------------------+----------+
|temperature|device-id|           timestamp|temp_trend|
+-----------+---------+--------------------+----------+
|         12|     1492|2023-04-06 17:07:...|      NULL|
|         12|     1492|2023-04-06 17:07:...|      12.0|
|         13|      256|2023-04-06 17:07:...|      12.5|
|         14|      257|2023-04-06 17:07:...|      13.5|
|         15|      258|2023-04-06 17:07:...|      14.5|
+-----------+---------+--------------------+----------+
```

Exemplo 2: calcular as médias móveis simples para uma série temporal de temperaturas com classificação

O exemplo calcula duas médias móveis simples das temperaturas usando dois e três pontos de dados classificados em ordem decrescente por device-id.

Consulta PPL:

```
os> source=t | trendline sort - device-id sma(2, temperature) as temp_trend_2 sma(3,
 temperature) as temp_trend_3;
fetched rows / total rows = 5/5
+-----------+---------+--------------------+------------+------------------+
|temperature|device-id|           timestamp|temp_trend_2|      temp_trend_3|
+-----------+---------+--------------------+------------+------------------+
|         15|      258|2023-04-06 17:07:...|        NULL|              NULL|
|         14|      257|2023-04-06 17:07:...|        14.5|              NULL|
|         13|      256|2023-04-06 17:07:...|        13.5|              14.0|
|         12|     1492|2023-04-06 17:07:...|        12.5|              13.0|
|         12|     1492|2023-04-06 17:07:...|        12.0|12.333333333333334|
+-----------+---------+--------------------+------------+------------------+
```

Comando where

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

O comando `where` usa uma expressão bool para filtrar o resultado da pesquisa. Ele só retorna o resultado quando a expressão bool é avaliada como verdadeira.

Sintaxe

Use a seguinte sintaxe:

```
where <boolean-expression>
```

bool-expression

- Opcional.
- Qualquer expressão que possa ser avaliada como um valor booleano.

Exemplo 1: filtrar conjunto de resultados com condição

O exemplo mostra como buscar documentos do índice de contas que atendam a condições específicas.

Consulta PPL:

```
os> source=accounts | where account_number=1 or gender="F" | fields account_number,
 gender;
fetched rows / total rows = 2/2
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Exemplos adicionais

Filtros com condições lógicas

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`
- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`
- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`

- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a between 1 and 4`: nota: isso retorna um >= 1 e um <= 4, ou seja, [1, 4]
- `source = table | where b not between '2024-09-10' and '2025-09-10'`: nota: isso retorna b >= "**********" e b <= "2025-09-10"
- `source = table | where cidrmatch(ip, '**********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`

```
source = table | eval status_category =
    case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code')
    | where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code'
    ) = 'Incorrect HTTP status code'
```

```
source = table
    | eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
    | where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor =
 8, 'even' else 'odd') = 'even'
    |  stats count() by factor
```

field summary

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a esse comando PPL, consulte. the section called “Comandos”

Use o comando `fieldsummary` para calcular estatísticas básicas para cada campo (count, distinct count, min, max, avg, stddev, mean) e determinar o tipo de dados de cada campo. Esse comando pode ser usado com qualquer barra anterior e a levará em consideração.

Sintaxe

Use a sintaxe a seguir. Para casos de uso de CloudWatch registros, somente um campo em uma consulta é compatível.

```
... | fieldsummary <field-list> (nulls=true/false)
```

includefields

- Lista de todas as colunas a serem coletadas com estatísticas em um conjunto unificado de resultados.

Nulos

- Opcional.
- Se definido como verdadeiro, inclui valores nulos nos cálculos de agregação (substitua nulo por zero para valores numéricos).

Exemplo 1

Consulta PPL:

```
os> source = t | where status_code != 200 | fieldsummary includefields= status_code
 nulls=true
+------------------+-------------+------------+------------+------------+------------
+------------+------------+----------------|
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
```

```
|-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "status_code"    |      2      |         2         | 301   |   403   |  352.0 |  352.0
  |  72.12489168102785  |  0    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
```

Exemplo 2

Consulta PPL:

```
os> source = t | fieldsummary includefields= id, status_code, request_path nulls=true
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
|-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
|       "id"       |      6      |         6         | 1     |   6    |  3.5   |   3.5
 |  1.8708286933869707  |  0    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "status_code"    |      4      |         3         | 200   |   403   |  184.0 |  184.0
  |  161.16699413961905 |  2    | "int"  |
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
| "request_path"   |      2      |         2         | /about| /home   |  0.0    |  0.0
    |      0           |  2    |"string"|
+-----------------+-------------+-----------+-----------+-----------+-----------
+-----------+-----------+----------------|
```

Comando expand

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

Use o comando `expand` para nivelar um campo do tipo Array <Any> ou Map<Any>, produzindo linhas individuais para cada elemento ou par chave-valor.

Sintaxe

Use a seguinte sintaxe:

```
expand <field> [As alias]
```

Campo

- O campo a ser expandido (explodido).
- O campo deve ser de um tipo compatível.

alias

- Opcional.
- O nome a ser usado em vez do nome original do campo.

Diretrizes de uso

O comando expand produz uma linha para cada elemento na matriz ou no campo de mapa especificado, onde:

- Os elementos da matriz se tornam linhas individuais.
- Os pares de chave-valor do mapa são divididos em linhas separadas, com cada par chave-valor representado como uma linha.
- Quando um alias é fornecido, os valores explodidos são representados sob o alias em vez do nome original do campo.

Você pode usar esse comando em combinação com outros, como stats, eval e parse, para manipular ou extrair dados após a expansão.

Exemplos

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`

- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`
- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

Você pode usar o comando expand em combinação com outros, como eval, stats etc. O uso de vários comandos expand criará um produto cartesiano de todos os elementos internos em cada matriz ou mapa composto.

Consulta SQL push-down eficaz

O comando expand é traduzido em uma operação SQL equivalente usando LATERAL VIEW explode, permitindo a explosão eficiente de matrizes ou mapas no nível da consulta SQL.

```
SELECT customer exploded_productId
FROM table
LATERAL VIEW explode(productId) AS exploded_productId
```

O comando explode oferece a seguinte funcionalidade:

- É uma operação de coluna que retorna uma nova coluna.
- Cria uma nova linha para cada elemento da coluna explodida.
- Os nulos internos são ignorados como parte do campo explodido (nenhuma linha é created/exploded para nulos).

Funções PPL

Tópicos

- Funções PPL de condição
- Funções PPL de hash criptográfico
- Funções PPL de data e hora
- Expressões PPL

- Funções PPL de endereço IP
- Funções PPL de JSON
- Funções PPL do Lambda
- Funções PPL matemáticas
- Funções PPL de string
- Funções PPL de conversão de tipo

Funções PPL de condição

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

ISNULL

Descrição: `isnull(field)` retornará verdadeiro se o campo for nulo.

Tipo de argumento:

- Todos os tipos de dados compatíveis.

Tipo de retorno:

- BOOLEAN

Exemplo:

```
os> source=accounts | eval result = isnull(employer) | fields result, employer,
 firstname
fetched rows / total rows = 4/4
+----------+-------------+-------------+
| result   | employer    | firstname   |
|----------+-------------+-------------|
| False    | AnyCompany  | Mary        |
| False    | ExampleCorp | Jane        |
```

```
| False    | ExampleOrg  | Nikki       |
| True     | null        | Juan        |
+----------+-------------+-------------+
```

ISNOTNULL

Descrição: `isnotnull(field)` retornará verdadeiro se o campo não for nulo.

Tipo de argumento:

- Todos os tipos de dados compatíveis.

Tipo de retorno:

- BOOLEAN

Exemplo:

```
os> source=accounts | where not isnotnull(employer) | fields account_number, employer
fetched rows / total rows = 1/1
+------------------+------------+
| account_number   | employer   |
|------------------+------------|
| 18               | null       |
+------------------+------------+
```

EXISTS

Exemplo:

```
os> source=accounts | where exists(email) | fields account_number, email
fetched rows / total rows = 1/1
```

IFNULL

Descrição: `ifnull(field1, field2)` retornará `field2` se `field1` for nulo.

Tipo de argumento:

- Todos os tipos de dados compatíveis.

- Se os dois parâmetros tiverem tipos diferentes, a função não passará na verificação semântica.

Tipo de retorno:

- Any

Exemplo:

```
os> source=accounts | eval result = ifnull(employer, 'default') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
+------------+------------+-------------+
| result     | employer   | firstname   |
|------------+------------+-------------|
| AnyCompany | AnyCompany | Mary        |
| ExampleCorp| ExampleCorp| Jane        |
| ExampleOrg | ExampleOrg | Nikki       |
| default    | null       | Juan        |
+------------+------------+-------------+
```

NULLIF

Descrição: `nullif(field1, field2)` retornará nulo se dois parâmetros forem iguais, caso contrário, retornará field1.

Tipo de argumento:

- Todos os tipos de dados compatíveis.
- Se os dois parâmetros tiverem tipos diferentes, a função não passará na verificação semântica.

Tipo de retorno:

- Any

Exemplo:

```
os> source=accounts | eval result = nullif(employer, 'AnyCompany') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
```

```
+----------------+----------------+-------------+
| result         | employer       | firstname   |
|----------------+----------------+-------------|
| null           | AnyCompany     | Mary        |
| ExampleCorp    | ExampleCorp    | Jane        |
| ExampleOrg     | ExampleOrg     | Nikki       |
| null           | null           | Juan        |
+----------------+----------------+-------------+
```

IF

Descrição: `if(condition, expr1, expr2)` retornará `expr1` se a condição for verdadeiro, caso contrário, retornará `expr2`.

Tipo de argumento:

- Todos os tipos de dados compatíveis.
- Se os dois parâmetros tiverem tipos diferentes, a função não passará na verificação semântica.

Tipo de retorno:

- Any

Exemplo:

```
os> source=accounts | eval result = if(true, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+----------+
| result   | firstname | lastname   |
|----------+-------------+----------|
| Jane     | Jane      | Doe        |
| Mary     | Mary      | Major      |
| Pat      | Pat       | Candella   |
| Dale     | Jorge     | Souza      |
+----------+-----------+------------+

os> source=accounts | eval result = if(false, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
```

```
| result   | firstname   | lastname   |
|----------+-------------+------------|
| Doe      | Jane        | Doe        |
| Major    | Mary        | Major      |
| Candella | Pat         | Candella   |
| Souza    | Jorge       | Souza      |
+----------+-------------+------------+

os> source=accounts | eval is_vip = if(age > 30 AND isnotnull(employer), true, false) |
 fields is_vip, firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
| is_vip   | firstname   | lastname   |
|----------+-------------+------------|
| True     | Jane        | Doe        |
| True     | Mary        | Major      |
| False    | Pat         | Candella   |
| False    | Jorge       | Souza      |
+----------+-------------+------------+
```

Funções PPL de hash criptográfico

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

MD5

MD5 calcula o MD5 resumo e retorna o valor como uma string hexadecimal de 32 caracteres.

Uso: `md5('hello')`

Tipo de argumento:

- STRING

Tipo de retorno:

- STRING

Exemplo:

```
os> source=people | eval `MD5('hello')` = MD5('hello') | fields `MD5('hello')`
fetched rows / total rows = 1/1
+----------------------------------+
| MD5('hello')                     |
|----------------------------------|
| <32 character hex string>        |
+----------------------------------+
```

SHA1

SHA1 retorna o resultado da string hexadecimal de SHA-1.

Uso: `sha1('hello')`

Tipo de argumento:

- STRING

Tipo de retorno:

- STRING

Exemplo:

```
os> source=people | eval `SHA1('hello')` = SHA1('hello') | fields `SHA1('hello')`
fetched rows / total rows = 1/1
+------------------------------------------+
| SHA1('hello')                            |
|------------------------------------------|
| <40-character SHA-1 hash result>         |
+------------------------------------------+
```

SHA2

SHA2 retorna o resultado da string hexadecimal da família SHA-2 de funções de hash (SHA-224, SHA-256, SHA-384 e SHA-512). O numBits indica o tamanho em bits desejado do resultado, que deve ter um valor de 224, 256, 384, 512

Uso:

- `sha2('hello',256)`
- `sha2('hello',512)`

Tipo de argumento:

- STRING, INTEGER

Tipo de retorno:

- STRING

Exemplo:

```
os> source=people | eval `SHA2('hello',256)` = SHA2('hello',256) | fields
 `SHA2('hello',256)`
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',256)                                                |
|------------------------------------------------------------------|
| <64-character SHA-256 hash result>                               |
+------------------------------------------------------------------+

os> source=people | eval `SHA2('hello',512)` = SHA2('hello',512) | fields
 `SHA2('hello',512)`
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',512)                                                |
                                          |
|------------------------------------------------------------------|
| <128-character SHA-512 hash result>                              |
+------------------------------------------------------------------+
```

Funções PPL de data e hora

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

DAY

Uso: `DAY(date)` extrai o dia do mês de uma data, no intervalo de 1 a 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `DAYOFMONTH`, `DAY_OF_MONTH`

Exemplo:

```
os> source=people | eval `DAY(DATE('2020-08-26'))` = DAY(DATE('2020-08-26')) | fields
 `DAY(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------+
| DAY(DATE('2020-08-26'))   |
|---------------------------|
| 26                        |
+---------------------------+
```

DAYOFMONTH

Uso: `DAYOFMONTH(date)` extrai o dia do mês de uma data, no intervalo de 1 a 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `DAY`, `DAY_OF_MONTH`

Exemplo:

```
os> source=people | eval `DAYOFMONTH(DATE('2020-08-26'))` =
 DAYOFMONTH(DATE('2020-08-26')) | fields `DAYOFMONTH(DATE('2020-08-26'))`
```

```
fetched rows / total rows = 1/1
+----------------------------------+
| DAYOFMONTH(DATE('2020-08-26'))   |
|----------------------------------|
| 26                               |
+----------------------------------+
```

DAY_OF_MONTH

Uso: DAY_OF_MONTH(DATE) extrai o dia do mês de uma data, no intervalo de 1 a 31.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: DAY, DAYOFMONTH

Exemplo:

```
os> source=people | eval `DAY_OF_MONTH(DATE('2020-08-26'))` =
 DAY_OF_MONTH(DATE('2020-08-26')) | fields `DAY_OF_MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| DAY_OF_MONTH(DATE('2020-08-26'))  |
|-----------------------------------|
| 26                                |
+-----------------------------------+
```

DAYOFWEEK

Uso: DAYOFWEEK(DATE) retorna o índice do dia da semana para uma data (1 = domingo, 2 = segunda-feira,..., 7 = sábado).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: DAY_OF_WEEK

Exemplo:

```
os> source=people | eval `DAYOFWEEK(DATE('2020-08-26'))` =
 DAYOFWEEK(DATE('2020-08-26')) | fields `DAYOFWEEK(DATE('2020-08-26'))`
```

```
fetched rows / total rows = 1/1
+---------------------------------+
| DAYOFWEEK(DATE('2020-08-26'))   |
|---------------------------------|
| 4                               |
+---------------------------------+
```

DAY_OF_WEEK

Uso: `DAY_OF_WEEK(DATE)` retorna o índice do dia da semana para uma data (1 = domingo, 2 = segunda-feira,..., 7 = sábado).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `DAYOFWEEK`

Exemplo:

```
os> source=people | eval `DAY_OF_WEEK(DATE('2020-08-26'))` =
 DAY_OF_WEEK(DATE('2020-08-26')) | fields `DAY_OF_WEEK(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAY_OF_WEEK(DATE('2020-08-26'))   |
|----------------------------------|
| 4                                |
+----------------------------------+
```

DAYOFYEAR

Uso: `DAYOFYEAR(DATE)` retorna o dia do ano de uma data, no intervalo de 1 a 366.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `DAY_OF_YEAR`

Exemplo:

```
os> source=people | eval `DAYOFYEAR(DATE('2020-08-26'))` =
 DAYOFYEAR(DATE('2020-08-26')) | fields `DAYOFYEAR(DATE('2020-08-26'))`
```

```
fetched rows / total rows = 1/1
+--------------------------------+
| DAYOFYEAR(DATE('2020-08-26'))   |
|--------------------------------|
| 239                            |
+--------------------------------+
```

DAY_OF_YEAR

Uso: DAY_OF_YEAR(DATE) retorna o dia do ano de uma data, no intervalo de 1 a 366.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: DAYOFYEAR

Exemplo:

```
os> source=people | eval `DAY_OF_YEAR(DATE('2020-08-26'))` =
 DAY_OF_YEAR(DATE('2020-08-26')) | fields `DAY_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAY_OF_YEAR(DATE('2020-08-26'))   |
|----------------------------------|
| 239                              |
+----------------------------------+
```

DAYNAME

Uso: DAYNAME(DATE) retorna o nome do dia da semana de uma data, incluindo segunda-feira, terça-feira, quarta-feira, quinta-feira, sexta-feira, sábado e domingo.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `DAYNAME(DATE('2020-08-26'))` = DAYNAME(DATE('2020-08-26')) |
 fields `DAYNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
```

```
+------------------------------+
| DAYNAME(DATE('2020-08-26'))  |
|------------------------------|
| Wednesday                    |
+------------------------------+
```

FROM_UNIXTIME

Uso: FROM_UNIXTIME retorna uma representação do argumento fornecido como um valor de timestamp ou string de caracteres. Essa função realiza uma conversão inversa da função UNIX_TIMESTAMP.

Se você fornecer um segundo argumento, FROM_UNIXTIME o use para formatar o resultado semelhante à função DATE_FORMAT.

Se o timestamp estiver fora do intervalo 1970-01-01 00:00:00 a 3001-01-18 23:59:59.999999 (0 a 32536771199.999999 hora epoch), a função retornará. NULL

Tipo de argumento: DOUBLE, STRING

Retorno tipo mapa:

DOUBLE -> TIMESTAMP

DUPLO, STRING -> STRING

Exemplos:

```
os> source=people | eval `FROM_UNIXTIME(1220249547)` = FROM_UNIXTIME(1220249547) |
 fields `FROM_UNIXTIME(1220249547)`
fetched rows / total rows = 1/1
+----------------------------+
| FROM_UNIXTIME(1220249547)  |
|----------------------------|
| 2008-09-01 06:12:27        |
+----------------------------+

os> source=people | eval `FROM_UNIXTIME(1220249547, 'HH:mm:ss')` =
 FROM_UNIXTIME(1220249547, 'HH:mm:ss') | fields `FROM_UNIXTIME(1220249547, 'HH:mm:ss')`
fetched rows / total rows = 1/1
+---------------------------------------+
| FROM_UNIXTIME(1220249547, 'HH:mm:ss')  |
```

```
|----------------------------------------|
| 06:12:27                               |
+----------------------------------------+
```

HOUR

Uso: `HOUR(TIME)` extrai o valor da hora.

Ao contrário de uma hora do dia padrão, o valor de hora nessa função pode ter um intervalo maior do que 23. Como resultado, o valor de retorno de `HOUR(TIME)` pode ser maior do que 23.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `HOUR_OF_DAY`

Exemplo:

```
os> source=people | eval `HOUR(TIME('01:02:03'))` = HOUR(TIME('01:02:03')) | fields
 `HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------+
| HOUR(TIME('01:02:03'))  |
|-------------------------|
| 1                       |
+-------------------------+
```

HOUR_OF_DAY

Uso: `HOUR_OF_DAY(TIME)` extrai o valor de hora da hora fornecida.

Ao contrário de uma hora do dia padrão, o valor de hora nessa função pode ter um intervalo maior do que 23. Como resultado, o valor de retorno de `HOUR_OF_DAY(TIME)` pode ser maior do que 23.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `HOUR`

Exemplo:

```
os> source=people | eval `HOUR_OF_DAY(TIME('01:02:03'))` =
 HOUR_OF_DAY(TIME('01:02:03')) | fields `HOUR_OF_DAY(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+--------------------------------+
| HOUR_OF_DAY(TIME('01:02:03'))   |
|--------------------------------|
| 1                              |
+--------------------------------+
```

LAST_DAY

Uso: LAST_DAY retorna o último dia do mês como um valor DATE para o argumento de data fornecido.

Tipo de argumento: DATE/STRING/TIMESTAMP/TIME

Tipo de retorno: DATE

Exemplo:

```
os> source=people | eval `last_day('2023-02-06')` = last_day('2023-02-06') | fields
 `last_day('2023-02-06')`
fetched rows / total rows = 1/1
+-------------------------+
| last_day('2023-02-06')   |
|-------------------------|
| 2023-02-28              |
+-------------------------+
```

LOCALTIMESTAMP

Uso: LOCALTIMESTAMP() é sinônimo de NOW().

Exemplo:

```
> source=people | eval `LOCALTIMESTAMP()` = LOCALTIMESTAMP() | fields
 `LOCALTIMESTAMP()`
fetched rows / total rows = 1/1
+--------------------+
| LOCALTIMESTAMP()    |
|--------------------|
| 2022-08-02 15:54:19 |
```

```
+---------------------+
```

LOCALTIME

Uso: `LOCALTIME()` é sinônimo de `NOW()`.

Exemplo:

```
> source=people | eval `LOCALTIME()` = LOCALTIME() | fields `LOCALTIME()`
fetched rows / total rows = 1/1
+---------------------+
| LOCALTIME()         |
|---------------------|
| 2022-08-02 15:54:19 |
+---------------------+
```

MAKE_DATE

Uso: `MAKE_DATE` retorna um valor de data com base nos valores de ano, mês e dia fornecidos. Todos os argumentos são arredondados para números inteiros.

Especificações: 1. MAKE_DATE(INTEGER, INTEGER, INTEGER) -> DATE

Tipo de argumento: INTEGER, INTEGER, INTEGER

Tipo de retorno: DATE

Exemplo:

```
os> source=people | eval `MAKE_DATE(1945, 5, 9)` = MAKEDATE(1945, 5, 9) | fields
 `MAKEDATE(1945, 5, 9)`
fetched rows / total rows = 1/1
+------------------------+
| MAKEDATE(1945, 5, 9)   |
|------------------------|
| 1945-05-09             |
+------------------------+
```

MINUTE

Uso: `MINUTE(TIME)` retorna o componente minuto da hora fornecida, como um número inteiro no intervalo de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `MINUTE_OF_HOUR`

Exemplo:

```
os> source=people | eval `MINUTE(TIME('01:02:03'))` =  MINUTE(TIME('01:02:03')) |
 fields `MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+----------------------------+
| MINUTE(TIME('01:02:03'))   |
|----------------------------|
| 2                          |
+----------------------------+
```

`MINUTE_OF_HOUR`

Uso: `MINUTE_OF_HOUR(TIME)` retorna o componente minuto da hora fornecida, como um número inteiro no intervalo de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `MINUTE`

Exemplo:

```
os> source=people | eval `MINUTE_OF_HOUR(TIME('01:02:03'))` =
 MINUTE_OF_HOUR(TIME('01:02:03')) | fields `MINUTE_OF_HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| MINUTE_OF_HOUR(TIME('01:02:03'))   |
|-----------------------------------|
| 2                                 |
+-----------------------------------+
```

`MONTH`

Uso: `MONTH(DATE)` retorna o mês da data fornecida como um número inteiro, no intervalo de 1 a 12 (onde 1 representa janeiro e 12 representa dezembro).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: MONTH_OF_YEAR

Exemplo:

```
os> source=people | eval `MONTH(DATE('2020-08-26'))` =  MONTH(DATE('2020-08-26')) |
 fields `MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------+
| MONTH(DATE('2020-08-26'))   |
|-----------------------------|
| 8                           |
+-----------------------------+
```

MONTHNAME

Uso: MONTHNAME(DATE) retorna o mês da data fornecida como um número inteiro, no intervalo de 1 a 12 (onde 1 representa janeiro e 12 representa dezembro).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: MONTH_OF_YEAR

Exemplo:

```
os> source=people | eval `MONTHNAME(DATE('2020-08-26'))` =
 MONTHNAME(DATE('2020-08-26')) | fields `MONTHNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------------+
| MONTHNAME(DATE('2020-08-26'))   |
|---------------------------------|
| August                          |
+---------------------------------+
```

MONTH_OF_YEAR

Uso: MONTH_OF_YEAR(DATE) retorna o mês da data fornecida como um número inteiro, no intervalo de 1 a 12 (onde 1 representa janeiro e 12 representa dezembro).

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `MONTH`

Exemplo:

```
os> source=people | eval `MONTH_OF_YEAR(DATE('2020-08-26'))` =
 MONTH_OF_YEAR(DATE('2020-08-26')) | fields `MONTH_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| MONTH_OF_YEAR(DATE('2020-08-26'))   |
|-------------------------------------|
| 8                                   |
+-------------------------------------+
```

`NOW`

Uso: `NOW` retorna a data e a hora atuais como um `TIMESTAMP` valor no formato YYYY-MM-DD 'hh:mm:ss'. O valor é expresso no fuso horário do cluster.

> **Note**
>
> `NOW()` retorna uma hora constante que indica quando a instrução começou a ser executada. Isso difere de `SYSDATE()`, que retorna a hora exata da execução.

Tipo de retorno: TIMESTAMP

Especificação: NOW() -> TIMESTAMP

Exemplo:

```
os> source=people | eval `value_1` = NOW(), `value_2` = NOW() | fields `value_1`,
 `value_2`
fetched rows / total rows = 1/1
+---------------------+---------------------+
| value_1             | value_2             |
|---------------------+---------------------|
| 2022-08-02 15:39:05 | 2022-08-02 15:39:05 |
```

```
+--------------------+---------------------+
```

QUARTER

Uso: `QUARTER(DATE)` retorna o trimestre do ano para a data fornecida como um número inteiro, no intervalo de 1 a 4.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `QUARTER(DATE('2020-08-26'))` = QUARTER(DATE('2020-08-26')) |
 fields `QUARTER(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------+
| QUARTER(DATE('2020-08-26'))   |
|-------------------------------|
| 3                             |
+-------------------------------+
```

SECOND

Uso: `SECOND(TIME)` retorna o componente segundo da hora fornecida, como um número inteiro no intervalo de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `SECOND_OF_MINUTE`

Exemplo:

```
os> source=people | eval `SECOND(TIME('01:02:03'))` = SECOND(TIME('01:02:03')) | fields
 `SECOND(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+---------------------------+
| SECOND(TIME('01:02:03'))  |
|---------------------------|
| 3                         |
```

```
+---------------------------+
```

SECOND_OF_MINUTE

Uso: `SECOND_OF_MINUTE(TIME)` retorna o componente segundo da hora fornecida, como um número inteiro no intervalo de 0 a 59.

Tipo de argumento: STRING/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Sinônimos: `SECOND`

Exemplo:

```
os> source=people | eval `SECOND_OF_MINUTE(TIME('01:02:03'))` =
 SECOND_OF_MINUTE(TIME('01:02:03')) | fields `SECOND_OF_MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| SECOND_OF_MINUTE(TIME('01:02:03'))  |
|-------------------------------------|
| 3                                   |
+-------------------------------------+
```

SUBDATE

Uso: `SUBDATE(DATE, DAYS)` subtrai o segundo argumento (como `DATE` ou `DAYS`) da data fornecida.

Tipo de argumento: DATE/TIMESTAMP, LONG

Mapa do tipo de retorno: (DATE, LONG) -> DATE

Antônimos: `ADDDATE`

Exemplo:

```
os> source=people | eval `'2008-01-02' - 31d` = SUBDATE(DATE('2008-01-02'), 31),
 `'2020-08-26' - 1` = SUBDATE(DATE('2020-08-26'), 1), `ts '2020-08-26 01:01:01' -
 1` = SUBDATE(TIMESTAMP('2020-08-26 01:01:01'), 1) | fields `'2008-01-02' - 31d`,
 `'2020-08-26' - 1`, `ts '2020-08-26 01:01:01' - 1`
fetched rows / total rows = 1/1
+---------------------+--------------------+-------------------------------+
```

```
| '2008-01-02' - 31d   | '2020-08-26' - 1   | ts '2020-08-26 01:01:01' - 1   |
|----------------------+--------------------+--------------------------------|
| 2007-12-02 00:00:00  | 2020-08-25         | 2020-08-25 01:01:01            |
+----------------------+--------------------+--------------------------------+
```

SYSDATE

Uso: `SYSDATE()` retorna a data e a hora atuais como um `TIMESTAMP` valor no formato 'YYYY-MM-DD hh:mm:ss.nnnnnnn'.

`SYSDATE()` retorna a hora exata em que é executada. Isso difere de NOW (), que retorna uma hora constante indicando quando a instrução começou a ser executada.

Tipo de argumento opcional: INTEGER (0 a 6): especifica o número de dígitos para frações de segundo no valor de retorno.

Tipo de retorno: TIMESTAMP

Exemplo:

```
os> source=people | eval `SYSDATE()` = SYSDATE() | fields `SYSDATE()`
fetched rows / total rows = 1/1
+----------------------------+
| SYSDATE()                  |
|----------------------------|
| 2022-08-02 15:39:05.123456 |
+----------------------------+
```

TIMESTAMP

Uso: `TIMESTAMP(EXPR)` constrói um tipo de timestamp com a string de entrada `expr` como timestamp.

Com um único argumento, `TIMESTAMP(expr)` constrói um timestamp a partir da entrada. Se `expr` for uma string, será interpretada como um timestamp. Para argumentos não string, a função converte `expr` em um timestamp usando o fuso horário UTC. Quando `expr` é um valor `TIME`, a função aplica a data de hoje antes da conversão.

Quando usado com dois argumentos, `TIMESTAMP(expr1, expr2)` adiciona a expressão de hora (`expr2`) à expressão de data ou timestamp (`expr1`) e retorna o resultado como um valor de timestamp.

Tipo de argumento: STRING/DATE/TIME/TIMESTAMP

Retorno tipo mapa:

(STRING/DATE/TIME/TIMESTAMP) -> CARIMBO DE DATA/HORA

(STRING/DATE/TIME/TIMESTAMP, STRING/DATE/TIME/TIMESTAMP) -> CARIMBO DE DATA/HORA

Exemplo:

```
os> source=people | eval `TIMESTAMP('2020-08-26 13:49:00')` = TIMESTAMP('2020-08-26
 13:49:00'), `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))` =
 TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42')) | fields `TIMESTAMP('2020-08-26
 13:49:00')`, `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))`
fetched rows / total rows = 1/1
+-----------------------------------
+-----------------------------------------------------+
| TIMESTAMP('2020-08-26 13:49:00')   | TIMESTAMP('2020-08-26 13:49:00',
 TIME('12:15:42'))   |
|-----------------------------------
+-----------------------------------------------------|
| 2020-08-26 13:49:00                | 2020-08-27 02:04:42
     |
+-----------------------------------
+-----------------------------------------------------+
```

UNIX_TIMESTAMP

Uso: UNIX_TIMESTAMP converte um argumento de data fornecido em hora Unix (segundos desde Epoch, que começou no início de 1970). Se nenhum argumento for fornecido, retornará a hora atual Unix.

O argumento de data pode ser uma DATE, uma string TIMESTAMP ou um número em um desses formatos: YYMMDD, YYMMDDhhmmss, YYYYMMDD ou YYYYMMDDhhmmss. Se o argumento incluir um componente de hora, poderá, opcionalmente, incluir segundos fracionários.

Se o argumento estiver em um formato inválido ou fora do intervalo 1970-01-01 00:00:00 a 3001-01-18 23:59:59.999999 (0 a 32536771199.999999 no horário epoch), a função retornará NULL.

A função aceita DATE, TIMESTAMP ou DOUBLE como tipos de argumento, ou nenhum argumento. Ela sempre retorna um valor DOUBLE representando o timestamp Unix.

Para a conversão inversa, você pode usar a função FROM_UNIXTIME.

Tipo de argumento:<NONE>/DOUBLE/DATE/TIMESTAMP

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `UNIX_TIMESTAMP(double)` = UNIX_TIMESTAMP(20771122143845),
 `UNIX_TIMESTAMP(timestamp)` = UNIX_TIMESTAMP(TIMESTAMP('1996-11-15 17:05:42')) |
 fields `UNIX_TIMESTAMP(double)`, `UNIX_TIMESTAMP(timestamp)`
fetched rows / total rows = 1/1
+-------------------------+----------------------------+
| UNIX_TIMESTAMP(double)  | UNIX_TIMESTAMP(timestamp)  |
|-------------------------+----------------------------|
| 3404817525.0            | 848077542.0                |
+-------------------------+----------------------------+
```

WEEK

Uso: `WEEK(DATE)` retorna o número da semana de uma data fornecida.

Tipo de argumento: DATE/TIMESTAMP/STRING

Tipo de retorno: INTEGER

Sinônimos: `WEEK_OF_YEAR`

Exemplo:

```
os> source=people | eval `WEEK(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20')) | fields
 `WEEK(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+---------------------------+
| WEEK(DATE('2008-02-20'))  |
|---------------------------|
| 8                         |
+---------------------------+
```

WEEKDAY

Uso: `WEEKDAY(DATE)` retorna o índice do dia da semana da data (0 = segunda-feira, 1 = terça-feira,..., 6 = domingo).

É semelhante à função `dayofweek`, mas retorna índices diferentes para cada dia.

Tipo de argumento: STRING/DATE/TIME/TIMESTAMP

Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `weekday(DATE('2020-08-26'))` = weekday(DATE('2020-08-26'))
 | eval `weekday(DATE('2020-08-27'))` = weekday(DATE('2020-08-27')) | fields
 `weekday(DATE('2020-08-26'))`, `weekday(DATE('2020-08-27'))`
fetched rows / total rows = 1/1
+------------------------------+------------------------------+
| weekday(DATE('2020-08-26'))  | weekday(DATE('2020-08-27'))  |
|------------------------------+------------------------------|
| 2                            | 3                            |
+------------------------------+------------------------------+
```

WEEK_OF_YEAR

Uso: `WEEK_OF_YEAR(DATE)` retorna o número da semana da data fornecida.

Tipo de argumento: DATE/TIMESTAMP/STRING

Tipo de retorno: INTEGER

Sinônimos: `WEEK`

Exemplo:

```
os> source=people | eval `WEEK_OF_YEAR(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20'))|
 fields `WEEK_OF_YEAR(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| WEEK_OF_YEAR(DATE('2008-02-20'))  |
|-----------------------------------|
| 8                                 |
+-----------------------------------+
```

YEAR

Uso: `YEAR(DATE)` retorna o ano da data, no intervalo de 1000 a 9999, ou 0 para a data “zero”.

Tipo de argumento: STRING/DATE/TIMESTAMP

Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `YEAR(DATE('2020-08-26'))` = YEAR(DATE('2020-08-26')) | fields
 `YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------+
| YEAR(DATE('2020-08-26'))   |
|----------------------------|
| 2020                       |
+----------------------------+
```

DATE_ADD

Uso: `DATE_ADD(date, INTERVAL expr unit)` adiciona o intervalo especificado à data fornecida.

Tipo de argumento: DATE, INTERVAL

Tipo de retorno: DATE

Antônimos: `DATE_SUB`

Exemplo:

```
os> source=people | eval `'2020-08-26' + 1d` = DATE_ADD(DATE('2020-08-26'), INTERVAL 1
 DAY) | fields `'2020-08-26' + 1d`
fetched rows / total rows = 1/1
+---------------------+
| '2020-08-26' + 1d   |
|---------------------|
| 2020-08-27          |
+---------------------+
```

DATE_SUB

Uso: `DATE_SUB(date, INTERVAL expr unit)` subtrai o intervalo expr da data.

Tipo de argumento: DATE, INTERVAL

Tipo de retorno: DATE

Antônimos: `DATE_ADD`

Exemplo:

```
os> source=people | eval `'2008-01-02' - 31d` = DATE_SUB(DATE('2008-01-02'), INTERVAL
 31 DAY) | fields `'2008-01-02' - 31d`
fetched rows / total rows = 1/1
+---------------------+
| '2008-01-02' - 31d  |
|---------------------|
| 2007-12-02          |
+---------------------+
```

`TIMESTAMPADD`

Uso: retorna um valor `TIMESTAMP` após adicionar um intervalo de tempo especificado a uma data fornecida.

Argumentos:

- intervalo: INTERVAL (SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, QUARTER, YEAR)
- inteiro: INTEGER
- data: DATA, TIMESTAMP ou STRING

Se você fornecer uma `STRING` como argumento de data, formate-a como `TIMESTAMP` válido. A função converte automaticamente um argumento `DATE` em um `TIMESTAMP`.

Exemplos:

```
os> source=people | eval `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')` =
 TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | eval `TIMESTAMPADD(QUARTER, -1,
 '2000-01-01 00:00:00')` = TIMESTAMPADD(QUARTER, -1, '2000-01-01 00:00:00') | fields
 `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')`, `TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00')`
fetched rows / total rows = 1/1
+-----------------------------------------------
+-----------------------------------------------+
```

```
| TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00') |
|----------------------------------------------
+----------------------------------------------|
| 2000-01-18 00:00:00                          | 1999-10-01 00:00:00
           |
+---------------------------------------------
+-----------------------------------------------+
```

TIMESTAMPDIFF

Uso: `TIMESTAMPDIFF(interval, start, end)` retorna a diferença entre o início e o fim date/times em unidades de intervalo especificadas.

Argumentos:

- intervalo: INTERVAL (SECOND, MINUTE, HOUR, DAY, WEEK, MONTH, QUARTER, YEAR)
- início: DATE, TIMESTAMP ou STRING
- fim: DATE, TIMESTAMP ou STRING

A função converte automaticamente argumentos em `TIMESTAMP` quando apropriado. Formate os argumentos `STRING` como `TIMESTAMP`s válidos.

Exemplos:

```
os> source=people | eval `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06
 00:00:00')` = TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 eval `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00'))` = TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00')) | fields `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00',
 '2001-03-06 00:00:00')`, `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00'))`
fetched rows / total rows = 1/1
+-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
| TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00')) |
|-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------|
```

```
| 4                                                                  | -23
                                                                        |
+--------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
```

UTC_TIMESTAMP

Uso: UTC_TIMESTAMP retorna o timestamp UTC atual como um valor no formato “YYYY-MM-DD hh:mm:ss”.

Tipo de retorno: TIMESTAMP

Especificação: UTC_TIMESTAMP () -> TIMESTAMP

Exemplo:

```
> source=people | eval `UTC_TIMESTAMP()` = UTC_TIMESTAMP() | fields `UTC_TIMESTAMP()`
fetched rows / total rows = 1/1
+---------------------+
| UTC_TIMESTAMP()     |
|---------------------|
| 2022-10-03 17:54:28 |
+---------------------+
```

CURRENT_TIMEZONE

Uso: CURRENT_TIMEZONE retorna o fuso horário local atual.

Tipo de retorno: STRING

Exemplo:

```
> source=people | eval `CURRENT_TIMEZONE()` = CURRENT_TIMEZONE() | fields
 `CURRENT_TIMEZONE()`
fetched rows / total rows = 1/1
+------------------------+
| CURRENT_TIMEZONE()     |
|------------------------|
| America/Chicago        |
+------------------------+
```

Expressões PPL

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

Expressões, especialmente expressões de valor, retornam um valor escalar. As expressões têm diferentes tipos e formatos. Por exemplo, há valores literais como expressões atômicas, além de expressões aritméticas, predicativas e de função definidas a partir deles. Você pode usar expressões em cláusulas diferentes, como usar expressões aritméticas nos comandos `Filter` e `Stats`.

Operadores

Uma expressão aritmética é uma expressão formada por literais numéricos e operadores aritméticos binários da seguinte forma:

1. `+`: adicionar.
2. `-`: subtrair.
3. `*`: multiplicar.
4. `/`: dividir (para números inteiros, o resultado é um número inteiro com a parte fracionária descartada)
5. `%`: módulo (use somente com números inteiros; o resultado é o resto da divisão)

Precedência

Use parênteses para controlar a precedência dos operadores aritméticos. Caso contrário, os operadores com maior precedência serão executados primeiro.

Conversão de tipo

A conversão de tipo implícita é realizada ao pesquisar assinaturas de operadores. Por exemplo, um número inteiro + um número real corresponde à assinatura `+(double,double)`, o que resulta em um número real. Essa regra também se aplica às chamadas de funções.

Exemplo de diferentes tipos de expressões aritméticas:

```
os> source=accounts | where age > (25 + 5) | fields age ;
```

```
fetched rows / total rows = 3/3
+-------+
| age   |
|-------|
| 32    |
| 36    |
| 33    |
+-------+
```

Operadores de predicado

Um operador de predicado é uma expressão que deve ser avaliada como sendo verdadeira. A comparação de valores `MISSING` e `NULL` segue estas regras:

- Um valor `MISSING` só é igual a um valor `MISSING` e é menor do que outros valores.
- Um valor `NULL` é igual a um valor `NULL`, é maior que um valor `MISSING`, mas é menor que todos os outros valores.

Operadores

Operadores de predicado

Name (Nome)	Description
`>`	Operador maior que
`>=`	Operador maior que ou igual a
`<`	Operador menor que
`!=`	Operador não igual a
`<=`	Operador menor que ou igual a
`=`	Operador igual a
`LIKE`	Correspondência a padrão simples
`IN`	Teste de valor NULL
`AND`	Operador AND

Name (Nome)	Description
OR	Operador OU
XOR	Operador XOR
NOT	Teste de valor NOT NULL

Você pode comparar datas e horas. Ao comparar diferentes tipos de data e hora (por exemplo DATE e TIME), ambos são convertidos em DATETIME. As seguintes regras se aplicam à conversão:

- TIME aplica-se até a data de hoje.
- DATE é interpretado à meia-noite.

Operador de predicado básico

Exemplo de operadores de comparação:

```
os> source=accounts | where age > 33 | fields age ;
fetched rows / total rows = 1/1
+-------+
| age   |
|-------|
| 36    |
+-------+
```

IN

Exemplo do campo de teste do operador IN em listas de valores:

```
os> source=accounts | where age in (32, 33) | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

OR

Exemplo do operador OR:

```
os> source=accounts | where age = 32 OR age = 33 | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

NOT

Exemplo do operador NOT:

```
os> source=accounts | where age not in (32, 33) | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 36    |
| 28    |
+-------+
```

Funções PPL de endereço IP

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

CIDRMATCH

Uso: `CIDRMATCH(ip, cidr)` verifica se o endereço IP especificado está dentro do intervalo cidr fornecido.

Tipo de argumento:

- STRING, STRING
- Tipo de retorno: BOOLEAN

Exemplo:

```
os> source=ips | where cidrmatch(ip, '***********/24') | fields ip
fetched rows / total rows = 1/1
+--------------+
| ip           |
|--------------|
| ***********  |
+--------------+

os> source=ipsv6 | where cidrmatch(ip, '2003:db8::/32') | fields ip
fetched rows / total rows = 1/1
+-----------------------------------------+
| ip                                      |
|-----------------------------------------|
| 2003:0db8:****:****:****:****:****:0000 |
+-----------------------------------------+
```

> **Note**
>
> - `ip`pode ser um IPv4 ou um IPv6 endereço.
> - `cidr`pode ser um IPv4 ou um IPv6 bloco.
> - `ip`e `cidr` deve ser ambos IPv4 ou ambos IPv6.
> - `ip` e `cidr` devem ser válidos e não vazios/não nulos.

Funções PPL de JSON

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

`JSON`

Uso: `json(value)` avalia se uma string pode ser analisada como formato JSON. A função retornará a string original se for JSON válido ou retornará nulo se for inválida.

Tipo de argumento: STRING

Tipo de retorno: STRING/NULL. Uma expressão STRING de um formato de objeto JSON válido.

Exemplos:

```
os> source=people | eval `valid_json()` = json('[1,2,3,{"f1":1,"f2":[5,6]},4]') |
 fields valid_json
fetched rows / total rows = 1/1
+-----------------------------------+
| valid_json                        |
+-----------------------------------+
| [1,2,3,{"f1":1,"f2":[5,6]},4]   |
+-----------------------------------+

os> source=people | eval `invalid_json()` = json('{"invalid": "json"') | fields
 invalid_json
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_OBJECT

Uso: `json_object(<key>, <value>[, <key>, <value>]...)` retorna um objeto JSON de membros de pares chave-valor.

Tipo de argumento:

- Uma **<key>** deve ser STRING.
- Um **<value>** pode ser qualquer tipo de dados.

Tipo de retorno: JSON_OBJECT. Uma StructType expressão de um objeto JSON válido.

Exemplos:

```
os> source=people | eval result = json_object('key', 123.45) | fields result
fetched rows / total rows = 1/1
+------------------+
| result           |
+------------------+
| {"key":123.45}   |
```

```
+------------------+

os> source=people | eval result = json_object('outer', json_object('inner', 123.45)) |
 fields result
fetched rows / total rows = 1/1
+------------------------------+
| result                       |
+------------------------------+
| {"outer":{"inner":123.45}}   |
+------------------------------+
```

JSON_ARRAY

Uso: `json_array(<value>...)` cria um JSON ARRAY usando uma lista de valores.

Tipo de argumento: um `<value>` pode ser qualquer tipo de valor, como string, número ou booleano.

Tipo de retorno: ARRAY. Uma matriz de qualquer tipo de dado compatível com uma matriz JSON válida.

Exemplos:

```
os> source=people | eval `json_array` = json_array(1, 2, 0, -1, 1.1, -0.11)
fetched rows / total rows = 1/1
+------------------------------+
| json_array                   |
+------------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11] |
+------------------------------+

os> source=people | eval `json_array_object` = json_object("array", json_array(1, 2, 0,
 -1, 1.1, -0.11))
fetched rows / total rows = 1/1
+---------------------------------------+
| json_array_object                     |
+---------------------------------------+
| {"array":[1.0,2.0,0.0,-1.0,1.1,-0.11]} |
+---------------------------------------+
```

TO_JSON_STRING

Uso: `to_json_string(json0bject)` retorna uma string JSON com um valor de objeto json fornecido.

Tipo de argumento: JSON_OBJECT

Tipo de retorno: STRING

Exemplos:

```
os> source=people | eval `json_string` = to_json_string(json_array(1, 2, 0, -1, 1.1,
 -0.11)) | fields json_string
fetched rows / total rows = 1/1
+--------------------------------+
| json_string                    |
+--------------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11]   |
+--------------------------------+

os> source=people | eval `json_string` = to_json_string(json_object('key', 123.45)) |
 fields json_string
fetched rows / total rows = 1/1
+-----------------+
| json_string     |
+-----------------+
| {'key', 123.45} |
+-----------------+
```

ARRAY_LENGTH

Usage: `array_length(jsonArray)` retorna o número de elementos na matriz mais externa.

Tipo de argumento: ARRAY. Um objeto ARRAY ou JSON_ARRAY.

Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `json_array` = json_array_length(json_array(1,2,3,4)),
 `empty_array` = json_array_length(json_array())
fetched rows / total rows = 1/1
+--------------+---------------+
| json_array   | empty_array   |
+--------------+---------------+
| 4            | 0             |
+--------------+---------------+
```

JSON_EXTRACT

Uso: `json_extract(jsonStr, path)` extrai um objeto JSON de uma string JSON com base no caminho JSON especificado. A função retornará nulo se a string JSON de entrada for inválida.

Tipo de argumento: STRING, STRING

Tipo de retorno: STRING

- Uma expressão STRING de um formato de objeto JSON válido.
- NULL é retornado no caso de um JSON inválido.

Exemplos:

```
os> source=people | eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}',
 '$a')
fetched rows / total rows = 1/1
+----------------------------------+
| json_extract('{"a":"b"}', 'a')   |
+----------------------------------+
| b                                |
+----------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')
fetched rows / total rows = 1/1
+------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[1].b')   |
+------------------------------------------------------------+
| 2.0                                                        |
+------------------------------------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')
fetched rows / total rows = 1/1
+------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[*].b')   |
+------------------------------------------------------------+
| [1.0,2.0]                                                  |
+------------------------------------------------------------+

os> source=people | eval `invalid_json` = json_extract('{"invalid": "json"')
```

```
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_KEYS

Uso: `json_keys(jsonStr)` retorna todas as chaves do objeto JSON mais externo como uma matriz.

Tipo de argumento: STRING. Uma expressão STRING de um formato de objeto JSON válido.

Tipo de retorno: ARRAY[STRING]. A função retorna `NULL` para qualquer outra string JSON válida, uma string vazia ou um JSON inválido.

Exemplos:

```
os> source=people | eval `keys` = json_keys('{"f1":"abc","f2":{"f3":"a","f4":"b"}}')
fetched rows / total rows = 1/1
+------------+
| keus       |
+------------+
| [f1, f2]   |
+------------+

os> source=people | eval `keys` = json_keys('[1,2,3,{"f1":1,"f2":[5,6]},4]')
fetched rows / total rows = 1/1
+--------+
| keys   |
+--------+
| null   |
+--------+
```

JSON_VALID

Uso: `json_valid(jsonStr)` avalia se uma string JSON usa uma sintaxe JSON válida e retorna VERDADEIRO ou FALSO.

Tipo de argumento: STRING

Tipo de retorno: BOOLEAN

Exemplos:

```
os> source=people | eval `valid_json` = json_valid('[1,2,3,4]'), `invalid_json` =
 json_valid('{"invalid": "json"') | feilds `valid_json`, `invalid_json`
fetched rows / total rows = 1/1
+--------------+----------------+
| valid_json   | invalid_json   |
+--------------+----------------+
| True         | False          |
+--------------+----------------+

os> source=accounts | where json_valid('[1,2,3,4]') and isnull(email) | fields
 account_number, email
fetched rows / total rows = 1/1
+------------------+---------+
| account_number   | email   |
|------------------+---------|
| 13               | null    |
+------------------+---------+
```

Funções PPL do Lambda

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

`EXISTS`

Uso: `exists(array, lambda)` avalia se um predicado Lambda contém um ou mais elementos da matriz.

Tipo de argumento: ARRAY, LAMBDA

Tipo de retorno: BOOLEAN. Retornará `TRUE` se pelo menos um elemento na matriz satisfizer o predicado Lambda, do contrário `FALSE`.

Exemplos:

```
os> source=people | eval array = json_array(1, -1, 2), result = exists(array, x -> x >
0) | fields result
```

```
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| true      |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = exists(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| false     |
+-----------+
```

FILTER

Uso: `filter(array, lambda)` filtra a matriz de entrada usando a função do Lambda fornecida.

Tipo de argumento: ARRAY, LAMBDA

Tipo de retorno: ARRAY. Uma ARRAY que contém todos os elementos na matriz de entrada que satisfazem o predicado lambda.

Exemplos:

```
 os> source=people | eval array = json_array(1, -1, 2), result = filter(array, x -> x >
 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| [1, 2]    |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = filter(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| []        |
```

```
+-----------+
```

TRANSFORM

Uso: `transform(array, lambda)` transforma elementos em uma matriz usando a função do Lambda transform. O segundo argumento implica o índice do elemento se estiver usando a função do Lambda binary. Isso é semelhante a um `map` em programação funcional.

Tipo de argumento: ARRAY, LAMBDA

Tipo de retorno: ARRAY. Uma ARRAY que contém o resultado da aplicação da função do lambda transform a cada elemento na matriz de entrada.

Exemplos:

```
os> source=people | eval array = json_array(1, 2, 3), result = transform(array, x -> x
 + 1) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [2, 3, 4]    |
+--------------+

os> source=people | eval array = json_array(1, 2, 3), result = transform(array, (x, i)
 -> x + i) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [1, 3, 5]    |
+--------------+
```

REDUCE

Uso: `reduce(array, start, merge_lambda, finish_lambda)` reduz uma matriz a um único valor aplicando funções do lambda. A função aplica merge_lambda ao valor inicial e a todos os elementos da matriz e, em seguida, aplica `finish_lambda` ao resultado.

Tipo de argumento: ARRAY, ANY, LAMBDA, LAMBDA

Tipo de retorno: ANY. O resultado final da aplicação das funções Lambda ao valor inicial e à matriz de entrada.

Exemplos:

```
 os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 6         |
+-----------+

os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 10, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 16        |
+-----------+

os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x, acc -> acc * 10) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 60        |
+-----------+
```

Funções PPL matemáticas

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

ABS

Uso: `ABS(x)` calcula o valor absoluto de x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de devolução: INTEGER/LONG/FLOAT/DOUBLE

Exemplo:

```
os> source=people | eval `ABS(-1)` = ABS(-1) | fields `ABS(-1)`
fetched rows / total rows = 1/1
+-----------+
| ABS(-1)   |
|-----------|
| 1         |
+-----------+
```

ACOS

Uso: `ACOS(x)` calcula o arco cosseno de x. Retornará `NULL` se x não estiver no intervalo de -1 a 1.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `ACOS(0)` = ACOS(0) | fields `ACOS(0)`
fetched rows / total rows = 1/1
+--------------------+
| ACOS(0)            |
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

ASIN

Uso: `asin(x)` calcula o arco seno de x. Retornará `NULL` se x não estiver no intervalo de -1 a 1.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `ASIN(0)` = ASIN(0) | fields `ASIN(0)`
fetched rows / total rows = 1/1
+-----------+
```

```
| ASIN(0)   |
|-----------|
| 0.0       |
+-----------+
```

ATAN

Uso: `ATAN(x)` calcula o arco tangente de x. `atan(y, x)` calcula o arco tangente de y/x, exceto que os sinais dos argumentos determinam o quadrante do resultado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `ATAN(2)` = ATAN(2), `ATAN(2, 3)` = ATAN(2, 3) | fields
 `ATAN(2)`, `ATAN(2, 3)`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| ATAN(2)            | ATAN(2, 3)         |
|--------------------+--------------------|
| 1.1071487177940904 | 0.5880026035475675 |
+--------------------+--------------------+
```

ATAN2

Uso: `ATAN2(y, x)` calcula o arco tangente de y/x, exceto que os sinais dos argumentos determinam o quadrante do resultado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `ATAN2(2, 3)` = ATAN2(2, 3) | fields `ATAN2(2, 3)`
fetched rows / total rows = 1/1
+--------------------+
| ATAN2(2, 3)        |
|--------------------|
| 0.5880026035475675 |
+--------------------+
```

CBRT

Usage: CBRT calcula a raiz cúbica de um número.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE:

INTEGER/LONG/FLOAT/DOUBLE-> DUPLO

Exemplo:

```
opensearchsql> source=location | eval `CBRT(8)` = CBRT(8), `CBRT(9.261)` = CBRT(9.261),
 `CBRT(-27)` = CBRT(-27) | fields `CBRT(8)`, `CBRT(9.261)`, `CBRT(-27)`;
fetched rows / total rows = 2/2
+-----------+---------------+-------------+
| CBRT(8)   | CBRT(9.261)   | CBRT(-27)   |
|-----------+---------------+-------------|
| 2.0       | 2.1           | -3.0        |
| 2.0       | 2.1           | -3.0        |
+-----------+---------------+-------------+
```

CEIL

Uso: um alias da função CEILING. CEILING(T) assume o teto do valor T.

Limitação: CEILING só funciona conforme o esperado quando o tipo duplo IEEE 754 exibe um decimal quando armazenado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: LONG

Exemplo:

```
os> source=people | eval `CEILING(0)` = CEILING(0), `CEILING(50.00005)` =
 CEILING(50.00005), `CEILING(-50.00005)` = CEILING(-50.00005) | fields `CEILING(0)`,
 `CEILING(50.00005)`, `CEILING(-50.00005)`
fetched rows / total rows = 1/1
+--------------+---------------------+----------------------+
| CEILING(0)   | CEILING(50.00005)   | CEILING(-50.00005)   |
|--------------+---------------------+----------------------|
| 0            | 51                  | -50                  |
+--------------+---------------------+----------------------+
```

```
os> source=people | eval `CEILING(3147483647.12345)` = CEILING(3147483647.12345),
 `CEILING(113147483647.12345)` = CEILING(113147483647.12345),
 `CEILING(3147483647.00001)` = CEILING(3147483647.00001) | fields
 `CEILING(3147483647.12345)`, `CEILING(113147483647.12345)`,
 `CEILING(3147483647.00001)`
fetched rows / total rows = 1/1
+-----------------------------+-------------------------------
+-----------------------------+
| CEILING(3147483647.12345)   | CEILING(113147483647.12345)   |
 CEILING(3147483647.00001)   |
|-----------------------------+-------------------------------
+-----------------------------|
| 3147483648                  | 113147483648                  | 3147483648
     |
+-----------------------------+-------------------------------
+-----------------------------+
```

CONV

Uso: `CONV(x, a, b)` converte o número x de uma base para a base b.

Tipo de argumento: x: STRING, a: INTEGER, b: INTEGER

Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `CONV('12', 10, 16)` = CONV('12', 10, 16), `CONV('2C', 16,
 10)` = CONV('2C', 16, 10), `CONV(12, 10, 2)` = CONV(12, 10, 2), `CONV(1111, 2, 10)` =
 CONV(1111, 2, 10) | fields `CONV('12', 10, 16)`, `CONV('2C', 16, 10)`, `CONV(12, 10,
 2)`, `CONV(1111, 2, 10)`
fetched rows / total rows = 1/1
+----------------------+----------------------+-------------------
+----------------------+
| CONV('12', 10, 16)   | CONV('2C', 16, 10)   | CONV(12, 10, 2)   | CONV(1111, 2, 10)
 |
|----------------------+----------------------+-------------------
+----------------------|
| c                    | 44                   | 1100              | 15
 |
+----------------------+----------------------+-------------------
+----------------------+
```

COS

Uso: `COS(x)` calcula o cosseno de x, onde x é fornecido em radianos.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `COS(0)` = COS(0) | fields `COS(0)`
fetched rows / total rows = 1/1
+----------+
| COS(0)   |
|----------|
| 1.0      |
+----------+
```

COT

Uso: `COT(x)` calcula a cotangente de x. Ele retornará out-of-range um erro se x for igual a 0.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `COT(1)` = COT(1) | fields `COT(1)`
fetched rows / total rows = 1/1
+--------------------+
| COT(1)             |
|--------------------|
| 0.6420926159343306 |
+--------------------+
```

CRC32

Uso: `CRC32` calcula um valor de verificação de redundância cíclica e retorna um valor não assinado de 32 bits.

Tipo de argumento: STRING

Tipo de retorno: LONG

Exemplo:

```
os> source=people | eval `CRC32('MySQL')` = CRC32('MySQL') | fields `CRC32('MySQL')`
fetched rows / total rows = 1/1
+------------------+
| CRC32('MySQL')   |
|------------------|
| 3259397556       |
+------------------+
```

DEGREES

Uso: `DEGREES(x)` converte x de radianos em graus.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `DEGREES(1.57)` = DEGREES(1.57) | fields `DEGREES(1.57)`
fetched rows / total rows  = 1/1
+-------------------+
| DEGREES(1.57)     |
|-------------------|
| 89.95437383553924 |
+-------------------+
```

E

Uso: `E()` retorna o número de Euler.

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `E()` = E() | fields `E()`
fetched rows / total rows = 1/1
+-------------------+
| E()               |
|-------------------|
| 2.718281828459045 |
+-------------------+
```

EXP

Uso: `EXP(x)` retorna e elevado à potência de x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `EXP(2)` = EXP(2) | fields `EXP(2)`
fetched rows / total rows = 1/1
+------------------+
| EXP(2)           |
|------------------|
| 7.38905609893065 |
+------------------+
```

FLOOR

Uso: `FLOOR(T)` assume o piso do valor T.

Limitação: `FLOOR` só funciona conforme o esperado quando o tipo duplo IEEE 754 exibe um decimal quando armazenado.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: LONG

Exemplo:

```
os> source=people | eval `FLOOR(0)` = FLOOR(0), `FLOOR(50.00005)` = FLOOR(50.00005),
 `FLOOR(-50.00005)` = FLOOR(-50.00005) | fields `FLOOR(0)`, `FLOOR(50.00005)`,
 `FLOOR(-50.00005)`
fetched rows / total rows = 1/1
+------------+------------------+--------------------+
| FLOOR(0)   | FLOOR(50.00005)   | FLOOR(-50.00005)   |
|------------+------------------+--------------------|
| 0          | 50                | -51                |
+------------+------------------+--------------------+

os> source=people | eval `FLOOR(3147483647.12345)` = FLOOR(3147483647.12345),
 `FLOOR(113147483647.12345)` = FLOOR(113147483647.12345), `FLOOR(3147483647.00001)`
```

```
 = FLOOR(3147483647.00001) | fields `FLOOR(3147483647.12345)`,
 `FLOOR(113147483647.12345)`, `FLOOR(3147483647.00001)`
fetched rows / total rows = 1/1
+---------------------------+-----------------------------+---------------------------+
| FLOOR(3147483647.12345)   | FLOOR(113147483647.12345)   | FLOOR(3147483647.00001)   |
|---------------------------+-----------------------------+---------------------------|
| 3147483647                | 113147483647                | 3147483647                |
+---------------------------+-----------------------------+---------------------------+

os> source=people | eval `FLOOR(282474973688888.022)` = FLOOR(282474973688888.022),
 `FLOOR(9223372036854775807.022)` = FLOOR(9223372036854775807.022),
 `FLOOR(9223372036854775807.0000001)` = FLOOR(9223372036854775807.0000001)
 | fields `FLOOR(282474973688888.022)`, `FLOOR(9223372036854775807.022)`,
 `FLOOR(9223372036854775807.0000001)`
fetched rows / total rows = 1/1
+-----------------------------+----------------------------------
+-------------------------------------+
| FLOOR(282474973688888.022)   | FLOOR(9223372036854775807.022)   |
 FLOOR(9223372036854775807.0000001)   |
|-----------------------------+----------------------------------
+-------------------------------------|
| 282474973688888              | 9223372036854775807              | 9223372036854775807
                  |
+-----------------------------+----------------------------------
+-------------------------------------+
```

LN

Uso: `LN(x)` retorna o logaritmo natural de x.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `LN(2)` = LN(2) | fields `LN(2)`
fetched rows / total rows = 1/1
+--------------------+
| LN(2)              |
|--------------------|
| 0.6931471805599453 |
+--------------------+
```

LOG

Uso: `LOG(x)` retorna o logaritmo natural de x que é o algoritmo de base e de x. log(B, x) é equivalente a log(x)/log(B).

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `LOG(2)` = LOG(2), `LOG(2, 8)` = LOG(2, 8) | fields `LOG(2)`,
 `LOG(2, 8)`
fetched rows / total rows = 1/1
+--------------------+-------------+
| LOG(2)             | LOG(2, 8)   |
|--------------------+-------------|
| 0.6931471805599453 | 3.0         |
+--------------------+-------------+
```

LOG2

Uso: `LOG2(x)` é equivalente a `log(x)/log(2)`.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `LOG2(8)` = LOG2(8) | fields `LOG2(8)`
fetched rows / total rows = 1/1
+-----------+
| LOG2(8)   |
|-----------|
| 3.0       |
+-----------+
```

LOG10

Uso: `LOG10(x)` é equivalente a `log(x)/log(10)`.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `LOG10(100)` = LOG10(100) | fields `LOG10(100)`
fetched rows / total rows = 1/1
+--------------+
| LOG10(100)   |
|--------------|
| 2.0          |
+--------------+
```

MOD

Uso: `MOD(n, m)` calcula o resto do número n dividido por m.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: tipo mais amplo entre os tipos de n e m se m for um valor diferente de zero. Se m for igual a 0, retornará NULL.

Exemplo:

```
os> source=people | eval `MOD(3, 2)` = MOD(3, 2), `MOD(3.1, 2)` = MOD(3.1, 2) | fields
 `MOD(3, 2)`, `MOD(3.1, 2)`
fetched rows / total rows = 1/1
+-------------+---------------+
| MOD(3, 2)   | MOD(3.1, 2)   |
|-------------+---------------|
| 1           | 1.1           |
+-------------+---------------+
```

PI

Uso: `PI()` retorna a constante pi.

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `PI()` = PI() | fields `PI()`
fetched rows / total rows = 1/1
+-------------------+
```

```
| PI()              |
|-------------------|
| 3.141592653589793 |
+-------------------+
```

POW

Uso: POW(x, y) calcula o valor de x elevado à potência de y. Entradas incorretas retornam um resultado NULL.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Sinônimos: POWER(_, _)

Exemplo:

```
os> source=people | eval `POW(3, 2)` = POW(3, 2), `POW(-3, 2)` = POW(-3, 2), `POW(3,
 -2)` = POW(3, -2) | fields `POW(3, 2)`, `POW(-3, 2)`, `POW(3, -2)`
fetched rows / total rows = 1/1
+-------------+--------------+--------------------+
| POW(3, 2)   | POW(-3, 2)   | POW(3, -2)         |
|-------------+--------------+--------------------|
| 9.0         | 9.0          | 0.1111111111111111 |
+-------------+--------------+--------------------+
```

POWER

Uso: POWER(x, y) calcula o valor de x elevado à potência de y. Entradas incorretas retornam um resultado NULL.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Sinônimos: POW(_, _)

Exemplo:

```
os> source=people | eval `POWER(3, 2)` = POWER(3, 2), `POWER(-3, 2)` = POWER(-3, 2),
 `POWER(3, -2)` = POWER(3, -2) | fields `POWER(3, 2)`, `POWER(-3, 2)`, `POWER(3, -2)`
fetched rows / total rows = 1/1
```

```
+---------------+----------------+--------------------+
| POWER(3, 2)   | POWER(-3, 2)   | POWER(3, -2)       |
|---------------+----------------+--------------------|
| 9.0           | 9.0            | 0.1111111111111111 |
+---------------+----------------+--------------------+
```

RADIANS

Uso: `RADIANS(x)` converte x de graus em radianos.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `RADIANS(90)` = RADIANS(90) | fields `RADIANS(90)`
fetched rows / total rows  = 1/1
+--------------------+
| RADIANS(90)        |
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

RAND

Uso: `RAND()`/`RAND(N)` retorna um valor de ponto flutuante aleatório no intervalo 0 <= valor < 1,0. Se você especificar o inteiro N, a função inicializará a semente antes da execução. Uma implicação desse comportamento é que, com um argumento N idêntico, `rand(N)` retorna o mesmo valor toda vez, produzindo uma sequência repetível de valores de coluna.

Tipo de argumento: INTEGER

Tipo de retorno: FLOAT

Exemplo:

```
os> source=people | eval `RAND(3)` = RAND(3) | fields `RAND(3)`
fetched rows / total rows = 1/1
+------------+
| RAND(3)    |
|------------|
| 0.73105735 |
```

```
+------------+
```

ROUND

Uso: `ROUND(x, d)` arredonda o argumento x para d casas decimais. Se você não especificar d, o padrão será 0.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Retorno tipo mapa:

- (INTEGER/LONG [,INTEGER]) -> LONG
- (FLOAT/DOUBLE [,INTEGER]) -> LONG

Exemplo:

```
os> source=people | eval `ROUND(12.34)` = ROUND(12.34), `ROUND(12.34, 1)` =
 ROUND(12.34, 1), `ROUND(12.34, -1)` = ROUND(12.34, -1), `ROUND(12, 1)` = ROUND(12, 1)
 | fields `ROUND(12.34)`, `ROUND(12.34, 1)`, `ROUND(12.34, -1)`, `ROUND(12, 1)`
fetched rows / total rows = 1/1
+----------------+-------------------+--------------------+----------------+
| ROUND(12.34)   | ROUND(12.34, 1)   | ROUND(12.34, -1)   | ROUND(12, 1)   |
|----------------+-------------------+--------------------+----------------|
| 12.0           | 12.3              | 10.0               | 12             |
+----------------+-------------------+--------------------+----------------+
```

SIGN

Uso: `SIGN` retorna o sinal do argumento como -1, 0 ou 1, dependendo de o número ser negativo, zero ou positivo.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `SIGN(1)` = SIGN(1), `SIGN(0)` = SIGN(0), `SIGN(-1.1)` =
 SIGN(-1.1) | fields `SIGN(1)`, `SIGN(0)`, `SIGN(-1.1)`
fetched rows / total rows = 1/1
+-----------+-----------+--------------+
```

```
| SIGN(1)   | SIGN(0)   | SIGN(-1.1)   |
|-----------+-----------+--------------|
| 1         | 0         | -1           |
+-----------+-----------+--------------+
```

SIN

Uso: `sin(x)` calcula o seno de x, onde x é fornecido em radianos.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Tipo de retorno: DOUBLE

Exemplo:

```
os> source=people | eval `SIN(0)` = SIN(0) | fields `SIN(0)`
fetched rows / total rows = 1/1
+----------+
| SIN(0)   |
|----------|
| 0.0      |
+----------+
```

SQRT

Uso: SQRT calcula a raiz quadrada de um número não negativo.

Tipo de argumento: INTEGER/LONG/FLOAT/DOUBLE

Retorno tipo mapa:

- (Não negativo) INTEGER/LONG/FLOAT/DOUBLE -> DUPLO
- (Negativo) INTEGER/LONG/FLOAT/DOUBLE -> NULL

Exemplo:

```
os> source=people | eval `SQRT(4)` = SQRT(4), `SQRT(4.41)` = SQRT(4.41) | fields
 `SQRT(4)`, `SQRT(4.41)`
fetched rows / total rows = 1/1
+-----------+--------------+
| SQRT(4)   | SQRT(4.41)   |
|-----------+--------------|
```

```
| 2.0       | 2.1          |
+-----------+--------------+
```

Funções PPL de string

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

`CONCAT`

Uso: `CONCAT(str1, str2, ...., str_9)` soma até 9 strings.

Tipo de argumento:

- STRING, STRING, ..., STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `CONCAT('hello', 'world')` = CONCAT('hello', 'world'),
 `CONCAT('hello ', 'whole ', 'world', '!')` = CONCAT('hello ', 'whole ', 'world', '!')
 | fields `CONCAT('hello', 'world')`, `CONCAT('hello ', 'whole ', 'world', '!')`
fetched rows / total rows = 1/1
+----------------------------+---------------------------------------------+
| CONCAT('hello', 'world')   | CONCAT('hello ', 'whole ', 'world', '!')    |
|----------------------------+---------------------------------------------|
| helloworld                 | hello whole world!                          |
+----------------------------+---------------------------------------------+
```

`CONCAT_WS`

Uso: `CONCAT_WS(sep, str1, str2)` concatena duas ou mais strings usando um separador especificado entre elas.

Tipo de argumento:

- STRING, STRING, ..., STRING

- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `CONCAT_WS(',', 'hello', 'world')` = CONCAT_WS(',', 'hello',
 'world') | fields `CONCAT_WS(',', 'hello', 'world')`
fetched rows / total rows = 1/1
+-----------------------------------+
| CONCAT_WS(',', 'hello', 'world')   |
|-----------------------------------|
| hello,world                       |
+-----------------------------------+
```

LENGTH

Uso: `length(str)` retorna o comprimento da string de entrada medido em bytes.

Tipo de argumento:

- STRING
- Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `LENGTH('helloworld')` = LENGTH('helloworld') | fields
 `LENGTH('helloworld')`
fetched rows / total rows = 1/1
+------------------------+
| LENGTH('helloworld')   |
|------------------------|
| 10                     |
+------------------------+
```

LOWER

Uso: `lower(string)` converte a string de entrada para minúsculas.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `LOWER('helloworld')` = LOWER('helloworld'),
 `LOWER('HELLOWORLD')` = LOWER('HELLOWORLD') | fields `LOWER('helloworld')`,
 `LOWER('HELLOWORLD')`
fetched rows / total rows = 1/1
+----------------------+----------------------+
| LOWER('helloworld')   | LOWER('HELLOWORLD')   |
|----------------------+----------------------|
| helloworld            | helloworld            |
+----------------------+----------------------+
```

LTRIM

Uso: `ltrim(str)` remove os caracteres de espaço antes da string de entrada.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `LTRIM('   hello')` = LTRIM('   hello'), `LTRIM('hello   ')` =
 LTRIM('hello   ') | fields `LTRIM('   hello')`, `LTRIM('hello   ')`
fetched rows / total rows = 1/1
+---------------------+---------------------+
| LTRIM('   hello')   | LTRIM('hello   ')   |
|---------------------+---------------------|
| hello               | hello               |
+---------------------+---------------------+
```

POSITION

Uso: `POSITION(substr IN str)` retorna a posição da primeira ocorrência de substring na string. Retornará 0 se a substring não estiver na string. Retornará NULL se qualquer argumento de entrada for NULL.

Tipo de argumento:

- STRING, STRING

- Tipo de retorno: INTEGER

Exemplo:

```
os> source=people | eval `POSITION('world' IN 'helloworld')` = POSITION('world'
 IN 'helloworld'), `POSITION('invalid' IN 'helloworld')`= POSITION('invalid' IN
 'helloworld')  | fields `POSITION('world' IN 'helloworld')`, `POSITION('invalid' IN
 'helloworld')`
fetched rows / total rows = 1/1
+-------------------------------------+---------------------------------------+
| POSITION('world' IN 'helloworld')   | POSITION('invalid' IN 'helloworld')   |
|-------------------------------------+---------------------------------------|
| 6                                   | 0                                     |
+-------------------------------------+---------------------------------------+
```

REVERSE

Uso: `REVERSE(str)` retorna a string inversa da string de entrada.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `REVERSE('abcde')` = REVERSE('abcde') | fields
 `REVERSE('abcde')`
fetched rows / total rows = 1/1
+--------------------+
| REVERSE('abcde')   |
|--------------------|
| edcba              |
+--------------------+
```

RIGHT

Uso: `right(str, len)` retorna os caracteres mais à direita da string de entrada. Retornará 0 se a substring não estiver na string. Retornará NULL se qualquer argumento de entrada for NULL.

Tipo de argumento:

- STRING, INTEGER
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `RIGHT('helloworld', 5)` = RIGHT('helloworld', 5),
 `RIGHT('HELLOWORLD', 0)` = RIGHT('HELLOWORLD', 0) | fields `RIGHT('helloworld', 5)`,
 `RIGHT('HELLOWORLD', 0)`
fetched rows / total rows = 1/1
+-------------------------+-------------------------+
| RIGHT('helloworld', 5)  | RIGHT('HELLOWORLD', 0)  |
|-------------------------+-------------------------|
| world                   |                         |
+-------------------------+-------------------------+
```

RTRIM

Uso: `rtrim(str)` corta os caracteres de espaço depois da string de entrada.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `RTRIM('   hello')` = RTRIM('   hello'), `RTRIM('hello   ')` =
 RTRIM('hello   ') | fields `RTRIM('   hello')`, `RTRIM('hello   ')`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| RTRIM('   hello')  | RTRIM('hello   ')  |
|--------------------+--------------------|
|    hello           | hello              |
+--------------------+--------------------+
```

SUBSTRING

Uso: `substring(str, start)` ou `substring(str, start, length)` retorna uma substring da string de entrada. Sem comprimento especificado, retorna a string inteira a partir da posição inicial.

Tipo de argumento:

- STRING, INTEIRO, INTEIRO
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `SUBSTRING('helloworld', 5)` = SUBSTRING('helloworld',
 5), `SUBSTRING('helloworld', 5, 3)` = SUBSTRING('helloworld', 5, 3) | fields
 `SUBSTRING('helloworld', 5)`, `SUBSTRING('helloworld', 5, 3)`
fetched rows / total rows = 1/1
+------------------------------+---------------------------------+
| SUBSTRING('helloworld', 5)   | SUBSTRING('helloworld', 5, 3)   |
|------------------------------+---------------------------------|
| oworld                       | owo                             |
+------------------------------+---------------------------------+
```

`TRIM`

Uso: `trim(string)` remove os espaços em branco antes e depois da string de entrada.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `TRIM('   hello')` = TRIM('   hello'), `TRIM('hello   ')` =
 TRIM('hello   ') | fields `TRIM('   hello')`, `TRIM('hello   ')`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| TRIM('   hello')   | TRIM('hello   ')   |
|--------------------+--------------------|
| hello              | hello              |
+--------------------+--------------------+
```

`UPPER`

Uso: `upper(string)` converte a string de entrada para maiúsculas.

Tipo de argumento:

- STRING
- Tipo de retorno: STRING

Exemplo:

```
os> source=people | eval `UPPER('helloworld')` = UPPER('helloworld'),
 `UPPER('HELLOWORLD')` = UPPER('HELLOWORLD') | fields `UPPER('helloworld')`,
 `UPPER('HELLOWORLD')`
fetched rows / total rows = 1/1
+----------------------+----------------------+
| UPPER('helloworld')   | UPPER('HELLOWORLD')   |
|----------------------+----------------------|
| HELLOWORLD            | HELLOWORLD            |
+----------------------+----------------------+
```

Funções PPL de conversão de tipo

> **Note**
>
> Para ver quais integrações AWS de fontes de dados oferecem suporte a essa função PPL, consulte. the section called “Funções”

`TRIM`

Uso: `cast(expr as dateType)` converte `expr` em `dataType` e retorna o valor de `dataType`.

As seguintes regras de conversão se aplicam:

Regras de conversão de tipo

Src/Target	STRING	NUMBER	BOOLEAN	TIMESTAMP	DATE	TIME
STRING		Note1	Note1	TIMESTAMP()	DATE()	TIME()
NUMBER	Note1		v!=0	N/D	N/D	N/D

Src/Target	STRING	NUMBER	BOOLEAN	TIMESTAMP	DATE	TIME
BOOLEAN	Note1	v?1:0		N/D	N/D	N/D
TIMESTAMP	Note1	N/D	N/D		DATE()	TIME()
DATE	Note1	N/D	N/D	N/D		N/D
TIME	Note1	N/D	N/D	N/D	N/D	

Exemplo de conversão em string:

```
os> source=people | eval `cbool` = CAST(true as string), `cint` = CAST(1 as string),
 `cdate` = CAST(CAST('2012-08-07' as date) as string) | fields `cbool`, `cint`, `cdate`
fetched rows / total rows = 1/1
+---------+--------+------------+
| cbool   | cint   | cdate      |
|---------+--------+------------|
| true    | 1      | 2012-08-07 |
+---------+--------+------------+
```

Exemplo de conversão em número:

```
os> source=people | eval `cbool` = CAST(true as int), `cstring` = CAST('1' as int) |
 fields `cbool`, `cstring`
fetched rows / total rows = 1/1
+---------+-----------+
| cbool   | cstring   |
|---------+-----------|
| 1       | 1         |
+---------+-----------+
```

Exemplo de conversão em data:

```
os> source=people | eval `cdate` = CAST('2012-08-07' as date), `ctime` =
 CAST('01:01:01' as time), `ctimestamp` = CAST('2012-08-07 01:01:01' as timestamp) |
 fields `cdate`, `ctime`, `ctimestamp`
fetched rows / total rows = 1/1
+------------+----------+---------------------+
| cdate      | ctime    | ctimestamp          |
|------------+----------+---------------------|
```

```
| 2012-08-07 | 01:01:01 | 2012-08-07 01:01:01 |
+------------+----------+---------------------+
```

Exemplo de conversão encadeada:

```
os> source=people | eval `cbool` = CAST(CAST(true as string) as boolean) | fields
 `cbool`
fetched rows / total rows = 1/1
+---------+
| cbool   |
|---------|
| True    |
+---------+
```

Monitoramento de domínios OpenSearch do Amazon Service

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho do Amazon OpenSearch Service e de suas outras AWS soluções. AWS fornece as seguintes ferramentas para monitorar seus recursos OpenSearch de serviço, relatar problemas e tomar ações automáticas quando apropriado:

Amazon CloudWatch

A Amazon CloudWatch monitora seus recursos de OpenSearch serviço em tempo real. É possível coletar e rastrear métricas, criar painéis personalizados e definir alarmes que notificam você ou realizam ações quando uma métrica atinge um determinado limite. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.

CloudWatch Registros da Amazon

O Amazon CloudWatch Logs permite monitorar, armazenar e acessar seus arquivos de OpenSearch log. CloudWatch O Logs monitora as informações nos arquivos de log e pode notificá-lo quando determinados limites são atingidos. Para obter mais informações, consulte o Guia do usuário do Amazon CloudWatch Logs.

Amazon EventBridge

A Amazon EventBridge fornece um fluxo quase em tempo real de eventos do sistema que descrevem as mudanças em seus domínios OpenSearch de serviço. Você pode criar regras que observem determinados eventos e acionem ações automatizadas em outros AWS serviços quando esses eventos ocorrerem. Para obter mais informações, consulte o Guia EventBridge do usuário da Amazon.

AWS CloudTrail

AWS CloudTrail captura as chamadas de API de configuração feitas ao OpenSearch Serviço como eventos. Ele pode enviar esses eventos para um bucket do Amazon S3 especificado por você. Usando essas informações, você pode identificar quais usuários e contas fizeram solicitações, o endereço IP de origem de onde as solicitações foram feitas e quando elas ocorreram. Para obter mais informações, consulte o Guia do usuário do AWS CloudTrail .

Tópicos

- Monitorando métricas de OpenSearch cluster com a Amazon CloudWatch
- OpenSearch Registros de monitoramento com o Amazon CloudWatch Logs
- Monitorando registros de auditoria no Amazon OpenSearch Service
- Eventos do OpenSearch Serviço de Monitoramento com a Amazon EventBridge
- Monitorando chamadas OpenSearch de API do Amazon Service com AWS CloudTrail

Monitorando métricas de OpenSearch cluster com a Amazon CloudWatch

O Amazon OpenSearch Service publica dados de seus domínios na Amazon. CloudWatch CloudWatch permite recuperar estatísticas sobre esses pontos de dados como um conjunto ordenado de dados de séries temporais, conhecido como métricas. OpenSearch O serviço envia a maioria das métricas CloudWatch em intervalos de 60 segundos. Se você usar volumes magnéticos do EBS ou de uso geral, as métricas do volume do EBS serão atualizadas somente a cada cinco minutos. Todas as métricas cumulativas (por exemplo, `ThreadpoolWriteRejected` e `ThreadpoolSearchRejected`) estão na memória e perderão o estado. As métricas serão redefinidas durante a queda do nó, a rejeição do nó, a substituição do nó e a blue/green implantação. Para obter mais informações sobre a Amazon CloudWatch, consulte o Guia CloudWatch do usuário da Amazon.

O console OpenSearch de serviço exibe uma série de gráficos com base nos dados brutos de CloudWatch. Dependendo de suas necessidades, talvez você prefira visualizar os dados do cluster em CloudWatch vez dos gráficos no console. O serviço mantém as métricas arquivadas por duas semanas e depois as descarta. As métricas são fornecidas sem custo adicional, mas CloudWatch ainda cobram pela criação de painéis e alarmes. Para obter mais informações, consulte os CloudWatchpreços da Amazon.

OpenSearch O serviço publica as seguintes métricas para CloudWatch:

- the section called “Métricas de cluster”
- the section called “Métricas de nó principal dedicado”
- the section called “Métricas de volume do EBS”
- the section called “Métricas de instância”
- the section called “Métricas quentes”

- the section called “Métricas do nó Coordenador dedicado”
- the section called “Métricas de armazenamento de baixa atividade”
- the section called “Métricas de alerta”
- the section called “Métricas de detecção de anomalias”
- the section called “Métricas de pesquisa assíncrona”
- the section called “Métricas de SQL”
- the section called “Métricas de k-NN”
- the section called “Métricas de pesquisa entre clusters”
- the section called “Métricas de replicação entre clusters”
- the section called “Métricas de Learning to Rank”
- the section called “Métricas da Piped Processing Language”

Visualizando métricas em CloudWatch

CloudWatch as métricas são agrupadas primeiro pelo namespace do serviço e depois pelas várias combinações de dimensões em cada namespace.

Para visualizar métricas usando o CloudWatch console

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. No painel de navegação à esquerda, localize Métricas e escolha Todas as métricas. Selecione o OpenSearchServicenamespace ES/.
3. Escolha uma dimensão para visualizar as métricas correspondentes. As métricas para nós individuais estão na dimensão `ClientId, DomainName, NodeId`. As métricas de cluster estão na dimensão `Per-Domain, Per-Client Metrics`. Algumas métricas de nó são agregadas no nível do cluster e, portanto, incluídas em ambas as dimensões. As métricas de fragmentos estão na dimensão `ClientId, DomainName, NodeId, ShardRole`.

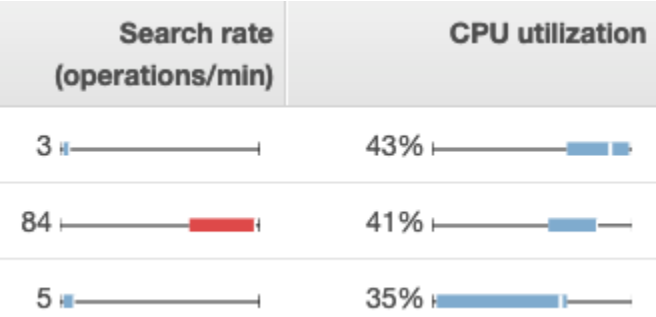
Para ver uma lista de métricas usando o AWS CLI

Execute este comando: .

```
aws cloudwatch list-metrics --namespace "AWS/ES"
```

Interpretando prontuários de saúde em serviço OpenSearch

Para visualizar métricas no OpenSearch Serviço, use as guias Integridade do cluster e Integridade da instância. A guia Integridade da instância usa gráficos de caixa para fornecer at-a-glance visibilidade da integridade de cada OpenSearch nó:



- Cada caixa colorida mostra a faixa de valores do nó ao longo do período de tempo especificado.
- As caixas azuis representam valores que são consistentes com outros nós. As caixas vermelhas representam exceções.
- A linha branca dentro de cada caixa de seleção mostra o valor atual do nó.
- As “caixas estreitas” em cada lado de cada caixa mostram os valores mínimo e máximo de todos os nós ao longo do período de tempo.

Se você fizer alterações de configuração para seu domínio, a lista de instâncias individuais nas guias Integridade do cluster e Integridade da instância geralmente duplicarão de tamanho por um breve período antes de retornar para o número correto. Para obter uma explicação sobre esse comportamento, consulte the section called “Alterações de configuração”.

Métricas de cluster

O Amazon OpenSearch Service fornece as seguintes métricas para clusters.

Métrica	Description
`ClusterStatus.green`	Um valor 1 indica que todos os fragmentos de índice estão alocados a nós no cluster. Estatística relevante: máximo

Métrica	Description
`ClusterStatus.yellow`	Um valor 1 indica que os fragmentos principais de todos os índices estão alocados a nós no cluster, mas os fragmentos de réplica de pelo menos um índice não estão. Para saber mais, consulte the section called “Status de cluster amarelo”. Estatística relevante: máximo
`ClusterStatus.red`	Um valor 1 indica que os fragmentos principais e de réplica de pelo menos um índice não estão alocados a nós no cluster. Para saber mais, consulte the section called “Status de cluster vermelho”. Estatística relevante: máximo
`Shards.active`	O número total de fragmentos ativos primários e de réplica. Estatística relevante: máximo, soma
`Shards.unassigned`	O número de fragmentos que não estão alocados a nós no cluster. Estatística relevante: máximo, soma
`Shards.delayedUnassigned`	O número de fragmentos cuja alocação de nó foi atrasada pelas configurações de tempo limite. Estatística relevante: máximo, soma
`Shards.activePrimary`	O número de fragmentos primários ativos. Estatística relevante: máximo, soma
`Shards.initializing`	O número de fragmentos que estão em inicialização. Estatísticas relevantes: soma
`Shards.relocating`	O número de fragmentos que estão em relocação. Estatísticas relevantes: soma

Métrica	Description
`Nodes`	O número de nós no cluster OpenSearch de serviços, incluindo nós mestres dedicados e nós quentes. Para obter mais informaçõ es, consulte the section called “Alterações de configuração”. Estatística relevante: máximo
`SearchableDocument s`	O número total de documentos pesquisáveis em todos os nós de dados no cluster. Estatísticas relevantes: mínimo, máximo, média
`DeletedDocuments`	O número total de documentos marcados para exclusão em todos os nós de dados no cluster. Esses documentos não aparecem mais nos resultados da pesquisa, mas OpenSearch apenas removem documentos excluídos do disco durante a mesclagem de segmentos. Essa métrica aumenta após solicitações e diminuições de exclusão após fusões de segmento. Estatísticas relevantes: mínimo, máximo, média
`CPUUtilization`	A porcentagem de utilização da CPU para nós de dados no cluster. Maximum (Máximo) mostra o nó com a maior utilização da CPU. Average (Médio) representa todos os nós no cluster. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: máximo, média

Métrica	Description
`FreeStorageSpace`	O espaço livre para nós de dados no cluster. `Sum` mostra o espaço livre total para o cluster, mas é necessário deixar o período em um minuto para obter um valor exato. `Minimum` e `Maximum` mostram os nós com o menor e o maior espaço livre, respectivamente. Essa métrica também está disponível para nós individuais. OpenSearch O serviço lança um `ClusterBlockException` quando essa métrica atinge`0`. Para recuperar, você deve excluir índices, adicionar instâncias maiores ou adicionar armazenamento EBS às instâncias existentes. Para saber mais, consulte the section called "Falta de espaço de armazenamento disponível". O console OpenSearch de serviço exibe esse valor em GiB. O CloudWatch console da Amazon o exibe em MiB. **Note** `FreeStorageSpace` sempre serão menores do que os valores `_cat/allocation` APIs fornecidos pelo OpenSearch `_cluster/stats` e. OpenSearch O serviço reserva uma porcentagem do espaço de armazenamento em cada instância para operações internas. Para saber mais, consulte Cálculo de requisitos de armazenamento. Estatísticas relevantes: mínima, máxima, média, soma
`ClusterUsedSpace`	O total de espaço usado para o cluster. Você deve deixar o período em um minuto para receber um valor preciso. O console OpenSearch de serviço exibe esse valor em GiB. O CloudWatch console da Amazon o exibe em MiB. Estatísticas relevantes: mínimo, máximo

Métrica	Description
`ClusterIndexWritesBlocked`	Indica se o cluster está aceitando ou bloqueando solicitações de gravação recebidas. Um valor de 0 significa que o cluster está aceitando solicitações. Um valor de 1 significa que ele está bloqueando solicitações. Alguns fatores comuns são: `FreeStorageSpace` é muito baixo ou `JVMMemoryPressure` é muito alto. Para aliviar esse problema, considere adicionar mais espaço em disco ou escalonar o cluster. Estatística relevante: máximo
`JVMMemoryPressure`	A porcentagem máxima do heap Java usada para todos os nós de dados no cluster. OpenSearch O serviço usa metade da RAM de uma instância para o heap Java, até um tamanho de heap de 32 GiB. Você pode dimensionar instâncias verticalmente até 64 GiB de RAM, sendo que nesse ponto você poderá dimension ar horizontalmente adicionando instâncias. Consulte the section called “ CloudWatch Alarmes recomendados”. Estatística relevante: máximo **Note** A lógica dessa métrica foi alterada no software de serviço R20220323. Para saber mais, consulte as notas de lançamento.
`OldGenJVMMemoryPressure`	A porcentagem máxima do heap do Java usada para a “geração antiga” em todos os nós de dados no cluster. Essa métrica também está disponível a nível de nós. Estatística relevante: máximo

Métrica	Description
`AutomatedSnapshotFailure`	O número de snapshots automatizados com falha para o cluster. Um valor de 1 indica que nenhum snapshot automatizado foi feito para o domínio nas últimas 36 horas. Estatísticas relevantes: mínimo, máximo
`CPUCreditBalance`	Os créditos de CPU ainda disponíveis para nós de dados no cluster. Um crédito de CPU oferece a performance de um núcleo de CPU completo por um minuto. Para saber mais, consulte Créditos de CPU no Guia do desenvolvedor do Amazon EC2. Essa métrica está disponível somente para os tipos de instância T2 Estatísticas relevantes: mínimo
`OpenSearchDashboardsHealthyNodes`	Uma verificação de saúde para OpenSearch painéis. Se mínimo, máximo e média forem todos iguais a 1, o Dashboards está se comportando normalmente. Se você tiver 10 nós com máximo de 1, mínimo de 0 e média de 0,7, isso significa que 7 nós (70%) são íntegros e 3 nós (30%) não são íntegros. Estatísticas relevantes: mínimo, máximo, média
`OpensearchDashboardsReportingFailedRequestSysErrCount`	O número de solicitações para gerar relatórios de OpenSearch painéis que falharam devido a problemas no servidor ou limitações de recursos. Estatísticas relevantes: soma
`OpensearchDashboardsReportingFailedRequestUserErrCount`	O número de solicitações para gerar relatórios de OpenSearch painéis que falharam devido a problemas do cliente. Estatísticas relevantes: soma
`OpensearchDashboardsReportingRequestCount`	O número total de solicitações para gerar relatórios de OpenSearch painéis. Estatísticas relevantes: soma

Métrica	Description
`OpensearchDashboardsReportingSuccessCount`	O número de solicitações bem-sucedidas para gerar relatórios de OpenSearch painéis. Estatísticas relevantes: soma
`KMSKeyError`	Um valor de 1 indica que a AWS KMS chave usada para criptografar dados em repouso foi desativada. Para restaurar o domínio de operações normais, reabilite a chave. O console exibe essa métrica somente para domínios que criptografam dados em repouso. Estatísticas relevantes: mínimo, máximo
`KMSKeyInaccessible`	Um valor de 1 indica que a AWS KMS chave usada para criptografar dados em repouso foi excluída ou revogada em suas concessões ao OpenSearch Serviço. Você não pode recuperar os domínios que estejam nesse estado. Mas, se tiver um snapshot manual, você poderá usá-lo para migrar os dados do domínio para um novo domínio. O console exibe essa métrica somente para domínios que criptografam dados em repouso. Estatísticas relevantes: mínimo, máximo
`InvalidHostHeaderRequests`	O número de solicitações HTTP feitas ao OpenSearch cluster que incluíram um cabeçalho de host inválido (ou ausente). As solicitações válidas incluem o nome do host do domínio como valor do cabeçalho do host. OpenSearch O serviço rejeita solicitações inválidas de domínios de acesso público que não tenham uma política de acesso restritiva. Recomendamos aplicar uma política de acesso restritiva a todos os domínios. Se você visualizar grandes valores para esta métrica, confirme que os clientes do OpenSearch incluem o nome de host do domínio (e não, por exemplo, seu endereço IP) em suas solicitações. Estatísticas relevantes: soma

Métrica	Description
`OpenSearchRequests (previously ElasticsearchRequests)`	O número de solicitações feitas ao OpenSearch cluster. Estatísticas relevantes: soma
`2xx, 3xx, 4xx, 5xx`	O número de solicitações a um domínio que resultaram no determinado código de resposta HTTP (2xx, 3xx, 4xx, 5xx). Estatísticas relevantes: soma
`ThroughputThrottle`	Indica se os discos estão sob controle de utilização ou não. O controle de utilização ocorre quando o throughput combinado de `ReadThroughputMicroBursting` e `WriteThroughputMicroBursting` é maior que o throughput máximo de `MaxProvisionedThroughput`. `MaxProvisionedThroughput` é o valor mais baixo do throughput da instância ou do throughput do volume provisionado. Um valor de 1 indica que os discos estão sob controle de utilização. Um valor de 0 indica comportamento normal. Para obter informações sobre o throughput de instâncias, consulte Instâncias otimizadas para Amazon EBS. Para obter informações sobre o throughput de volume, consulte os tipos de volume do Amazon EBS. Estatísticas relevantes: mínimo, máximo

Métrica	Description
`IopsThrottle`	Indica se o número de input/output operações por segundo (IOPS) no domínio foi reduzido ou não. O controle de utilização ocorre quando o IOPS do nó de dados viola o limite máximo permitido do volume do EBS ou da instância EC2 do nó de dados. Para obter informações sobre o IOPSS de instâncias, consulte Instâncias otimizadas para Amazon EBS. Para obter informações sobre o IOPS de volume, consulte os tipos de volume do Amazon EBS. Estatísticas relevantes: mínimo, máximo
`HighSwapUsage`	Um valor 1 indica que a troca devido a falhas da página provavelm ente causou picos no uso do disco subjacente durante um período específico. Estatística relevante: máximo

Métricas de nó principal dedicado

O Amazon OpenSearch Service fornece as seguintes métricas para nós mestres dedicados.

Métrica	Description
`MasterCPUUtilizati on`	A porcentagem máxima de recursos da CPU usados pelos nós principais dedicados. Recomendamos aumentar o tamanho do tipo de instância quando essa métrica atingir 60%. Estatística relevante: máximo
`MasterFreeStorageS pace`	Essa métrica não é relevante e pode ser ignorada. O serviço não usa nós principais como nós de dados.
`MasterJVMMemoryPre ssure`	A porcentagem máxima do heap Java usada para todos os nós principais dedicados no cluster. Recomendamos a mudança para um tipo de instância maior quando essa métrica atingir 85%.

Métrica	Description
	Estatística relevante: máximo **Note** A lógica dessa métrica foi alterada no software de serviço R20220323. Para saber mais, consulte as notas de lançamento.
`MasterOldGenJVMMemoryPressure`	A porcentagem máxima do heap do Java usada para a “geração antiga” por nó principal. Estatística relevante: máximo
`MasterCPUCreditBalance`	Os créditos de CPU ainda disponíveis para nós principais dedicados no cluster. Um crédito de CPU oferece a performance de um núcleo de CPU completo por um minuto. Para saber mais, consulte Créditos de CPU no Guia do desenvolvedor do Amazon EC2. Essa métrica está disponível somente para os tipos de instância T2 Estatísticas relevantes: mínimo
`MasterReachableFromNode`	Uma verificação de integridade exceções `MasterNotDiscovered` . Um valor de 1 indica comportamento normal. Um valor de 0 indica que `/_cluster/health/` está falhando. Falhas significam que o nó principal está inacessível a partir do nó de origem. Geralmente são o resultado de um problema de conectividade de rede ou de AWS dependência. Estatística relevante: máximo
`MasterSysMemoryUtilization`	O percentual de memória do nó principal que está em uso. Estatística relevante: máximo

Métricas do nó Coordenador dedicado

O Amazon OpenSearch Service fornece as seguintes métricas para nós coordenadores dedicados.

Métrica	Description
`CoordinatorCPUUtilization`	A porcentagem máxima de recursos da CPU usados pelos nós coordenadores dedicados. Recomendamos aumentar o tamanho do tipo de instância quando essa métrica atingir 80%. Estatística relevante: máximo
`CoordinatorJVMMemoryPressure`	A porcentagem máxima do heap Java usada para todos os nós coordenadores dedicados no cluster. Recomendamos a mudança para um tipo de instância maior quando essa métrica atingir 85%. Estatística relevante: máximo
`CoordinatorOldGenJVMMemoryPressure`	A porcentagem máxima do heap do Java usada para a “geração antiga” por nó principal. Estatística relevante: máximo
`CoordinatorSysMemoryUtilization`	A porcentagem de memória do nó coordenador que está em uso. Estatística relevante: máximo
`CoordinatorFreeStorageSpace`	Essa métrica indica que o serviço não usa nós coordenadores como nós de dados.

Métricas de volume do EBS

O Amazon OpenSearch Service fornece as seguintes métricas para volumes do EBS.

Métrica	Description
`ReadLatency`	A latência, em segundos, para operações de leitura em volumes do EBS. Esta métrica também está disponível para nós individuais.

Métrica	Description
	Estatísticas relevantes: mínimo, máximo, média
`WriteLatency`	A latência, em segundos, para operações de gravação em volumes do EBS. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média
`ReadThroughput`	O throughput, em bytes por segundo, para operações de leitura em volumes do EBS. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média
`ReadThroughputMicroBursting`	O throughput, em bytes por segundo, para operações de leitura em volumes do EBS quando a microintermitência é levada em consideração. Esta métrica também está disponível para nós individuais. A microintermitência ocorre quando um volume do EBS aumenta o IOPS ou a taxa de throughput por períodos de tempo significativamente mais curtos (menos de um minuto). Estatísticas relevantes: mínimo, máximo, média
`WriteThroughput`	O throughput, em bytes por segundo, para operações de gravação em volumes do EBS. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média
`WriteThroughputMicroBursting`	O throughput, em bytes por segundo, para operações de gravação em volumes do EBS quando a microintermitência é levada em consideração. Esta métrica também está disponível para nós individuais. A microintermitência ocorre quando um volume do EBS aumenta o IOPS ou a taxa de throughput por períodos de tempo significativamente mais curtos (menos de um minuto). Estatísticas relevantes: mínimo, máximo, média

Métrica	Description
`DiskQueueDepth`	O número de solicitações pendentes de entrada e saída (E/S) de um volume do EBS. Estatísticas relevantes: mínimo, máximo, média
`ReadIOPS`	O número de operações de entrada e saída (E/S) por segundo para operações de leitura em volumes do EBS. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média
`ReadIOPSMicroBursting`	O número de operações de entrada e saída (E/S) por segundo para operações de leitura em volumes do EBS quando a microintermitência é levando em consideração. Esta métrica também está disponível para nós individuais. A microintermitência ocorre quando um volume do EBS aumenta o IOPS ou a taxa de throughput por períodos de tempo significativamente mais curtos (menos de um minuto). Estatísticas relevantes: mínimo, máximo, média
`WriteIOPS`	O número de operações de entrada e saída (E/S) por segundo para operações de gravação em volumes do EBS. Esta métrica também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média
`WriteIOPS MicroBursting`	O número de operações de entrada e saída (E/S) por segundo para operações de gravação em volumes do EBS quando a microintermitência é levando em consideração. Esta métrica também está disponível para nós individuais. A microintermitência ocorre quando um volume do EBS aumenta o IOPS ou a taxa de throughput por períodos de tempo significativamente mais curtos (menos de um minuto). Estatísticas relevantes: mínimo, máximo, média

Métrica	Description
`BurstBalance`	A porcentagem de créditos de entrada e saída (E/S) restantes no bucket de intermitência para um volume do EBS. Um valor de 100 significa que o volume acumulou o número máximo de créditos. Se essa porcentagem cair abaixo de 70%, consulte the section called “O saldo de intermitência do EBS está baixo”. O saldo intermitente permanece em 0 para domínios com tipos de volume gp3 e domínios com volume gp2 cujo tamanho de volume seja superior a 1000 GiB. Estatísticas relevantes: mínimo, máximo, média
`VolumeStalledIOcheck`	O status dos volumes do EBS para determinar quando eles estão danificados. A métrica é um valor binário que retorna um status 0 (aprovado) ou 1 (reprovado) dependendo do volume do EBS poder realizar as operações de entrada e saída. `VolumeStalledIOcheck` também está disponível para nós individuais. Estatísticas relevantes: mínimo, máximo, média

Métricas de instância

O Amazon OpenSearch Service fornece as seguintes métricas para cada instância em um domínio. OpenSearch O serviço também agrega essas métricas de instância para fornecer informações sobre a integridade geral do cluster. Você pode verificar esse comportamento usando a estatística Contagem de amostras no console. Cada métrica na tabela a seguir tem estatísticas relevantes para o nó e o cluster.

> **Important**
>
> Versões diferentes do Elasticsearch usam grupos de threads diferentes para processar chamadas para a API `_index`. As versões 1.5 e 2.3 do Elasticsearch usam o grupo de threads de índice. Elasticsearch 5. x, 6.0 e 6.2 usam o pool de threads em massa. OpenSearch e o Elasticsearch 6.3 e versões posteriores usam o pool de threads de gravação. Atualmente, o console OpenSearch de serviço não inclui um gráfico para o pool de threads em massa.

Use `GET _cluster/settings?include_defaults=true` para verificar o grupo de threads e os tamanhos de fila para seu cluster.

Métrica	Description
`FetchLatency`	A diferença de tempo total, em milissegundos, obtida por todas as operações de busca de fragmento em um nó entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo
`FetchRate`	O número total de operações de busca de fragmentos por minuto para todos os fragmentos em um nó de dados. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma
`ScrollTotal`	O número total de operações de rolagem de fragmentos por minuto para todos os fragmentos em um nó de dados. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma
`ScrollCurrent`	O número de operações de rolagem de fragmentos que estão em execução no momento. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma
`OpenContexts`	O número de contextos de pesquisa abertos. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma

Métrica	Description
`ThreadCount`	O número total de threads atualmente sendo utilizados pelo OpenSearch processo. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma
`ShardReactivateCount`	O número total de vezes que todos os fragmentos foram ativados a partir de um estado ocioso. Estatísticas do nó relevante: soma, máximo Estatísticas do cluster relevante: soma, máximo
`ConcurrentSearchRate`	O número total de solicitações de pesquisa usando a pesquisa simultânea de segmentos por minuto para todos os fragmentos em um nó de dados. Uma única chamada para a API `_search` pode retornar resultados de muitos fragmentos diferentes. Se cinco desses fragmentos estiverem em um nó, o nó reportará 5 para essa métrica, mesmo se o cliente só fizer uma solicitação. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma
`ConcurrentSearchLatency`	A diferença no tempo total, em milissegundos, obtida por todas as pesquisas usando a pesquisa simultânea de segmentos em um nó entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo

Métrica	Description
`IndexingLatency`	A diferença no tempo total, em milissegundos, obtida por todas as operações de indexação em um nó entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo
`IndexingRate`	O número de operações de indexação por minuto. Uma única chamada para a API `_bulk` que adiciona dois documentos e atualiza duas contagens tem quatro operações, que podem ser espalhadas entre um ou mais nós. Se esse índice tiver uma ou mais réplicas e estiver em um OpenSearch domínio sem instâncias otimizadas, outros nós no cluster também registrarão um total de quatro operações de indexação. Para OpenSearch domínios com instâncias otimizadas, outros nós com réplicas não registram nenhuma operação. Exclusões de documento não são consideradas para essa métrica. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma
`SearchLatency`	A diferença no tempo total, em milissegundos, obtida por todas as pesquisas em um nó entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo

Métrica	Description
`SearchRate`	O número total de solicitações de pesquisa por minuto para todos os fragmentos em um nó de dados. Uma única chamada para a API `_search` pode retornar resultados de muitos fragmentos diferentes. Se cinco desses fragmentos estiverem em um nó, o nó reportará 5 para essa métrica, mesmo se o cliente só fizer uma solicitação. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma
`SegmentCount`	O número de segmentos em um nó de dados. Quanto mais segmentos você tiver, maior será a duração de cada pesquisa. OpenSearch ocasionalmente mescla segmentos menores em um maior. Estatísticas de nós relevantes: máximo, média Estatísticas do cluster relevante: soma, máximo, média
`SysMemoryUtilization`	O percentual de memória da instância que está em uso. Valores altos para essa métrica são normais e geralmente não represent am um problema com seu cluster. Para obter um melhor indicador de possíveis problemas de performance e estabilidade, consulte a métrica `JVMMemoryPressure` . Estatísticas do nó relevante: mínimo, máximo, média Estatísticas relevantes de cluster: mínimo, máximo, média, soma
`JVMGCYoungCollectionCount`	O número de vezes que a coleta de lixo “nova geração” foi executada. Um grande número de execuções crescente é uma parte normal das operações do cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média

Métrica	Description
JVMGCYoungCollectionTime	A quantidade de tempo, em milissegundos, que o cluster gastou executando a coleta de lixo “nova geração”. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
JVMGCOldCollectionCount	O número de vezes que a coleta de lixo “geração antiga” foi executada. Em um cluster com recursos suficientes, esse número deve permanecer pequeno e com crescimento com pouca frequência. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
JVMGCOldCollectionTime	A quantidade de tempo, em milissegundos, que o cluster gastou executando a coleta de lixo “geração antiga”. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
OpenSearchDashboardsConcurrentConnections	O número de conexões simultâneas ativas com os OpenSearch painéis. Se esse número continuar a crescer, considere escalar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
OpenSearchDashboardsHealthyNode	Uma verificação de saúde para o nó individual dos OpenSearch painéis. Um valor de 1 indica comportamento normal. Um valor de 0 indica que Dashboards está inacessível. Estatísticas do nó relevante: mínimo Estatísticas relevantes de cluster: mínimo, máximo, média, soma

Métrica	Description
OpenSearchDashboardsHeapTotal	A quantidade de memória de pilha alocada aos OpenSearch painéis em MiB. Diferentes tipos de instância do EC2 podem afetar a alocação exata de memória. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
OpenSearchDashboardsHeapUsed	A quantidade absoluta de memória de pilha usada pelos OpenSearch painéis em MiB. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
OpenSearchDashboardsHeapUtilization	A porcentagem máxima de memória de pilha disponível usada pelos OpenSearch painéis. Se esse valor aumentar acima de 80%, considere escalar seu cluster. Estatísticas do nó relevante: máximo Estatísticas relevantes de cluster: mínimo, máximo, média, soma
OpenSearchDashboardsOS1MinuteLoad	A média de carga de CPU de um minuto para OpenSearch painéis. A carga da CPU deve, idealmente, permanecer abaixo de 1,00. Embora picos temporários não sejam um problema, recomenda mos aumentar o tamanho do tipo de instância se essa métrica estiver consistentemente acima de 1,00. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo

Métrica	Description
`OpenSearchDashboardsRequestTotal`	A contagem total de solicitações HTTP feitas aos OpenSearch painéis. Se o sistema estiver lento ou você observar números elevados de solicitações de painéis, considere aumentar o tamanho do tipo de instância. Estatísticas de nós relevantes: soma Estatísticas do cluster relevante: soma
`OpenSearchDashboardsResponseTimesMaxInMillis`	O tempo máximo, em milissegundos, necessário para que os OpenSearch painéis respondam a uma solicitação. Se as solicitações demorarem consistentemente muito tempo para retornar resultados, considere aumentar o tamanho do tipo de instância. Estatísticas do nó relevante: máximo Estatísticas de cluster relevantes máximo, média
`SearchTaskCancelled`	O número de cancelamentos do nó coordenador. Estatísticas de nós relevantes: soma Estatísticas do cluster relevante: soma
`SearchShardTaskCancelled`	O número de cancelamentos de nós de dados. Estatísticas de nós relevantes: soma Estatísticas do cluster relevante: soma,
`ThreadpoolForce_mergeQueue`	O número de tarefas na fila no grupo de thread de união de força. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média

Métrica	Description
`ThreadpoolForce_mergeRejected`	O número de tarefas rejeitadas no grupo de thread de união de força. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`ThreadpoolForce_mergeThreads`	O tamanho do grupo de threads de união de força. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolIndexQueue`	O número de tarefas na fila no grupo de thread de índice. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. O tamanho máximo da fila de índice é de 200. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`ThreadpoolIndexRejected`	O número de tarefas rejeitadas no grupo de thread de índice. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`ThreadpoolIndexThreads`	O tamanho do grupo de threads de índice. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma

Métrica	Description
`ThreadpoolSearchQueue`	O número de tarefas na fila no grupo de thread de pesquisa. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. O tamanho da fila de pesquisa máximo é 1.000. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`ThreadpoolSearchRejected`	O número de tarefas rejeitadas no grupo de thread de pesquisa. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`ThreadpoolSearchThreads`	O tamanho do grupo de threads de pesquisa. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`Threadpoolsql-workerQueue`	O número de tarefas na fila no grupo de threads de pesquisa SQL. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`Threadpoolsql-workerRejected`	O número de tarefas rejeitadas no grupo de threads de pesquisa SQL. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma

Métrica	Description
`Threadpoolsql-workerThreads`	O tamanho do grupo de threads de pesquisa SQL. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolBulkQueue`	O número de tarefas na fila no grupo de thread em massa. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`ThreadpoolBulkRejected`	O número de tarefas rejeitadas no grupo de thread em massa. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`ThreadpoolBulkThreads`	O tamanho do grupo de threads em massa. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolIndexSearcherQueue`	O número de tarefas na fila no grupo de threads de buscador de índice. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`ThreadpoolIndexSearcherRejected`	O número de tarefas rejeitadas no grupo de thread de buscador de índice. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma

Métrica	Description
`ThreadpoolIndexSearcherThreads`	O tamanho do grupo de threads de buscador de pesquisa. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolWriteThreads`	O tamanho do grupo de threads de gravação. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolWriteQueue`	O número de tarefas na fila no grupo de threads de gravação. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma
`ThreadpoolWriteRejected`	O número de tarefas rejeitadas no grupo de threads de gravação. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma **Note** Como o tamanho padrão da fila de gravação foi aumentado de 200 para 10000 na versão 7.1, essa métrica não é mais o único indicador de rejeições do Serviço. OpenSearch Use as métricas `CoordinatingWriteRejected`, `PrimaryWriteRejected` e `ReplicaWriteRejected` para monitorar rejeições nas versões 7.1 e posteriores.

Métrica	Description
`CoordinatingWriteRejected`	O número total de rejeições ocorreu no nó de coordenação devido à pressão de indexação desde a última inicialização do processo de OpenSearch serviço. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma Esta métrica está disponível na versão 7.1 e posteriores.
`PrimaryWriteRejected`	O número total de rejeições ocorreu nos fragmentos primários devido à pressão de indexação desde a última inicialização do processo de OpenSearch serviço. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma Esta métrica está disponível na versão 7.1 e posteriores.
`ReplicaWriteRejected`	O número total de rejeições ocorreu nos fragmentos de réplica devido à pressão de indexação desde a última OpenSearch inicialização do processo de serviço. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma Esta métrica está disponível na versão 7.1 e posteriores.
`WorkloadManagementEnabled`	Indica se o atributo de gerenciamento da workload está habilitado. Um valor de 1 significa que está habilitado e um valor de 0 significa que está `monitor_only` ou desabilitado. Estatísticas do nó relevante: máximo, mínimo Estatísticas do cluster relevante: média, soma Esta métrica está disponível na versão 7.1 e posteriores.

Métrica	Description
`SoftQueryGroupCount`	Número de grupos de consulta no modo flexível no domínio. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma Esta métrica está disponível na versão 7.1 e posteriores.
`EnforcedQueryGroupCount`	Número de grupos de consulta no modo imposto no domínio. Estatísticas de nós relevantes: médio, máximo Estatísticas do cluster relevante: média, máxima, soma Esta métrica está disponível na versão 7.1 e posteriores.

Métricas quentes

O Amazon OpenSearch Service fornece as seguintes métricas para arquitetura de armazenamento multicamada e UltraWarm

> **Note**
>
> As métricas relacionadas à indexação quente são aplicáveis somente à arquitetura de armazenamento multicamada

Métrica	Description
`WarmIndexingLatency`	A diferença no tempo total, em milissegundos, obtida por todas as operações de indexação em um nó quente entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo

Métrica	Description
`WarmIndexingRate`	O número de operações de indexação em aquecimento por minuto. Uma única chamada para a API `_bulk` que adiciona dois documentos e atualiza duas contagens tem quatro operações, que podem ser espalhadas entre um ou mais nós. Se esse índice tiver uma ou mais réplicas e estiver em um OpenSearch domínio sem instâncias otimizadas, outros nós no cluster também registrarão um total de quatro operações de indexação. Para OpenSearch domínios com instâncias otimizadas, outros nós com réplicas não registram nenhuma operação. Exclusões de documento não são consideradas para essa métrica. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma
`WarmThreadpoolIndexingQueue`	O número de tarefas na fila no grupo de thread de índice. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. O tamanho máximo da fila de índice é de 200. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, máxima, soma
`WarmThreadpoolIndexingRejected`	O número de tarefas rejeitadas no grupo de thread de índice. Se esse número continuar a crescer, considere escalonar seu cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`WarmThreadpoolIndexingThreads`	O tamanho do grupo de threads de índice. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma

Métrica	Description
`WarmCPUUtilization`	A porcentagem de uso da CPU para nós quentes no cluster. Maximum (Máximo) mostra o nó com a maior utilização da CPU. A média represent a todos os nós quentes no cluster. Essa métrica também está disponível para nós Warm individuais. Estatísticas relevantes: máximo, média
`WarmFreeStorageSpace`	A quantidade de espaço de armazenamento de alta atividade livre em MiB. Como o Warm usa o Amazon S3 em vez de discos conectados, essa `Sum` é a única estatística relevante. Você deve deixar o período em um minuto para receber um valor preciso. Estatísticas relevantes: soma
`WarmSearchableDocuments`	O número total de documentos pesquisáveis em todos os índices warm no cluster. Você deve deixar o período em um minuto para receber um valor preciso. Estatísticas relevantes: soma
`WarmSearchLatency`	A diferença no tempo total, em milissegundos, obtida por todas as pesquisas em um Warm entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máximo
`WarmSearchRate`	O número total de solicitações de pesquisa por minuto para todos os fragmentos em um nó Warm. Uma única chamada para a API `_search` pode retornar resultados de muitos fragmentos diferentes. Se cinco desses fragmentos estiverem em um nó, o nó reportará 5 para essa métrica, mesmo se o cliente só fizer uma solicitação. Estatísticas do nó relevante: média Estatísticas do cluster relevante: média, máxima, soma

Métrica	Description
`WarmStorageSpaceUtilization`	A quantidade total de espaço de armazenamento de alta atividade, em MiB, que o cluster está usando. Estatística relevante: máximo
`HotStorageSpaceUtilization`	A quantidade total de espaço de armazenamento de atividade muito alta que o cluster está usando. Estatística relevante: máximo
`WarmSysMemoryUtilization`	A porcentagem de memória do nó de alta atividade que está em uso. Estatística relevante: máximo
`HotToWarmMigrationQueueSize`	O número de índices aguardando no momento para a migração do armazenamento quente para o armazenamento warm. Estatística relevante: máximo
`WarmToHotMigrationQueueSize`	O número de índices aguardando no momento para a migração do armazenamento warm para o armazenamento quente. Estatística relevante: máximo
`HotToWarmMigrationFailureCount`	O número total de migrações de atividade muito alta para alta atividade que falharam. Essa métrica está disponível somente para UltraWarm nós. Estatísticas relevantes: soma
`HotToWarmMigrationForceMergeLatency`	A latência média da etapa de forçar mesclagem do processo de migração. Se esse estágio sempre demorar muito, considere aumentar. Essa métrica está disponível somente para UltraWarm nós. `index.ultrawarm.migration.force_merge.max_num_segments` . Estatística relevante: média

Métrica	Description
`HotToWarmMigrationSnapshotLatency`	A latência média da etapa de snapshot do processo de migração. Se esse estágio demorar muito de forma consistente, certifique-se de que os fragmentos estejam adequadamente dimensionados e distribuídos por todo o cluster. Essa métrica está disponível somente para UltraWarm nós. Estatística relevante: média
`HotToWarmMigrationProcessingLatency`	A latência média de migrações de atividade muito alta para alta atividade bem-sucedidas, nãoincluindo tempo gasto na fila. Esse valor é a soma do tempo necessário para concluir os estágios de forçar mesclagem, snapshot e realocação de fragmentos do processo de migração. Essa métrica está disponível somente para UltraWarm nós. Estatística relevante: média
`HotToWarmMigrationSuccessCount`	O número total de migrações de atividade muito alta para alta atividade bem-sucedidas. Estatísticas relevantes: soma
`HotToWarmMigrationSuccessLatency`	A latência média de migrações de atividade muito alta para alta atividade bem-sucedidas, incluindo tempo gasto na fila. Estatística relevante: média
`WarmThreadpoolSearchThreads`	O tamanho do pool de tópicos de pesquisa do Warm. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: média, soma

Métrica	Description
`WarmThreadpoolSearchRejected`	O número de tarefas rejeitadas no pool de tópicos de pesquisa do Warm. Se esse número aumentar continuamente, considere adicionar mais nós quentes. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`WarmThreadpoolSearchQueue`	O número de tarefas em fila no pool de tópicos de pesquisa do Warm. Se o tamanho da fila for consistentemente alto, considere adicionar mais nós quentes. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`WarmJVMMemoryPressure`	A porcentagem máxima do heap Java usada para os nós Warm. Estatística relevante: máximo **Note** A lógica dessa métrica foi alterada no software de serviço R20220323. Para saber mais, consulte as notas de lançamento.
`WarmOldGenJVMMemoryPressure`	A porcentagem máxima do heap Java usada para a “geração antiga” por nó Warm. Estatística relevante: máximo
`WarmJVMGCYoungCollectionCount`	O número de vezes que a coleta de lixo da “geração jovem” foi executada em nós Warm. Um grande número de execuções crescente é uma parte normal das operações do cluster. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média

Métrica	Description
`WarmJVMGCYoungCollectionTime`	A quantidade de tempo, em milissegundos, que o cluster gastou realizando a coleta de lixo da “geração jovem” nos nós do Warm. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`WarmJVMGCOldCollectionCount`	O número de vezes que a coleta de lixo da “velha geração” foi executada em nós Warm. Em um cluster com recursos suficientes, esse número deve permanecer pequeno e com crescimento com pouca frequência. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média
`WarmConcurrentSearchRate`	O número total de solicitações de pesquisa usando a pesquisa simultânea por segmento por minuto para todos os fragmentos em um nó Warm. Uma única chamada para a API `_search` pode retornar resultados de muitos fragmentos diferentes. Se cinco desses fragmentos estiverem em um nó, o nó reportará 5 para essa métrica, mesmo se o cliente só fizer uma solicitação. Estatísticas do nó relevante: média Estatísticas do cluster relevante: soma, máximo, média
`WarmConcurrentSearchLatency`	A diferença no tempo total, em milissegundos, obtida por todas as pesquisas usando a pesquisa simultânea de segmentos em um nó quente entre o minuto N e o minuto (N-1). Estatísticas do nó relevante: média Estatísticas de cluster relevantes máximo, média
`WarmThreadpoolIndexSearcherQueue`	O número de tarefas em fila no pool de threads do Warm index searcher. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, máximo, média

Métrica	Description
`WarmThreadpoolIndexSearcherRejected`	O número de tarefas rejeitadas no pool de tópicos do Warm index searcher. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma
`WarmThreadpoolIndexSearcherThreads`	O tamanho do pool de tópicos do Warm Index Searcher. Estatísticas do nó relevante: máximo Estatísticas do cluster relevante: soma, média

Métricas de armazenamento de baixa atividade

O Amazon OpenSearch Service fornece as seguintes métricas para armazenamento a frio.

Métrica	Description
`ColdStorageSpaceUtilization`	A quantidade total de espaço de armazenamento de baixa atividade, em MiB, que o cluster está usando. Estatísticas relevantes: máx.
`ColdToWarmMigrationFailureCount`	O número total de migrações de baixa atividade para alta atividade que falharam. Estatísticas relevantes: soma
`ColdToWarmMigrationLatency`	A quantidade de tempo necessária para que as migrações de baixa atividade para alta atividade sejam concluídas. Estatística relevante: média
`ColdToWarmMigrationQueueSize`	O número de índices aguardando no momento para a migração do armazenamento frio para o armazenamento warm. Estatística relevante: máximo

Métrica	Description
`ColdToWarmMigrationSuccessCount`	O número total de migrações de baixa atividade para alta atividade bem-sucedidas. Estatísticas relevantes: soma
`WarmToColdMigrationFailureCount`	O número total de migrações de alta atividade para baixa atividade que falharam. Estatísticas relevantes: soma
`WarmToColdMigrationLatency`	A quantidade de tempo necessária para que as migrações de alta atividade para baixa atividade sejam concluídas. Estatística relevante: média
`WarmToColdMigrationQueueSize`	O número de índices aguardando atualmente para migrar do armazenamento warm para o armazenamento frio. Estatística relevante: máximo
`WarmToColdMigrationSuccessCount`	O número total de migrações de alta atividade para baixa atividade bem-sucedidas. Estatísticas relevantes: soma

OpenSearch Métricas de instâncias otimizadas (OR1)

O Amazon OpenSearch Service fornece as seguintes métricas para OR1instâncias.

Métrica	Description
`RemoteStorageUsedSpace`	A quantidade total de espaço do Amazon S3, em MiB, que o cluster está usando. Estatísticas relevantes: soma
`RemoteStorageWriteRejected`	O número total de solicitações rejeitadas nos fragmentos primários devido à pressão de armazenamento e replicação

Métrica	Description
	remotos. Isso é calculado a partir da última inicialização do processo de OpenSearch serviço. Estatísticas relevantes: soma
`ReplicationLagMaxTime`	A quantidade de tempo, em milissegundos, que os fragmentos de réplica ficam atrasados em relação aos fragmentos primários . Estatística relevante: máximo

Métricas de alerta

O Amazon OpenSearch Service fornece as seguintes métricas para alertas.

Métrica	Description
`AlertingDegraded`	Um valor de 1 significa que o índice de alerta é vermelho ou um ou mais nós não estão na programação. Um valor de 0 indica comportamento normal. Estatística relevante: máximo
`AlertingIndexExists`	Um valor de 1 significa que o índice `.opensearch-alerting-config` existe. Um valor de 0 significa que não. Até que você use o recurso de alerta pela primeira vez, esse valor permanecerá como 0. Estatística relevante: máximo
`AlertingIndexStatus.green`	A integridade do índice. Um valor de 1 significa verde. Um valor de 0 significa que o índice não existe ou não está verde. Estatística relevante: máximo
`AlertingIndexStatus.red`	A integridade do índice. Um valor de 1 significa vermelho. Um valor de 0 significa que o índice não existe ou não está vermelho.

Métrica	Description
	Estatística relevante: máximo
`AlertingIndexStatus.yellow`	A integridade do índice. Um valor de 1 significa amarelo. Um valor de 0 significa que o índice não existe ou não está amarelo. Estatística relevante: máximo
`AlertingNodesNotOnSchedule`	Um valor de 1 significa que alguns trabalhos não estão sendo executados de acordo com a programação. Um valor de 0 significa que todos os trabalhos de alerta estão sendo executados de acordo com a programação (ou que não existem trabalhos de alerta). Verifique o console OpenSearch de serviços ou faça uma `_nodes/stats` solicitação para ver se algum nó mostra alto uso de recursos. Estatística relevante: máximo
`AlertingNodesOnSchedule`	Um valor de 1 significa que todos os trabalhos de alerta estão em execução de acordo com a programação (ou que não existem trabalhos de alerta). Um valor de 0 significa que alguns trabalhos não estão sendo executados de acordo com a programação. Estatística relevante: máximo
`AlertingScheduledJobEnabled`	Um valor de 1 significa que a configuração do cluster `opensearch.scheduled_jobs.enabled` é verdadeira. Um valor de 0 significa que é falsa e os trabalhos programados estão desabilitados. Estatística relevante: máximo

Métricas de detecção de anomalias

O Amazon OpenSearch Service fornece as seguintes métricas para detecção de anomalias.

Métrica	Description
`ADPluginUnhealthy`	Um valor de 1 significa que o plug-in de detecção de anomalias não está funcionando corretamente, seja por causa de um alto número de

Métrica	Description
	falhas, seja porque um dos índices que ele usa é vermelho. Um valor de 0 indica que o plug-in está funcionando conforme esperado. Estatística relevante: máximo
`ADExecute RequestCount`	O número de solicitações para detectar anomalias. Estatísticas relevantes: soma
`ADExecute FailureCount`	O número de solicitações com falha para detecção de anomalias. Estatísticas relevantes: soma
`ADHCExecu teFailureCount`	O número de solicitações de detecção de anomalias para detectores de alta cardinalidade que falharam. Estatísticas relevantes: soma
`ADHCExecu teRequestCount`	O número de solicitações de detecção de anomalias para detectores de alta cardinalidade. Estatísticas relevantes: soma
`ADAnomaly ResultsIn dexStatus IndexExists`	Um valor de 1 significa que o índice para o qual o alias `.opensearch-anomaly-results` aponta existe. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo
`ADAnomaly ResultsIn dexStatus.red`	Um valor de 1 significa que o índice para o qual o alias `.opensearch-anomaly-results` aponta é vermelho. Um valor de 0 significa que não é. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo

Métrica	Description
`ADAnomaly Detectors IndexStat usIndexExists`	Um valor de 1 significa que o índice `.opensearch-anomaly-detectors` existe. Um valor de 0 significa que não. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo
`ADAnomaly Detectors IndexStat us.red`	Um valor de 1 significa que o índice `.opensearch-anomaly-detectors` é vermelho. Um valor de 0 significa que não é. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo
`ADModelsC heckpoint IndexStat usIndexExists`	Um valor de 1 significa que o índice `.opensearch-anomaly-checkpoints` existe. Um valor de 0 significa que não. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo
`ADModelsC heckpoint IndexStat us.red`	Um valor de 1 significa que o índice `.opensearch-anomaly-checkpoints` é vermelho. Um valor de 0 significa que não é. Até que o recurso de detecção de anomalias seja usado pela primeira vez, esse valor permanecerá 0. Estatística relevante: máximo

Métricas de pesquisa assíncrona

O Amazon OpenSearch Service fornece as seguintes métricas para pesquisa assíncrona.

Estatísticas de nó coordenador de pesquisa assíncrona (por nó coordenador)

Métrica	Description
`AsynchronousSearchSubmissionRate`	O número de pesquisas assíncronas enviadas no último minuto.
`AsynchronousSearchInitializedRate`	O número de pesquisas assíncronas inicializadas no último minuto.
`AsynchronousSearchRunningCurrent`	O número de pesquisas assíncronas atualmente em execução.
`AsynchronousSearchCompletionRate`	O número de pesquisas assíncronas concluídas com êxito no último minuto.
`AsynchronousSearchFailureRate`	O número de pesquisas assíncronas que foram concluídas e falharam no último minuto.
`AsynchronousSearchPersistRate`	O número de pesquisas assíncronas que persistiram no último minuto.
`AsynchronousSearchPersistFailedRate`	O número de pesquisas assíncronas que falharam ao persistir no último minuto.
`AsynchronousSearchRejected`	O número total de pesquisas assíncronas rejeitadas desde o momento de ativação do nó.

Métrica	Description
`AsynchronousSearchCancelled`	O número total de pesquisas assíncronas canceladas desde o momento de ativação do nó.
`AsynchronousSearchMaxRunningTime`	A duração da pesquisa assíncrona de execução mais longa em um nó no último minuto.

Estatísticas de cluster de pesquisa assíncrona

Métrica	Description
`AsynchronousSearchStoreHealth`	A integridade do armazenamento no índice persistido (vermelho/não vermelho) no último minuto.
`AsynchronousSearchStoreSize`	O tamanho do índice do sistema em todos os fragmentos no último minuto.
`AsynchronousSearchStoredResponseCount`	O número de respostas armazenadas no índice do sistema no último minuto.

Métricas do Auto-Tune

O Amazon OpenSearch Service fornece as seguintes métricas para o Auto-Tune.

Métrica	Description
`AutoTuneChangesHistoryHeapSize`	O histórico de alterações em MiB para valores de ajuste do tamanho da pilha.

Métrica	Description
`AutoTuneChangesHistoryJVMYoungGenArgs`	O histórico de alterações dos argumentos da JVM. YongGen
`AutoTuneFailed`	Um booleano que indica se a alteração do Auto-Tune falhou.
`AutoTuneSucceeded`	Um booleano que indica se a alteração do Auto-Tune foi bem-sucedida.
`AutoTuneValue`	O histórico de alterações da fila (contagem) e o histórico de alterações dos ajustes do cache (em MiB) para alterações sem interrupções.

Métricas do multi-AZ com modo de espera

O Amazon OpenSearch Service fornece as seguintes métricas para Multi-AZ com Standby.

Métricas em nível de nó para nós de dados em zonas de disponibilidade ativas

Métrica	Description
`CPUUtilization`	A porcentagem de utilização da CPU para nós de dados no cluster. Maximum (Máximo) mostra o nó com a maior utilização da CPU. Average (Médio) representa todos os nós no cluster. Esta métrica também está disponível para nós individuais.
`FreeStorageSpace`	O espaço livre para nós de dados no cluster. `Sum` mostra o espaço livre total para o cluster, mas é necessário deixar o período em um minuto para obter um valor exato. `Minimum` e `Maximum` mostram os nós com o menor e o maior espaço livre, respectivamente. Essa métrica também está disponível para nós individuais. OpenSearch O serviço lança um `ClusterBlockException` quando essa métrica atinge`0`. Para recuperar, você deve excluir índices, adicionar instâncias maiores ou adicionar armazenamento EBS às instâncias existentes. Para saber mais, consulte the section called “Falta de espaço de armazenamento disponível”.

Métrica	Description
	O console OpenSearch de serviço exibe esse valor em GiB. O CloudWatch console da Amazon o exibe em MiB.
`JVMMemory Pressure`	A porcentagem máxima do heap Java usada para todos os nós de dados no cluster. OpenSearch O serviço usa metade da RAM de uma instância para o heap Java, até um tamanho de heap de 32 GiB. Você pode dimensionar instâncias verticalmente até 64 GiB de RAM, sendo que nesse ponto você poderá dimensionar horizontalmente adicionan do instâncias. Consulte the section called “ CloudWatch Alarmes recomendados”.
`SysMemory Utilization`	O percentual de memória da instância que está em uso. Valores altos para essa métrica são normais e geralmente não representam um problema com seu cluster. Para obter um melhor indicador de possíveis problemas de performance e estabilidade, consulte a métrica `JVMMemoryPressure` .
`IndexingLatency`	A diferença no tempo total, em milissegundos, obtida por todas as operações de indexação em um nó entre o minuto N e o minuto (N-1).
`IndexingRate`	O número de operações de indexação por minuto.
`SearchLatency`	A diferença no tempo total, em milissegundos, obtida por todas as pesquisas em um nó entre o minuto N e o minuto (N-1).
`SearchRate`	O número total de solicitações de pesquisa por minuto para todos os fragmentos em um nó de dados.
`Threadpoo lSearchQueue`	O número de tarefas na fila no grupo de thread de pesquisa. Se o tamanho da fila é consistentemente alto, considere escalonar seu cluster. O tamanho da fila de pesquisa máximo é 1.000.
`Threadpoo lWriteQueue`	O número de tarefas na fila no grupo de threads de gravação.

Métrica	Description
`ThreadpoolSearchRejected`	O número de tarefas rejeitadas no grupo de thread de pesquisa. Se esse número continuar a crescer, considere escalonar seu cluster.
`ThreadpoolWriteRejected`	O número de tarefas rejeitadas no grupo de threads de gravação.

Métricas no nível do cluster para clusters em zonas de disponibilidade ativas

Métrica	Description
`DataNodes`	O número total de fragmentos ativos e em espera.
`DataNodesShards.active`	O número total de fragmentos ativos primários e de réplica.
`DataNodesShards.unassigned`	O número de fragmentos que não estão alocados a nós no cluster.
`DataNodesShards.initializing`	O número de fragmentos que estão em inicialização.
`DataNodesShards.relocating`	O número de fragmentos que estão em relocação.

Métricas de alternação da zona de disponibilidade

Se `ActiveReads.`*`Availability-Zone`* `= 1`, então a zona está ativa. Se `ActiveReads.`*`Availability-Zone`* `= 0`, então a zona está em modo de espera.

Métricas pontuais

O Amazon OpenSearch Service fornece as seguintes métricas para pesquisas pontuais (PIT).

Estatísticas de nó coordenador de PIT (por nó coordenador)

Métrica	Description
`CurrentPointInTime`	O número de contextos de pesquisa PIT ativos no nó.
`TotalPointInTime`	O número de contextos de pesquisa de PIT expirados desde o momento de ativação do nó.
`AvgPointInTimeAliveTime`	A média de manutenção ativa dos contextos de pesquisa de PIT desde o momento de ativação do nó.
`HasActivePointInTime`	Um valor de 1 indica que há contextos PIT ativos nos nós desde o tempo de atividade do nó. Um valor de zero significa que não há.
`HasUsedPointInTime`	Um valor de 1 indica que há contextos PIT expirados nos nós desde o tempo de atividade do nó. Um valor de zero significa que não há.

Métricas de SQL

O Amazon OpenSearch Service fornece as seguintes métricas para suporte a SQL.

Métrica	Description
`SQLFailedRequestCountByCusErr`	O número de solicitações com falha para a API `_sql` devido a um problema do cliente. Por exemplo, uma solicitação pode retornar o código de status HTTP 400 devido a um `IndexNotFoundException` . Estatísticas relevantes: soma
`SQLFailedRequestCountBySysErr`	O número de solicitações com falha para a API `_sql` devido a um problema de servidor ou limitação de recurso. Por exemplo, uma solicitação pode retornar o código de status HTTP 503 devido a um `VerificationException` . Estatísticas relevantes: soma

Métrica	Description
`SQLRequestCount`	O número de solicitações para a API `_sql`. Estatísticas relevantes: soma
`SQLDefaultCursorRequestCount`	Semelhante a `SQLRequestCount` , mas conta apenas solicitações de paginação. Estatísticas relevantes: soma
`SQLUnhealthy`	Um valor de 1 indica que, em resposta a determinadas solicitações, o plug-in do SQL está retornando códigos de resposta 5xx ou passando DSL de consulta inválida para o OpenSearch. Outras solicitações devem continuar a ter êxito. Um valor de 0 indica que não há falhas recentes. Se você vir um valor sustentado de 1, solucione o problema das solicitações que seus clientes estão fazendo ao plug-in. Estatística relevante: máximo

Métricas de k-NN

O Amazon OpenSearch Service inclui as seguintes métricas para o plug-in k-near neighbor (k-NN).

Métrica	Description
`KNNCacheCapacityReached`	Métrica por nó para determinar se a capacidade do cache foi atingida. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatística relevante: máximo
`KNNCircuitBreakerTriggered`	Métrica por cluster para determinar se o disjuntor foi acionado. Se algum nó retornar um valor 1 para `KNNCacheCapacityReached` , esse valor também retornará 1. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatística relevante: máximo

Métrica	Description
KNNEvictionCount	Métrica por nó para o número de gráficos que foram removidos do cache devido a restrições de memória ou tempo ocioso. Remoções explícitas que ocorrem devido à exclusão do índice não são contadas. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma
KNNGraphIndexErrors	Métrica por nó para o número de solicitações para adicionar o campo knn_vector de um documento a um gráfico que produziram erros. Estatísticas relevantes: soma
KNNGraphIndexRequests	Métrica por nó para o número de solicitações para adicionar o campo knn_vector de um documento a um gráfico. Estatísticas relevantes: soma
KNNGraphMemoryUsage	Métrica por nó para o tamanho do cache atual (tamanho total de todos os gráficos na memória) em kilobytes. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatística relevante: média
KNNGraphQueryErrors	Métrica por nó para o número de consultas de gráfico que produziram erros. Estatísticas relevantes: soma
KNNGraphQueryRequests	Métrica por nó para o número de consultas de gráfico. Estatísticas relevantes: soma

Métrica	Description
KNNHitCount	Métrica por nó para o número de acertos de cache. Um acerto de cache ocorre quando um usuário consulta um gráfico que já está carregado na memória. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma
KNNLoadExceptionCount	Métrica por nó para o número de vezes que uma exceção ocorreu ao tentar carregar um gráfico no cache. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma
KNNLoadSuccessCount	Métrica por nó para o número de vezes que o plug-in carregou com êxito um gráfico no cache. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma
KNNMissCount	Métrica por nó para o número de perdas do cache. Uma perda de cache ocorre quando um usuário consulta um gráfico que ainda não está carregado na memória. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma
KNNQueryRequests	Métrica por nó para o número de solicitações de consulta recebidas pelo plug-in k-NN. Estatísticas relevantes: soma
KNNRemoteBuildEnabled	Valor binário que especifica se o recurso está ativado. Estatísticas relevantes: Binário
KNNRemoteIndexBuildFailureCount	Número total de falhas de compilação. Estatísticas relevantes: soma

Métrica	Description
KNNRemoteIndexBuildSuccessCount	Número total de construções bem-sucedidas. Estatísticas relevantes: soma
KNNScriptCompilationErrors	Métrica por nó para o número de erros durante a compilação de scripts. Essa estatística só é relevante para a pesquisa de scripts de pontuação k-NN. Estatísticas relevantes: soma
KNNScriptCompilations	Métrica por nó para o número de vezes que o script k-NN foi compilado. Esse valor normalmente deve ser 1 ou 0, mas se o cache que contém os scripts compilados estiver preenchido, o script k-NN poderá ser recompilado. Essa estatística só é relevante para a pesquisa de scripts de pontuação k-NN. Estatísticas relevantes: soma
KNNScriptQueryErrors	Métrica por nó para o número de erros durante consultas de scripts. Essa estatística só é relevante para a pesquisa de scripts de pontuação k-NN. Estatísticas relevantes: soma
KNNScriptQueryRequests	Métrica por nó para o número total de consultas de scripts. Essa estatística só é relevante para a pesquisa de scripts de pontuação k-NN. Estatísticas relevantes: soma
KNNTotalLoadTime	O tempo em nanossegundos que o algoritmo k-NN demorou para carregar gráficos no cache. Essa métrica só é relevante para pesquisas k-NN aproximadas. Estatísticas relevantes: soma

Métrica	Description
`VectorIndexBuildAcceleration0CU`	O número de unidades de OpenSearch computação (OCUs) usadas para acelerar a indexação vetorial. Estatísticas relevantes: soma

Métricas de pesquisa entre clusters

O Amazon OpenSearch Service fornece as seguintes métricas para pesquisa entre clusters.

Métricas de domínio de origem

Métrica	Dimensão	Description
`CrossClusterOutboundConnections`	`ConnectionId`	Número de nós conectados. Se sua resposta incluir um ou mais domínios ignorados, use essa métrica para rastrear quaisquer conexões não íntegras. Se esse número cair para 0, a conexão não estará íntegra.
`CrossClusterOutboundRequests`	`ConnectionId`	Número de solicitações de pesquisa enviadas para o domínio de destino. Use para verificar se a carga de solicitações de pesquisa entre clusters está sobrecarregando o domínio, correlacione qualquer pico nessa métrica com qualquer pico de JVM/CPU.

Métrica de domínio de destino

Métrica	Dimensão	Description
`CrossClusterInboundRequests`	`ConnectionId`	Número de solicitações de conexão de entrada recebidas do domínio de origem.

Adicione um CloudWatch alarme no caso de você perder uma conexão inesperadamente. Para ver as etapas para criar um alarme, consulte Criar um CloudWatch alarme com base em um limite estático.

Métricas de replicação entre clusters

O Amazon OpenSearch Service fornece as seguintes métricas para replicação entre clusters.

Métrica	Description
`ReplicationRate`	A taxa média de operações de replicação por segundo. Essa métrica é semelhante à métrica do `IndexingRate` .
`LeaderCheckPoint`	Para uma conexão específica, a soma dos valores do ponto de verificação líder em todos os índices de replicação. Você pode usar essa métrica para medir a latência de replicação.
`FollowerCheckPoint`	Para uma conexão específica, a soma dos valores do ponto de verificação seguidor em todos os índices de replicação. Você pode usar essa métrica para medir a latência de replicação.
`ReplicationNumSyncingIndices`	O número de índices que têm um status de replicação de `SYNCING`.
`ReplicationNumBootstrappingIndices`	O número de índices que têm um status de replicação de `BOOTSTRAPPING` .
`ReplicationNumPausedIndices`	O número de índices que têm um status de replicação de `PAUSED`.
`ReplicationNumFailedIndices`	O número de índices que têm um status de replicação de `FAILED`.

Métrica	Description
`CrossClusterOutboundReplicationRequests`	O número de solicitações de transporte de replicação no domínio seguidor. Solicitações de transporte são internas e ocorrem sempre que uma operação de API de replicação é chamada. Também ocorrem quando as pesquisas do domínio do seguidor mudam do domínio líder.
`CrossClusterInboundReplicationRequests`	O número de solicitações de transporte de replicação no domínio líder. Solicitações de transporte são internas e ocorrem sempre que uma operação de API de replicação é chamada.
`AutoFollowNumSuccessStartReplication`	O número de índices seguidores que foram criados com êxito por uma regra de replicação para uma conexão específica.
`AutoFollowNumFailedStartReplication`	O número de índices seguidores que falharam ao serem criados por uma regra de replicação quando havia um padrão de correspondência. Esse problema pode surgir devido a um problema de rede no cluster remoto ou devido a um problema de segurança (ou seja, a função associada não tem permissão para iniciar a replicação).
`AutoFollowLeaderCallFailure`	Se houve alguma consulta com falha entre o índice seguidor e o índice líder para extrair novos dados. Um valor de 1 significa que houve uma ou mais chamadas com falha no último minuto.

Métricas de Learning to Rank

O Amazon OpenSearch Service fornece as seguintes métricas para Learning to Rank.

Métrica	Description
`LTRRequestTotalCount`	Contagem total de solicitações de classificação.

Métrica	Description
`LTRRequestErrorCount`	Contagem total de solicitações malsucedidas.
`LTRStatus.red`	Rastreia se um dos índices necessários para executar o plug-in é vermelho.
`LTRMemoryUsage`	Memória total usada pelo plug-in.
`LTRFeatureMemoryUsageInBytes`	A quantidade de memória, em bytes, usada pelos campos de recursos do Learning to Rank.
`LTRFeaturesetMemoryUsageInBytes`	A quantidade de memória, em bytes, usada por todos os conjuntos de recursos do Learning to Rank.
`LTRModelMemoryUsageInBytes`	A quantidade de memória, em bytes, usada por todos os modelos do Learning to Rank.

Métricas da Piped Processing Language

O Amazon OpenSearch Service fornece as seguintes métricas para a linguagem de processamento canalizada.

Métrica	Description
`PPLFailedRequestCountByCusErr`	O número de solicitações com falha para a API `_ppl` devido a um problema do cliente. Por exemplo, uma solicitação pode retornar o código de status HTTP 400 devido a um `IndexNotFoundException` .
`PPLFailedRequestCountBySysErr`	O número de solicitações com falha para a API `_ppl` devido a um problema de servidor ou limitação de recurso. Por exemplo, uma

Métrica	Description
	solicitação pode retornar o código de status HTTP 503 devido a um `VerificationException` .
`PPLRequestCount`	O número de solicitações para a API `_ppl`.

OpenSearch Registros de monitoramento com o Amazon CloudWatch Logs

O Amazon OpenSearch Service expõe os seguintes OpenSearch registros por meio do Amazon CloudWatch Logs:

- Logs de erro
- Logs lentos de solicitação de pesquisa
- Logs lentos de fragmentos
- Logs de auditoria

Os logs lentos de fragmentos de pesquisa, logs lentos de fragmentos de indexação e logs de erros são úteis para solucionar problemas de desempenho e estabilidade. Os logs de auditoria rastreiam a atividade do usuário para fins de conformidade Por padrão, todos os logs são desabilitados. Se ativado, o CloudWatch preço padrão se aplica.

> **Note**
>
> Os registros de erros estão disponíveis somente para as versões 5.1 OpenSearch e posteriores do Elasticsearch. Os registros lentos estão disponíveis para todas as versões OpenSearch e para o Elasticsearch.

Para seus registros, OpenSearch usa o Apache Log4j 2 e seus níveis de log integrados (do menos ao mais severo) de`TRACE`,,`DEBUG`, `INFO``WARN`, e. `ERROR` `FATAL`

Se você ativar os registros de erros, o OpenSearch Serviço publicará linhas de registro de `WARN``ERROR`, e `FATAL` para CloudWatch. OpenSearch O serviço também publica várias exceções do `DEBUG` nível, incluindo as seguintes:

- `org.opensearch.index.mapper.MapperParsingException`
- `org.opensearch.index.query.QueryShardException`
- `org.opensearch.action.search.SearchPhaseExecutionException`
- `org.opensearch.common.util.concurrent.OpenSearchRejectedExecutionException`
- `java.lang.IllegalArgumentException`

Os logs de erros podem ajudar a solucionar problemas em muitas situações, incluindo:

- Problemas de compilação de script Painless
- Consultas inválidas
- Indexação de problemas
- Falhas de snapshots
- Falhas de migração do Index State Management

> **Note**
>
> Nem todos os erros são registrados nos logs de erros.

> **Note**
>
> OpenSearch O serviço não registra todos os erros que ocorrem.

Tópicos

- Habilitação da publicação de logs (console)
- Habilitação da publicação de logs (AWS CLI)
- Habilitação da publicação de logs (AWS SDKs)
- Habilitação da publicação de logs (CloudFormation)
- Como definir limites de log lento para solicitações de pesquisa
- Como definir limites de logs lentos de fragmentos
- Teste logs lentos
- Visualizar logs

Habilitação da publicação de logs (console)

O console OpenSearch de serviço é a maneira mais simples de permitir a publicação de registros no CloudWatch.

Para habilitar a publicação de registros em CloudWatch (console)

1. Acesse aws.amazon.com e escolha Fazer login e forneça suas credenciais.
2. Em Analytics, escolha Amazon OpenSearch Service.
3. Selecione o domínio que deseja atualizar.
4. Na guia Logs, selecione um tipo de log e escolha Habilitar.
5. Crie um novo grupo de CloudWatch registros ou escolha um existente.

> **Note**
>
> Se você planejar habilitar vários logs, recomendamos publicar cada um em seu próprio grupo de logs. Essa separação torna os logs mais fáceis de serem encontrados.

6. Escolha uma política de acesso que contenha as permissões apropriadas ou crie uma política usando o JSON fornecido pelo console:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
      ],
      "Resource": "arn:aws:logs:us-east-1:111122223333:log-
group:cw_log_group_name:*"
    }
  ]
```

```
}
```

Recomendamos que você adicione as chaves de condição `aws:SourceAccount` e `aws:SourceArn` na política para se proteger contra o problema confused deputy. A conta de origem é o proprietário do domínio e o ARN de origem é o ARN do domínio. Para adicionar essas chaves de condição, o seu domínio deve estar no software de serviço R20211203 ou superior.

Por exemplo, você poderia adicionar o bloco de condições a seguir na política:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

> **Important**
>
> CloudWatch O Logs oferece suporte a 10 políticas de recursos por região. Se você planeja habilitar registros para vários domínios OpenSearch de serviço, crie e reutilize uma política mais ampla que inclua vários grupos de registros para evitar atingir esse limite. Para obter as etapas sobre como atualizar a política, consulte the section called "Habilitação da publicação de logs (AWS CLI)".

7. Escolha Habilitar.

   O status de seu domínio muda de Active para Processing. O status deve retornar para Ativo antes que a publicação de logs seja habilitada. Essa alteração geralmente leva 30 minutos, mas pode demorar mais, dependendo da configuração do domínio.

Se você tiver habilitado um dos logs lentos de fragmentos, consulte the section called "Como definir limites de logs lentos de fragmentos". Se você habilitou os logs de auditoria, consulte the section called "Etapa 2: ativar os registros de auditoria nos OpenSearch painéis". Se tiver habilitado apenas logs de erros, você não precisará executar nenhuma etapa de configuração adicional.

Habilitação da publicação de logs (AWS CLI)

Antes de habilitar a publicação de registros, você precisa de um grupo de CloudWatch registros. Se você ainda não tem, pode criar um usando o seguinte comando:

```
aws logs create-log-group --log-group-name my-log-group
```

Digite o comando seguinte para localizar o ARN do grupo de log e anote-o:

```
aws logs describe-log-groups --log-group-name my-log-group
```

Agora você pode conceder permissões ao OpenSearch Serviço para gravar no grupo de registros. Você deve fornecer o ARN do grupo de log quase no final do comando:

```
aws logs put-resource-policy \
  --policy-name my-policy \
  --policy-document '{ "Version": "2012-10-17",        "Statement": [{ "Sid":
 "", "Effect": "Allow", "Principal": { "Service": "es.amazonaws.com"}, "Action":
[ "logs:PutLogEvents","logs:CreateLogStream"],"Resource": "cw_log_group_arn:*"}]}'
```

> **Important**
>
> CloudWatch O Logs oferece suporte a 10 políticas de recursos por região. Se você planeja habilitar a fragmentação de registros lentos para vários domínios de OpenSearch serviço, crie e reutilize uma política mais ampla que inclua vários grupos de registros para evitar atingir esse limite.

Se você precisar revisar essa política posteriormente, use o comando `aws logs describe-resource-policies`. Para atualizar a política, emita o mesmo comando `aws logs put-resource-policy` com um novo documento de política.

Por fim, você pode usar a `--log-publishing-options` opção de habilitar a publicação. A sintaxe para essa opção é a mesma para os comandos `create-domain` e `update-domain-config`.

Parâmetro	Valores válidos
`--log-publishing-options`	`SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn=` *cw_log_group_arn* `,Enabled=true|false}`
	`INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn=` *cw_log_group_arn* `,Enabled=true|false}`
	`ES_APPLICATION_LOGS={CloudWatchLogsLogGroupArn=` *cw_log_group_arn* `,Enabled=true|false}`
	`AUDIT_LOGS={CloudWatchLogsLogGroupArn=` *cw_log_group_arn* `,Enabled=true|false}`

> **Note**
>
> Se você planejar habilitar vários logs, recomendamos publicar cada um em seu próprio grupo de logs. Essa separação torna os logs mais fáceis de serem encontrados.

Exemplo

O exemplo a seguir habilita a publicação de pesquisa e logs lentos de fragmentos de indexação no domínio especificado:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --log-publishing-options
 "SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-log-
group,Enabled=true},INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-other-log-group,Enabled=true}"
```

Para desativar a publicação em CloudWatch, execute o mesmo comando com`Enabled=false`.

Se você tiver habilitado um dos logs lentos de fragmentos, consulte the section called “Como definir limites de logs lentos de fragmentos”. Se você habilitou os logs de auditoria, consulte the section called “Etapa 2: ativar os registros de auditoria nos OpenSearch painéis”. Se tiver habilitado apenas logs de erros, você não precisará executar nenhuma etapa de configuração adicional.

Habilitação da publicação de logs (AWS SDKs)

Antes de habilitar a publicação de registros, você deve primeiro criar um grupo de CloudWatch registros, obter seu ARN e conceder permissões ao OpenSearch Serviço para gravar nele. As operações relevantes estão documentadas na Referência da API Amazon CloudWatch Logs:

- `CreateLogGroup`
- `DescribeLogGroup`
- `PutResourcePolicy`

Você pode acessar essas operações usando AWS SDKso.

O AWS SDKs (exceto o Android e o iOS SDKs) suporta todas as operações definidas na Amazon OpenSearch Service API Reference, incluindo a `--log-publishing-options` opção de `CreateDomain` `UpdateDomainConfig` e.

Se você tiver habilitado um dos logs lentos de fragmentos, consulte the section called “Como definir limites de logs lentos de fragmentos”. Se tiver habilitado apenas logs de erros, você não precisará executar nenhuma etapa de configuração adicional.

Habilitação da publicação de logs (CloudFormation)

Neste exemplo, usamos CloudFormation para criar um grupo de registros chamado`opensearch-logs`, atribuir as permissões apropriadas e, em seguida, criar um domínio com a publicação de registros ativada para registros de aplicativos, fragmentar registros lentos de pesquisa e indexar registros lentos.

Antes de habilitar a publicação de registros, você precisa criar um grupo de CloudWatch registros:

```
Resources:
  OpenSearchLogGroup:
    Type: AWS::Logs::LogGroup
    Properties:
      LogGroupName: opensearch-logs
Outputs:
  Arn:
    Value:
      'Fn::GetAtt':
        - OpenSearchLogGroup
        - Arn
```

O modelo gera o ARN do grupo de logs. Neste caso, o ARN é `arn:aws:logs:us-east-1:123456789012:log-group:opensearch-logs`.

Usando o ARN, crie uma política de recursos que dê ao OpenSearch serviço permissões para gravar no grupo de registros:

```
Resources:
 OpenSearchLogPolicy:
   Type: AWS::Logs::ResourcePolicy
   Properties:
     PolicyName: my-policy
     PolicyDocument: "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Sid\": \"\",
 \"Effect\": \"Allow\", \"Principal\": { \"Service\": \"es.amazonaws.com\"}, \"Action
\":[ \"logs:PutLogEvents\",\"logs:CreateLogStream\"],\"Resource\": \"arn:aws:logs:us-
east-1:123456789012:log-group:opensearch-logs:*\"}]}"
```

Por fim, crie a CloudFormation pilha a seguir, que gera um domínio OpenSearch de serviço com publicação de registros. A política de acesso permite que o usuário faça todas as solicitações HTTP para o domínio. Conta da AWS

```
Resources:
  OpenSearchServiceDomain:
    Type: "AWS::OpenSearchService::Domain"
    Properties:
      DomainName: my-domain
      EngineVersion: "OpenSearch_1.0"
      ClusterConfig:
        InstanceCount: 2
        InstanceType: "r6g.xlarge.search"
        DedicatedMasterEnabled: true
        DedicatedMasterCount: 3
        DedicatedMasterType: "r6g.xlarge.search"
      EBSOptions:
        EBSEnabled: true
        VolumeSize: 10
        VolumeType: "gp2"
      AccessPolicies:
        Version: "2012-10-17"
        Statement:
            Effect: "Allow"
            Principal:
                AWS: "arn:aws:iam::123456789012:user/es-user"
```

```
            Action: "es:*"
            Resource: "arn:aws:es:us-east-1:123456789012:domain/my-domain/*"
      LogPublishingOptions:
        ES_APPLICATION_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        SEARCH_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        INDEX_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
```

Para obter informações detalhadas sobre sintaxe, consulte as opções de publicação de logs no Manual do usuário do CloudFormation .

Como definir limites de log lento para solicitações de pesquisa

Os registros lentos da solicitação de pesquisa estão disponíveis para pesquisa em domínios OpenSearch de serviço executados na versão 2.13 e posterior. Os limites de log lento da solicitação de pesquisa são configurados para o tempo de duração total da solicitação. Isso é diferente dos logs lentos de solicitações de fragmentos, que são configurados para o tempo de duração de cada fragmento.

Você pode especificar logs lentos de solicitação de pesquisa com configurações de cluster. Isso é diferente dos logs lentos de fragmentos, que você ativa com as configurações de índice. Por exemplo, você pode especificar as seguintes configurações por meio da API OpenSearch REST:

```
PUT domain-endpoint/_cluster/settings
{
  "transient": {
    "cluster.search.request.slowlog.threshold.warn": "5s",
    "cluster.search.request.slowlog.threshold.info": "2s"
  }
}
```

> **Note**
>
> O PUT `/_cluster/settings` endpoint é restrito no Amazon OpenSearch Service. O envio dessa solicitação retornará o erro: `Your request: '/_cluster/settings' payload is not allowed.` para definir limites de log lentos de fragmentação, use configurações em nível de índice. Por exemplo:
>
> ```
> PUT domain-endpoint/index/_settings
> {
>   "index.search.slowlog.threshold.query.warn": "5s",
>   "index.search.slowlog.threshold.query.info": "2s"
> }
> ```

Como definir limites de logs lentos de fragmentos

OpenSearch desativa a fragmentação de registros lentos por padrão. Depois de habilitar a publicação de fragmentos de registros lentos no CloudWatch, você ainda precisa especificar limites de registro para cada OpenSearch índice. Esses limites definem exatamente o que deve ser registrado e em que nível de log.

Por exemplo, você pode especificar essas configurações por meio da API OpenSearch REST:

```
PUT domain-endpoint/index/_settings
{
  "index.search.slowlog.threshold.query.warn": "5s",
  "index.search.slowlog.threshold.query.info": "2s"
}
```

Teste logs lentos

Para testar se a solicitação de pesquisa e os registros lentos de fragmentação estão sendo publicados com êxito, considere começar com valores muito baixos para verificar se os registros aparecem e CloudWatch, em seguida, aumentar os limites para níveis mais úteis.

Se os logs não aparecerem, verifique o seguinte:

- O grupo CloudWatch de registros existe? Verifique o CloudWatch console.

- O OpenSearch serviço tem permissões para gravar no grupo de registros? Verifique o console OpenSearch de serviço.
- O domínio do OpenSearch serviço está configurado para publicar no grupo de registros? Verifique o console de OpenSearch serviço, use a AWS CLI `describe-domain-config` opção ou ligue `DescribeDomainConfig` usando um dos SDKs.
- Os limites de OpenSearch registro são baixos o suficiente para que suas solicitações os excedam?

  Para revisar os limites de logs lentos de sua solicitação de pesquisa para um domínio, use o seguinte comando:

  ```
  GET domain-endpoint/_cluster/settings?flat_settings
  ```

  Para revisar seus limites de logs lentos de fragmentos para um índice, use o seguinte comando:

  ```
  GET domain-endpoint/index/_settings?pretty
  ```

Se você quer desabilitar logs lentos para um índice, redefina todos os limites que você mudou para os valores padrão de `-1`.

Desabilitar a publicação para CloudWatch usar o console de OpenSearch serviço ou AWS CLI não interrompe OpenSearch a geração de registros; apenas interrompe a publicação desses registros. Verifique suas configurações de índice se você não precisar mais dos logs lentos de fragmentos e suas configurações de domínio se não precisar mais dos logs lentos de solicitação de pesquisa.

Visualizar logs

Visualizar o aplicativo e os logins lentos CloudWatch é como visualizar qualquer outro CloudWatch registro. Para obter mais informações, consulte Exibir dados de registro no Guia do usuário do Amazon CloudWatch Logs.

Algumas considerações sobre a visualização de logs:

- OpenSearch O serviço publica somente os primeiros 255.000 caracteres de cada linha para. CloudWatch O conteúdo restante ficará truncado. Para logs de auditoria, o limite é de 10.000 caracteres por mensagem.
- Em CloudWatch, os nomes dos fluxos de log têm sufixos de`-index-slow-logs`, `-search-slow-logs-application-logs`, e `-audit-logs` para ajudar a identificar seu conteúdo.

Monitorando registros de auditoria no Amazon OpenSearch Service

Se o seu domínio do Amazon OpenSearch Service usa controle de acesso refinado, você pode habilitar registros de auditoria para seus dados. Os registros de auditoria são altamente personalizáveis e permitem que você acompanhe a atividade do usuário em seus OpenSearch clusters, incluindo sucesso e falhas de autenticação, solicitações OpenSearch, alterações de índice e consultas de pesquisa recebidas. A configuração padrão monitora um conjunto popular de ações do usuário, mas recomendamos adaptar as configurações às suas necessidades exatas.

Assim como os registros de OpenSearch aplicativos e os registros lentos, o OpenSearch Service publica registros de auditoria no CloudWatch Logs. Se ativado, o CloudWatch preço padrão se aplica.

> **Note**
>
> Para ativar os registros de auditoria, sua função de usuário deve ser mapeada para a `security_manager` função, o que lhe dá acesso à API OpenSearch `plugins/_security` REST. Para saber mais, consulte the section called “Modificação do usuário primário”.

Tópicos

- Limitações
- Habilitação dos logs de auditoria
- Ative o registro de auditoria usando o AWS CLI
- Habilitar o registro de auditoria em log usando a API de configuração
- Camadas e categorias do log de auditoria
- Configurações do log de auditoria
- Exemplo de log de auditoria
- Configuração de logs de auditoria usando a API REST

Limitações

Os logs de auditoria têm as seguintes limitações:

- Os logs de auditoria não incluem solicitações de pesquisa entre clusters que foram rejeitadas pela política de acesso ao domínio do destino.

- O tamanho máximo de cada mensagem do log de auditoria é 10.000 caracteres. A mensagem do log de auditoria será truncada se exceder esse limite.

Habilitação dos logs de auditoria

A habilitação dos logs de auditoria é um processo em duas etapas. Primeiro, você configura seu domínio para publicar registros de auditoria no CloudWatch Logs. Em seguida, você ativa os registros de auditoria nos OpenSearch painéis e os configura para atender às suas necessidades.

> **Important**
>
> Se você encontrar um erro ao seguir essas etapas, consulte the section called “Não é possível habilitar logs de auditoria” para obter informações de solução de problemas.

Etapa 1: ativar registros de log e configurar uma política de acesso

Estas etapas descrevem como habilitar logs de auditoria usando o console. Você também pode habilitá-los usando o AWS CLI, ou a API OpenSearch de serviço.

Para habilitar registros de auditoria para um domínio OpenSearch de serviço (console)

1. Escolha o domínio para abrir sua configuração e, em seguida, acesse a guia Logs.
2. Selecione Logs de auditoria e, em seguida, Habilitar.
3. Crie um grupo de CloudWatch registros ou escolha um existente.
4. Escolha uma política de acesso que contenha as permissões apropriadas ou crie uma política usando o JSON fornecido pelo console:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
```

```
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
      ],
      "Resource": "arn:aws:logs:us-east-1:111122223333:log-group:/aws/
opensearch/domains/domain-name/*"
    }
  ]
}
```

Recomendamos que você adicione as chaves de condição `aws:SourceAccount` e `aws:SourceArn` na política para se proteger contra o problema confused deputy. A conta de origem é o proprietário do domínio e o ARN de origem é o ARN do domínio. Para adicionar essas chaves de condição, o seu domínio deve estar no software de serviço R20211203 ou superior.

Por exemplo, você poderia adicionar o bloco de condições a seguir na política:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

5. Escolha Habilitar.

Etapa 2: ativar os registros de auditoria nos OpenSearch painéis

Depois de habilitar os registros de auditoria no console de OpenSearch serviços, você também deve habilitá-los nos OpenSearch painéis e configurá-los para atender às suas necessidades.

1. Abra OpenSearch Painéis e escolha Segurança no menu do lado esquerdo.
2. Escolha Logs de auditoria.
3. Escolha Habilitar log de auditoria.

A interface do usuário do Dashboars oferece controle total das configurações do log de auditoria em Configurações gerais e Configurações de compatibilidade. Para obter uma descrição de todas as opções de configuração, consulte Configurações de log de auditoria.

Ative o registro de auditoria usando o AWS CLI

O AWS CLI comando a seguir ativa registros de auditoria em um domínio existente:

```
aws opensearch update-domain-config --domain-name my-domain --log-publishing-options
 "AUDIT_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-
group:my-log-group,Enabled=true}"
```

Você também pode habilitar os logs de auditoria ao criar um domínio. Para saber mais, consulte a Referência de comandos da AWS CLI.

Habilitar o registro de auditoria em log usando a API de configuração

A seguinte solicitação para a API de configuração habilita os logs de auditoria em um domínio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "LogPublishingOptions": {
    "AUDIT_LOGS": {
      "CloudWatchLogsLogGroupArn":"arn:aws:logs:us-east-1:123456789012:log-
group1:sample-domain",
      "Enabled":true
    }
  }
}
```

Para obter mais informações, consulte a referência da Amazon OpenSearch Service API.

Camadas e categorias do log de auditoria

A comunicação do cluster ocorre em duas camadas separadas: a camada REST e a camada de transporte.

- A camada REST abrange a comunicação com clientes HTTP, como curl, Logstash, OpenSearch Dashboards, o cliente REST de alto nível Java, a biblioteca Python Requests — todas as solicitações HTTP que chegam ao cluster.

- A camada de transporte cobre a comunicação entre nós. Por exemplo, depois que uma solicitação de pesquisa chega ao cluster (sobre a camada REST), o nó de coordenação que atende à solicitação envia a consulta para outros nós, recebe suas respostas, coleta os documentos necessários e os reúne na resposta final. Operações como alocação de fragmentos e rebalanceamento também ocorrem sobre a camada de transporte.

Você pode habilitar ou desabilitar logs de auditoria para camadas inteiras, bem como categorias de auditoria individuais para uma camada. A tabela a seguir contém um resumo das categorias de auditoria e das camadas para as quais elas estão disponíveis.

Categoria	Description	Disponível para REST	Disponível para transporte
FAILED_LOGIN	Uma solicitação continha credenciais inválidas, e a autenticação falhou.	Sim	Sim
MISSING_PRIVILEGES	Um usuário não tinha os privilégios necessários para fazer a solicitação.	Sim	Sim
GRANTED_PRIVILEGES	Um usuário tinha os privilégios necessários para fazer a solicitação.	Sim	Sim
OPENSEARCH_SECURITY_INDEX_ATTEMPT	Uma solicitação tentou modificar o índice `.opendistro_security` .	Não	Sim
AUTHENTICATED	Uma solicitação continha credenciais válidas e a autenticação foi bem-sucedida.	Sim	Sim
INDEX_EVENT	Uma solicitação executou uma operação	Não	Sim

Categoria	Description	Disponível para REST	Disponível para transporte
	administrativa em um índice, como criar um, definir um alias ou executar uma mesclagem forçada. A lista completa de `indices:admin/` ações que essa categoria inclui está disponível na OpenSearch documentação.		

Além dessas categorias padrão, o controle de acesso refinado oferece várias categorias adicionais projetadas para atender aos requisitos de conformidade de dados.

Categoria	Description
COMPLIANCE_DOC_READ	Uma solicitação executou um evento de leitura em um documento em um índice.
COMPLIANCE_DOC_WRITE	Uma solicitação executou um evento de gravação em um documento em um índice.
COMPLIANCE_INTERNAL_CONFIG_READ	Uma solicitação executou um evento de leitura no índice `.opendistro_security` .
COMPLIANCE_INTERNAL_CONFIG_WRITE	Uma solicitação executou um evento de gravação no índice `.opendistro_security` .

Você pode ter qualquer combinação de categorias e atributos de mensagem. Por exemplo, se você enviar uma solicitação REST para indexar um documento, poderá ver as seguintes linhas nos logs de auditoria:

- AUTHENTICATED na camada REST (autenticação)
- GRANTED_PRIVILEGE na camada de transporte (autorização)
- COMPLIANCE_DOC_WRITE (documento gravado em um índice)

Configurações do log de auditoria

Há várias opções de configuração para os logs de auditoria.

Configurações gerais

As configurações gerais permitem habilitar ou desabilitar categorias individuais ou camadas inteiras. Recomendamos enfaticamente manter GRANTED_PRIVILEGES e AUTHENTICATED como categorias excluídas. Caso contrário, essas categorias serão registradas para cada solicitação válida para o cluster.

Nome	Configuração de backend	Description
Camada REST	enable_rest	Habilite ou desabilite eventos que ocorrem na camada REST.
Categorias desabilitadas de REST	disabled_ rest_categories	Especifique categorias de auditoria a serem ignoradas na camada REST. Modificar essas categorias pode aumentar drasticamente o tamanho dos logs de auditoria.
Transport Layer	enable_tr ansport	Habilite ou desabilite eventos que acontecem na camada de transporte.
Categorias desabilitadas de transporte	disabled_ transport _categories	Especifique categorias de auditoria que devem ser ignoradas na camada de transporte. Modificar essas categorias pode aumentar drasticamente o tamanho dos logs de auditoria.

As configurações de atributo permitem personalizar a quantidade de detalhes em cada linha de log.

Nome	Configuração de backend	Description
Solicitações em massa	resolve_bulk_requests	Habilitar essa configuração gera um log para cada documento em uma solicitação em massa, o que pode aumentar drasticamente o tamanho dos logs de auditoria.
Corpo da solicitação	log_request_body	Inclua o corpo da solicitação das solicitações.
Resolver índices	resolve_índices	Resolva aliases em índices.

Use as configurações de ignorar para excluir um conjunto de usuários ou caminhos de API:

Nome	Configuração de backend	Description
Usuários ignorados	ignore_users	Especifique os usuários que não deseja incluir.
Solicitações ignoradas	ignore_requests	Especifique padrões de solicitação que não deseja incluir.

Configurações de conformidade

As configurações de conformidade permitem ajustar o acesso ao índice, ao documento ou ao nível de campo.

Nome	Configuração de backend	Description
Log de compatibilidade	enable_compliance	Habilite ou desabilite o log de compatibilidade

Você pode especificar as configurações a seguir para o log de eventos de leitura e gravação.

Nome	Configuração de backend	Description
Log de configuração interno	internal_config	Habilite ou desabilite o log de eventos no índice `.opendistro_security` .

Você pode especificar as configurações a seguir para eventos de leitura.

Nome	Configuração de backend	Description
Ler metadados	read_metadata_only	Incluir apenas metadados para eventos de leitura. Não inclua campos de documento.
Usuários ignorados	read_ignore_users	Não inclua determinados usuários para eventos de leitura.
Campos observados	read_watched_fields	Especifique os índices e campos a serem observados para eventos de leitura. A adição de campos observados gera um log por acesso ao documento, o que pode aumentar drasticamente o tamanho dos logs de auditoria. Os campos observados oferecem suporte a padrões de índice e padrões de campo: `{ "index-name-pattern": [ "field-name-pattern" ], "logs*": [ "message" ], "twitter": [ "id", "user*" ]`

Nome	Configuração de backend	Description
		}

Você pode especificar as configurações a seguir para eventos de gravação.

Nome	Configuração de backend	Description
Metadados de gravação	write_metadata_only	Inclua metadados somente para eventos de gravação. Não inclua campos de documento.
Diferenças de log	write_log_diffs	Se write_metadata_only for false (falso), inclua somente as diferenças entre eventos de gravação.
Usuários ignorados	write_ignore_users	Não inclua determinados usuários para eventos de gravação.
Observar índices	write_watched_índices	Especifique os índices ou padrões de índice para observar eventos de gravação. A adição de campos observados gera um log por acesso ao documento, o que pode aumentar drasticamente o tamanho dos logs de auditoria.

Exemplo de log de auditoria

Esta seção inclui um exemplo de configuração, solicitação de pesquisa e o log de auditoria resultante para todos os eventos de leitura e gravação de um índice.

Etapa 1: Configurar logs de auditoria

Depois de habilitar a publicação de registros de auditoria em um grupo de CloudWatch registros, navegue até a página de registro de auditoria de OpenSearch painéis e escolha Habilitar registro de auditoria.

1. Em Configurações gerais, escolha Configurar e certifique-se de que a opção Camada REST esteja habilitada.

2. Em Configurações de compatibilidade, escolha Configurar.
3. Em Gravação, em Campos observados, adicione `accounts` para todos os eventos de gravação neste índice.
4. Em Ler, em Campos observados, adicione os campos `ssn` e `id-` do índice `accounts`:

```
{
  "accounts-": [
    "ssn",
    "id-"
  ]
}
```

Etapa 2: Executar eventos de leitura e gravação

1. Navegue até OpenSearch Painéis, escolha Dev Tools e indexe um documento de amostra:

```
PUT accounts/_doc/0
{
  "ssn": "123",
  "id-": "456"
}
```

2. Para testar um evento de leitura, envie a seguinte solicitação:

```
GET accounts/_search
{
  "query": {
    "match_all": {}
  }
}
```

Etapa 3: Observar os logs

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. No painel de navegação, escolha Grupos de logs.
3. Escolha o grupo de logs que você especificou ao habilitar os logs de auditoria. Dentro do grupo de registros, o OpenSearch Service cria um fluxo de registros para cada nó em seu domínio.

4. Em Fluxos de log, escolha Pesquisar tudo.
5. Para os eventos de leitura e gravação, consulte os logs correspondentes. Um atraso de 5 segundos antes do log ser exibido é normal.

   Exemplo de gravação de log de auditoria

```
{
  "audit_compliance_operation": "CREATE",
  "audit_cluster_name": "824471164578:audit-test",
  "audit_node_name": "be217225a0b77c2bd76147d3ed3ff83c",
  "audit_category": "COMPLIANCE_DOC_WRITE",
  "audit_request_origin": "REST",
  "audit_compliance_doc_version": 1,
  "audit_node_id": "3xNJhm4XS_yTzEgDWcGRjA",
  "@timestamp": "2020-08-23T05:28:02.285+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "3.236.145.227",
  "audit_trace_doc_id": "lxnJGXQBqZSlDB91r_uZ",
  "audit_request_effective_user": "admin",
  "audit_trace_shard_id": 8,
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

   Exemplo de leitura de log de auditoria

```
{
  "audit_cluster_name": "824471164578:audit-docs",
  "audit_node_name": "806f6050cb45437e2401b07534a1452f",
  "audit_category": "COMPLIANCE_DOC_READ",
  "audit_request_origin": "REST",
  "audit_node_id": "saSevm9ASte0-pjAtYi2UA",
  "@timestamp": "2020-08-31T17:57:05.015+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "54.240.197.228",
  "audit_trace_doc_id": "config:7.7.0",
  "audit_request_effective_user": "admin",
  "audit_trace_shard_id": 0,
```

```
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

Para incluir o corpo da solicitação, retorne às configurações de conformidade nos OpenSearch painéis e desative a opção Gravar metadados. Para excluir eventos por um usuário específico, adicione o usuário a Usuários Ignorados.

Para obter uma descrição de cada campo do log de auditoria, consulte Referência de campos do log de auditoria. Para obter informações sobre como pesquisar e analisar seus dados de registro de auditoria, consulte Análise de dados de log com o CloudWatch Logs Insights no Guia do usuário do Amazon CloudWatch Logs.

Configuração de logs de auditoria usando a API REST

Recomendamos o uso de OpenSearch painéis para configurar registros de auditoria, mas você também pode usar a API REST de controle de acesso refinada. Esta seção contém uma solicitação de exemplo. A documentação completa sobre a API REST está disponível na OpenSearch documentação.

```
PUT _opendistro/_security/api/audit/config
{
  "enabled": true,
  "audit": {
    "enable_rest": true,
    "disabled_rest_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "enable_transport": true,
    "disabled_transport_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "resolve_bulk_requests": true,
    "log_request_body": true,
    "resolve_indices": true,
```

```
    "exclude_sensitive_headers": true,
    "ignore_users": [
      "kibanaserver"
    ],
    "ignore_requests": [
      "SearchRequest",
      "indices:data/read/*",
      "/_cluster/health"
    ]
  },
  "compliance": {
    "enabled": true,
    "internal_config": true,
    "external_config": false,
    "read_metadata_only": true,
    "read_watched_fields": {
      "read-index-1": [
        "field-1",
        "field-2"
      ],
      "read-index-2": [
        "field-3"
      ]
    },
    "read_ignore_users": [
      "read-ignore-1"
    ],
    "write_metadata_only": true,
    "write_log_diffs": false,
    "write_watched_indices": [
      "write-index-1",
      "write-index-2",
      "log-*",
      "*"
    ],
    "write_ignore_users": [
      "write-ignore-1"
    ]
  }
}
```

Eventos do OpenSearch Serviço de Monitoramento com a Amazon EventBridge

O Amazon OpenSearch Service se integra EventBridge à Amazon para notificá-lo sobre determinados eventos que afetam seus domínios. Os eventos dos AWS serviços são entregues quase EventBridge em tempo real. Os mesmos eventos também são enviados para a Amazon CloudWatch Events, a antecessora da Amazon EventBridge. Você pode escrever regras simples para indicar quais eventos são do seu interesse, e as ações automatizadas a serem tomadas quando um evento corresponder à regra. Ações que podem ser automaticamente acionadas incluem:

- Invocando uma função AWS Lambda
- Invocação de um Run Command do Amazon EC2
- Transmitir o evento Amazon Kinesis Data Streams
- Ativando uma máquina de estado AWS Step Functions
- Notificar um tópico do Amazon SNS ou uma fila do Amazon SQS

Para obter mais informações, consulte Comece a usar a Amazon EventBridge no Guia EventBridge do usuário da Amazon.

Tópicos

- Eventos de atualização de software de serviço
- Auto-Tune de eventos
- Eventos de integridade do cluster
- Eventos de endpoint da VPC
- Eventos de desativação do nó
- Eventos de retirada do nó degradado
- Eventos de erro de domínio
- Tutorial: Ouvindo EventBridge eventos do Amazon OpenSearch Service
- Tutorial: Envio de alertas do Amazon SNS para atualizações de software disponíveis

Eventos de atualização de software de serviço

OpenSearch O serviço envia eventos para EventBridge quando ocorre um dos seguintes eventos de atualização do software de serviço.

Atualização do software de serviço disponível

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço está disponível.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update R20220928 available. Service Software
 Deployment Mechanism:
                   Blue/Green. For more information on deployment configuration,
 please
                   see: https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Atualização de software de serviço agendada

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço é agendada. Para atualizações opcionais, você recebe a notificação na data agendada e tem a opção de reagendar a qualquer momento. Para as atualizações obrigatórias, você recebe a notificação três dias antes da data agendada e tem a opção de reagendar dentro da janela obrigatória.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Scheduled",
    "severity": "High",
    "description": "A new service software update [R20200330-p1] has been scheduled at
 [21st May 2023 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
  }
}
```

Atualização do software de serviço reagendada

OpenSearch O serviço envia esse evento quando uma atualização opcional do software do serviço é reagendada. Para obter mais informações, consulte the section called “Atualizações opcionais x obrigatórias”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
```

```
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Rescheduled",
    "severity": "High",
    "description": "The service software update [R20200330-p1], which was originally
 scheduled for
                    [21st May 2023 12:40 GMT], has been rescheduled to [23rd May 2023
 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
  }
}
```

Atualização do software de serviço iniciada

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço é iniciada.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Started",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] started.
  }
}
```

Atualização do software de serviço concluída

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço é concluída.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Completed",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] completed."
  }
}
```

Atualização de software de serviço cancelada

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço é cancelada.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
```

```
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled as a
                    newer update is available. Please schedule the latest update."
  }
}
```

Atualização do software de serviço agendada cancelada

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço que estava agendada anteriormente para o domínio é cancelada.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled."
  }
}
```

Atualização de software de serviço não executada

OpenSearch O serviço envia esse evento quando não consegue iniciar uma atualização do software do serviço.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Unexecuted",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] cannot be
 started. Reason: [reason]"
  }
}
```

Falha na atualização do software de serviço

OpenSearch O serviço envia esse evento quando uma atualização do software do serviço falha.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Failed",
    "severity": "High",
```

```
    "description": "Installation of service software update [R20200330-p1] failed.
 [reason].
  }
}
```

Atualização do software de serviço necessária

OpenSearch O serviço envia esse evento quando é necessária uma atualização do software do serviço. Para obter mais informações, consulte the section called “Atualizações opcionais x obrigatórias”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Required",
    "severity": "High",
    "description": "Service software update [R20200330-p1] available. Update
                    will be automatically installed after [21st May 2023] if no
                    action is taken. Service Software Deployment Mechanism: Blue/Green.
                    For more information on deployment configuration, please see:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Auto-Tune de eventos

OpenSearch O serviço envia eventos para EventBridge quando um dos seguintes eventos de ajuste automático ocorrer.

Auto-Tune pendente

OpenSearch O serviço envia esse evento quando o Auto-Tune identifica recomendações de ajuste para melhorar o desempenho e a disponibilidade do cluster. Você verá esse evento somente para domínios com o Auto-Tune desabilitado.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Pending",
    "description": "Auto-Tune recommends the following new settings for your
 domain: { JVM Heap size : 60%}. Enable Auto-Tune to improve cluster stability and
 performance.",
    "scheduleTime": "{iso8601-timestamp}"
  }
}
```

Auto-Tune iniciado

OpenSearch O serviço envia esse evento quando o Auto-Tune começa a aplicar novas configurações ao seu domínio.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
```

```
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Started",
    "scheduleTime": "{iso8601-timestamp}",
    "startTime": "{iso8601-timestamp}",
    "description" : "Auto-Tune is applying the following settings to your domain: { JVM
 Heap size : 60%}."
  }
}
```

O Auto-Tune requer uma implantação programada blue/green

OpenSearch O serviço envia esse evento quando o Auto-Tune identifica recomendações de ajuste que exigem uma blue/green implantação programada.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Pending",
    "startTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has identified the following settings for your domain
 that require a blue/green deployment: { JVM Heap size : 60%}.
                    You can schedule the deployment for your preferred time."
  }
```

```
}
```

Auto-Tune cancelado

OpenSearch O serviço envia esse evento quando a programação do Auto-Tune é cancelada porque não há recomendações de ajuste pendentes.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Cancelled",
    "scheduleTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has cancelled the upcoming blue/green deployment."
  }
}
```

Auto-Tune concluído

OpenSearch O serviço envia esse evento quando o Auto-Tune conclui a blue/green implantação e o cluster está operacional com as novas configurações de JVM em vigor.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
```

```
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "completionTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has completed the blue/green deployment and successfully
 applied the following settings: { JVM Heap size : 60%}."
  }
}
```

Auto-Tune desabilitado e alterações revertidas

OpenSearch O serviço envia esse evento quando o Auto-Tune é desativado e as alterações aplicadas são revertidas.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": [ "arn:aws:es:us-east-1:123456789012:domain/test-domain" ],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. All settings have been reverted. Auto-
Tune will continue to evaluate
                    cluster performance and provide recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Auto-Tune desabilitado e alterações mantidas

OpenSearch O serviço envia esse evento quando o Auto-Tune é desativado e as alterações aplicadas são mantidas.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. The most-recent settings by Auto-Tune
 have been retained.
                    Auto-Tune will continue to evaluate cluster performance and provide
 recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Eventos de integridade do cluster

OpenSearch O serviço envia determinados eventos para EventBridge quando a integridade do seu cluster está comprometida.

Recuperação de cluster vermelho iniciada

OpenSearch O serviço envia esse evento após o status do cluster ficar vermelho continuamente por mais de uma hora. Tenta restaurar automaticamente um ou mais índices vermelhos de um snapshot para corrigir o status do cluster.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Started",
      "severity":"High",
      "description":"Your cluster status is red. We have started automatic snapshot
 restore for the red indices.
                    No action is needed from your side. Red indices [red-index-0, red-
index-1]"
   }
}
```

Recuperação de cluster vermelho parcialmente concluída

OpenSearch O serviço envia esse evento quando só conseguiu restaurar um subconjunto de índices vermelhos de um snapshot ao tentar corrigir o status de um cluster vermelho.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
```

```
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Partially Restored",
      "severity":"High",
      "description":"Your cluster status is red. We were able to restore the following
 Red indices from
                   snapshot: [red-index-0]. Indices not restored: [red-index-1].
 Please refer https://docs.aws.amazon.com/opensearch-service/latest/developerguide/
handling-errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
}
```

Falha na recuperação de cluster vermelho

OpenSearch O serviço envia esse evento quando não consegue restaurar nenhum índice ao tentar corrigir o status de um cluster vermelho.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Failed",
      "severity":"High",
      "description":"Your cluster status is red. We were unable to restore the Red
 indices automatically.
                   Indices not restored: [red-index-0, red-index-1]. Please refer
 https://docs.aws.amazon.com/opensearch-service/latest/developerguide/handling-
errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
```

```
}
```

Fragmentos a serem excluídos

OpenSearch O serviço envia esse evento quando tenta corrigir automaticamente o status do cluster vermelho depois de ficar vermelho continuamente por 14 dias, mas um ou mais índices permanecem vermelhos. Depois de mais 7 dias (21 dias no total em vermelho contínuo), o OpenSearch Serviço continua excluindo fragmentos não atribuídos em todos os índices vermelhos.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:36:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "severity":"Medium",
      "description":"Your cluster status is red. Please fix the red indices as soon as
 possible.
                    If not fixed by 2022-04-12 01:51:47+00:00, we will delete all
 unassigned shards,
                    the unit of storage and compute, for these red indices to recover
 your domain and make it green.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.
                    test_data, test_data1",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) to be deleted"
   }
}
```

Fragmentos excluídos

OpenSearch O serviço envia esse evento após o status do cluster ficar vermelho continuamente por 21 dias. Continua excluindo os fragmentos não atribuídos (armazenamento e computação) em todos os índices vermelhos. Para obter detalhes, consulte the section called “Correção automática de clusters vermelhos”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:54:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "severity":"High",
      "description":"We have deleted unassinged shards, the unit of storage and
 compute, in
                    red indices: index-1, index-2 because these indices were red for
 more than
                    21 days and could not be restored with the automated restore
 process.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) deleted"
   }
}
```

Aviso de alta contagem de fragmentos

OpenSearch O serviço envia esse evento quando a contagem média de fragmentos em seus nós de dados ativos excede 90% do limite padrão recomendado de 1.000. Embora as versões posteriores

do Elasticsearch OpenSearch suportem um limite máximo configurável de fragmentos por nó, recomendamos que você não tenha mais do que 1.000 fragmentos por nó. Consulte Como escolher o número de fragmentos.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Low",
     "description":"One or more data nodes have close to 1000 shards. To ensure optimum
 performance and stability of your
                    cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
  }
}
```

Limite de contagem de fragmentos excedido

OpenSearch O serviço envia esse evento quando a contagem média de fragmentos em seus nós de dados ativos excede o limite padrão recomendado de 1.000. Embora as versões posteriores do Elasticsearch OpenSearch suportem um limite máximo configurável de fragmentos por nó, recomendamos que você não tenha mais do que 1.000 fragmentos por nó. Consulte Como escolher o número de fragmentos.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
```

```
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes have more than 1000 shards. To ensure
 optimum performance and stability of your
                   cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
  }
}
```

Pouco espaço em disco

OpenSearch O serviço envia esse evento quando um ou mais nós em seu cluster têm menos de 25% do espaço de armazenamento disponível ou menos de 25 GB.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Low Disk Space",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes in your cluster has less than 25% of storage
 space or less than 25GB.
```

```
                    Your cluster will be blocked for writes at 20% or 20GB. Please refer
 to the documentation for more information - https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/handling-errors.html#troubleshooting-cluster-block"
  }
}
```

Violação de marca d'água de baixo disco

OpenSearch O serviço envia esse evento quando todos os nós em seu cluster têm menos de 10% do espaço de armazenamento disponível ou menos de 10 GB. Quando todos os nós violarem a marca d'água de disco embaixo, qualquer novo índice resultará em um cluster amarelo e, quando todos os nós ficarem abaixo da marca d'água do disco no alto, resultará em um cluster vermelho.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Low Disk Watermark Breach",
     "status":"Warning",
     "severity":"Medium",
     "description":"Low Disk Watermark threshold is about to be breached. Once the
 threshold is breached, new index creation will be blocked on all
                    nodes to prevent the cluster status from turning red. Please
 increase disk size to suit your storage needs. For more information,
                    see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#troubleshooting-cluster-block".
  }
}
```

Saldo de intermitência do EBS abaixo de 70%

OpenSearch O serviço envia esse evento quando o saldo de estouro do EBS em um ou mais nós de dados fica abaixo de 70%. O esgotamento do balanceamento intermitente do EBS pode causar indisponibilidade generalizada do cluster e limitação de I/O solicitações, o que pode levar a altas latências e tempos limite nas solicitações de indexação e pesquisa. Para obter as etapas de correção para esse problema, consulte the section called “O saldo de intermitência do EBS está baixo”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"Medium",
     "description":"EBS burst balance on one or more data nodes is below 70%.
                    Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
                    to fix this issue."
  }
}
```

Saldo de intermitência do EBS abaixo de 20%

OpenSearch O serviço envia esse evento quando o saldo de estouro do EBS em um ou mais nós de dados fica abaixo de 20%. O esgotamento do balanceamento intermitente do EBS pode causar indisponibilidade generalizada do cluster e limitação de I/O solicitações, o que pode levar a altas latências e tempos limite nas solicitações de indexação e pesquisa. Para obter as etapas de correção para esse problema, consulte the section called “O saldo de intermitência do EBS está baixo”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"High",
     "description":"EBS burst balance on one or more data nodes is below 20%.
                    Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
                    to fix this issue.
  }
}
```

Controle de utilização do throughput do disco

OpenSearch O serviço envia esse evento quando as solicitações de leitura e gravação para seu domínio estão sendo limitadas devido às limitações de taxa de transferência dos volumes do EBS ou da instância EC2. Se você receber essa notificação, considere escalar seus volumes ou instâncias seguindo as melhores práticas AWS recomendadas. Se o seu tipo de volume for `gp2`, aumente o tamanho do volume. Se o seu tipo de volume for `gp3`, forneça mais throughput. Você também pode verificar se a base da instância e a taxa máxima de throughput do EBS são maiores ou iguais à taxa de throughput do volume provisionado e se pode aumentar a escala verticalmente.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
```

```
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Disk Throughput Throttle",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is experiencing throttling due to instance or volume
 throughput limitations.
                   Please consider scaling your domain to suit your throughput needs.
 In July 2023, we improved
                   the accuracy of throughput throttle calculation by replacing ‘Max
 volume throughput’ with
                   ‘Provisioned volume throughput’. Please refer to the documentation
 for more information."
   }
}
```

Tamanho de fragmento grande

OpenSearch O serviço envia esse evento quando um ou mais fragmentos em seu cluster excedem 50 GiB ou 65 GiB. Para garantir o desempenho e a estabilidade ideais do cluster, reduza o tamanho dos fragmentos.

Para saber mais, consulte práticas recomendadas de fragmentação.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Large Shard Size",
     "status":"Warning",
```

```
     "severity":"Medium",
     "description":"One or more shards are larger than 65GiB. To ensure optimum cluster
 performance and stability, reduce shard sizes.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-large-shard-
size."
  }
}
```

Uso elevado do JVM

OpenSearch O serviço envia esse evento quando a `JVMMemoryPressure` métrica do seu domínio ultrapassa 80%. Se exceder 92% por 30 minutos, todas as operações de gravação em seu cluster serão bloqueadas. Para garantir a estabilidade ideal do cluster, reduza o tráfego para o cluster ou escale seu domínio para fornecer memória suficiente para sua workload.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High JVM Usage",
     "status":"Warning",
     "severity":"High",
     "description":"JVM memory pressure has exceeded 80%. If it exceeds 92% for 30
 minutes, all write operations to your cluster
                    will be blocked. To ensure optimum cluster stability, reduce
 traffic to the cluster or use larger instance types.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-high-jvm."
  }
}
```

GC insuficiente

OpenSearch O serviço envia esse evento quando a JVM máxima está acima de 70% e a diferença entre a máxima e a mínima é menor que 30%. Isso pode indicar que a JVM não consegue recuperar memória suficiente durante os ciclos de coleta de resíduos para sua workload. Isso pode levar a respostas cada vez mais lentas e latências mais altas; e, em alguns casos, até mesmo a queda de nós devido a verificações de integridade expiradas. Para garantir a estabilidade ideal do cluster, reduza o tráfego para o cluster ou escale seu domínio para fornecer memória suficiente para sua workload.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Insufficient GC",
     "status":"Warning",
     "severity":"Medium",
     "description":"Maximum JVM is above 70% and JVM range is less than 30%. This may
 indicate insufficient garbage collection for your workload.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-insufficient-
gc."
  }
}
```

Aviso de roteamento de índice personalizado

OpenSearch O serviço envia esse evento quando seu domínio está em estado de processamento e contém índices com configurações personalizadas de index.routing.allocation que podem fazer com que as implantações azul-esverdeadas parem. Verifique se as configurações foram aplicadas corretamente.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Custom Index Routing Warning",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is in processing state and contains indice(s) with
 custom index.routing.allocation
                    settings which can cause blue-green deployments to get stuck.
 Verify settings are applied properly.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-index-routing."
  }
}
```

Falha no bloqueio de fragmentos

OpenSearch O serviço envia esse evento quando seu domínio não está íntegro devido a fragmentos não atribuídos com. `[ShardLockObtainFailedException]` Para obter mais informações, consulte Como resolvo a exceção de bloqueio por fragmentos na memória no Amazon OpenSearch Service?

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
```

```
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Failed Shard Lock",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is unhealthy due to unassigned shards with
 [ShardLockObtainFailedException]. For more information,
                    see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html#monitoring-events-failed-shard-lock."
}
```

Eventos de endpoint da VPC

OpenSearch O serviço envia determinados eventos EventBridge relacionados aos endpoints AWS PrivateLink da interface.

Falha na criação de endpoint da VPC

OpenSearch O serviço envia esse evento quando não consegue criar um VPC endpoint solicitado. Esse erro pode ocorrer porque você atingiu o limite do número de endpoints da VPC permitido em uma região. Você também verá esse erro se uma sub-rede ou grupo de segurança especificado não existir.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Create Validation",
```

```
      "status":"Failed",
      "severity":"High",
      "description":"Unable to create VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: You've reached the limit on the
                    number of VPC endpoints that you can create in the AWS Region."
   }
}
```

Falha na atualização de endpoint da VPC

OpenSearch O serviço envia esse evento quando não consegue excluir um VPC endpoint solicitado.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Update Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to update VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: <failure message>."
   }
}
```

Falha na exclusão de endpoint da VPC

OpenSearch O serviço envia esse evento quando não consegue excluir um VPC endpoint solicitado.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Delete Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to delete VPC endpoint aos-0d4c74c0342343 for domain
                     arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: Specified subnet doesn't exist."
    }
}
```

Eventos de desativação do nó

OpenSearch O serviço envia eventos para EventBridge quando ocorrer um dos seguintes eventos de desativação do nó.

Desativação do nó agendada

OpenSearch O serviço envia esse evento quando a desativação de um nó é agendada.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
```

```
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Scheduled",
    "severity": "Medium",
    "description": "An automated action to retire and replace a node has been scheduled
 on your domain.
                    The node will be replaced in the next off-peak window. For more
 information, see
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html."
  }
}
```

Desativação do nó concluída

OpenSearch O serviço envia esse evento quando a desativação do nó é concluída.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Completed",
    "severity": "Medium",
    "description": "The node has been retired and replaced with a new node."
  }
}
```

Falha na desativação do nó

OpenSearch O serviço envia esse evento quando a desativação de um nó falha.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Failed",
    "severity": "Medium",
    "description": "Node retirement failed. No actions are required from your end. We
 will automatically
                   retry replacing the node."
  }
}
```

Eventos de retirada do nó degradado

OpenSearch O serviço envia esses eventos quando a substituição de um nó é necessária devido à degradação do hardware em um nó.

Notificação de retirada do nó degradado

OpenSearch O serviço envia esse evento quando a ação automatizada para retirar e substituir um nó degradado é agendada para seu domínio.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"db233454-aad1-7676-3b15-10a84b052baa",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
```

```
   "account":"123456789012",
   "time":"2024-01-11T08:16:06Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"An automated action to retire and replace a node has
 been scheduled on your domain. For more information, please see https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/monitoring-events.html.",
      "event":"Degraded Node Retirement Notification",
      "status":"Scheduled"
   }
}
```

Conclusão da retirada do nó degradado

OpenSearch O serviço envia esse evento quando um nó degradado é retirado e substituído por um novo nó.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"7444215c-90f9-a52d-bcda-e85973a9a762",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T10:20:30Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"The node has been retired and replaced with a new node.",
      "event":"Degraded Node Retirement Notification",
      "status":"Completed"
   }
}
```

Falha na retirada do nó degradado

OpenSearch O serviço envia esse evento se a retirada do nó degradado falhar.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"c328e9bb-93b9-c0b2-b17a-df527fdf96b6",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T08:31:38Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"Node retirement failed. No actions are required from your end. We
 will automatically re-try replacing the node.",
      "event":"Degraded Node Retirement Notification",
      "status":"Failed"
   }
}
```

Eventos de erro de domínio

OpenSearch O serviço envia eventos para EventBridge quando ocorrer um dos seguintes erros de domínio.

Falha na validação da atualização do domínio

OpenSearch O serviço envia esse evento se encontrar uma ou mais falhas de validação ao tentar atualizar ou realizar uma alteração na configuração em um domínio. Para obter ajuda para resolver essas falhas, consulte the section called “Solução de problemas de erros de validação”.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Domain Update Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Domain Update Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to perform updates to your domain due to the following
 validation failures: <failures>
                   Please see the documentation for more information https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/managedomains-
configuration-changes.html#validation"
   }
}
```

Chave do KMS inacessível

OpenSearch O serviço envia esse evento quando não consegue acessar sua AWS KMS chave.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Domain Error Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"KMS Key Inaccessible",
     "status":"Error",
```

```
     "severity":"High",
     "description":"The KMS key associated with this domain is inaccessible. You are at
 risk of losing access to your domain.
                    For more information, please refer to https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Isolamento de domínios

OpenSearch O serviço envia esse evento quando seu domínio fica isolado e não consegue receber, ler ou gravar solicitações porque não pode ser acessado pela rede.

Exemplo

O exemplo a seguir mostra um evento desse tipo.

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2023-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Domain Isolation Notification",
     "status":"Error",
     "severity":"High",
     "description":"Your OpenSearch Service domain has been isolated. An isolated
 domain is unreachable by network and cannot receive, read, or write requests. For more
 information and assistance, please contact AWS Support at https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Tutorial: Ouvindo EventBridge eventos do Amazon OpenSearch Service

Neste tutorial, você configura uma AWS Lambda função simples que escuta os eventos do Amazon OpenSearch Service e os grava em um stream de CloudWatch logs do Logs.

Pré-requisitos

Este tutorial pressupõe que você tenha um domínio de OpenSearch serviço existente. Se você ainda não criou um domínio, siga as etapas em Criação e gerenciamento de domínios para criar um.

Etapa 1: Criar a função do Lambda

Neste procedimento, você cria uma função Lambda simples para servir como destino para mensagens de eventos OpenSearch de serviço.

Para criar uma função Lambda de destino

1. Abra o AWS Lambda console em https://console.aws.amazon.com/lambda/.
2. Escolha Criar função e Criar desde o início.
3. Em Nome da função, insira event-handler.
4. Para Runtime, escolha Python 3.8.
5. Escolha Criar Função.
6. Na seção Function code, edite o código de exemplo de acordo com o exemplo a seguir.

```
import json

def lambda_handler(event, context):
    if event["source"] != "aws.es":
        raise ValueError("Function only supports input from events with a source
 type of: aws.es")

    print(json.dumps(event))
```

   Essa é uma função simples do Python 3.8 que imprime os eventos enviados pelo Service. OpenSearch Se tudo estiver configurado corretamente, no final deste tutorial, os detalhes do evento aparecerão no stream de CloudWatch registros de registros associado a essa função Lambda.
7. Escolha Implantar.

Etapa 2: registrar uma regra de evento

Nesta etapa, você cria uma EventBridge regra que captura eventos dos seus domínios de OpenSearch serviço. Essa regra captura todos os eventos na conta em que ela está definida. As

mensagens de eventos em si contêm informações sobre a fonte do evento, inclusive o domínio do qual ele se originou. Você pode usar essas informações para filtrar e classificar eventos de forma programática.

Para criar uma EventBridge regra

1. Abra o EventBridge console em https://console.aws.amazon.com/events/.
2. Escolha Criar regra.
3. Nomeie a regra como event-rule.
4. Escolha Próximo.
5. Para o padrão do evento, selecione AWS services, Amazon OpenSearch Service e All Events. Esse padrão se aplica a todos os seus domínios de OpenSearch serviço e a todos os eventos OpenSearch de serviço. Como alternativa, você pode criar um padrão mais específico para filtrar alguns resultados.
6. Pressione Próximo.
7. Em Target (Destino), escolha Função do Lambda. No menu suspenso de função, escolha manipulador de eventos.
8. Pressione Próximo.
9. Ignore as tags e pressione Próximo novamente.
10. Revise a configuração e selecione Criar regra.

Etapa 3: Testar sua configuração

Na próxima vez que você receber uma notificação na seção Notificações do console de OpenSearch serviço, se tudo estiver configurado corretamente, sua função Lambda será acionada e gravará os dados do evento em um stream de CloudWatch registros da função.

Para testar sua configuração

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. No painel de navegação, escolha Logs e selecione o grupo de logs para sua função Lambda (por exemplo,aws/lambda/event/-handler).
3. Selecione um fluxo de log para visualizar os dados do evento.

Tutorial: Envio de alertas do Amazon SNS para atualizações de software disponíveis

Neste tutorial, você configura uma regra de EventBridge evento da Amazon que captura notificações de atualizações de software de serviço disponíveis no Amazon OpenSearch Service e envia uma notificação por e-mail por meio do Amazon Simple Notification Service (Amazon SNS).

Pré-requisitos

Este tutorial pressupõe que você tenha um domínio de OpenSearch serviço existente. Se você ainda não criou um domínio, siga as etapas em Criação e gerenciamento de domínios para criar um.

Etapa 1: Criar e se inscrever em um tópico do Amazon SNS

Configure um tópico do Amazon SNS para funcionar como um destino de evento para a nova regra de evento.

Para criar um destino do Amazon SNS

1. Abra o console do Amazon SNS em https://console.aws.amazon.com/sns/ v3/home.
2. Escolha Tópicos e Criar tópico.
3. Para o tipo de trabalho, escolha Padrão e nomeie o trabalho como software-update.
4. Escolha Criar tópico.
5. Após o tópico ser criado, escolha Criar assinatura.
6. Em Protocolo, escolha Email. Em Endpoint, insira um endereço de e-mail ao qual tenha acesso e escolha Criar assinatura.
7. Verifique sua conta de e-mail e espere para receber uma mensagem de e-mail de confirmação de assinatura. Quando você recebê-la, escolha Confirmar assinatura.

Etapa 2: Registrar uma regra de evento

Em seguida, registre uma regra de eventos que captura apenas eventos de atualização de software de serviço.

Para criar uma regra de evento

1. Abra o EventBridge console em https://console.aws.amazon.com/events/.

2. Escolha Criar regra.
3. Nomeie a regra como softwareupdate-rule.
4. Escolha Próximo.
5. Para o padrão do evento, selecione AWS serviços, Amazon OpenSearch Service (cluster gerenciado) e Amazon OpenSearch Service Software Update Notification. Esse padrão corresponde a qualquer evento de atualização de software de OpenSearch serviço do Service. Para obter mais informações sobre padrões de eventos, consulte os padrões de EventBridge eventos da Amazon no Guia EventBridge do usuário da Amazon.
6. Opcionalmente, você pode filtrar apenas por gravidades específicas. Para as gravidades de cada evento, consulte the section called “Eventos de atualização de software de serviço”.
7. Escolha Próximo.
8. Em Target (Destino), escolha Tópico do SNS e selecione software-update.
9. Escolha Próximo.
10. Ignore as tags e selecione Próximo.
11. Revise a configuração da regra e selecione Criar regra.

Na próxima vez que você receber uma notificação do OpenSearch Serviço sobre uma atualização de software de serviço disponível, se tudo estiver configurado corretamente, o Amazon SNS deverá enviar um alerta por e-mail sobre a atualização.

Monitorando chamadas OpenSearch de API do Amazon Service com AWS CloudTrail

O Amazon OpenSearch Service se integra com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no OpenSearch Serviço. CloudTrail captura todas as chamadas de API de configuração para OpenSearch Service as events.

> **Note**
>
> CloudTrail captura apenas chamadas para a API de configuração, como `CreateDomain` e. `GetUpgradeStatus` CloudTrail não captura chamadas para o. OpenSearch APIs, como `_search` `_bulk` e. Para essas chamadas, consulte the section called “Monitoramento de logs de auditoria”.

As chamadas capturadas incluem chamadas do console OpenSearch de serviço ou AWS CLI de um AWS SDK. Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para OpenSearch o Service. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao OpenSearch Serviço, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais CloudTrail, consulte o Guia AWS CloudTrail do usuário.

Informações OpenSearch do Amazon Service em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre no OpenSearch Serviço, essa atividade é registrada em um CloudTrail evento junto com outros eventos do AWS serviço no Histórico de eventos. É possível visualizar, pesquisar e baixar eventos recentes em sua Conta da AWS conta. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em sua Conta da AWS conta, incluindo eventos do OpenSearch Service, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as Regiões da AWS. A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para saber mais, consulte:

- Criando uma trilha para o seu Conta da AWS
- AWS integrações de serviços com CloudTrail o Logs
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações da API de configuração do OpenSearch serviço são registradas CloudTrail e documentadas na Amazon OpenSearch Service API Reference.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais de usuário root ou AWS Identity and Access Management (IAM)
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado
- Se a solicitação foi feita por outro AWS serviço

Para obter mais informações, consulte Elemento userIdentity do CloudTrail .

Entendendo as entradas do arquivo de log do Amazon OpenSearch Service

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreateDomain` operação:


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/test-user",
    "accountId": "123456789012",
    "accessKeyId": "access-key",
    "userName": "test-user",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T21:59:11Z"
      }
    },
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2018-08-21T22:00:05Z",
  "eventSource": "es.amazonaws.com",
  "eventName": "CreateDomain",
```

```
  "awsRegion": "us-west-1",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "engineVersion": "OpenSearch_1.0",
    "clusterConfig": {
      "instanceType": "m4.large.search",
      "instanceCount": 1
    },
    "snapshotOptions": {
      "automatedSnapshotStartHour": 0
    },
    "domainName": "test-domain",
    "encryptionAtRestOptions": {},
    "eBSOptions": {
      "eBSEnabled": true,
      "volumeSize": 10,
      "volumeType": "gp2"
    },
    "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}",
    "advancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    }
  },
  "responseElements": {
    "domainStatus": {
      "created": true,
      "clusterConfig": {
        "zoneAwarenessEnabled": false,
        "instanceType": "m4.large.search",
        "dedicatedMasterEnabled": false,
        "instanceCount": 1
      },
      "cognitoOptions": {
        "enabled": false
      },
      "encryptionAtRestOptions": {
        "enabled": false
      },
      "advancedOptions": {
        "rest.action.multi.allow_explicit_index": "true"
      },
```

```
      "upgradeProcessing": false,
      "snapshotOptions": {
        "automatedSnapshotStartHour": 0
      },
      "eBSOptions": {
        "eBSEnabled": true,
        "volumeSize": 10,
        "volumeType": "gp2"
      },
      "engineVersion": "OpenSearch_1.0",
      "processing": true,
      "aRN": "arn:aws:es:us-west-1:123456789012:domain/test-domain",
      "domainId": "123456789012/test-domain",
      "deleted": false,
      "domainName": "test-domain",
      "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},\"Action\":\"es:*\",
\"Resource\":\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}"
    }
  },
  "requestID": "12345678-1234-1234-1234-987654321098",
  "eventID": "87654321-4321-4321-4321-987654321098",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Monitoramento operacional unificado com o Cluster Insights

O Amazon OpenSearch Service agora inclui o Cluster Insights, uma solução de monitoramento que fornece visibilidade operacional abrangente de seus clusters por meio de um único painel. Isso elimina a complexidade de ter que analisar e correlacionar vários registros e métricas para identificar riscos potenciais à disponibilidade ou ao desempenho do cluster. A solução automatiza a consolidação de dados operacionais críticos entre nós, índices e fragmentos, transformando a solução de problemas complexa em um processo simplificado. Você pode detectar problemas como fragmentos grandes e marcas d'água com pouco disco, visualizar métricas detalhadas nos níveis de nó, índice e fragmento e acessar as melhores práticas de segurança e resiliência.

> **Note**
>
> O Cluster Insights está disponível por meio da interface de usuário do OpenSearch serviço sem custo adicional para todos os usuários que executam a OpenSearch versão 2.17 ou posterior. Além disso, os domínios com OpenSearch as versões 2.17 e 2.19 também precisam estar na atualização mais recente da versão do software de serviço.

Benefícios

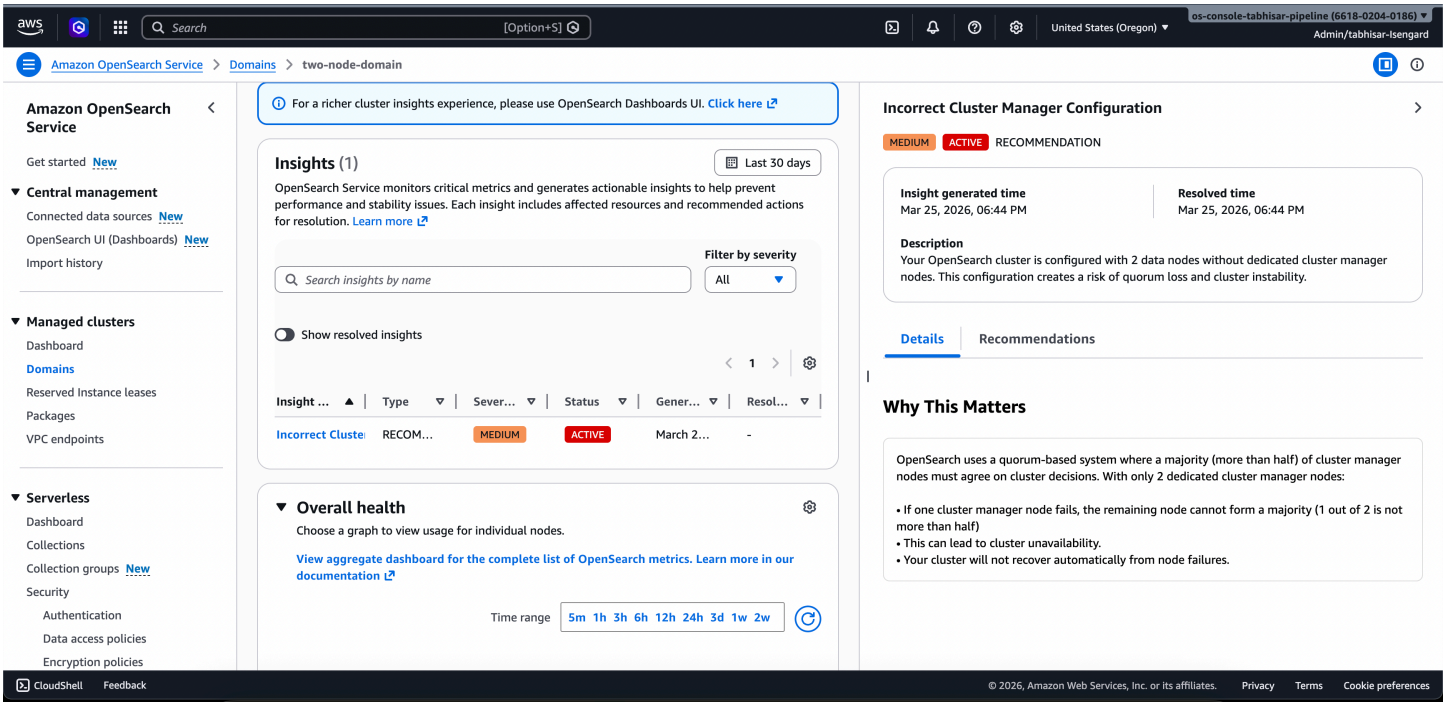
- Monitoramento proativo - monitore a integridade do cluster de forma proativa com métricas de desempenho detalhadas em todos os componentes, desde nós e índices individuais até fragmentos e consultas de pesquisa.
- Visibilidade unificada - consolide os dados de monitoramento em um único painel
- Recomendações acionáveis — Obtenha step-by-step orientação para a resolução de problemas
- Cobertura abrangente — monitore a segurança, a estabilidade e a resiliência em seus OpenSearch clusters
- Otimização de consultas - identifique consultas que consomem muitos recursos e otimize o desempenho

Com o Cluster Insights, você pode manter o desempenho ideal do cluster, reduzir a sobrecarga operacional e garantir as melhores práticas consistentes em seus OpenSearch clusters

Acesse o Cluster Insights por meio do console

Analise as recomendações de desempenho e resiliência e faça as alterações de configuração necessárias, tudo no mesmo console. No console, na guia Integridade do cluster, você pode acessar o Cluster Insights que lista todos os Insights ativos. Clique em qualquer Insight para ver as recomendações.

Tela 1: Cluster Insights na guia Cluster Health



Acesse o Cluster Insights e métricas detalhadas por meio da OpenSearch interface

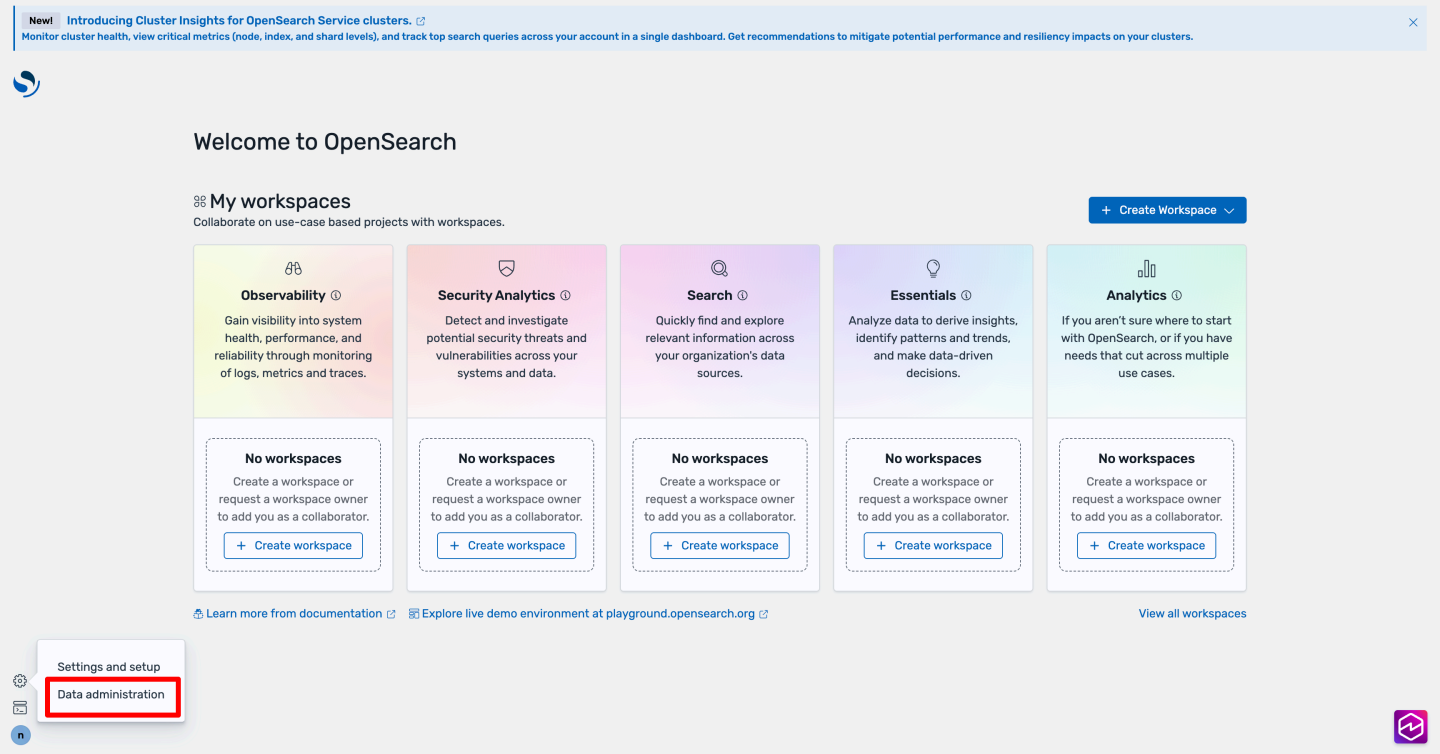
Você pode visualizar insights de um cluster de OpenSearch serviços específico por meio da OpenSearch interface do usuário (painéis). Na OpenSearch interface do usuário, um aplicativo é simplesmente uma construção organizacional, como uma pasta. Cada aplicativo pode se conectar e exibir insights para vários clusters OpenSearch de serviços. O acesso ao Cluster Insights requer uma função administrativa no aplicativo de OpenSearch interface do usuário.

> **Note**
>
> O acesso ao Cluster Insights requer uma função administrativa no aplicativo de OpenSearch interface do usuário.

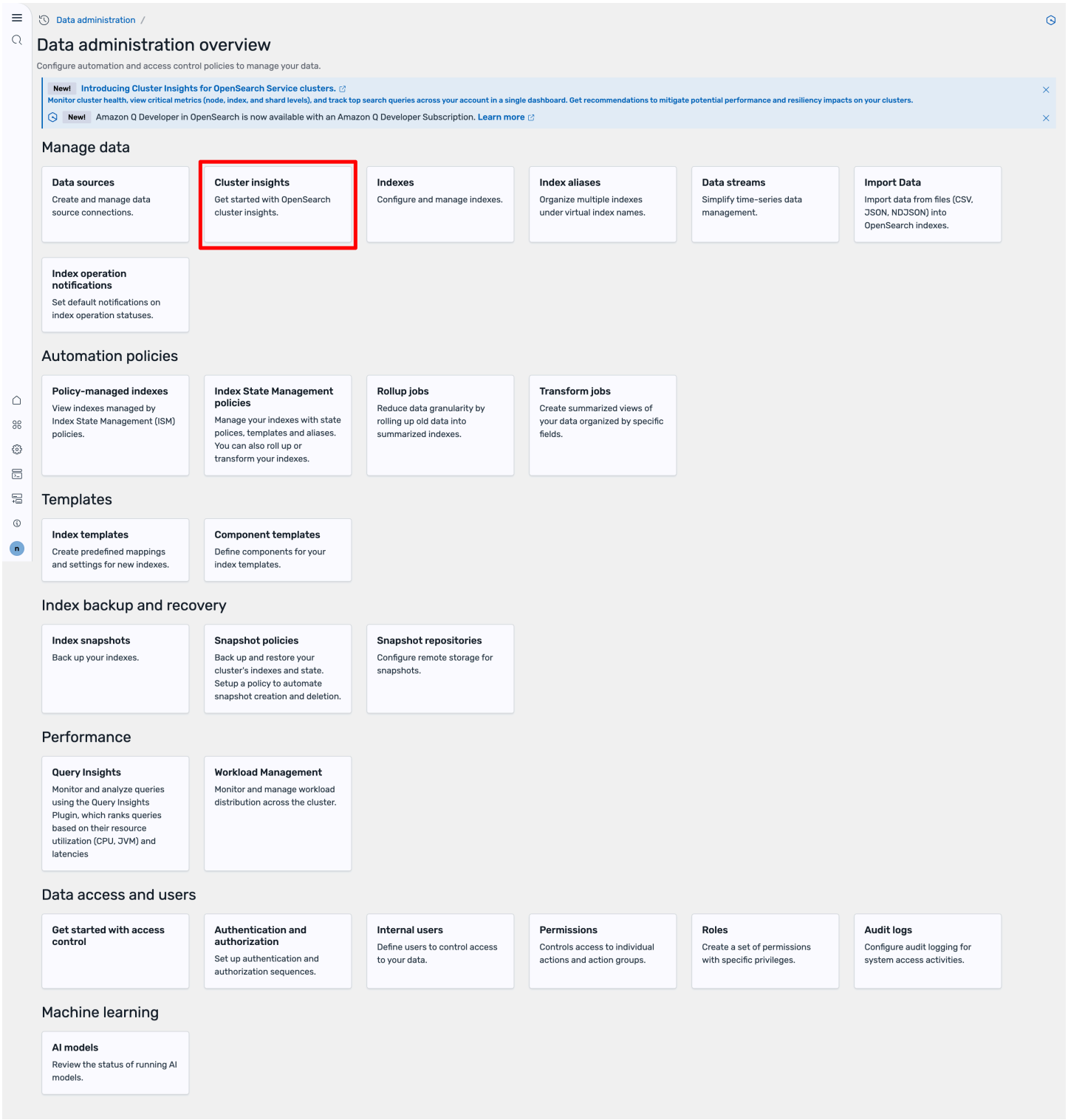
Crie e configure um aplicativo para visualizar o Cluster Insights

1. Abra o console OpenSearch de serviço em https://console.aws.amazon.com/aos/home
2. Escolha OpenSearch UI (painéis) na navegação à esquerda
3. Conclua as etapas a seguir para criar e configurar um aplicativo:
    1. Criar um aplicativo OpenSearch de serviço
    2. Associar fontes de dados
4. Depois de concluir as duas etapas acima, você pode visualizar o Cluster Insights no painel de OpenSearch interface do usuário na seção Configurações > Administrador de dados > Cluster Insights. O ícone Configurações está localizado na parte inferior esquerda da tela da OpenSearch interface do usuário.

Tela 2: Acesse o administrador de dados a partir da interface OpenSearch



Tela 3: Cluster Insights na seção Gerenciar dados



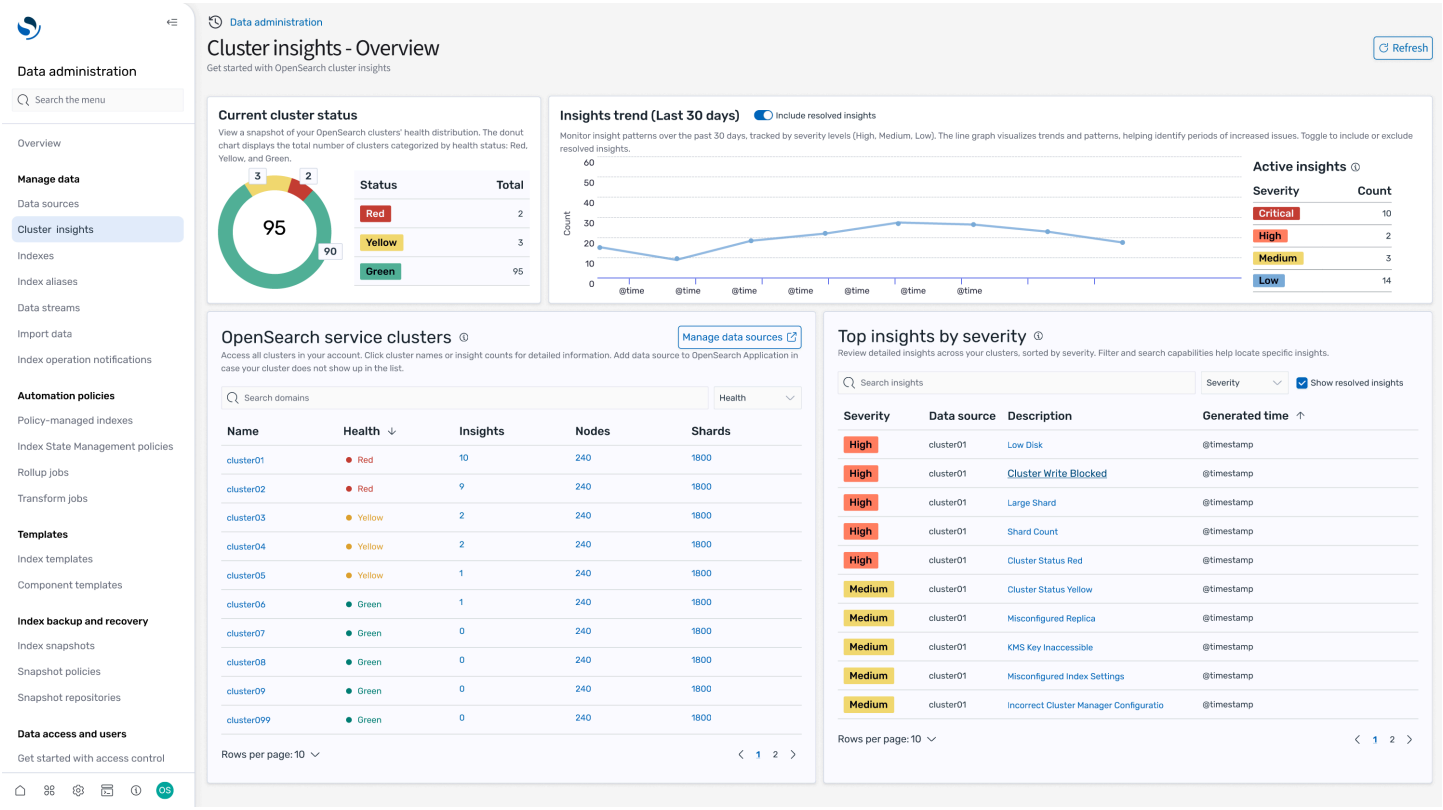
Compreendendo o Cluster Insights

Esta seção descreve os vários insights disponíveis no Cluster Insights.

Painel de visão geral

A página Visão geral do Cluster Insights, conforme mostrado na captura de tela a seguir, fornece uma visão de alto nível da integridade do seu cluster no nível do aplicativo e compreende as seguintes seções:

Tela 4: página inicial do Cluster Insights no aplicativo de OpenSearch interface do usuário.



Status atual do cluster

Um gráfico de rosca exibe o status de integridade do seu cluster:

- Verde - Todos os fragmentos e réplicas primários são alocados aos nós
- Amarelo - Todos os fragmentos primários são alocados, mas algumas réplicas não
- Vermelho - Pelo menos um fragmento primário não está alocado em nenhum nó

Tendência de insights

O gráfico de tendências acompanha os padrões de problemas nos últimos 30 dias, ajudando você a identificar problemas emergentes e monitorar o progresso da resolução.

Insights abertos atuais

Uma contagem organizada por severidade dos insights abertos nos últimos 30 dias.

OpenSearch Clusters de serviços

Esta seção lista todos os seus OpenSearch clusters com estatísticas importantes, incluindo contagem de nós, contagem de fragmentos e consultas ativas.

Principais insights por severidade

Você pode revisar os insights em todos os domínios do seu aplicativo. Esta seção prioriza os problemas que precisam de atenção imediata (Críticos e de Alta Gravidade). Cada insight inclui uma descrição e recomendações específicas, que podem ajudar você a se concentrar primeiro em questões críticas.

Detalhes dos insights

Cada insight na seção Principais insights por gravidade é interativo e fornece uma análise detalhada. Por exemplo, quando você escolhe o insight Large Shard Size:

1. Você vê quantos fragmentos excedem o limite e quais índices são afetados.
2. Um mapa de recursos identifica cada fragmento superdimensionado com seu índice, ID e tamanho atual.
3. A guia de recomendações fornece orientação sobre step-by-step remediação.
4. A guia Histórico exibe um cronograma das ações de remediação de recursos.

Detalhes do cluster

Quando você seleciona um cluster específico na seção OpenSearch Service Clusters, OpenSearch exibe insights desse cluster nas seguintes guias: Integridade do cluster, visualização de nós, visualização de índice, visualização de fragmentos e visualização de consulta. A guia Integridade do cluster exibe as seguintes informações:

Visão geral do

As principais informações incluem integridade do cluster, contagem de fragmentos, contagem de nós, contagem de índices e estatísticas de documentos.

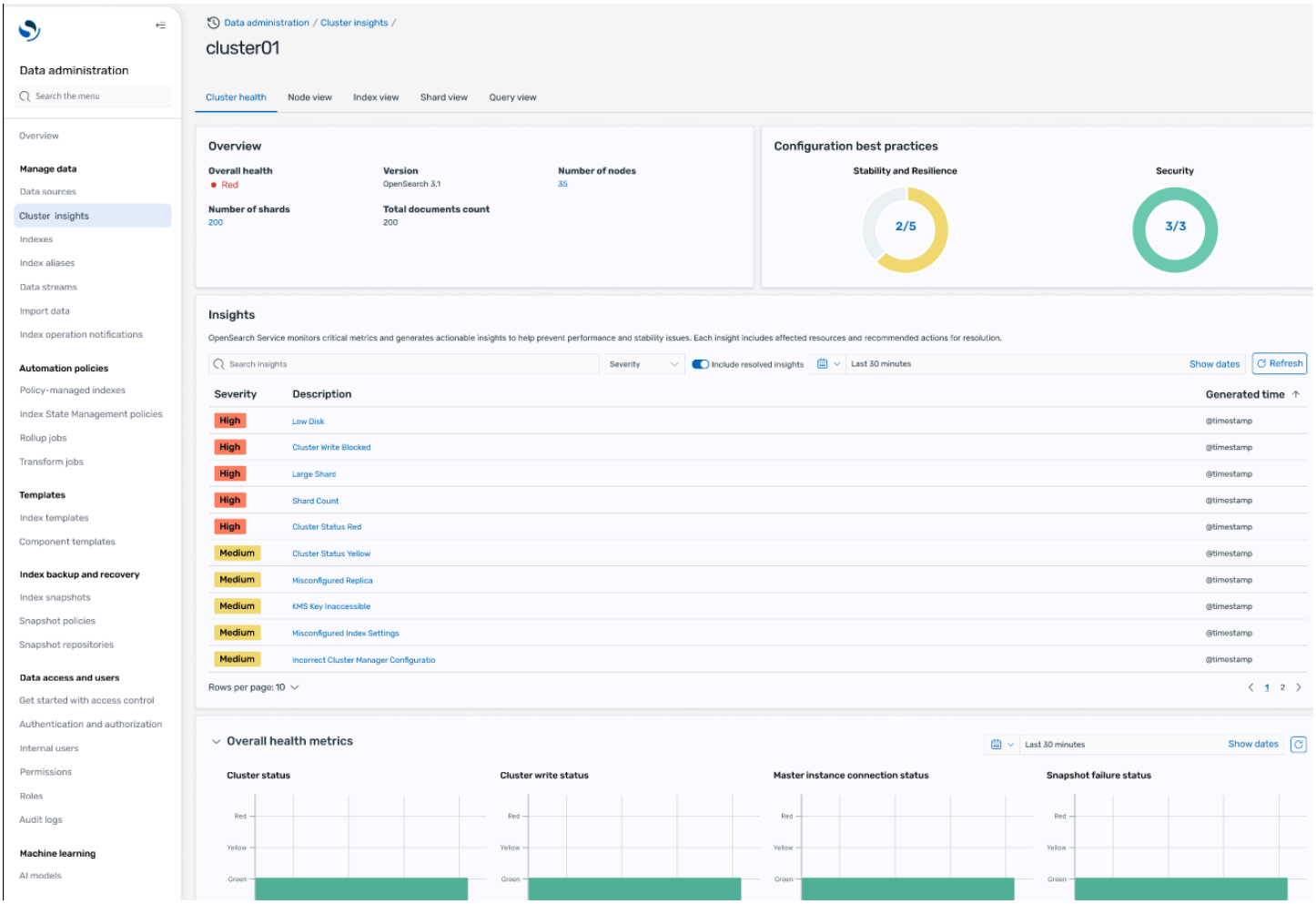
Práticas recomendadas para configuração

Os gráficos de rosca mostram a conformidade com as configurações recomendadas de resiliência e segurança.

Insights

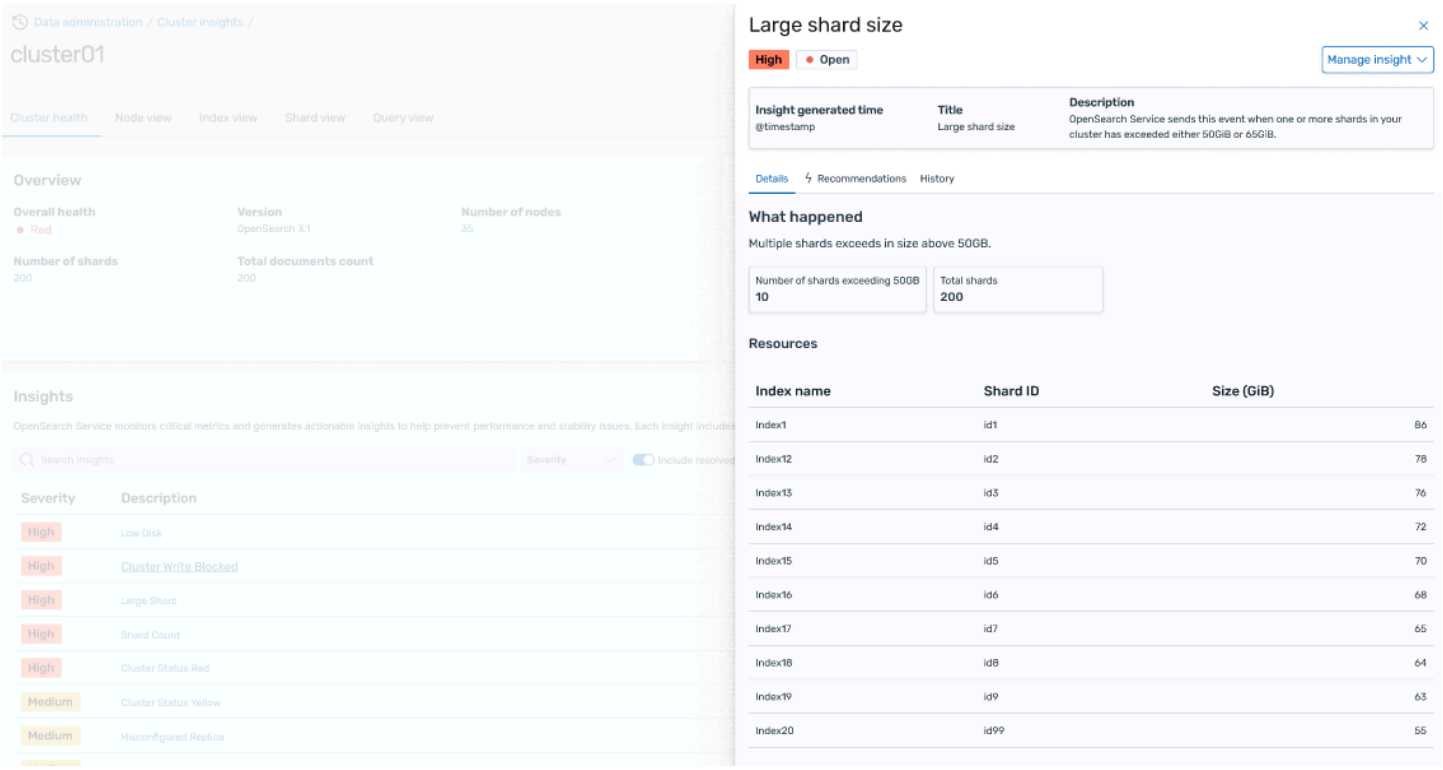
Uma tabela lista os insights recentes gerados para o cluster, com as mesmas orientações detalhadas de detalhamento e remediação disponíveis na página de visão geral.

Tela 5: A visão geral do Cluster Health fornece as principais métricas, melhores práticas e insights



Ao clicar em qualquer informação, você pode ver detalhes, recursos e recomendações impactados. Além disso, você também pode ver o histórico dos recursos fixos.

Tela 6: detalhes do Insight. Fornece detalhes, recomendações e cronograma histórico.



Seção de métricas

Os gráficos interativos desta seção exibem as seguintes métricas de cluster:

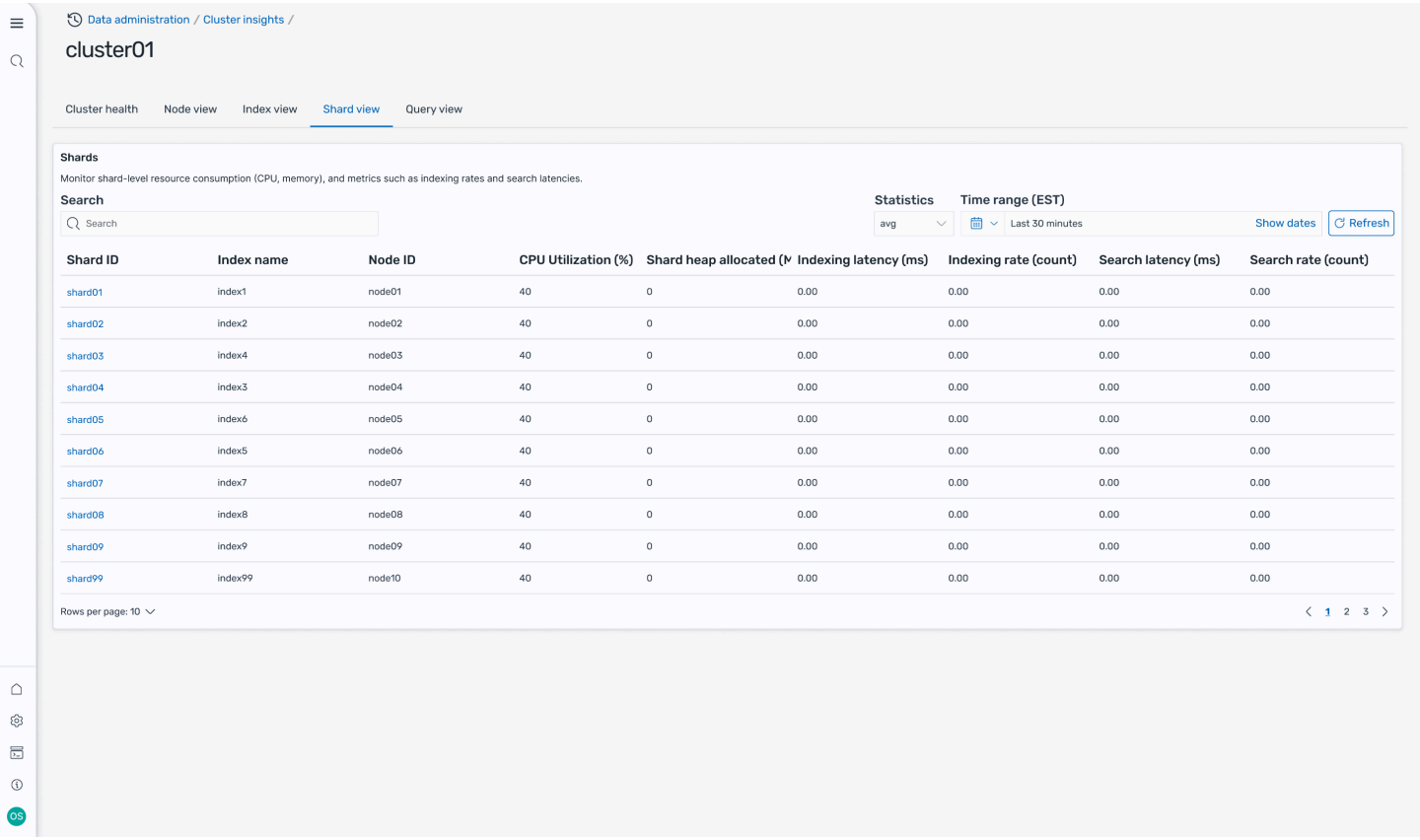
- Métricas gerais de integridade do cluster, como status do cluster, status de gravação e documentos pesquisáveis
- KPIs (Indicadores-chave de desempenho), como taxas e latências de indexação e pesquisa
- Métricas de utilização de recursos, como JVM e utilização de CPU

Visualizações Node, Index e Shard

As visualizações Node, Index e Shard usam OpenSearch estatísticas para fornecer visibilidade detalhada das operações do cluster. Você pode ver:

- Métricas em tempo real, como utilização da CPU e pressão da memória JVM
- Pesquisa e indexação de dados de desempenho
- Pontos de acesso de recursos em todos os componentes do cluster
- Diagnóstico granular em nível de nó
- Maior pilha de fragmentos alocada

Tela 7: métricas de nível de Node, Index e Shard



Visualização da consulta

> **Note**
>
> O recurso Query View é compatível com as OpenSearch versões 2.19 ou posteriores.

A página Visualização da consulta ajuda você a monitorar consultas que consomem muitos recursos com:

Configuração de acesso

A visualização das N principais consultas requer permissões de controle de acesso refinadas. Verifique o seguinte:

- O controle de acesso refinado está ativado em seu domínio.
- Sua função do IAM (ou usuário interno) é mapeada para uma OpenSearch função com as permissões de cluster necessárias para insights de consulta.

- Para obter acesso total de administrador, mapeie o ARN da sua função do IAM como uma função de back-end para as funções all_access e security_manager. Você pode fazer isso em OpenSearch Painéis em Segurança > Funções > selecionar a função > Usuários mapeados > Gerenciar mapeamento ou usando a API de segurança (PUT _plugins/_ security/api/rolesmapping/ all _access).

Sem mapeamentos de função adequados, os usuários podem receber erros 403 Proibidos ao tentar acessar os dados do Query Insights. Para obter detalhes, consulte Controle de acesso refinado.

Painéis ao vivo

Veja as estatísticas de execução, o uso da CPU e da memória e o progresso da conclusão de cada consulta.

Principais N consultas

Uma tabela classificada mostra as consultas mais significativas com detalhes que incluem:

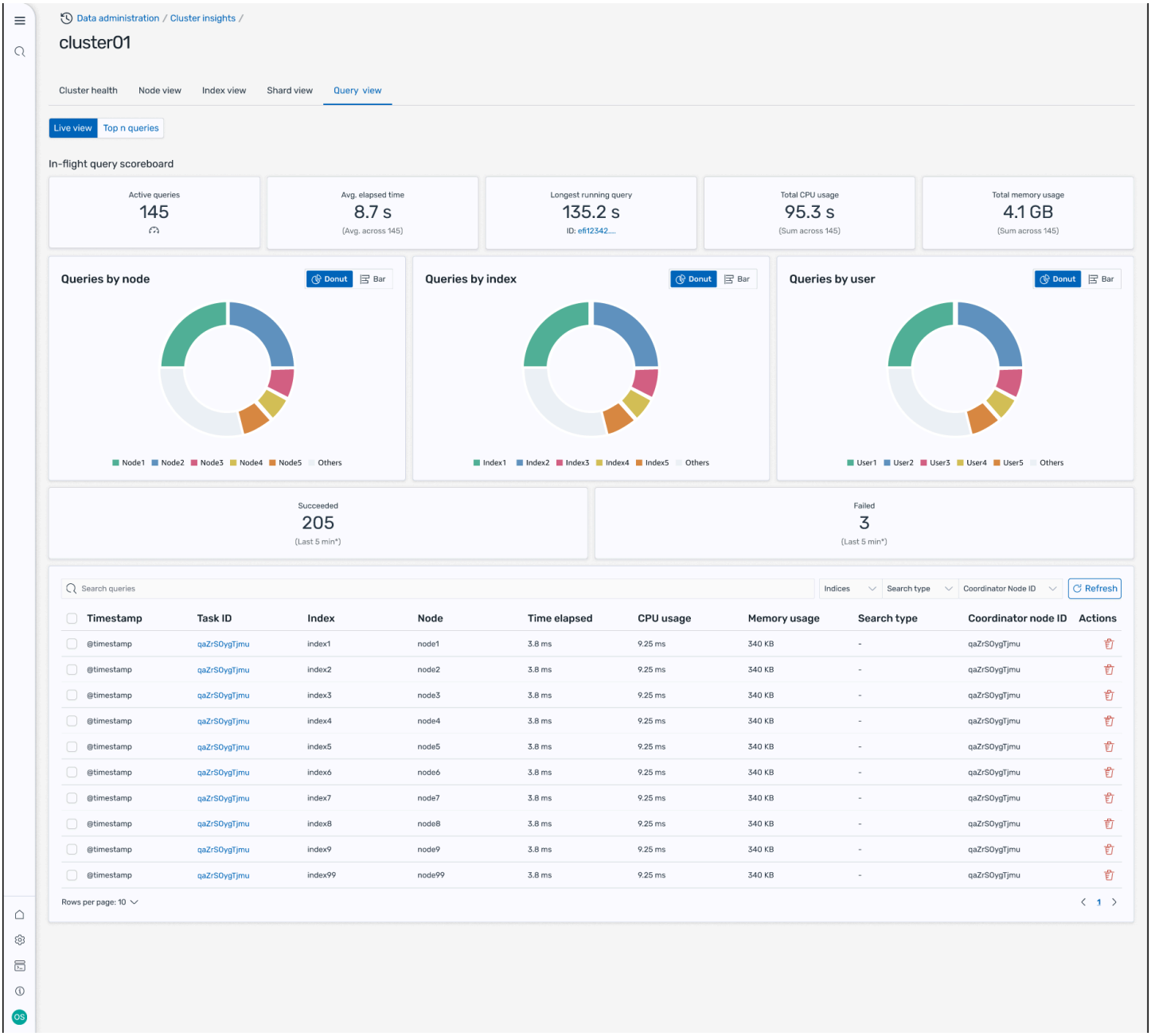
- Contagem de consultas
- Latência, CPU e uso de memória
- Tipo de pesquisa e nó coordenador
- Índices alvo e contagem de fragmentos

Detalhes da consulta

Clique duas vezes em qualquer consulta para ver:

- Carga útil exata da consulta e etapas de execução
- Detalhamento da latência para cada fase (expandir, consultar, buscar)
- Recomendações de otimização

Tela 8: visualização ao vivo durante o voo. Você também pode ver as consultas Top-N



Acesse Insights por meio de EventBridge eventos da Amazon

Você pode monitorar insights por meio de EventBridge eventos da Amazon. Para obter detalhes adicionais, verifique as notificações.

Catálogo de insights

O Cluster Insights fornece os seguintes insights acionáveis sobre clusters afetados

Insights	Description
Disco baixo	Uma visão de alta severidade é gerada quando o cluster FreeStorage cai abaixo de 25% da capacidade total ou 25 GB, qualquer que seja o limite atingido primeiro.
Gravação de cluster bloqueada	Uma visão crítica da gravidade é gerada quando o cluster fica abaixo FreeStorage de 20% da capacidade total ou 20 GB, qualquer que seja o limite atingido primeiro.
Fragmento grande	Um insight com baixa severidade é gerado quando o tamanho do fragmento está entre 50 GB e 100 GB. A severidade aumenta para Média quando o tamanho do fragmento está entre 100 GB e 200 GB, e aumenta ainda mais para Alta quando o tamanho do fragmento excede 200 GB.
Contagem de fragmentos	Um insight de alta severidade é gerado quando o cluster excede 25 fragmentos por GB de heap configurado ou quando o número de fragmentos por nó atinge ou excede 900. Para a OpenSearch versão 2.17 e superior, esse insight será gerado com base na configuração do gerenciador de cluster, conforme detalhado nesta documentação.
Status do cluster vermelho	Uma visão crítica da gravidade é gerada quando a substituição de um nó degradado no OpenSearch cluster resulta em um status de cluster vermelho. Isso requer uma ação imediata para restaurar o cluster para um estado verde.
Status do cluster: amarelo	Um insight de severidade média é gerado quando a substituição de um nó degradado no OpenSearch cluster resulta em um status de cluster amarelo.
Réplica mal configurada	Um insight de severidade média é gerado quando o cluster não tem réplicas configuradas ou quando o número de réplicas excede o número total de nós de dados.
Chave KMS inacessível	Uma visão crítica da gravidade é gerada quando o OpenSearch cluster perde o acesso à chave KMS usada para criptografia

Insights	Description
	em repouso. Isso impede que o cluster acesse dados criptografados e afeta as operações do cluster, exigindo ação imediata para restaurar o acesso.
Configurações de índice mal configuradas	Uma visão de severidade média é gerada quando a configuração do índice é “index.routing.allocation”. require/include/exclude. **<**attribute**>**"existe em qualquer índice no OpenSearch cluster. Essa configuração pode impedir implantações azul-esverdeadas bem-sucedidas durante as atualizações do cluster.
Configuração incorreta do Cluster Manager	Uma visão crítica da gravidade é gerada quando o OpenSearch cluster tem 2 nós de dados sem nós dedicados do gerenciador de cluster ou 2 nós dedicados do gerenciador de cluster. Essas configurações criam um risco de perda de quorum e instabilidade do cluster.
Estratégia de fragmentação abaixo do ideal	(a) Uma visão de severidade média é gerada indicando que um ou mais índices têm contagens de fragmentos inferiores a 80% de seus nós de dados. Isso pode resultar em uma distribuição de carga de trabalho abaixo do ideal em todo o cluster. (b) Uma visão de severidade média é gerada indicando que um ou mais índices têm fragmentos com 20% mais dados do que o tamanho médio de fragmentos por nó. Isso pode resultar em uma distribuição de carga de trabalho abaixo do ideal em todo o cluster.

Insights	Description
Sobrecarga do cluster	(a) Uma visão de alta gravidade da sobrecarga do cluster indica que mais de 70% dos nós de dados no cluster estão experimen tando uma utilização elevada na CPU, memória, E/S de disco, taxa de transferência de disco ou utilização do disco. Isso indica que o cluster pode estar se aproximando dos limites de capacidade. (b) Uma visão de gravidade crítica indica que mais de 70% dos nós de dados estão experimentando alta utilização de recursos em CPU, memória, E/S de disco, taxa de transferência de disco ou utilização de disco, e observa-se limitação ou rejeição de solicitaç ões. Isso requer ação imediata para escalar o cluster.

Segurança no Amazon OpenSearch Service

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isto como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos programas de conformidade da AWS. Para saber mais sobre os programas de conformidade que se aplicam ao Amazon OpenSearch Service, consulte AWS Services in Scope by Compliance Program.
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o OpenSearch Serviço. Os tópicos a seguir mostram como configurar o OpenSearch Serviço para atender aos seus objetivos de segurança e conformidade. Você também aprende a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos do OpenSearch Serviço.

Tópicos

- Proteção de dados no Amazon OpenSearch Service
- Identity and Access Management no Amazon OpenSearch Service
- Prevenção contra o ataque do “substituto confuso” em todos os serviços
- Controle de acesso refinado no Amazon Service OpenSearch
- Validação de conformidade para o Amazon OpenSearch Service
- Resiliência no Amazon Service OpenSearch
- Autenticação e autorização JWT para Amazon Service OpenSearch
- Segurança de infraestrutura no Amazon OpenSearch Service
- Autenticação SAML para painéis OpenSearch

- Suporte confiável de propagação de identidade do IAM Identity Center para OpenSearch
- Configurando a autenticação do Amazon Cognito para painéis OpenSearch
- Usando funções vinculadas a serviços para o Amazon Service OpenSearch

Proteção de dados no Amazon OpenSearch Service

O modelo de responsabilidade AWS compartilhada de se aplica à proteção de dados no Amazon OpenSearch Service. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para saber mais sobre a privacidade de dados, consulte as Data Privacy FAQ. Para saber mais sobre a proteção de dados na Europa, consulte a postagem do blog AWS Shared Responsibility Model and RGPD no Blog de segurança da AWS .

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com Centro de Identidade do AWS IAM ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com AWS os recursos. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar AWS atividades, consulte Como trabalhar com CloudTrail trilhas no Guia AWS CloudTrail do usuário.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sensíveis armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para saber mais sobre os endpoints FIPS disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sensíveis, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o OpenSearch Serviço ou outro Serviços da AWS usando o console, a API ou AWS SDKs. AWS CLI Quaisquer dados inseridos em tags ou em campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, é fortemente recomendável que não sejam incluídas informações de credenciais no URL para validar a solicitação nesse servidor.

Criptografia de dados em repouso para o Amazon OpenSearch Service

OpenSearch Os domínios de serviço oferecem criptografia de dados em repouso, um recurso de segurança que ajuda a impedir o acesso não autorizado aos seus dados. O recurso usa AWS Key Management Service (AWS KMS) para armazenar e gerenciar suas chaves de criptografia e o algoritmo Advanced Encryption Standard com chaves de 256 bits (AES-256) para realizar a criptografia. Se habilitado, o recurso criptografa os seguintes aspectos de um domínio:

- Todos os índices (incluindo aqueles em UltraWarm armazenamento)
- OpenSearch troncos
- Arquivos de troca
- Todos os outros dados no diretório da aplicação
- Snapshots automatizados

Os seguintes itens não são criptografados quando você ativa a criptografia de dados em repouso, mas você pode executar etapas adicionais para protegê-los:

- Instantâneos manuais: no momento, você não pode usar AWS KMS chaves para criptografar instantâneos manuais. No entanto, você pode usar a criptografia no lado do servidor com chaves gerenciadas pelo S3 ou chaves do KMS para criptografar o bucket que você usa como repositório de snapshots. Para instruções, consulte the section called “Registro de um repositório de snapshots manuais”.
- Registros lentos e registros de erros: se você publicar registros e quiser criptografá-los, poderá criptografar o grupo de CloudWatch registros de registros usando a mesma AWS KMS chave do domínio de OpenSearch serviço. Para obter mais informações, consulte Criptografar dados de log em CloudWatch registros usando AWS Key Management Service o Guia do usuário do Amazon CloudWatch Logs.

> **Note**
>
> Você não pode habilitar a criptografia em repouso em um domínio existente se UltraWarm o armazenamento a frio estiver habilitado no domínio. Primeiro, você deve desativar UltraWarm o armazenamento a frio, ativar a criptografia em repouso e, em seguida, reativá-lo UltraWarm ou reativá-lo. Se você quiser manter os índices em UltraWarm um armazenamento refrigerado, você deve movê-los para o armazenamento a quente antes de desativar UltraWarm ou armazenar a frio.

OpenSearch O serviço suporta somente chaves KMS de criptografia simétrica, não chaves assimétricas. Para saber como criar chaves simétricas, consulte Criar uma chave do KMS no Guia do usuário do AWS Key Management Service .

Independentemente de a criptografia em repouso estar ativada, todos os domínios criptografam automaticamente pacotes personalizados usando AES-256 e chaves gerenciadas por serviços. OpenSearch

Permissões

Para usar o console OpenSearch de serviço para configurar a criptografia de dados em repouso, você deve ter permissões de leitura AWS KMS, como a seguinte política baseada em identidade:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:List*",
        "kms:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```

Se quiser usar uma chave diferente da chave AWS própria, você também deve ter permissões para criar concessões para a chave. Essas permissões normalmente assumem a forma de uma política baseada em recursos que você especifica ao criar a chave.

Se você quiser manter sua chave exclusiva para o OpenSearch Serviço, você pode adicionar a ViaService condição kms: a essa política de chaves:

```
"Condition": {
  "StringEquals": {
    "kms:ViaService": "es.us-west-1.amazonaws.com"
  },
  "Bool": {
    "kms:GrantIsForAWSResource": "true"
  }
}
```

Para saber mais, consulte Políticas de chaves no AWS KMS no Guia do desenvolvedor do AWS Key Management Service .

Ativação da criptografia de dados em repouso

A criptografia de dados em repouso em novos domínios requer o Elasticsearch 5.1 OpenSearch ou posterior. Habilitá-lo em domínios existentes requer o Elasticsearch 6.7 OpenSearch ou posterior.

Para habilitar a criptografia de dados em repouso (console)

1. Abra o domínio no AWS console e escolha Ações e Editar configuração de segurança.
2. Em Criptografia, selecione Habilitar criptografia de dados em repouso.
3. Escolha uma AWS KMS chave para usar e, em seguida, escolha Salvar alterações.

Também é possível habilitar a criptografia por meio da API de configuração. A solicitação a seguir permite a criptografia de dados em repouso em um domínio existente:

```
{
   "ClusterConfig":{
      "EncryptionAtRestOptions":{
         "Enabled": true,
         "KmsKeyId":"arn:aws:kms:us-east-1:123456789012:alias/my-key"
      }
   }
```

```
}
```

A chave do KMS foi desabilitada ou excluída

Se você desabilitar ou excluir a chave usada para criptografar um domínio, o domínio ficará inacessível. OpenSearch O serviço envia uma notificação informando que não consegue acessar a chave KMS. Habilite novamente a chave imediatamente para acessar o seu domínio.

A equipe OpenSearch de serviço não poderá ajudá-lo a recuperar seus dados se sua chave for excluída. AWS KMS exclui as chaves somente após um período de espera de pelo menos sete dias. Se a exclusão da sua chave estiver pendente, cancele-a ou tire um snapshot manual do domínio para evitar a perda de dados.

Desativação da criptografia de dados em repouso

Depois de configurar um domínio para criptografar dados em repouso, você não pode desativar a configuração. Em vez disso, você pode tirar um snapshot manual do domínio existente, criar outro domínio, migrar seus dados e excluir o domínio anterior.

Monitoramento de domínios que criptografam dados em repouso

Domínios que criptografam dados em repouso têm duas métricas adicionais: `KMSKeyError` e `KMSKeyInaccessible`. Essas métricas serão exibidas somente se o domínio encontrar um problema com sua chave de criptografia. Para obter descrições completas dessas métricas, consulte the section called “Métricas de cluster”. Você pode visualizá-los usando o console do OpenSearch Service ou o CloudWatch console da Amazon.

> **Tip**
>
> Cada métrica representa um problema significativo para um domínio, por isso recomendamos que você crie CloudWatch alarmes para ambos. Para obter mais informações, consulte the section called “ CloudWatch Alarmes recomendados”.

Outras considerações

- A rotação automática de chaves preserva as propriedades de suas AWS KMS chaves, portanto, a rotação não afeta sua capacidade de acessar seus OpenSearch dados. Os domínios OpenSearch de serviço criptografados não oferecem suporte à rotação manual de chaves, o que envolve a

criação de uma nova chave e a atualização de qualquer referência à chave antiga. Para saber mais, consulte Girar AWS KMS chaves no Guia do AWS Key Management Service desenvolvedor.

- Certos tipos de instâncias não oferecem suporte à criptografia de dados em repouso. Para obter detalhes, consulte the section called “Tipos de instâncias compatíveis”.
- Domínios que criptografam dados em repouso usam outro nome de repositório para seus snapshots automatizados. Para saber mais, consulte the section called “Restauração de snapshots”.
- Embora seja altamente recomendável habilitar a criptografia em repouso, esse recurso pode adicionar sobrecarga adicional à CPU e acrescentar alguns milissegundos de latência. Contudo, a maioria dos casos de uso não é afetada por essas diferenças, e a magnitude do impacto depende da configuração do cluster, dos clientes e do perfil de uso.

Node-to-node criptografia para Amazon OpenSearch Service

Node-to-node a criptografia fornece uma camada adicional de segurança além dos recursos padrão do Amazon OpenSearch Service.

Cada domínio OpenSearch de serviço, independentemente de usar o acesso à VPC, reside em sua própria VPC dedicada. Essa arquitetura impede que possíveis invasores interceptem o tráfego entre os OpenSearch nós e mantém o cluster seguro. Por padrão, no entanto, o tráfego dentro da VPC não é criptografado. Node-to-nodea criptografia habilita a criptografia TLS 1.2 para todas as comunicações dentro da VPC.

Se você enviar dados para o OpenSearch Serviço via HTTPS, a node-to-node criptografia ajuda a garantir que seus dados permaneçam criptografados enquanto OpenSearch os distribuem (e redistribuem) por todo o cluster. Se os dados chegarem sem criptografia via HTTP, o OpenSearch Service os criptografará depois que chegarem ao cluster. Você pode exigir que todo o tráfego para o domínio chegue por HTTPS usando o console ou a API de configuração. AWS CLI

Node-to-node a criptografia é necessária se você ativar o controle de acesso refinado.

Ativando a node-to-node criptografia

Node-to-node a criptografia em novos domínios requer qualquer versão do OpenSearch Elasticsearch 6.0 ou posterior. Habilitar a node-to-node criptografia em domínios existentes requer qualquer versão do OpenSearch Elasticsearch 6.7 ou posterior. Escolha o domínio existente no console do AWS , Ações e Editar configuração de segurança.

Como alternativa, você pode usar a API de configuração AWS CLI ou. Para obter mais informações, consulte a Referência de AWS CLI comandos e a referência da API de OpenSearch serviços.

Desativando a criptografia node-to-node

Depois de configurar um domínio para usar node-to-node criptografia, você não pode desativar a configuração. Em vez disso, você pode tirar um snapshot manual do domínio criptografado, criar outro domínio, migrar seus dados e excluir o domínio anterior.

Identity and Access Management no Amazon OpenSearch Service

O Amazon OpenSearch Service oferece várias maneiras de controlar o acesso aos seus domínios. Esta seção aborda os diversos tipos de políticas, como elass interagem entre si e como você pode criar suas próprias políticas personalizadas.

> **Important**
>
> O suporte à VPC introduz algumas considerações adicionais sobre o controle de acesso ao OpenSearch serviço. Para obter mais informações, consulte the section called “Sobre políticas de acesso em domínios da VPC”.

Tipos de políticas

OpenSearch O serviço oferece suporte a três tipos de políticas de acesso:

- the section called “Políticas baseadas em recursos”
- the section called “Políticas baseadas em identidade”
- the section called “Políticas baseadas em IP”

Políticas baseadas em recursos

Quando um domínio é criado, uma política baseada em recurso é adicionada, muitas vezes chamada de política de acesso ao domínio. Essas políticas especificam que ações uma entidade principal pode executar nos sub-recursos do domínio (com exceção da pesquisa entre clusters). Os sub-recursos incluem OpenSearch índices e. APIs O elemento de política JSON Entidade principal no

IAM especifica as contas, os usuários ou os perfis que têm acesso permitido. O elemento de política JSON Recurso especifica quais sub-recursos essas entidades principais podem acessar.

Por exemplo, a política baseada em recurso a seguir concede ao `test-user` (`es:*`) acesso total aos sub-recursos em `test-domain`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:*"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
}
```

Duas considerações importantes se aplicam a essa política:

- Esses privilégios se aplicam apenas a esse domínio. A menos que você crie políticas semelhantes em outros domínios, `test-user` só poderá acessar `test-domain`.
- O terminador `/*` no elemento `Resource` é significativo e indica que as políticas baseadas em recursos só se aplicam aos sub-recursos do domínio, e não ao próprio domínio. Em políticas baseadas em recursos, a ação `es:*` é equivalente a `es:ESHttp*`.

  Por exemplo, o `test-user` pode fazer solicitações em relação a um índice (`GET https://search-test-domain.us-west-1.es.amazonaws.com/test-index`), mas não pode atualizar a configuração do domínio (`POST https://es.us-west-1.amazonaws.com/2021-01-01/opensearch/domain/test-domain/config`).

Observe a diferença entre os dois endpoints. O acesso à API de configuração requer uma política baseada em identidade.

Você pode especificar um nome de índice parcial adicionando um curinga. Este exemplo identifica todos os índices que começam com `commerce`:

```
arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce*
```

Nesse caso, o curinga significa que o `test-user` pode fazer solicitações para índices em `test-domain` que tenham nomes que comecem com `commerce`.

Para restringir ainda mais o `test-user`, você pode aplicar a seguinte política:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-data/_search"
    }
  ]
}
```

Agora, o `test-user` só pode executar uma operação: pesquisar no índice `commerce-data`. Todos os outros índices no domínio estão inacessíveis, e sem permissões para usar as ações `es:ESHttpPut` ou `es:ESHttpPost`, o `test-user` não pode adicionar ou modificar documentos.

Em seguida, você pode optar por configurar uma função para usuários avançados. Essa política dá `power-user-role` acesso aos métodos HTTP GET e PUT para todos URIs no índice:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:role/power-user-role"
        ]
      },
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPut"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-
data/*"
    }
  ]
}
```

Se o seu domínio estiver em uma VPC ou usar controle de acesso refinado, você poderá usar uma política de acesso a domínios abertos. Caso contrário, sua diretiva de acesso ao domínio deverá conter alguma restrição, seja por endereço IP ou entidade principal.

Para obter informações sobre todas as ações disponíveis, consulte the section called “Referência de elementos da política”. Para obter um controle muito mais granular sobre seus dados, use uma política de acesso a domínio aberto com controle de acesso refinado.

Políticas baseadas em identidade

Ao contrário das políticas baseadas em recursos, que fazem parte de cada domínio de OpenSearch serviço, você anexa políticas baseadas em identidade a usuários ou funções usando o serviço AWS Identity and Access Management (IAM). Assim como nas políticas baseadas em recursos, as políticas baseadas em identidade especificam quem pode acessar um serviço, quais ações podem ser executadas e, se aplicável, em quais recursos essas ações podem ser executadas.

As políticas baseadas em identidade tendem a ser mais genéricas, embora não exista essa exigência. Elas geralmente controlam somente as ações de API de configuração que um usuário pode realizar. Depois de implementar essas políticas, você pode usar políticas baseadas em recursos (ou controle de acesso refinado) no OpenSearch Service para oferecer aos usuários acesso a índices e. OpenSearch APIs

> **Note**
>
> Os usuários com a `AmazonOpenSearchServiceReadOnlyAccess` política AWS gerenciada não conseguem ver o status de integridade do cluster no console. Para permitir que eles vejam o status de integridade do cluster (e outros OpenSearch dados), adicione a `es:ESHttpGet` ação a uma política de acesso e anexe-a às suas contas ou funções.

Como as políticas baseadas em identidade são anexadas a usuários ou funções (principais), o JSON não especifica um principal. A política a seguir concede acesso a ações que começam com `Describe` e `List`. Essa combinação de ações fornece acesso somente leitura a configurações de domínio, mas não aos dados armazenados no próprio domínio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:Describe*",
        "es:List*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Um administrador pode ter acesso total ao OpenSearch Serviço e a todos os dados armazenados em todos os domínios:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

As políticas baseadas em identidade permitem que você use tags para controlar o acesso à API de configuração. A seguinte política, por exemplo, permitirá que as entidades principais anexadas visualizem e atualizem a configuração de um domínio se o domínio tiver a tag `team:devops`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:UpdateDomainConfig",
      "es:DescribeDomain",
      "es:DescribeDomainConfig"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/team": [
          "devops"
        ]
      }
    }
  }]
}
```

Você também pode usar tags para controlar o acesso à OpenSearch API. As políticas baseadas em tags para a OpenSearch API se aplicam somente aos métodos HTTP. Por exemplo, a política a seguir permite que os diretores anexados enviem solicitações GET e PUT para a OpenSearch API se o domínio tiver a `environment:production` tag:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:ESHttpGet",
      "es:ESHttpPut"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
      }
    }
  }]
}
```

Para um controle mais granular da OpenSearch API, considere usar um controle de acesso refinado.

> **Note**
>
> Depois de adicionar uma ou mais OpenSearch APIs políticas baseadas em tags, você deve realizar uma única operação de tag (como adicionar, remover ou modificar uma tag) para que as alterações entrem em vigor em um domínio. Você deve usar o software de serviço R20211203 ou posterior para incluir operações de OpenSearch API em políticas baseadas em tags.

OpenSearch O serviço oferece suporte às chaves de condição `TagKeys` globais `RequestTag` e da API de configuração, não da OpenSearch API. Essas condições aplicam-se somente a chamadas

de API que incluem tags dentro da solicitação, como `CreateDomain`, `AddTags` e `RemoveTags`. A política a seguir permite que as entidades principais anexadas criem domínios, mas somente se incluírem a tag `team:it` na solicitação:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "es:CreateDomain",
      "es:AddTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/team": [
          "it"
        ]
      }
    }
  }
}
```

Para saber mais sobre o uso de tags para controle de acesso e sobre as diferenças entre as políticas baseadas em recursos e as políticas baseadas em identidades, consulte Definir permissões com base em atributos com autorização ABAC no Guia do usuário do IAM.

Políticas baseadas em IP

As políticas baseadas em IP restringem o acesso a um domínio para um ou mais endereços IP ou blocos CIDR. Tecnicamente, as políticas baseadas em IP não são um tipo de política diferente. Mas na verdade, elas são apenas políticas baseadas em recursos que especificam uma entidade principal anônima e incluem uma condição especial. Para saber mais, consulte Elementos de política JSON do IAM: Condition no Guia do usuário do IAM.

O principal atrativo das políticas baseadas em IP é que elas permitem solicitações não assinadas a um domínio de OpenSearch serviço, o que permite usar clientes como curl e OpenSearch painéis

ou acessar o domínio por meio de um servidor proxy. Para saber mais, consulte the section called "Usando um proxy para acessar o OpenSearch serviço a partir de painéis".

> **Note**
>
> Se você ativou o acesso à VPC para seu domínio, não poderá configurar uma política baseada em IP. Em vez disso, você pode usar `security groups` para controlar quais endereços IP podem acessar o domínio. Para saber mais, consulte os seguintes tópicos:
>
> - the section called "Sobre políticas de acesso em domínios da VPC"
> - Controle o tráfego para seus AWS recursos usando grupos de segurança no Guia do usuário da Amazon VPC

A política a seguir concede a todas as solicitações HTTP que se originam no intervalo de IP especificado acesso ao `test-domain`:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
```

```
}
```

Se o seu domínio tiver um endpoint público e não usar controle de acesso refinado, recomendamos combinar entidades principais do IAM e endereços IP. Esta política concederá acesso HTTP ao `test-user` somente se a solicitação se originar do intervalo de IP especificado:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": [
        "arn:aws:iam::987654321098:user/test-user"
      ]
    },
    "Action": [
      "es:ESHttp*"
    ],
    "Condition": {
      "IpAddress": {
        "aws:SourceIp": [
          "192.0.2.0/24"
        ]
      }
    },
    "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
  }]
}
```

Fazendo e assinando solicitações OpenSearch de serviço

Mesmo se você configurar uma política de acesso totalmente aberta baseada em recursos, todas as solicitações para a API de configuração do OpenSearch serviço devem ser assinadas. Se suas políticas especificarem funções ou usuários do IAM, as solicitações para o OpenSearch APIs também deverão ser assinadas usando o AWS Signature versão 4. O método de assinatura é diferente dependendo da API:

- Para fazer chamadas para a API de configuração do OpenSearch serviço, recomendamos que você use uma das AWS SDKs. SDKsIsso simplifica muito o processo e pode economizar uma quantidade significativa de tempo em comparação com a criação e assinatura de suas próprias solicitações. Os endpoints da API de configuração usam o formato a seguir:

```
es.region.amazonaws.com/2021-01-01/
```

Por exemplo, a seguinte solicitação faz uma alteração de configuração no domínio `movies`, mas é necessário que você a assine (não recomendado):

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/movies/config
{
  "ClusterConfig": {
    "InstanceType": "c5.xlarge.search"
  }
}
```

Se você usa um dos SDKs, como o Boto 3, o SDK processa automaticamente a assinatura da solicitação:

```
import boto3

client = boto3.client(es)
response = client.update_domain_config(
  DomainName='movies',
  ClusterConfig={
    'InstanceType': 'c5.xlarge.search'
  }
)
```

Para obter um código de exemplo Java, consulte the section called “Usando o AWS SDKs”.

- Para fazer chamadas para o OpenSearch APIs, você deve assinar suas próprias solicitações. OpenSearch APIs Use o seguinte formato:

```
domain-id.region.es.amazonaws.com
```

Por exemplo, a seguinte solicitação procura o índice `movies` para thor:

```
GET https://my-domain.us-east-1.es.amazonaws.com/movies/_search?q=thor
```

> **Note**
>
> O serviço ignora os parâmetros passados URLs para solicitações HTTP POST assinadas com o Signature Version 4.

Quando há colisão de políticas

Quando as políticas discordam entre si ou não fazem nenhuma referência explícita a um usuário, surgem complexidades. Como o IAM funciona no Guia do usuário do IAM fornece um breve resumo da lógica de avaliação de políticas:

- Por padrão, todas as solicitações são negadas.
- Uma permissão explícita substitui esse padrão.
- Uma negar explícito substitui todas as permissões.

Por exemplo, se uma política baseada em recursos conceder acesso a um sub-recurso de domínio (um OpenSearch índice ou API), mas uma política baseada em identidade negar seu acesso, você terá acesso negado. Se uma política baseada em identidade concede acesso e uma política baseada em recursos não especifica se você deve ou não ter acesso, esse acesso é concedido. Consulte a tabela a seguir com o cruzamento de políticas para obter um resumo completo dos resultados para sub-recursos de domínios.

	Permitido na política baseada em recursos	Negado na política baseada em recursos	Nem permitido nem negado na política baseada em recursos
Allowed in identity-based policy	Permitir	Deny	Permitir
Denied in identity-based policy	Deny	Deny	Deny

	Permitido na política baseada em recursos	Negado na política baseada em recursos	Nem permitido nem negado na política baseada em recursos
Neither allowed nor denied in identity-based policy	Permitir	Deny	Deny

Referência de elementos da política

OpenSearch O serviço oferece suporte à maioria dos elementos de política na Referência de elementos de política do IAM, com exceção de`NotPrincipal`. A tabela a seguir mostra os elementos mais comuns.

Elemento da política de JSON	Resumo
`Version`	Versão atual da linguagem de política é `2012-10-17` . Todas as políticas de acesso devem especificar esse valor.
`Effect`	Esse elemento especifica se a declaração permite ou nega o acesso às ações especificadas. Os valores válidos são `Allow` ou `Deny`.
`Principal`	Esse elemento especifica a função Conta da AWS ou o usuário do IAM que tem acesso permitido ou negado a um recurso e pode assumir várias formas: • AWS contas: `"Principal":{"AWS": ["123456789012"]}` ou `"Principal":{"AWS": ["arn:aws:iam::123456789012:root"]}` • Usuários do IAM: `"Principal":{"AWS": ["arn:aws:iam::123456789012:user/test-user"]}` • Funções do IAM: `"Principal":{"AWS": ["arn:aws:iam::123456789012:role/test-role"]}`

Elemento da política de JSON	Resumo
	Important Especificar o curinga * permite o acesso anônimo ao domínio, o que não recomendamos, a menos que você adicione uma condição baseada em IP, use o suporte à VPC ou habilite o controle de acesso refinado. Além disso inspecione com cuidado as seguintes políticas para garantir que elas não concedam acesso amplo: • Políticas baseadas em identidade vinculadas a entidades principais da AWS associadas (por exemplo, perfis do IAM). • Políticas baseadas em recursos anexadas aos AWS recur sos associados (por exemplo, chaves AWS Key Management Service KMS)

Elemento da política de JSON	Resumo
`Action`	OpenSearch O serviço usa `ESHttp*` ações para métodos OpenSearch HTTP. O resto das ações se aplicam à API de configuração. Determinadas ações `es:` dão suporte a permissões no nível do recurso. Por exemplo, você pode conceder a um usuário permissões para excluir um determinado domínio sem conceder a esse usuário permissões para excluir qualquer domínio. Outras ações se aplicam apenas ao serviço em si. Por exemplo, `es:ListDomainNames` não faz sentido no contexto de um único domínio e, portanto, requer um curinga. Para obter uma lista de todas as ações disponíveis e se elas se aplicam aos sub-recursos do domínio (`test-domain/*`), à configuração do domínio (`test-domain`) ou somente ao serviço (`*`), consulte Ações, recursos e chaves de condição do Amazon OpenSearch Service na Referência de Autorização de Serviço Políticas baseadas em recursos são diferentes das permissões no nível do recurso. As políticas baseadas em recursos são políticas JSON completas anexadas aos domínios. As permissões no nível do recurso tornam possível a restrição de ações em domínios específicos ou sub-recursos. Na prática, você pode pensar na permissão no nível do recurso como uma seção opcional de um recurso ou uma política baseada em identidade. Embora as permissões no nível de recurso para `es:CreateDomain` possam parecer não intuitivas — afinal de contas, por que conceder permissões a um usuário para criar um domínio que já existe? —, o uso de um curinga permite aplicar um esquema de nomenclatura simples aos seus domínios, como `"Resource": "arn:aws:es:us-west-1:987654321098:domain/my-team-name-*"` . Naturalmente, nada impede que você inclua ações juntamente com elementos de recursos menos restritivos, como estes:

Elemento da política de JSON	Resumo
	JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "es:ESHttpGet", "es:DescribeDomain" ], "Resource": "*" } ] }` Para saber mais sobre o emparelhamento de ações e recursos, consulte o elemento `Resource` nesta tabela.

Elemento da política de JSON	Resumo
`Condition`	OpenSearch O serviço é compatível com a maioria das condições descritas nas chaves de contexto de condição AWS global no Guia do usuário do IAM. Exceções notáveis incluem a `aws:PrincipalTag` chave, que o OpenSearch Serviço não suporta. Ao configurar uma política baseada em IP, você especifica os endereços IP ou bloco CIDR como uma condição, como esta: `"Condition": { "IpAddress": { "aws:SourceIp": [ "192.0.2.0/32" ] } }` Conforme observado em the section called “Políticas baseadas em identidade”`aws:ResourceTag` , as chaves de `aws:TagKeys` condição`aws:RequestTag` , e se aplicam à API de configuração, bem como OpenSearch APIs a.

Elemento da política de JSON	Resumo

`Resource`	OpenSearch O serviço usa `Resource` elementos de três maneiras básicas:
	
	• Para ações que se aplicam ao próprio OpenSearch Serviço, como`es:ListDomainNames` , ou para permitir acesso total, use a seguinte sintaxe:
	
	`"Resource": "*"`
	
	• Para as ações que envolvem uma configuração de domínio, como `es:DescribeDomain` , você pode usar a seguinte sintaxe:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`"`
	
	• Para as ações que se aplicam a um sub-recurso de domínio, como `es:ESHttpGet` , você pode usar a seguinte sintaxe:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/*"`
	
	Você não precisa usar um curinga. OpenSearch O serviço permite que você defina uma política de acesso diferente para cada OpenSearch índice ou API. Por exemplo, você pode limitar as permissões de um usuário para o índice `test-index` :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index"`
	
	Em vez de acesso total ao `test-index` , você pode preferir limitar a política somente à API de pesquisa:
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index/_search"`

Elemento da política de JSON	Resumo
	Você pode até mesmo controlar o acesso a documentos individuais: `"Resource": "arn:aws:es:  region:aws-account-id:domain/domain-name /test-index/test-type/1"` Essencialmente, se OpenSearch expressar o sub-recurso como um URI, você pode controlar o acesso a ele usando uma política de acesso. Para ter ainda mais controle sobre quais recursos um usuário pode acessar, consulte the section called “Controle de acesso refinado”. Para obter detalhes sobre quais ações dão suporte a permissões no nível do recurso, consulte o elemento `Action` nesta tabela.

Opções avançadas e considerações sobre a API

OpenSearch O serviço tem várias opções avançadas, uma das quais tem implicações de controle de acesso:`rest.action.multi.allow_explicit_index`. Como sua configuração padrão é verdadeiro, ela permite que os usuários ignorem as permissões de sub-recursos em determinadas circunstâncias.

Por exemplo, considere a seguinte política baseada em recurso:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
```

```
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/test-index/*",
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/_bulk"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

Essa política concede acesso `test-user` total `test-index` e à API OpenSearch em massa. Ela também permite solicitações `GET` ao `restricted-index`.

A seguinte solicitação de indexação, como você pode esperar, falha devido a um erro de permissão:

```
PUT https://search-test-domain.us-west-1.es.amazonaws.com/restricted-index/movie/1
{
  "title": "Your Name",
  "director": "Makoto Shinkai",
  "year": "2016"
}
```

Ao contrário da API de índice, a API em massa permite criar, atualizar e excluir vários documentos em uma única chamada. Contudo, normalmente você especifica essas operações no corpo da solicitação, em vez de na URL da solicitação. Como o OpenSearch Service usa URLs para controlar o acesso aos sub-recursos do domínio, `test-user` pode, na verdade, usar a API em massa para

fazer alterações em. `restricted-index` Embora o usuário não tenha permissões POST no índice, a seguinte solicitação é bem-sucedida:

```
POST https://search-test-domain.us-west-1.es.amazonaws.com/_bulk
{ "index" : { "_index": "restricted-index", "_type" : "movie", "_id" : "1" } }
{ "title": "Your Name", "director": "Makoto Shinkai", "year": "2016" }
```

Nesta situação, a política de acesso não consegue cumprir o que pretendia. Para evitar que os usuários ignorem esses tipos de restrições, você pode alterar o `rest.action.multi.allow_explicit_index` para o valor falso. Se esse valor for falso, todas as chamadas para bulk, mget e msearch APIs que especificam nomes de índice no corpo da solicitação deixarão de funcionar. Em outras palavras, as chamadas para `_bulk` não funcionam mais, mas as chamadas para o `test-index/_bulk` funcionam. Este segundo endpoint contém um nome de índice, portanto, você não precisa especificar um no corpo da solicitação.

OpenSearch Os painéis dependem muito de mget e msearch, portanto, é improvável que funcionem corretamente após essa alteração. Para remediação parcial, você pode deixar `rest.action.multi.allow_explicit_index` como verdadeiro e negar a determinados usuários o acesso a um ou mais deles APIs.

Para obter informações sobre como alterar essa configuração, consulte the section called “Configurações avançadas do cluster”.

Da mesma forma, a seguinte política baseada em recursos contém dois problemas sutis:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    },
    {
      "Effect": "Deny",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

- Apesar da negação explícita, o `test-user` ainda pode fazer chamadas, como `GET https://search-test-domain.us-west-1.es.amazonaws.com/_all/_search` e `GET https://search-test-domain.us-west-1.es.amazonaws.com/*/_search` para acessar os documentos no `restricted-index`.
- Como o elemento `Resource` faz referência ao `restricted-index/*`, o `test-user` não tem permissões para acessar diretamente os documentos do índice. O usuário, no entanto, tem permissões para excluir todo o índice. Para evitar o acesso e a exclusão, a política deve especificar `restricted-index*`.

Em vez de misturar permissões amplas e negações focadas, a abordagem mais segura é seguir o princípio do privilégio mínimo e conceder apenas as permissões necessárias para executar uma tarefa. Para obter mais informações sobre como controlar o acesso a índices ou OpenSearch operações individuais, consultethe section called “Controle de acesso refinado”.

> **Important**
>
> A especificação de um caractere curinga * permite acesso anônimo ao domínio. Não é recomendável usar o caractere curinga. Além disso analise com cuidado as seguintes políticas para garantir que não concedam amplo acesso:
>
> - Políticas baseadas em identidade vinculadas aos AWS diretores associados (por exemplo, funções do IAM)
> - Políticas baseadas em recursos anexadas aos AWS recursos associados (por exemplo, chaves AWS Key Management Service KMS)

Configuração de políticas de acesso

- Para obter instruções sobre como criar ou modificar políticas baseadas em recursos e IP no OpenSearch Service, consulte. the section called “Configuração de políticas de acesso”
- Para obter instruções para criar ou modificar políticas baseadas em identidades no IAM, consulte Defina permissões personalizadas do IAM com políticas gerenciadas pelo cliente no Guia do usuário do IAM.

Exemplos adicionais de políticas

Embora este capítulo inclua muitos exemplos de políticas, o controle de AWS acesso é um assunto complexo que é melhor compreendido por meio de exemplos. Para saber mais, consulte Exemplo de políticas baseadas em identidade do IAM noManual do usuário do IAM.

Referência de permissões da Amazon OpenSearch Service API

Ao configurar o controle de acesso, você escreve políticas de permissão que podem ser anexadas a uma identidade do IAM (políticas baseadas em identidade). Para obter informações de referência detalhadas, consulte os seguintes tópicos na Referência de autorização do serviço:

- Ações, recursos e chaves de condição para o OpenSearch Serviço.
- Ações, recursos e chaves de condição para OpenSearch ingestão.

Essa referência contém informações sobre quais operações de API do podem ser usadas em uma política do IAM. Também inclui o AWS recurso para o qual você pode conceder as permissões e as chaves de condição que você pode incluir para um controle de acesso refinado.

Você especifica as ações no campo `Action` da política, o valor de recurso no campo `Resource` da política e as condições no campo `Condition` da política. Para especificar uma ação para o OpenSearch Serviço, use o `es:` prefixo seguido pelo nome da operação da API (por exemplo,`es:CreateDomain`). Para especificar uma ação para OpenSearch ingestão, use o `osis:` prefixo seguido pela operação da API (por exemplo,`osis:CreatePipeline`).

AWS políticas gerenciadas para o Amazon OpenSearch Service

Uma política AWS gerenciada é uma política autônoma criada e administrada por AWS. AWS as políticas gerenciadas são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas AWS gerenciadas podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque elas estão disponíveis para uso de todos os AWS clientes. Recomendamos que você reduza ainda mais as permissões definindo as políticas gerenciadas pelo cliente que são específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas AWS gerenciadas. Se AWS atualizar as permissões definidas em uma política AWS gerenciada, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está anexada. AWS é mais provável que atualize uma política AWS gerenciada quando uma nova AWS service (Serviço da AWS) for lançada ou novas operações de API forem disponibilizadas para serviços existentes.

Para saber mais, consulte AWS Políticas gerenciadas pela no Guia do usuário do IAM.

AmazonOpenSearchDirectQueryGlueCreateAccess

OpenSearch Concede ao Amazon Service Direct Query Service acesso ao `CreateDatabase` `CreatePartitionCreateTable`,, `BatchCreatePartition` AWS Glue API e.

Você pode encontrar a AmazonOpenSearchDirectQueryGlueCreateAccesspolítica no console do IAM.

AmazonOpenSearchServiceFullAccess

Concede acesso total às operações e aos recursos da API de configuração de OpenSearch serviços para um Conta da AWS.

Você pode encontrar a AmazonOpenSearchServiceFullAccesspolítica no console do IAM.

AmazonOpenSearchServiceReadOnlyAccess

Concede acesso somente de leitura a todos os recursos do OpenSearch Serviço para um. Conta da AWS

Você pode encontrar a AmazonOpenSearchServiceReadOnlyAccesspolítica no console do IAM.

AmazonOpenSearchServiceRolePolicy

Não é possível anexar a `AmazonOpenSearchServiceRolePolicy` às entidades do IAM. Essa política está vinculada a uma função vinculada ao serviço que permite que o OpenSearch Serviço acesse os recursos da conta. Para obter mais informações, consulte the section called “Permissões”.

Você pode encontrar a AmazonOpenSearchServiceRolePolicypolítica no console do IAM.

AmazonOpenSearchServiceCognitoAccess

Fornece as permissões mínimas do Amazon Cognito necessárias para ativar a autenticação Cognito.

Você pode encontrar a AmazonOpenSearchServiceCognitoAccesspolítica no console do IAM.

AmazonOpenSearchIngestionServiceRolePolicy

Não é possível anexar a `AmazonOpenSearchIngestionServiceRolePolicy` às entidades do IAM. Essa política está vinculada a uma função vinculada ao serviço que permite que a OpenSearch ingestão habilite o acesso à VPC para pipelines de ingestão, crie tags e publique métricas relacionadas à ingestão em sua conta. CloudWatch Para obter mais informações, consulte the section called “Uso de perfis vinculados ao serviço”.

Você pode encontrar a AmazonOpenSearchIngestionServiceRolePolicypolítica no console do IAM.

OpenSearchIngestionSelfManagedVpcePolicy

Não é possível anexar a `OpenSearchIngestionSelfManagedVpcePolicy` às entidades do IAM. Essa política está vinculada a uma função vinculada ao serviço que permite que a OpenSearch ingestão habilite o acesso autogerenciado à VPC para pipelines de ingestão, crie tags e publique métricas relacionadas à ingestão em sua conta. CloudWatch Para obter mais informações, consulte the section called “Uso de perfis vinculados ao serviço”.

Você pode encontrar a OpenSearchIngestionSelfManagedVpcePolicypolítica no console do IAM.

AmazonOpenSearchIngestionFullAccess

Concede acesso total às operações e aos recursos da API de OpenSearch ingestão para um Conta da AWS.

Você pode encontrar a AmazonOpenSearchIngestionFullAccesspolítica no console do IAM.

AmazonOpenSearchIngestionReadOnlyAccess

Concede acesso somente de leitura a todos os recursos OpenSearch de ingestão para um. Conta da AWS

Você pode encontrar a AmazonOpenSearchIngestionReadOnlyAccesspolítica no console do IAM.

AmazonOpenSearchServerlessServiceRolePolicy

Fornece as Amazon CloudWatch permissões mínimas necessárias para enviar dados métricos OpenSearch sem servidor para o. CloudWatch

Você pode encontrar a AmazonOpenSearchServerlessServiceRolePolicypolítica no console do IAM.

OpenSearch Atualizações de serviços para políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas do OpenSearch Serviço desde que esse serviço começou a monitorar as alterações.

Alteração	Descrição	Data
Atualizou o `AmazonOpenSearchIngestionServiceRolePolicy`	A atualização dá permissão ao OpenSearch Ingestion para modificar os VPC endpoints criados OpenSearch por para compartilhar pipelines entre eles. VPCs Para ver a política JSON, consulte o console do IAM.	28 de agosto de 2025
Atualizou o `AmazonOpenSearchServiceRolePolicy`	Adicionada a seguinte instrução à política. Quando o Amazon OpenSearch Service assume a função `AWSServiceRoleForAmazonOpenSearchService` vinculada ao serviço, essa nova declaração na	31 de março de 2025

Alteração	Descrição	Data
	política permite OpenSearch atualizar o escopo de acesso de qualquer Centro de Identidade do AWS IAM aplicativo que seja gerenciado apenas pelo. OpenSearch `{ "Effect": "Allow", "Action": "sso:PutApplicationAccessScope", "Resource": "arn:aws:sso::*:application/*/*", "Condition": { "StringEquals": { "aws:ResourceOrgID": "${aws:PrincipalOrgID}" } } }`	
Atualização do `AmazonOpenSearchServerlessServiceRolePolicy`	Adicionado o Sid `AllowAOSSCloudwatchMetrics` à política `AmazonOpenSearchServerlessServiceRolePolicy`. Um Sid é um ID de instrução que funciona como um identificador opcional da instrução da política.	12 de julho de 2024

Alteração	Descrição	Data
`OpenSearchIngestionSelfManagedVpcePolicy` adicionado	Uma nova política que permite que a OpenSearch ingestão habilite o acesso autogerenciado à VPC para pipelines de ingestão, crie tags e publique métricas relacionadas à ingestão em sua conta. CloudWatch Para ver a política JSON, consulte o console do IAM.	12 de junho de 2024
Adicionado `AmazonOpenSearchDirectQueryGlueCreateAccess`	OpenSearch Concede ao Amazon Service Direct Query Service acesso ao `CreateDatabase` `CreatePartition` `CreateTable` ,, `BatchCreatePartition` AWS Glue API e.	6 de maio de 2024
Atualização do `AmazonOpenSearchServiceRolePolicy` e do `AmazonElasticsearchServiceRolePolicy` .	Foram adicionadas as permissões necessárias para que a função vinculada ao serviço atribua e IPv6 cancele a atribuição de endereços. A política obsoleta do Elasticsearch também foi atualizada para garantir a compatibilidade com versões anteriores.	18 de outubro de 2023

Alteração	Descrição	Data
`AmazonOpenSearchIngestionServiceRolePolicy` adicionado	Uma nova política que permite que a OpenSearch ingestão habilite o acesso à VPC para pipelines de ingestão, crie tags e publique métricas relacionadas à CloudWatch ingestão em sua conta. Para ver a política JSON, consulte o console do IAM.	26 de abril de 2023
`AmazonOpenSearchIngestionFullAccess` adicionado	Uma nova política que concede acesso total às operações e aos recursos da API de OpenSearch ingestão para um Conta da AWS. Para ver a política JSON, consulte o console do IAM.	26 de abril de 2023
`AmazonOpenSearchIngestionReadOnlyAccess` adicionado	Uma nova política que concede acesso somente de leitura a todos os recursos de OpenSearch ingestão para um. Conta da AWS Para ver a política JSON, consulte o console do IAM.	26 de abril de 2023

Alteração	Descrição	Data
`AmazonOpenSearchServerlessServiceRolePolicy` adicionado	Uma nova política que fornece as permissões mínimas necessárias para enviar dados métricos OpenSearch sem servidor para o. Amazon CloudWatch Para ver a política JSON, consulte o console do IAM.	29 de novembro de 2022
Atualização do `AmazonOpenSearchServiceRolePolicy` e do `AmazonElasticsearchServiceRolePolicy` .	Foram adicionadas as permissões necessárias para que a função vinculada ao serviço crie VPC endpoints OpenSearch gerenciados por serviços. Algumas ações só podem ser executadas quando a solicitação contém a tag `OpenSearchManaged=true` . A política obsoleta do Elasticsearch também foi atualizada para garantir a compatibilidade com versões anteriores.	7 de novembro de 2022

Alteração	Descrição	Data
Atualização do `AmazonOpenSearchServiceRolePolicy` e do `AmazonElasticsearchServiceRolePolicy` .	Foi adicionado suporte para a `PutMetricData` ação, que é necessário para publicar métricas de OpenSearch cluster na Amazon CloudWatch. A política obsoleta do Elasticsearch também foi atualizada para garantir a compatibilidade com versões anteriores. Para ver a política JSON, consulte o console do IAM.	12 de setembro de 2022
Atualização do `AmazonOpenSearchServiceRolePolicy` e do `AmazonElasticsearchServiceRolePolicy` .	Adicionado suporte ao tipo de recurso `acm`. A política fornece a permissão mínima AWS Certificate Manager (ACM) somente leitura necessária para que a função vinculada ao serviço verifique e valide os recursos do ACM a fim de criar e atualizar domínios personalizados habilitados para endpoints. A política obsoleta do Elasticsearch também foi atualizada para garantir a compatibilidade com versões anteriores.	28 de julho de 2022

Alteração	Descrição	Data
Atualização do `AmazonOpenSearchServiceCognitoAccess` e do `AmazonESCognitoAccess` .	Foi adicionado suporte para a `UpdateUserPoolClient` ação, que é necessário para definir a configuração do grupo de usuários do Cognito durante a atualização do Elasticsearch para o. OpenSearch Permissões corrigidas para a ação `SetIdentityPoolRoles` para permitir o acesso a todos os recursos. A política obsoleta do Elasticsearch também foi atualizada para garantir a compatibilidade com versões anteriores.	20 de dezembro de 2021
Atualização do `AmazonOpenSearchServiceRolePolicy`	Adicionado suporte ao tipo de recurso `security-group` . A política fornece as permissões mínimas do Amazon EC2 e do Elastic Load Balancing necessárias para a função vinculada ao serviço habilitar o acesso à VPC.	9 de setembro de 2021

Alteração	Descrição	Data
• Adicionado `AmazonOpenSearchServiceFullAccess` • Defasada `AmazonESFullAccess`	Esta nova política destina-se a substituir a política antiga. Ambas as políticas fornecem acesso total à API OpenSearch de configuração do serviço e a todos os métodos HTTP do OpenSearch APIs. O controle de acesso refinado e as políticas baseadas em recursos ainda podem restringir o acesso.	7 de setembro de 2021
• Adicionado `AmazonOpenSearchServiceReadOnlyAccess` • Defasada `AmazonESReadOnlyAccess`	Esta nova política destina-se a substituir a política antiga. Ambas as políticas fornecem acesso somente de leitura à API de configuração do OpenSearch serviço (`es:Describe* es:List*`,, `ees:Get*`) e nenhum acesso aos métodos HTTP para o. OpenSearch APIs	7 de setembro de 2021
• Adicionado `AmazonOpenSearchServiceCognitoAccess` • Defasada `AmazonESCognitoAccess`	Esta nova política destina-se a substituir a política antiga. Ambas as políticas fornecem as permissões mínimas do Amazon Cognito necessárias para ativar a autenticação Cognito.	7 de setembro de 2021

Alteração	Descrição	Data
• Adicionado AmazonOpenSearchServiceRolePolicy • Defasada `AmazonElasticsearchServiceRolePolicy`	Esta nova política destina-se a substituir a política antiga. Ambas as políticas fornecem as permissões mínimas do Amazon EC2 e do Elastic Load Balancing necessárias para a função vinculada ao serviço habilitar o acesso à VPC.	7 de setembro de 2021
Início do rastreamento das alterações	O Amazon OpenSearch Service agora rastreia as alterações nas políticas AWS gerenciadas.	7 de setembro de 2021

Prevenção contra o ataque do “substituto confuso” em todos os serviços

O problema "confused deputy" é um problema de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executar a ação. Em AWS, a falsificação de identidade entre serviços pode resultar em um problema confuso de delegado. A personificação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Recomendamos usar as chaves de contexto de condição `aws:SourceAccount`global `aws:SourceArn`e as chaves de contexto nas políticas de recursos para limitar as permissões que a Amazon OpenSearch Service concede a outro serviço ao recurso. Se o valor de `aws:SourceArn` não contém ID da conta, como um ARN do bucket do Amazon S3, você deve usar ambas as chaves de contexto de condição global para limitar as permissões. Se você usa ambas as chaves de contexto de condição global, e o valor `aws:SourceArn` contém o ID da conta, o valor

`aws:SourceAccount` e a conta no valor `aws:SourceArn` deverão utilizar a mesma ID de conta quando na mesma declaração de política. Use `aws:SourceArn` se quiser apenas um recurso associado a acessibilidade de serviço. Use `aws:SourceAccount` se quiser permitir que qualquer recurso nessa conta seja associado ao uso entre serviços.

O valor de `aws:SourceArn` deve ser o ARN do domínio do OpenSearch serviço.

A maneira mais eficaz de se proteger do problema 'confused deputy' é usar a chave de contexto de condição global `aws:SourceArn` com o ARN completo do recurso. Se você não souber o ARN completo do recurso ou se especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com curingas (*) para as partes desconhecidas do ARN. Por exemplo, .`arn:aws:es:*:123456789012:*`

O exemplo a seguir mostra como você pode usar as chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e as chaves de contexto no OpenSearch Service para evitar o problema confuso do substituto.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":{
      "Sid":"ConfusedDeputyPreventionExamplePolicy",
      "Effect":"Allow",
      "Principal":{
         "Service":"es.amazonaws.com"
      },
      "Action":"sts:AssumeRole",
      "Condition":{
         "StringEquals":{
            "aws:SourceAccount":"111122223333"
         },
         "ArnLike":{
            "aws:SourceArn":"arn:aws:es:us-east-1:111122223333:domain/my-domain"
         }
      }
   }
}
```

Controle de acesso refinado no Amazon Service OpenSearch

O controle de acesso refinado oferece formas adicionais de controlar o acesso aos seus dados no Amazon Service. OpenSearch Por exemplo, dependendo de quem faz a solicitação, você pode querer que uma pesquisa retorne resultados de somente um índice. Talvez você queira ocultar determinados campos em seus documentos ou excluir determinados documentos completamente.

O controle de acesso refinado oferece os seguintes recursos:

- Controle de acesso com base em função
- Segurança no nível de índice, documento e campo
- OpenSearch Multilocação de painéis
- Autenticação básica HTTP para OpenSearch OpenSearch painéis

Tópicos

- Visão geral: controle de acesso refinado e segurança de serviços OpenSearch
- Principais conceitos
- Sobre o usuário-mestre
- Habilitar o controle de acesso detalhado
- Acessando OpenSearch painéis como usuário principal
- Gerenciar permissões
- Configurações recomendadas
- Limitações
- Modificação do usuário primário
- Usuários primários adicionais
- Snapshots manuais
- Integrações
- Diferenças de API REST
- Tutorial: Configuração de um domínio com um usuário primário do IAM e autenticação do Amazon Cognito
- Tutorial: Configuração de um domínio com o banco de dados interno do usuário e a autenticação básica HTTP

Visão geral: controle de acesso refinado e segurança de serviços OpenSearch

A segurança OpenSearch do Amazon Service tem três camadas principais:

Rede

A primeira camada de segurança é a rede, que determina se as solicitações chegam a um domínio OpenSearch de serviço. Se você escolher Acesso público ao criar um domínio, as solicitações de qualquer cliente conectado à Internet poderão chegar ao endpoint do domínio. Se você escolher o Acesso à VPC, os clientes devem se conectar à VPC (e os grupos de segurança associados devem permitir) para que uma solicitação chegue ao endpoint. Para saber mais, consulte the section called “Suporte à VPC”.

Política de acesso ao domínio

A segunda camada de segurança é a política de acesso ao domínio. Depois que uma solicitação chega a um endpoint do domínio, a política de acesso baseada em recursos permite ou nega o acesso da solicitação a um determinado URI. A política de acesso aceita ou rejeita solicitações na "borda" do domínio, antes que elas cheguem ao OpenSearch.

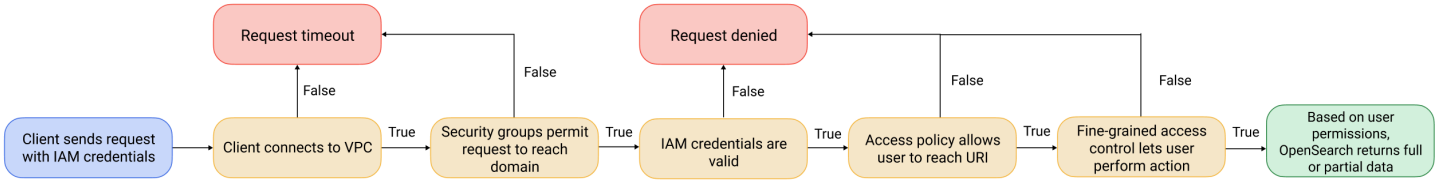
Controle de acesso refinado

A terceira e última camada de segurança é o controle de acesso refinado. Depois que uma política de acesso baseada em recursos permitir que uma solicitação chegue a um endpoint do domínio, o controle de acesso refinado avaliará as credenciais do usuário e autenticará o usuário ou negará a solicitação. Se o controle de acesso refinado autenticar o usuário, ele obterá todas as funções mapeadas para esse usuário e usará o conjunto completo de permissões para determinar como lidar com a solicitação.

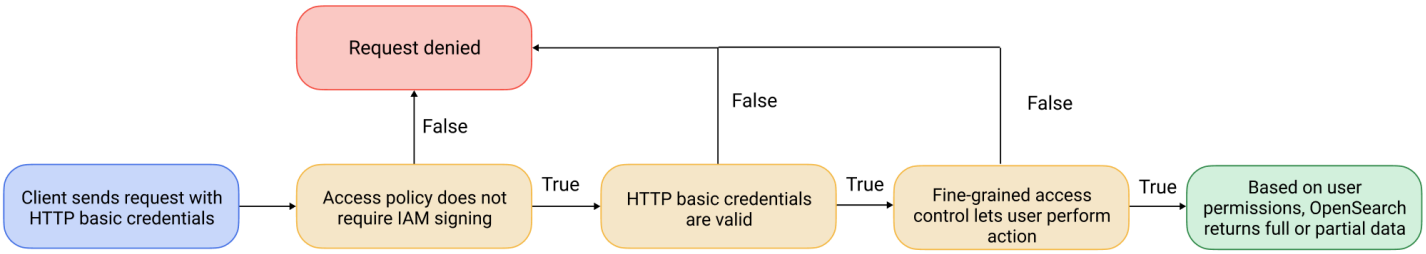
> **Note**
>
> Se uma política de acesso baseada em recursos contiver funções ou usuários do IAM, os clientes devem enviar solicitações assinadas usando o AWS Signature versão 4. Como tal, as políticas de acesso podem entrar em conflito com o controle de acesso refinado, especialmente se você usar o banco de dados interno de usuários e a autenticação básica HTTP. Não é possível assinar uma solicitação com um nome de usuário e

senha e credenciais do IAM. Em geral, se você habilitar o controle de acesso refinado, recomendamos usar uma política de acesso ao domínio que não exija solicitações assinadas.

O diagrama a seguir ilustra uma configuração comum: um domínio de acesso da VPC com controle de acesso refinado habilitado, uma política de acesso baseada no IAM e um usuário primário do IAM.



O diagrama ilustra a seguir outra configuração comum: um domínio de acesso público com controle de acesso refinado habilitado, uma política de acesso que não usa os principais do IAM e um usuário primário no banco de dados de usuários interno.



Exemplo

Considere uma solicitação de `GET` para `movies/_search?q=thor`. O usuário tem permissões para pesquisar o índice `movies`? Em caso afirmativo, o usuário tem as permissões para exibir todos os documentos dentro dele? A resposta deve omitir ou tornar algum campo anônimo? Para o usuário primário, a resposta pode ser semelhante a esta:

```
{
  "hits": {
    "total": 7,
    "max_score": 8.772789,
    "hits": [{
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
```

```
          "directors": [
            "Kenneth Branagh",
            "Joss Whedon"
          ],
          "release_date": "2011-04-21T00:00:00Z",
          "genres": [
            "Action",
            "Adventure",
            "Fantasy"
          ],
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor",
          "actors": [
            "Chris Hemsworth",
            "Anthony Hopkins",
            "Natalie Portman"
          ],
          "year": 2011
        }
      },
      ...
    ]
  }
}
```

Se um usuário com permissões mais limitadas emitir exatamente a mesmo solicitação, a resposta pode ser semelhante a esta:

```
{
  "hits": {
    "total": 2,
    "max_score": 8.772789,
    "hits": [{
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
          "year": 2011,
          "release_date":
 "3812a72c6dd23eef3c750c2d99e205cbd260389461e19d610406847397ecb357",
```

```
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor"
        }
      },
      ...
    ]
  }
}
```

A resposta tem menos ocorrências e menos campos para cada ocorrência. Além disso, o campo `release_date` torna-se anônimo. Se um usuário sem permissões fizer a mesma solicitação, o cluster retornará um erro:

```
{
  "error": {
    "root_cause": [{
      "type": "security_exception",
      "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
    }],
    "type": "security_exception",
    "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
  },
  "status": 403
}
```

Se um usuário fornecer credenciais inválidas, o cluster retornará uma exceção `Unauthorized`.

Principais conceitos

Ao começar a usar o controle de acesso refinado, considere os seguintes conceitos:

- Perfis: a principal maneira de usar o controle de acesso refinado. Nesse caso, as funções são distintas das funções do IAM. As funções contêm qualquer combinação de permissões: nível de cluster, específica de índice, nível de documento e nível de campo.
- Mapeamento: depois de configurar um perfil, você o mapeia para um ou mais usuários. Por exemplo, é possível mapear três funções para um único usuário: uma função que fornece acesso

ao Dashboards, uma que fornece acesso somente leitura ao `index1` e uma que fornece acesso de gravação ao `index2`. Ou, é possível incluir todas essas permissões em uma única função.

- Usuários — Pessoas ou aplicativos que fazem solicitações ao OpenSearch cluster. Os usuários têm credenciais, sejam chaves de acesso do IAM ou um nome de usuário e senha, que eles especificam quando fazem solicitações.

Sobre o usuário-mestre

O usuário principal no OpenSearch Service é uma combinação de nome de usuário e senha, ou um principal do IAM, que tem permissões completas para o OpenSearch cluster subjacente. Um usuário é considerado usuário principal se tiver todo o acesso ao OpenSearch cluster junto com a capacidade de criar usuários internos, funções e mapeamentos de funções nos OpenSearch painéis.

Um usuário mestre criado no console de OpenSearch serviço ou por meio da CLI é mapeado automaticamente para duas funções predefinidas:

- `all_access`: fornece total acesso a todas as operações em todo o cluster, permissão para gravar em todos os índices do cluster e permissão para gravar em todos os locatários.
- `security_manager`: fornece acesso ao plug-in de segurança e ao gerenciamento de usuários e permissões.

Com essas duas funções, o usuário obtém acesso à guia Segurança nos OpenSearch painéis, onde pode gerenciar usuários e permissões. Se você criar outro usuário interno e mapeá-lo apenas para o perfil `all_access`, esse usuário não terá acesso à guia Segurança. Você pode criar usuários-mestres adicionais mapeando-os explicitamente para os perfis `all_access` e `security_manager`. Para instruções, consulte the section called “Usuários primários adicionais”.

Ao criar um usuário-mestre para o domínio, você pode especificar uma entidade principal do IAM existente ou criar um usuário-mestre no banco de dados de usuários internos. Considere o seguinte ao decidir qual deles usar:

- IAM principal — Se você escolher um IAM principal para seu usuário principal, todas as solicitações para o cluster devem ser assinadas usando o AWS Signature Version 4.

  OpenSearch O serviço não leva em consideração nenhuma das permissões do diretor do IAM. O usuário ou perfil do IAM serve apenas para autenticação. As políticas sobre esse usuário ou perfil

não têm nenhuma relação com a autorização do usuário-mestre. A autorização é feita por meio de várias permissões no plug-in de OpenSearch segurança.

Por exemplo, você pode atribuir zero permissões do IAM a um principal do IAM e, desde que a máquina ou pessoa possa se autenticar para esse usuário ou função, ela terá o poder do usuário mestre no OpenSearch Serviço.

Recomendamos o IAM se você quiser usar os mesmos usuários em vários clusters, se quiser usar o Amazon Cognito para acessar painéis ou se tiver OpenSearch clientes que ofereçam suporte à assinatura do Signature versão 4.

- Banco de dados de usuários interno: se você criar um mestre no banco de dados de usuários interno (com uma combinação de nome de usuário e senha), poderá usar a autenticação HTTP básica (bem como as credenciais do IAM) para fazer solicitações ao cluster. A maioria dos clientes oferece suporte à autenticação básica, incluindo curl, que também oferece suporte à AWS Signature versão 4 com a opção --aws-sigv4. O banco de dados interno do usuário é armazenado em um OpenSearch índice, então você não pode compartilhá-lo com outros clusters.

  Recomendamos o banco de dados interno de usuários se você não precisar reutilizar usuários em vários clusters, se quiser usar a autenticação básica HTTP para acessar o Dashboards (em vez do Amazon Cognito) ou se você tiver clientes que sejam compatíveis somente com a autenticação básica. O banco de dados interno do usuário é a maneira mais simples de começar a usar o OpenSearch Service.

Habilitar o controle de acesso detalhado

Ative o controle de acesso refinado usando o console ou a API de AWS CLI configuração. Para obter as etapas, consulte Criação e gerenciamento de domínios.

O controle de acesso refinado requer o Elasticsearch OpenSearch 6.7 ou posterior. Também requer HTTPS para todo o tráfego para o domínio, criptografia de dados em repouso e node-to-node criptografia. Dependendo de como você configura os atributos avançados do controle de acesso detalhado, o processamento adicional de suas solicitações pode exigir atributos de computação e memória em nós de dados individuais. Depois que habilitar o controle de acesso refinado, não será possível desabilitá-lo.

Habilitação do controle de acesso refinado em domínios existentes

Você pode habilitar um controle de acesso refinado em domínios existentes em execução no Elasticsearch 6.7 OpenSearch ou posterior.

Para habilitar o controle de acesso refinado em um domínio existente (console)

1. Selecione o seu domínio e escolha Ações e, depois, Editar configurações de segurança.
2. Selecione Habilitar o controle de acesso refinado.
3. Escolha como criar o usuário primário:
   - Se você quiser usar o IAM para o gerenciamento de usuários, escolha Definir ARN do IAM como usuário primário e especifique o ARN para uma função do IAM.
   - Se quiser usar o banco de dados de usuário interno, escolha Criar usuário primário e especifique um nome de usuário e senha.
4. (Opcional) Selecione Habilitar o período de migração para política de acesso aberto/baseado em IP. Essa configuração viabiliza um período de transição de 30 dias durante o qual os usuários existentes podem continuar acessando o domínio sem interrupções e as políticas de acesso baseado em IP e aberto existentes continuarão funcionando com o seu domínio. Durante esse período de migração, recomendamos que os administradores criem as funções necessárias e as mapeiem para os usuários para o domínio. Se você usar políticas baseadas em identidade, em vez de uma política de acesso aberto ou baseado em IP, será possível desabilitar essa configuração.

   Você também precisa atualizar os seus clientes para trabalhar com controle de acesso refinado durante o período de migração. Por exemplo, se você mapear funções do IAM com controle de acesso refinado, você deve atualizar seus clientes para começar a assinar solicitações com o AWS Signature versão 4. Se você configurar a autenticação básica de HTTP com controle de acesso refinado, deverá atualizar os seus clientes para fornecer as credenciais de autenticação básicas apropriadas nas solicitações.

   Durante o período de migração, os usuários que acessarem o endpoint do OpenSearch Dashboards do domínio acessarão diretamente a página Discover em vez da página de login. Os administradores e usuários primários podem escolher Login para fazer login com credenciais de administrador e configurar mapeamentos de funções.

> **Important**
>
> OpenSearch O serviço desativa automaticamente o período de migração após 30 dias. Recomendamos encerrá-lo assim que você criar as funções necessárias e mapeá-las para os usuários. Após o término do período de migração, não será possível habilitá-lo novamente.

5. Escolha Salvar alterações.

A alteração aciona uma implantação azul-verde durante a qual a integridade do cluster fica vermelha, mas todas as operações do cluster permanecem inalteradas.

Para habilitar o controle de acesso refinado em um domínio existente (CLI)

Configure `AnonymousAuthEnabled` como `true` para habilitar o período de migração com controle de acesso refinado:

```
aws opensearch update-domain-config --domain-name test-domain --region us-east-1 \
      --advanced-security-options '{ "Enabled": true,
 "InternalUserDatabaseEnabled":true, "MasterUserOptions": {"MasterUserName":"master-
username","MasterUserPassword":"master-password"},"AnonymousAuthEnabled": true}'
```

Sobre o default_role

O controle de acesso refinado exige o mapeamento de funções. Se seu domínio usa políticas de acesso baseadas em identidade, o OpenSearch Service mapeia automaticamente seus usuários para uma nova função chamada default_role para ajudá-lo a migrar adequadamente os usuários existentes. Esse mapeamento temporário garante que os seus usuários ainda possam enviar com êxito solicitações GET e PUT assinadas pelo IAM até que você crie seus próprios mapeamentos de função.

A função não adiciona nenhuma vulnerabilidade ou falha de segurança ao seu domínio do OpenSearch Serviço. Recomendamos excluir a função padrão assim que você configurar suas próprias funções e mapeá-las adequadamente.

Cenários de migração

A tabela a seguir descreve o comportamento de cada método de autenticação antes e depois de habilitar o controle de acesso refinado em um domínio existente, assim como as etapas que os administradores devem seguir para mapear corretamente seus usuários para funções:

Método de autenticação	Antes de habilitar o controle de acesso refinado	Depois de habilitar o controle de acesso refinado	Tarefas do administrador
Políticas baseadas em identidade	Todos os usuários que cumprem a política do IAM podem acessar o domínio.	Não é necessário habilitar o período de migração. OpenSearch O serviço mapeia automaticamente todos os usuários que atendem à política do IAM para o default_role para que eles possam continuar acessando o domínio.	1. Crie mapeamentos de função personalizados no domínio. 2. Exclua a default_role.
Políticas baseadas em IP	Todos os usuários dos endereços IP ou blocos CIDR permitidos podem acessar o domínio.	Durante o período de migração de 30 dias, todos os usuários dos endereços IP ou blocos CIDR permitidos poderão continuar acessando o domínio.	1. Crie mapeamentos de função personalizados no domínio. 2. Atualize os seus clientes para fornecer credenciais de autenticação básicas ou credenciais do IAM, dependendo da sua configuração de mapeamento de função. 3. Encerre o período de migração. Os usuários dos endereços IP ou blocos CIDR permitidos enviando solicitações sem autenticação

Método de autenticação	Antes de habilitar o controle de acesso refinado	Depois de habilitar o controle de acesso refinado	Tarefas do administrador
			básica ou credenciais do IAM perderão o acesso ao domínio.
Políticas de acesso aberto	Todos os usuários na Internet podem acessar o domínio.	Durante o período de migração de 30 dias, todos os usuários na Internet podem continuar acessando o domínio.	1. Crie mapeamentos de função no domínio. 2. Atualize os seus clientes para fornecer credenciais de autentica ção básicas ou credenciais do IAM, dependendo da sua configuração de mapeamento de função. 3. Encerre o período de migração. Os usuários que enviarem solicitaç ões sem autenticação básica ou credenciais do IAM perderão o acesso ao domínio.

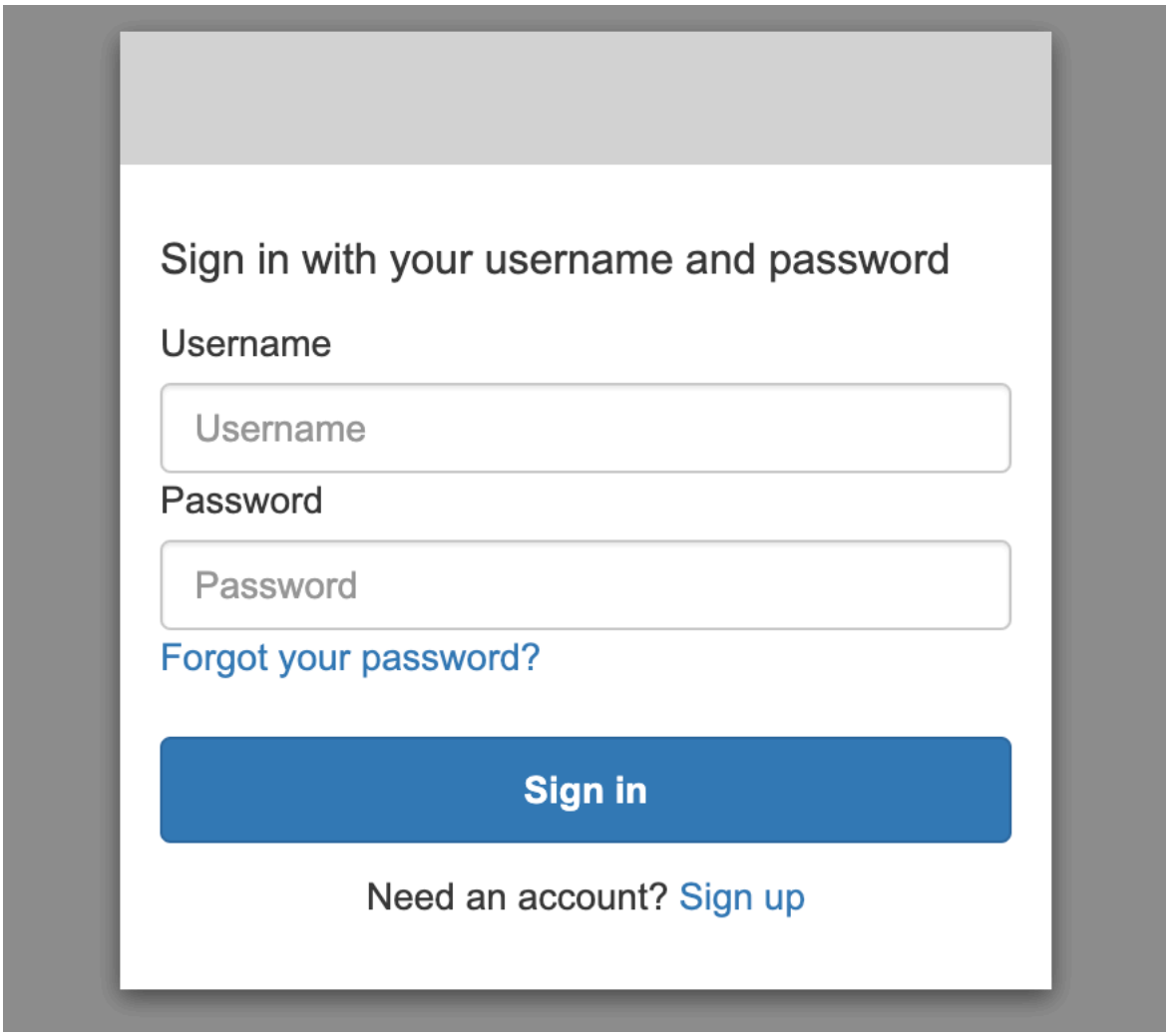
Acessando OpenSearch painéis como usuário principal

O controle de acesso refinado tem um plug-in de OpenSearch painéis que simplifica as tarefas de gerenciamento. Você pode usar o Dashboard para gerenciar usuários, funções, mapeamentos, grupos de ação e locatários. No entanto, a página de login do OpenSearch Dashboards e o método de autenticação subjacente diferem, dependendo de como você gerencia os usuários e configura seu domínio.

- Se desejar usar o IAM para o gerenciamento de usuários, use the section called “Autenticação do Amazon Cognito para painéis OpenSearch ” para acessar o Dashboards. Caso contrário, o Dashboards exibirá uma página de login não funcional. Consulte the section called “Limitações”.

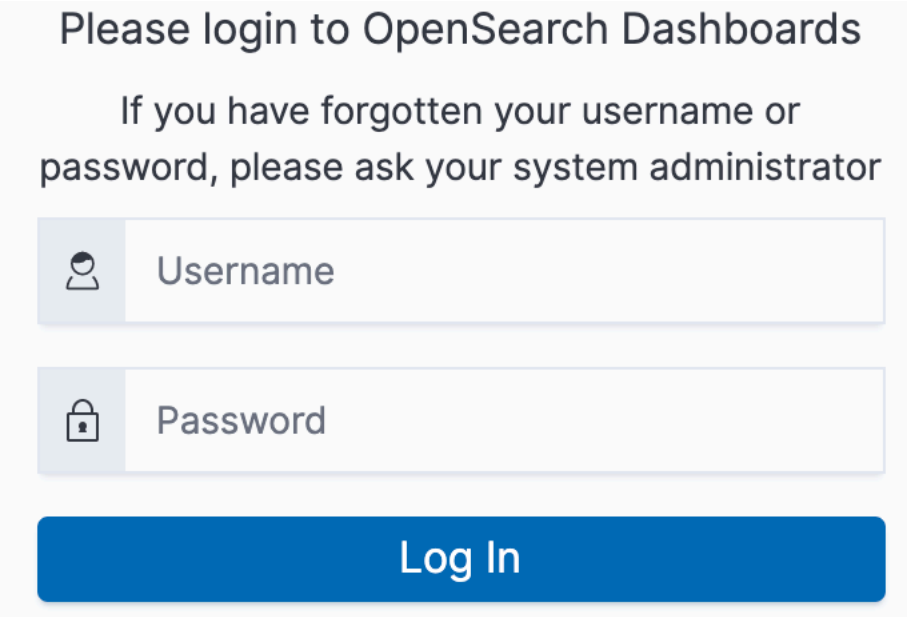
  Com a autenticação do Amazon Cognito, uma das funções assumidas do grupo de identidades deve corresponder à função do IAM especificada para o usuário primário. Para saber mais sobre

essa configuração, consulte the section called “(Opcional) Configuração de acesso granular” e the section called “Tutorial: Controle de acesso minucioso com autenticação Cognito”.



- Se você escolher usar o banco de dados de usuário interno, você pode fazer login no Painéis com seu nome de usuário primário e senha. Você deverá acessar o Dashboards via HTTPS. O Amazon Cognito e a autenticação SAML para Dashboards substituem essa tela de login.

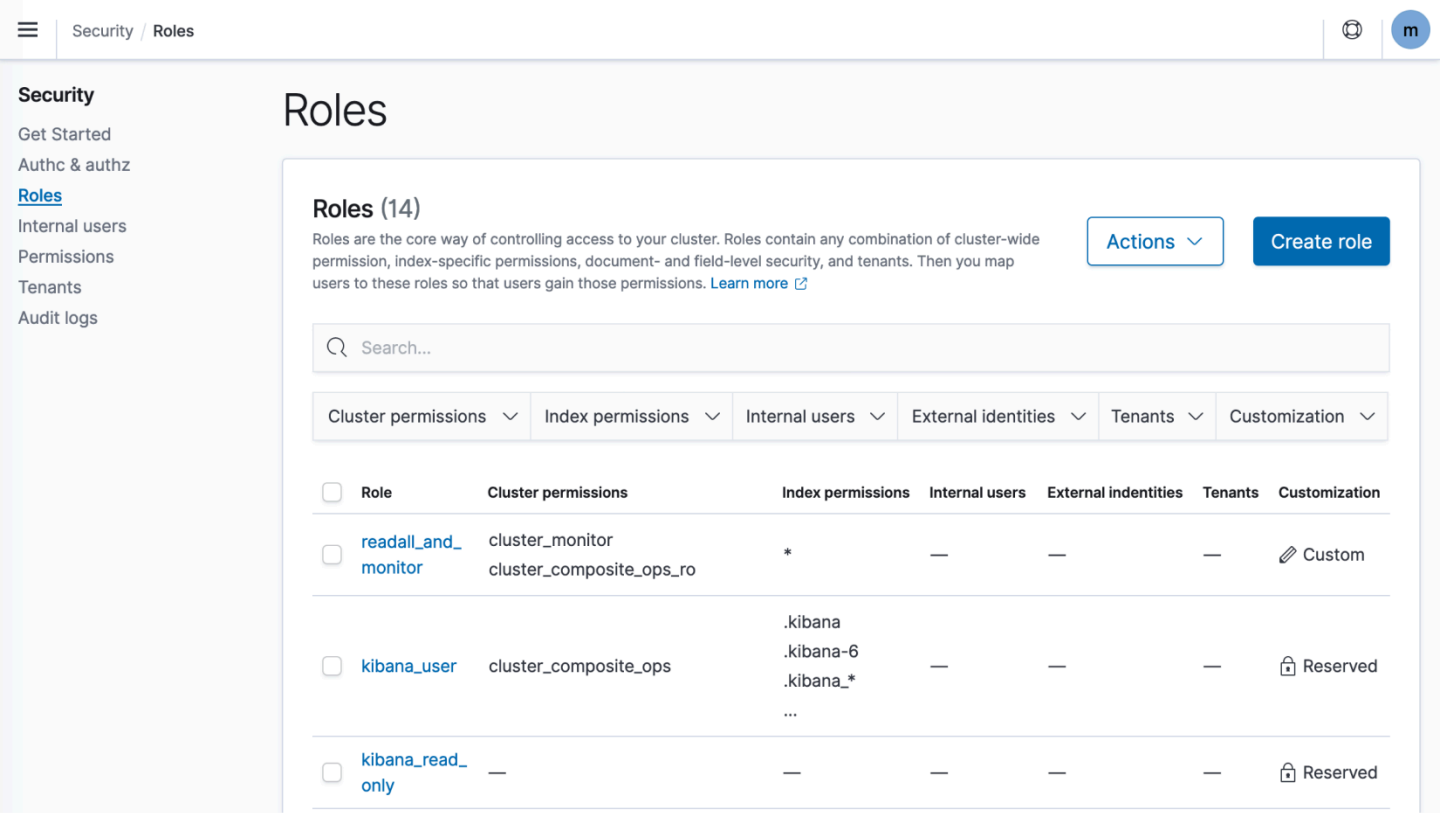
  Para saber mais sobre essa configuração, consulte the section called “Tutorial: Banco de dados interno de usuários com autenticação básica”.



- Se você optar por usar a autenticação SAML, poderá fazer login usando credenciais de um provedor de identidade externo. Para saber mais, consulte the section called “Autenticação SAML para painéis OpenSearch”.

Gerenciar permissões

Conforme observado em the section called “Principais conceitos”, você gerencia permissões de controle de acesso refinado usando funções, usuários e mapeamentos. Esta seção descreve como criar e aplicar esses recursos. Recomendamos fazer login no Dashboards como o usuário primário para executar essas operações.



> **Note**
>
> As permissões que você escolhe conceder aos usuários variam amplamente com base no caso de uso. Não podemos cobrir todos os cenários nesta documentação. Ao determinar quais permissões conceder aos seus usuários, faça referência às permissões de OpenSearch cluster e índice mencionadas nas seções a seguir e sempre siga o princípio do privilégio mínimo.

Criar funções

Você pode criar novas funções para um controle de acesso refinado usando OpenSearch painéis ou a `_plugins/_security` operação na API REST. Para saber mais, consulte Criar funções.

O controle de acesso refinado também inclui várias funções predefinidas. Clientes como OpenSearch Dashboards e Logstash fazem uma grande variedade de solicitações OpenSearch, o que pode dificultar a criação manual de funções com o conjunto mínimo de permissões. Por exemplo, a função `opensearch_dashboards_user` inclui as permissões de que um usuário precisa para criar padrões de índice, visualizações, painéis e locatários. Recomendamos mapeá-la em qualquer

função de usuário ou de backend que acesse o Dashboards, juntamente com funções adicionais que permitam o acesso a outros índices.

O Amazon OpenSearch Service não oferece as seguintes OpenSearch funções:

- `observability_full_access`
- `observability_read_access`
- `reports_read_access`
- `reports_full_access`

O Amazon OpenSearch Service oferece várias funções que não estão disponíveis com OpenSearch:

- `ultrawarm_manager`
- `ml_full_access`
- `cold_manager`
- `notifications_full_access`
- `notifications_read_access`

Segurança em nível de cluster

As permissões em nível de cluster incluem a capacidade de realizar solicitações amplas, como `_mget`, `_msearch` e `_bulk`, monitorar a integridade, obter snapshots e muito mais. Gerencie essas permissões usando a seção Permissões do cluster ao criar uma função. Para obter a lista completa das permissões no nível do cluster, consulte Permissões de cluster.

Em vez de usar permissões individuais, muitas vezes você pode alcançar a postura de segurança desejada usando uma combinação dos grupos de ação padrão. Para obter uma lista de grupos de ação no nível do cluster, consulte Nível do cluster.

Segurança em nível de índice

As permissões no nível do índice incluem a capacidade de criar novos índices, pesquisar índices, ler e escrever documentos, excluir documentos, gerenciar aliases e muito mais. Gerencie essas permissões usando a seção Permissões do índice ao criar uma função. Para obter a lista completa das permissões no nível do índice, consulte Permissões de índice.

Em vez de usar permissões individuais, muitas vezes você pode alcançar a postura de segurança desejada usando uma combinação dos grupos de ação padrão. Para obter uma lista de grupos de ação no nível do índice, consulte Nível do índice.

Segurança em nível de documento

A segurança no nível do documento permite restringir quais documentos em um índice um usuário pode ver. Ao criar uma função, especifique um padrão de índice e uma OpenSearch consulta. Qualquer usuário mapeado para essa função poderá ver somente os documentos que correspondam à consulta. A segurança no nível do documento afeta o número de ocorrências que você recebe ao pesquisar.

Para saber mais, consulte Segurança em nível de documento.

Segurança em nível de campo

A segurança no nível do campo permite controlar quais campos de documento um usuário pode ver. Ao criar uma função, adicione uma lista de campos a serem incluídos ou excluídos. Se você incluir campos, os usuários que você mapear para essa função poderão ver somente esses campos. Se você excluir campos, eles poderão ver todos os campos exceto os excluídos. A segurança no nível do campo afeta o número de campos incluídos em ocorrências ao pesquisar.

Para saber mais, consulte Segurança em nível de campo.

Mascaramento de campo

O mascaramento de campo é uma alternativa à segurança no nível do campo que permite que você torne os dados anônimos em um campo em vez de removê-los completamente. Ao criar uma função, adicione uma lista de campos a serem mascarados. O mascaramento de campo afeta a possibilidade de ver o conteúdo de um campo ao pesquisar.

> **Tip**
>
> Se você aplicar o mascaramento padrão a um campo, o OpenSearch Service usará um hash seguro e aleatório que pode causar resultados de agregação imprecisos. Para executar agregações em campos mascarados, use o mascaramento baseado em padrões.

Criação de usuários do

Se você ativou o banco de dados interno do usuário, poderá criar usuários usando OpenSearch painéis ou a `_plugins/_security` operação na API REST. Para saber mais, consulte Criar usuários.

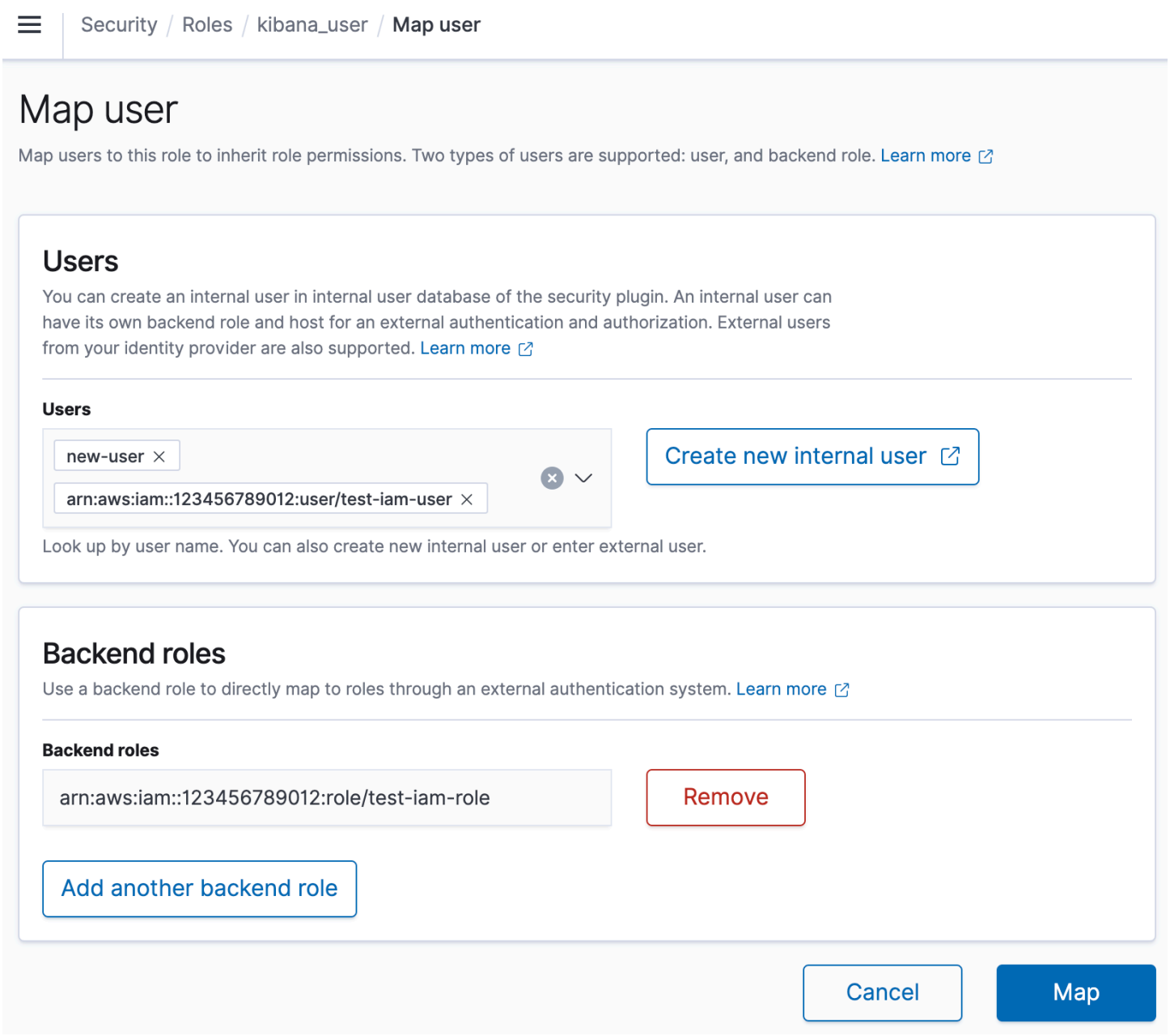
Se você escolheu o IAM para seu usuário primário, ignore esta parte do Dashboards. Crie perfis do IAM. Para saber mais, consulte o Manual do usuário do IAM.

Mapear funções em usuários

O mapeamento de função é o aspecto mais crítico do controle de acesso refinado. O controle de acesso refinado tem algumas funções predefinidas para ajudar a começar, mas a menos que você mapeie funções para os usuários, cada solicitação ao cluster terminará em um erro de permissões.

As funções de backend podem ajudar a simplificar o processo de mapeamento de funções. Em vez de mapear a mesma perfil para 100 usuários individuais, é possível mapear a perfil para uma única perfil de backend. Todos os 100 usuários compartilham. As funções de backend podem ser funções do IAM ou strings arbitrárias.

- Especifique usuários ARNs, usuários e cadeias de caracteres de usuário do Amazon Cognito na seção Usuários. As strings de usuário do Cognito assumem a forma de `Cognito/`*`user-pool-id`*`/`*`username`*.
- Especifique as funções de back-end e a função do IAM ARNs na seção Funções de back-end.



Você pode mapear funções para usuários usando OpenSearch painéis ou a `_plugins/_security` operação na API REST. Para saber mais sobre as funções de usuário, consulte Mapear usuários em funções.

Criação de grupos de ação

Grupos de ação são conjuntos de permissões que podem ser reutilizados em diferentes recursos. Você pode criar novos grupos de ação usando OpenSearch painéis ou a `_plugins/_security`

operação na API REST, embora os grupos de ação padrão sejam suficientes para a maioria dos casos de uso. Para saber mais sobre os grupos de ação padrão, consulte Grupos de ação padrão.

OpenSearch Multilocação de painéis

Locatários são espaços para salvar padrões de índice, visualizações, painéis e outros objetos do Dashboards. A locação múltipla dos Painéis permite que você compartilhe seu trabalho de forma segura com outros usuários dos Painéis (ou mantenha-o privado) e configure as locações dinamicamente. É possível controlar quais funções têm acesso a um locatário e se essas funções têm acesso de leitura ou gravação. O inquilino global é o padrão. Para saber mais, consulte Multilocação de OpenSearch painéis.

Como visualizar o locatário atual ou alterar locatários

1. Navegue até OpenSearch Painéis e faça login.
2. Selecione o ícone de usuário no canto superior direito e escolha Alternar locatários.
3. Verifique seu locatário antes de criar visualizações ou painéis. Se você deseja compartilhar seu trabalho com todos os outros usuários do Dashboards, escolha Global. Para compartilhar seu trabalho com um subconjunto de usuários do Dashboards, escolha um locatário compartilhado diferente. Caso contrário, escolha Privado.

> **Note**
>
> OpenSearch Os painéis mantêm um índice separado para cada inquilino e criam um modelo de índice chamado. `tenant_template` Não exclua nem modifique o `tenant_template` índice, pois isso pode causar mau funcionamento dos OpenSearch painéis se o mapeamento do índice do inquilino estiver configurado incorretamente.

Configurações recomendadas

Devido à forma como o controle de acesso refinadointerage com outros recursos de segurança, recomendamos várias configurações de controle de acesso refinado que funcionam bem na maioria dos casos de uso.

Description	Usuário primário	Política de acesso ao domínio

Use as credenciais do IAM para chamadas para o OpenSearch APIs e use a autenticação SAML para acessar os painéis. Gerencie funções de controle de acesso detalhado usando o Dashboards ou a API REST.	Usuário ou perfil do IAM	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		}
		]
		}

Use credenciais do IAM ou autenticação básica para chamadas para o. OpenSearch APIs Gerencie funções de controle de acesso detalhado usando o Dashboards ou a API REST.
 	
Essa configuração oferece muita flexibilidade, especialmente se você tiver OpenSearch clientes que oferecem suporte apenas à autenticação básica.
	Nome de usuário e senha	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		}
		]
		}

Description	Usuário primário	Política de acesso ao domínio
Se você tiver um provedor de identidade existente, use a Autenticação do SAML para acessar o Dashboards. Caso contrário, gerencie usuários do Dashboards no banco de dados interno de usuários.		
Use as credenciais do IAM para chamadas para o OpenSearch APIs e use o Amazon Cognito para acessar painéis. Gerencie funções de controle de acesso detalhado usando o Dashboards ou a API REST.	Usuário ou perfil do IAM	JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": "arn:aws:es: us-east-1 :111122223333 :domain/domain-name /*" } ] }`

Description	Usuário primário	Política de acesso ao domínio

Use as credenciais do IAM para chamadas para o OpenSearch APIs e bloqueie a maior parte do acesso aos painéis. Gerencie funções de controle de acesso refinado usando a API REST.	Usuário ou perfil do IAM	JSON
		{
		"Version":"2012-10-17",
		"Statement": [
		{
		"Effect": "Allow",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /*"
		},
		{
		"Effect": "Deny",
		"Principal": {
		"AWS": "*"
		},
		"Action": "es:ESHttp*",
		"Resource": "arn:aws:es: *us-east-1* :*111122223333* :domain/*domain-name* /_dashboards*"
		}
		]
		}

Limitações

O controle de acesso refinado tem várias limitações importantes:

- O aspecto `hosts` dos mapeamentos de função, que mapeia funções para os nomes de host ou endereços IP, não funcionará se o domínio estiver dentro de uma VPC. Ainda assim, é possível mapear funções para usuários e funções de backend.

- Se você escolher o IAM para o usuário primário e não habilitar a autenticação do Amazon Cognito ou SAML, o Dashboards exibirá uma página de login não funcional.
- Se você escolher o IAM para o usuário primário, ainda poderá criar usuários no banco de dados interno de usuários. No entanto, como a autenticação básica HTTP não está habilitada nesta configuração, quaisquer pedidos assinados com essas credenciais de utilizador serão rejeitados.
- Se utilizar o SQL para consultar um índice ao qual você não tenha acesso, receberá um erro “sem permissões”. Se o índice não existir, você receberá um erro “Índice não existe”. Essa diferença nas mensagens de erro significa que você pode confirmar a existência de um índice se adivinhar seu nome.

  Para minimizar o problema, não inclua informações confidenciais em nomes de índice. Para negar todo o acesso ao SQL, adicione o seguinte elemento à sua política de acesso ao domínio:

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": [
      "*"
    ]
  },
  "Action": [
    "es:*"
  ],
  "Resource": "arn:aws:es:us-east-1:123456789012:domain/my-domain/_plugins/_sql"
}
```

- Se a versão do seu domínio for 2.3 ou superior e você tiver um controle de acesso detalhado ativado, `max_clause_count` a configuração como 1 causará problemas com seu domínio. Recomendamos definir essa conta para um número maior.
- Se estiver habilitando o controle de acesso refinado em um domínio no qual o controle de acesso refinado não está configurado, para fontes de dados criadas para consulta direta, você mesmo precisará configurar os perfis de controle de acesso refinado. Para obter mais informações sobre como configurar funções de acesso refinadas, consulte Criação de integrações de fontes de dados do Amazon OpenSearch Service com o Amazon S3.

Modificação do usuário primário

Se você esquecer os detalhes do usuário primário, poderá reconfigurá-lo usando o console, a AWS CLI ou a API de configuração.

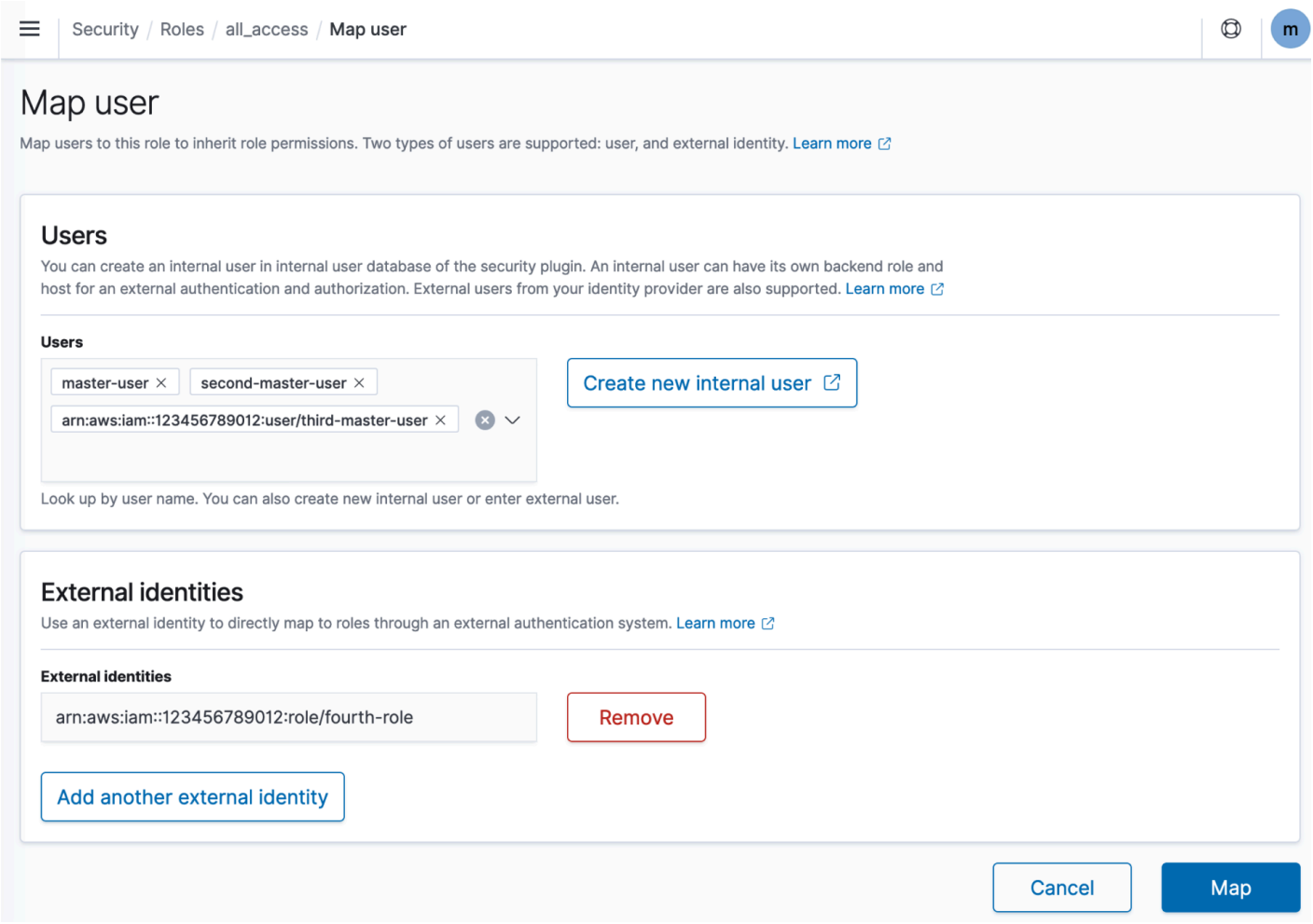
Como modificar o usuário primário (console)

1. Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/home/.
2. Escolha o seu domínio e escolha Ações, Editar configurações de segurança.
3. Escolha Definir ARN do IAM como usuário primário ou Criar novo usuário primário.
   - Se você usou anteriormente um usuário primário do IAM, o controle de acesso refinado mapeará novamente a função `all_access` para o novo ARN do IAM especificado.
   - Se você usou anteriormente o banco de dados interno de usuários, o controle de acesso refinado criará um novo usuário primário. É possível usar o novo usuário primário para excluir o antigo.
   - A mudança do banco de dados de usuário interno para um usuário primário do IAM não exclui os usuários do banco de dados interno de usuários. Em vez disso, ela apenas desabilita a autenticação básica HTTP. Exclua manualmente os usuários do banco de dados interno do usuário ou mantenha-os para o caso de precisar reativar a autenticação básica HTTP.
4. Escolha Salvar alterações.

Usuários primários adicionais

Você designa um usuário primário ao criar um domínio, mas, se desejar, pode usar esse usuário primário para criar usuários primários adicionais. Você tem duas opções: OpenSearch painéis ou a API REST.

- No Dashboards, escolha Segurança, Funções e mapeie o novo usuário primário nas funções `all_access` e `security_manager`.



- Para usar a API REST, envie as seguintes solicitações:

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

```
PUT _plugins/_security/api/rolesmapping/security_manager
{
```

```
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

Essas solicitações substituem os mapeamentos de função atuais, portanto, execute as solicitações `GET` primeiro para que você possa incluir todas as funções atuais nas solicitações `PUT`. A API REST será especialmente útil se você não conseguir acessar o Dashboards e quiser mapear uma função do IAM do Amazon Cognito na função `all_access`.

Snapshots manuais

O controle de acesso refinado apresenta algumas complicações adicionais quando são tirados snapshots manuais. Para registrar um repositório de snapshots, mesmo que use a autenticação básica HTTP para todos os outros fins, você deve mapear a função `manage_snapshots` em uma função do IAM que tenha permissões `iam:PassRole` para assumir `TheSnapshotRole`, conforme definido nos the section called “Pré-requisitos”.

Depois, use essa função do IAM para enviar uma solicitação assinada ao domínio, conforme descrito em the section called “Registro de um repositório de snapshots manuais”.

Integrações

Se você usa outros AWS serviços com o OpenSearch Service, deve fornecer as funções do IAM para esses serviços com as permissões apropriadas. Por exemplo, os fluxos de entrega do Firehose frequentemente usam um perfil do IAM denominado `firehose_delivery_role`. No Dashboards, crie uma função para o controle de acesso refinado e mapeie a função do IAM nela. Nesse caso, a nova função precisará das seguintes permissões:

```
{
  "cluster_permissions": [
    "cluster_composite_ops",
    "cluster_monitor"
  ],
```

```
  "index_permissions": [{
    "index_patterns": [
      "firehose-index*"
    ],
    "allowed_actions": [
      "create_index",
      "manage",
      "crud"
    ]
  }]
}
```

As permissões variam de acordo com as ações que cada serviço executa. Uma AWS IoT regra ou AWS Lambda função que indexa dados provavelmente precisa de permissões semelhantes às do Firehose, enquanto uma função Lambda que só realiza pesquisas pode usar um conjunto mais limitado.

Diferenças de API REST

A API REST do controle de acesso minucioso difere ligeiramente dependendo da sua versão do OpenSearch/Elasticsearch . Antes de fazer uma solicitação PUT, faça uma solicitação `GET` para verificar o corpo da solicitação esperada. Por exemplo, uma solicitação `GET` para `_plugins/_security/api/user` retornar todos os usuários, que poderá ser modificada e usada para fazer solicitações PUT válidas.

No Elasticsearch 6.x, as solicitações para criar usuários são semelhantes a:

```
PUT _opendistro/_security/api/user/new-user
{
  "password": "some-password",
  "roles": ["new-backend-role"]
}
```

No OpenSearch Elasticsearch 7.x, as solicitações têm a seguinte aparência (altere `_plugins` para `_opendistro` se estiver usando o Elasticsearch):

```
PUT _plugins/_security/api/user/new-user
{
  "password": "some-password",
  "backend_roles": ["new-backend-role"]
}
```

Além disso, os locatários são propriedades de funções no Elasticsearch 6.x:

```
GET _opendistro/_security/api/roles/all_access

{
  "all_access": {
    "cluster": ["UNLIMITED"],
    "tenants": {
      "admin_tenant": "RW"
    },
    "indices": {
      "*": {
        "*": ["UNLIMITED"]
      }
    },
    "readonly": "true"
  }
}
```

No OpenSearch Elasticsearch 7.x, eles são objetos com seu próprio URI (altere `_plugins` para `_opendistro` se estiver usando o Elasticsearch):

```
GET _plugins/_security/api/tenants

{
  "global_tenant": {
    "reserved": true,
    "hidden": false,
    "description": "Global tenant",
    "static": false
  }
}
```

Para obter a documentação sobre a API OpenSearch REST, consulte a referência da API do plug-in de segurança.

> **Tip**
>
> Se usar o banco de dados de usuário interno, você poderá usar curl para fazer solicitações e testar seu domínio. Teste os seguintes comandos de exemplo:
>
> ```
> curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_search'
> ```

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_plugins/
_security/api/user'
```

Tutorial: Configuração de um domínio com um usuário primário do IAM e autenticação do Amazon Cognito

Este tutorial aborda um caso de uso popular do Amazon OpenSearch Service para controle de acesso refinado: um usuário mestre do IAM com autenticação do Amazon Cognito para painéis. OpenSearch

No tutorial, configuraremos um perfil do IAM principal e um perfil do IAM limitado, que depois associaremos aos usuários no Amazon Cognito. O usuário principal pode então entrar nos OpenSearch painéis, mapear o usuário limitado para uma função e usar um controle de acesso refinado para limitar as permissões do usuário.



Embora essas etapas usem o grupo de usuários do Amazon Cognito para a autenticação, esse mesmo processo básico funciona para qualquer provedor de autenticação do Cognito que permita atribuir diferentes funções do IAM a usuários diferentes.

Você concluirá as seguintes etapas neste tutorial:

1. Criar perfis do IAM principais e limitado
2. Criar um domínio com a autenticação Cognito
3. Configurar um grupo de usuários e um banco de identidades do Cognito
4. Mapeie funções em OpenSearch painéis
5. Testas as permissões

Etapa 1: Criar perfis do IAM principal e limitado

Navegue até o console AWS Identity and Access Management (IAM) e crie duas funções separadas:

- `MasterUserRole`: o usuário primário, que terá permissões completas para o cluster e gerencia funções e mapeamentos de função.
- `LimitedUserRole`: um perfil mais restrito, à qual você concederá acesso limitado como usuário primário.

Para receber instruções passo a passo para criar perfis, consulte Criar um perfil usando políticas de confiança personalizadas no Guia do usuário do IAM.

Ambos os perfis devem ter a política de confiança a seguir, que permite que seu grupo de identidades do Cognito assuma os perfis:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Federated": "cognito-identity.amazonaws.com"
    },
    "Action": "sts:AssumeRoleWithWebIdentity",
    "Condition": {
      "StringEquals": {
        "cognito-identity.amazonaws.com:aud": "{identity-pool-id}"
      },
      "ForAnyValue:StringLike": {
        "cognito-identity.amazonaws.com:amr": "authenticated"
      }
    }
  }]
}
```

> **Note**
>
> Substitua `identity-pool-id` pelo identificador exclusivo do seu grupo de identidades do Amazon Cognito. Por exemplo, `.us-east-1:0c6cdba7-3c3c-443b-a958-fb9feb207aa6`

Etapa 2: Criar um domínio com a autenticação Cognito

Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa/ e crie um domínio com as seguintes configurações:

- OpenSearch 1.0 ou posterior, ou Elasticsearch 7.8 ou posterior
- Acesso público
- Controle de acesso minucioso habilitado com `MasterUserRole` como usuário primário (criado na etapa anterior)
- Autenticação do Amazon Cognito habilitada para painéis OpenSearch . Para obter instruções para habilitar a autenticação do Cognito e selecionar um usuário e um grupo de identidades, consulte the section called “Configuração de um domínio para uso da autenticação do Amazon Cognito”.
- A seguinte política de acesso ao domínio:

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/{domain-name}/*"
    }
  ]
}
```

- HTTPS necessário para todo o tráfego para o domínio

- Node-to-node criptografia
- Criptografia de dados em repouso

Etapa 3: Configurar usuários do Cognito

Enquanto seu domínio estiver sendo criado, configure os usuários principal e limitado no Amazon Cognito seguindo Criar um grupo de usuários no Guia do desenvolvedor do Amazon Cognito. Por fim, configure seu banco de identidades seguindo as etapas em Criar um grupo de identidades no Amazon Cognito. Os grupos de usuários e identidades devem estar na mesma Região da AWS.

Etapa 4: mapear funções em OpenSearch painéis

Agora que seus usuários estão configurados, você pode entrar no OpenSearch Dashboards como usuário principal e mapear usuários para funções.

1. Volte para o console OpenSearch de serviços e navegue até a URL dos OpenSearch painéis do domínio que você criou. O URL segue este formato: *`domain-endpoint`*`/_dashboards/`.
2. Faça login com as credenciais `master-user`.
3. Escolha Adicionar dados de amostras e adicione os dados de voo de amostra.
4. No painel de navegação à esquerda, escolha Segurança, Funções, Criar função.
5. Nomeie a função `new-role`.
6. Em Índice, especifique `opensearch_dashboards_sample_data_fli*` (`kibana_sample_data_fli*` nos domínios do Elasticsearch).
7. Em Permissões de índice, escolha ler.
8. Em Segurança em nível de documento, especifique a seguinte consulta:

```
{
  "match": {
    "FlightDelay": true
  }
}
```

9. Para segurança em nível de campo, escolha Excluir e especifique `FlightNum`.
10. Em Anonimização, especifique `Dest`.
11. Escolha Criar.

12. Escolha Usuários mapeados e Gerenciar mapeamento. Adicione o nome do recurso da Amazon (ARN) para `LimitedUserRole` como uma identidade externa e escolha Mapa.
13. Retorne à lista de funções e escolha opensearch_dashboards_user. Escolha Usuários mapeados e Gerenciar mapeamento. Adicione o ARN para `LimitedUserRole` como uma função de backend e escolha Mapa.

Etapa 5: Testar as permissões

Quando suas funções estiverem mapeadas corretamente, é possível fazer login como o usuário limitado e testá-las.

1. Em uma nova janela privada do navegador, navegue até o URL dos OpenSearch painéis do domínio, faça login usando `limited-user` as credenciais e escolha Explorar sozinho.
2. Escolha Ferramentas de desenvolvimento e execute a pesquisa padrão:

   ```
   GET _search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

   Observe o erro de permissões. `limited-user` não tem permissões para executar pesquisas em todo o cluster.
3. Execute outra pesquisa:

   ```
   GET opensearch_dashboards_sample_data_flights/_search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

   Observe que todos os documentos correspondentes têm um campo `FlightDelay` de `true`, um campo anônimo `Dest` e nenhum campo `FlightNum`.
4. Na janela original do navegador, conectado como `master-user`, escolha Ferramentas de desenvolvimento e execute as mesmas pesquisas. Observe a diferença nas permissões, número de ocorrências, documentos correspondentes e campos incluídos.

Tutorial: Configuração de um domínio com o banco de dados interno do usuário e a autenticação básica HTTP

Este tutorial aborda outro caso de uso de controle de acesso refinado e popular: um usuário principal no banco de dados de usuários interno e autenticação básica HTTP para painéis. OpenSearch O usuário principal pode então entrar nos OpenSearch painéis, criar um usuário interno, mapear o usuário para uma função e usar um controle de acesso refinado para limitar as permissões do usuário.

Você concluirá as seguintes etapas neste tutorial:

1. Crie um domínio com um usuário primário
2. Configurar um usuário interno nos OpenSearch painéis
3. Mapeie funções em OpenSearch painéis
4. Testas as permissões

Etapa 1: Criar um domínio

Navegue até o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa/ e crie um domínio com as seguintes configurações:

- OpenSearch 1.0 ou posterior, ou Elasticsearch 7.9 ou posterior
- Acesso público
- Controle de acesso refinado com um usuário primário no banco de dados interno de usuários (`TheMasterUser` para o restante deste tutorial)
- Autenticação do Amazon Cognito para Dashboards desabilitada
- A seguinte política de acesso:

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
```

```
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/{domain-name}/*"
    }
  ]
}
```

- HTTPS necessário para todo o tráfego para o domínio
- Node-to-node criptografia
- Criptografia de dados em repouso

Etapa 2: criar um usuário interno nos OpenSearch painéis

Agora que você tem um domínio, pode entrar no OpenSearch Dashboards e criar um usuário interno.

1. Volte para o console OpenSearch de serviços e navegue até a URL dos OpenSearch painéis do domínio que você criou. O URL segue este formato: *domain-endpoint*`/_dashboards/`.
2. Faça login com o `TheMasterUser`.
3. Escolha Adicionar dados de amostras e adicione os dados de voo de amostra.
4. No painel de navegação à esquerda, escolha Segurança, Usuários internos, Criar usuário interno.
5. Nomeie o usuário `new-user` e especifique uma senha. Em seguida, escolha Criar.

Etapa 3: mapear funções em OpenSearch painéis

Agora que seu usuário está configurado, você pode mapeá-lo para uma perfil.

1. Fique na seção Segurança dos OpenSearch Painéis e escolha Funções, Criar função.
2. Nomeie a função `new-role`.
3. Em Índice, especifique `opensearch_dashboards_sample_data_fli*`(`kibana_sample_data_fli*` nos domínios do Elasticsearch) para o padrão de índice.
4. Para o grupo de ações, escolha leitura.
5. Em Segurança em nível de documento, especifique a seguinte consulta:

```
{
  "match": {
    "FlightDelay": true
  }
}
```

6. Para segurança em nível de campo, escolha Excluir e especifique `FlightNum`.
7. Em Anonimização, especifique `Dest`.
8. Escolha Criar.
9. Escolha Usuários mapeados e Gerenciar mapeamento. Em seguida, adicione `new-user` a Usuários e escolha Mapa.
10. Retorne à lista de funções e escolha opensearch_dashboards_user. Escolha Usuários mapeados e Gerenciar mapeamento. Em seguida, adicione `new-user` a Usuários e escolha Mapa.

Etapa 4: Testar as permissões

Quando suas funções estiverem mapeadas corretamente, é possível fazer login como o usuário limitado e testá-las.

1. Em uma nova janela privada do navegador, navegue até o URL dos OpenSearch painéis do domínio, faça login usando `new-user` as credenciais e escolha Explorar sozinho.
2. Escolha Ferramentas de desenvolvimento e execute a pesquisa padrão:

```
GET _search
{
  "query": {
    "match_all": {}
  }
}
```

   Observe o erro de permissões. `new-user` não tem permissões para executar pesquisas em todo o cluster.

3. Execute outra pesquisa:

```
GET dashboards_sample_data_flights/_search
```

```
{
  "query": {
    "match_all": {}
  }
}
```

Observe que todos os documentos correspondentes têm um campo `FlightDelay` de `true`, um campo anônimo `Dest` e nenhum campo `FlightNum`.

4. Na janela original do navegador, conectado como `TheMasterUser`, escolha Ferramentas de desenvolvimento e execute as mesmas pesquisas. Observe a diferença nas permissões, número de ocorrências, documentos correspondentes e campos incluídos.

Validação de conformidade para o Amazon OpenSearch Service

Auditores terceirizados avaliam a segurança e a conformidade do Amazon OpenSearch Service como parte de vários programas de AWS conformidade. Esses programas incluem SOC, PCI e HIPAA.

Se você tiver requisitos de conformidade, considere usar qualquer versão do Elasticsearch 6.0 OpenSearch ou posterior. As versões anteriores do Elasticsearch não oferecem uma combinação de criptografia de dados em repouso e node-to-node criptografia e é improvável que atendam às suas necessidades. Você também pode considerar usar qualquer versão do Elasticsearch 6.7 OpenSearch ou posterior se o controle de acesso refinado for importante para seu caso de uso. Independentemente disso, escolher uma versão específica OpenSearch ou do Elasticsearch ao criar um domínio não garante a conformidade.

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a documentação AWS de segurança.

Resiliência no Amazon Service OpenSearch

A infraestrutura AWS global é construída em torno Regiões da AWS de zonas de disponibilidade. Regiões da AWS fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre zonas de disponibilidade Regiões da AWS e zonas de disponibilidade, consulte Infraestrutura AWS global.

Além da infraestrutura AWS global, o OpenSearch Service oferece vários recursos para ajudar a suportar suas necessidades de resiliência e backup de dados:

- Domínios e estilhaços de réplica Multi-AZ
- Snapshots automatizados e manuais

Autenticação e autorização JWT para Amazon Service OpenSearch

O Amazon OpenSearch Service agora permite que você use JSON Web Tokens (JWTs) para autenticação e autorização. JWTs são tokens de acesso baseados em JSON usados para conceder acesso de login único (SSO). Você pode usar o JWTs in OpenSearch Service para criar tokens de login único para validar solicitações para seu OpenSearch domínio do Service. Para usar JWTs, você deve ter um controle de acesso refinado ativado e fornecer uma chave pública válida formatada em RSA ou ECDSA PEM. Para obter mais informações sobre controle de acesso refinado, consulte Controle de acesso refinado no Amazon Service. OpenSearch

Você pode configurar JSON Web Tokens usando o console de OpenSearch serviço, o AWS Command Line Interface (AWS CLI) ou o. AWS SDKs

Considerações

Antes de usar JWTs com o Amazon OpenSearch Service, você deve considerar o seguinte:

- Devido ao tamanho das chaves públicas RSA na formatação PEM, recomendamos usar o console da AWS para configurar a autenticação e autorização do JWT.
- Você deve fornecer usuários e funções válidos ao especificar os campos de assuntos e funções para seus JWTs, caso contrário, as solicitações serão negadas.
- OpenSearch 2.11 é a versão compatível mais antiga que pode ser usada para autenticação JWT.

Modificar a política de acesso ao domínio

Antes de poder configurar a autenticação e autorização do JWT, é necessário atualizar sua política de acesso ao domínio para permitir que os usuários do JWT acessem o domínio. Caso contrário, todas as solicitações autorizadas do JWT recebidas serão negadas. A política de acesso ao domínio recomendada para fornecer acesso total aos subrecursos (/*) é:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
      {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Configurar autenticação e autorização do JWT

Você pode ativar a autenticação e autorização do JWT durante o processo de criação do domínio ou atualizando um domínio existente. As etapas de configuração variam um pouco, dependendo de qual opção você escolher.

As etapas a seguir explicam como configurar um domínio existente para autenticação e autorização do JWT no console OpenSearch de serviço:

1. Em Configuração do domínio, navegue até Autenticação e autorização do JWT para OpenSearch, selecione Habilitar autenticação e autorização do JWT.
2. Configure a chave pública a ser usada em seu domínio. Para fazer isso, você pode carregar um arquivo PEM contendo uma chave pública ou inseri-lo manualmente.

   > **Note**
   >
   > Se a chave enviada ou inserida não for válida, um aviso aparecerá acima da caixa de texto especificando o problema.

3. (Opcional) Em Configurações adicionais, você pode definir os seguintes campos opcionais

   - Chave de assunto — você pode deixar esse campo vazio para usar a `sub` chave padrão para seu JWTs.
   - Chave de funções — você pode deixar esse campo vazio para usar a `roles` chave padrão para sua JWTs.

   Depois de fazer as alterações, salve o seu domínio.

Usar um JWT para enviar uma solicitação de teste

Depois de criar um novo JWT com um par específico de assunto e perfil, você pode enviar uma solicitação de teste. Para fazer isso, use a chave privada para assinar sua solicitação por meio da ferramenta que criou o JWT. OpenSearch O serviço é capaz de validar a solicitação recebida verificando essa assinatura.

> **Note**
>
> Se você especificou uma chave de assunto ou chave de funções personalizada para seu JWT, você deve usar os nomes de reivindicações corretos para seu JWT.

Veja a seguir um exemplo de como usar um token JWT para acessar o OpenSearch Serviço por meio do endpoint de pesquisa do seu domínio:

```
curl -XGET "$search_endpoint" -H "Authorization: Bearer <JWT>"
```

Configurar a autenticação e autorização do JWT (AWS CLI)

O AWS CLI comando a seguir ativa a autenticação e autorização do JWT, OpenSearch desde que o domínio exista:

```
aws opensearch update-domain-config --domain-name <your_domain_name> --advanced-
security-options '{"JWTOptions":{"Enabled":true, "PublicKey": "<your_public_key>",
 "SubjectKey": "<your_subject_key>", "RolesKey": "<your_roles_key>"}}'
```

Configurar a autenticação e autorização do JWT (configuração via API)

A solicitação a seguir para a API de configuração permite a autenticação e autorização do JWT OpenSearch em um domínio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "JWTOptions": {
      "Enabled": true,
      "PublicKey": "public-key",
      "RolesKey": "optional-roles-key",
      "SubjectKey": "optional-subject-key"
   }
  }
}
```

Gerar um par de chaves

JWTs Para configurar seu OpenSearch domínio, você precisará fornecer uma chave pública no formato Privacy-Enhanced Mail (PEM). Atualmente, o Amazon OpenSearch Service oferece suporte a dois algoritmos de criptografia assimétrica ao usar JWTs: RSA e ECDSA.

Para criar um par de chaves RSA usando a biblioteca comum do openssl, siga estas etapas:

1. `openssl genrsa -out privatekey.pem 2048`
2. `openssl rsa -in privatekey.pem -pubout -out publickey.pem`

Neste exemplo, o `publickey.pem` arquivo contém a chave pública para uso com o Amazon OpenSearch Service, enquanto `privatekey.pem` contém a privada para assinar o JWTs envio para o serviço. Além disso, você tem a opção de converter a chave privada no `pkcs8` formato comumente usado, se precisar dela para gerar sua JWTs.

Se você usar o botão de upload para adicionar um arquivo PEM diretamente ao console, o arquivo deverá ter uma extensão `.pem`, outras extensões de arquivo, como `.crt`, `.cert` ou `.key` não têm suporte no momento.

Segurança de infraestrutura no Amazon OpenSearch Service

Como serviço gerenciado, é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte AWS Cloud Security. Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte Proteção de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Você usa chamadas de API AWS publicadas para acessar a API OpenSearch de configuração do serviço por meio da rede. Para configurar a versão mínima necessária do TLS para aceitar, especifique o valor `TLSSecurityPolicy` nas opções do endpoint do domínio:

```
aws opensearch update-domain-config --domain-name my-domain --domain-endpoint-options
 '{"TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"}'
```

Para obter detalhes, consulte a Referência de comandos da AWS CLI.

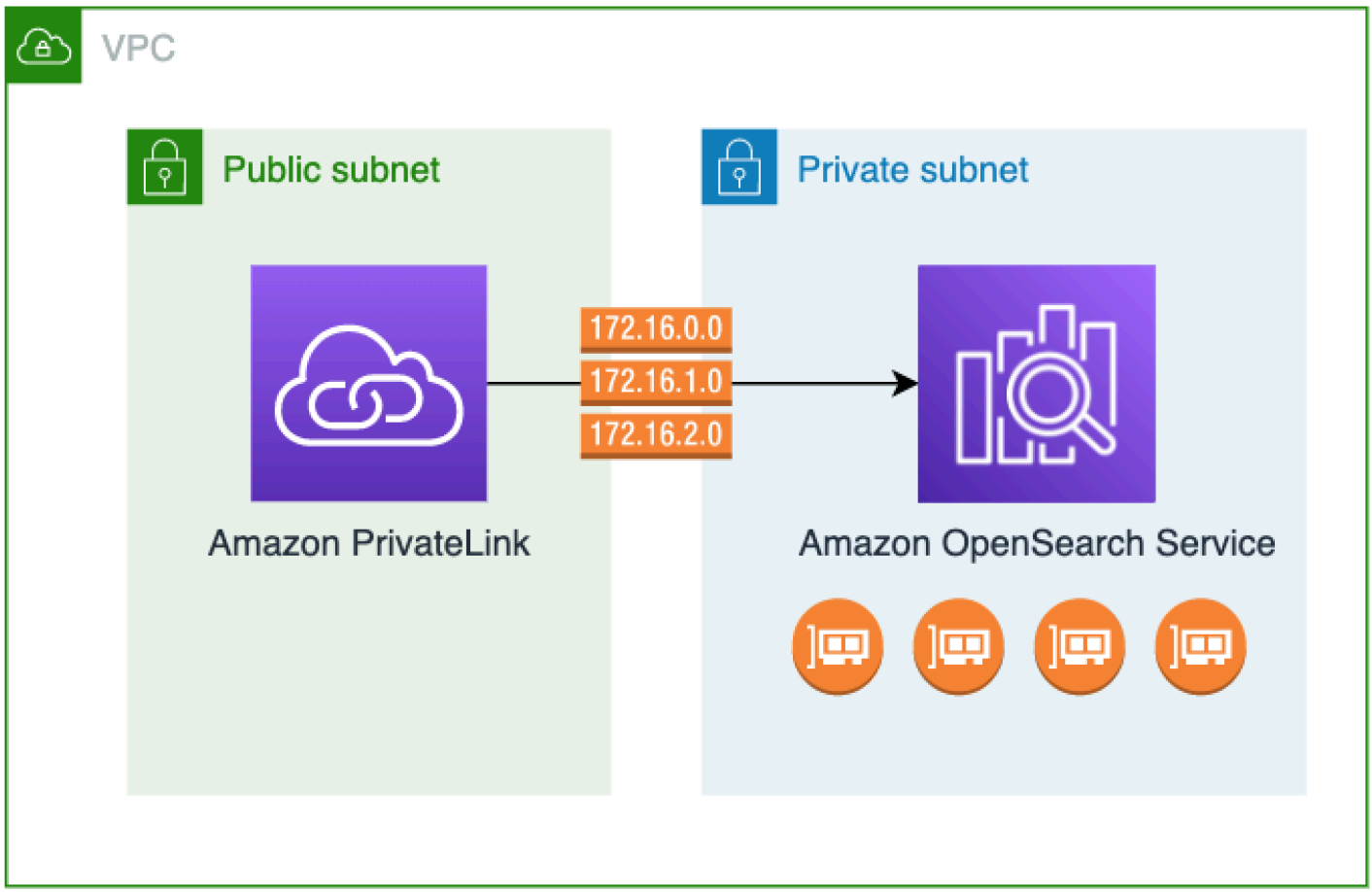
Dependendo da configuração do seu domínio, talvez você também precise assinar solicitações para OpenSearch APIs o. Para obter mais informações, consulte the section called “Fazendo e assinando solicitações OpenSearch de serviço”.

OpenSearch O serviço oferece suporte a domínios de acesso público, que podem receber solicitações de qualquer dispositivo conectado à Internet, e domínios de acesso VPC, que são isolados da Internet pública.

Acesse o Amazon OpenSearch Service usando um OpenSearch VPC endpoint gerenciado por serviços ()AWS PrivateLink

Você pode acessar um domínio do Amazon OpenSearch Service configurando um OpenSearch VPC endpoint gerenciado pelo serviço (desenvolvido por). AWS PrivateLink Esses endpoints criam uma conexão privada entre sua VPC e o Amazon OpenSearch Service. Você pode acessar os domínios do OpenSearch Service VPC como se estivessem em sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão. Direct Connect As instâncias em sua VPC não precisam de endereços IP públicos para acessar OpenSearch o Serviço.

Você pode configurar domínios OpenSearch de serviço para expor endpoints adicionais em execução em sub-redes públicas ou privadas dentro da mesma VPC, VPC diferente ou diferente. Contas da AWS Isso permite que você adicione uma camada adicional de segurança para acessar seus domínios, independentemente de onde eles sejam executados, sem nenhuma infraestrutura para gerenciar. O diagrama a seguir ilustra os endpoints de VPC OpenSearch gerenciados por serviços dentro da mesma VPC:



Você estabelece essa conexão privada criando um endpoint VPC OpenSearch de interface gerenciada por serviços, desenvolvido por. AWS PrivateLink Criaremos uma interface de rede de endpoint em cada sub-rede que você habilitar para o endpoint da VPC de interface. Essas são interfaces de rede gerenciadas por serviços que servem como ponto de entrada para o tráfego destinado ao Serviço. OpenSearch O preço padrão AWS PrivateLink do endpoint de interface se aplica aos endpoints VPC gerenciados por OpenSearch serviços cobrados abaixo. AWS PrivateLink

Você pode criar VPC endpoints para domínios que executam todas as versões do Elasticsearch legado e do OpenSearch Elasticsearch. Para saber mais, consulte Acessar os Serviços da AWS pelo AWS PrivateLink no Guia do AWS PrivateLink .

Considerações e limitações do serviço OpenSearch

Antes de configurar uma interface VPC endpoint para OpenSearch Service, consulte Acessar um AWS serviço usando uma interface VPC endpoint no Guia.AWS PrivateLink

Ao usar OpenSearch VPC endpoints gerenciados por serviços, considere o seguinte:

- É possível apenas usar endpoints da VPC de interface para se conectar a domínios da VPC. Não há suporte para domínios públicos.
- Os endpoints da VPC só podem se conectar a domínios dentro da mesma Região da AWS.
- O HTTPS é o único protocolo com suporte para endpoints da VPC. O HTTP não é permitido.
- OpenSearch O serviço oferece suporte para fazer chamadas para todas as operações de OpenSearch API suportadas por meio de uma interface VPC endpoint.
- É possível configurar no máximo 50 endpoints por conta, e no máximo 10 endpoints por domínio. Um único domínio pode ter no máximo 10 entidades principais autorizadas.
- No momento, você não pode usar AWS CloudFormation para criar endpoints VPC de interface.
- Você só pode criar endpoints VPC de interface por meio do console de OpenSearch serviço ou usando a OpenSearch API de serviço. Você não pode criar endpoints VPC de interface para OpenSearch serviço usando o console Amazon VPC.
- OpenSearch Os VPC endpoints gerenciados por serviços não podem ser acessados pela Internet. Um OpenSearch VPC endpoint gerenciado por serviços pode ser acessado somente dentro da VPC em que o endpoint é provisionado ou em qualquer ponto emparelhado VPCs com a VPC em que o endpoint é provisionado, conforme permitido pelas tabelas de rotas e grupos de segurança.
- As políticas de VPC endpoint não são compatíveis com o Service. OpenSearch Você pode associar um grupo de segurança às interfaces de rede do endpoint para controlar o tráfego para o OpenSearch serviço por meio da interface VPC endpoint.

- Sua função vinculada ao serviço deve estar na mesma AWS conta que você usa para criar o VPC endpoint.
- Para criar, atualizar e excluir o endpoint OpenSearch Service VPC, você deve ter as seguintes permissões do Amazon EC2, além das permissões do Amazon Service: OpenSearch
  - `ec2:CreateVpcEndpoint`
  - `ec2:DescribeVpcEndpoints`
  - `ec2:ModifyVpcEndpoint`
  - `ec2:DeleteVpcEndpoints`
  - `ec2:CreateTags`
  - `ec2:DescribeTags`
  - `ec2:DescribeSubnets`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeVpcs`

> **Note**
>
> Atualmente, você não pode limitar a criação de VPC endpoints ao Service. OpenSearch Estamos trabalhando para tornar isso possível em uma atualização futura.

Fornecimento de acesso a um domínio

Se a VPC que você deseja acessar seu domínio estiver em outra Conta da AWS, você precisará autorizá-la na conta do proprietário antes de criar uma interface VPC endpoint.

Para permitir que uma VPC em outra Conta da AWS acesse seu domínio

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa/.
2. No painel de navegação, escolha Domínios e abra o domínio ao qual você deseja fornecer acesso.
3. Acesse a guia VPC endpoints, que mostra as contas e os correspondentes VPCs que têm acesso ao seu domínio.
4. Escolha Autorizar entidade principal.

5. Insira o Conta da AWS ID da conta que acessará seu domínio. Essa etapa autoriza a conta especificada a criar endpoints da VPC no domínio.
6. Escolha Authorize.

Criação de uma endpoint da VPC de interface para um domínio de VPC

Você pode criar uma interface VPC endpoint para OpenSearch Service usando o console OpenSearch Service ou o AWS Command Line Interface ().AWS CLI

Para criar uma interface VPC endpoint para um domínio de serviço OpenSearch

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa/.
2. No painel de navegação à esquerda, escolha Endpoints da VPC.
3. Escolha Criar endpoint.
4. Selecione se deseja conectar um domínio no atual Conta da AWS ou em outro Conta da AWS.
5. Selecione o domínio ao qual você se conecta com esse endpoint. Se o domínio estiver no atual Conta da AWS, use o menu suspenso para escolher o domínio. Se o domínio estiver em uma conta diferente, insira o nome do recurso da Amazon (ARN) do domínio ao qual você deseja se conectar. Para escolher um domínio em uma conta diferente, o proprietário precisa fornecer acesso ao domínio.
6. Para VPC, selecione a VPC a partir da qual você acessará o Serviço. OpenSearch
7. Para Sub-redes, selecione uma ou mais sub-redes a partir das quais você acessará o Serviço. OpenSearch
8. Em Grupos de segurança, selecione os grupos de segurança para associar às interfaces de rede do endpoint. Essa é uma etapa crítica na qual você limita as portas, os protocolos e as origens para o tráfego de entrada que você está autorizando para o seu endpoint. As regras do grupo de segurança devem permitir que os recursos que usarão o VPC endpoint para se comunicar com o OpenSearch Serviço se comuniquem com a interface de rede do endpoint.
9. Escolha Criar endpoint. O endpoint deverá estar ativo em 2 a 5 minutos.

Trabalhando com endpoints OpenSearch VPC gerenciados por serviços usando a API de configuração

Use as seguintes operações de API para criar e gerenciar endpoints de OpenSearch VPC gerenciados por serviços.

- CreateVpcEndpoint
- ListVpcEndpoints
- UpdateVpcEndpoint
- DeleteVpcEndpoint

Use as seguintes operações de API para gerenciar o acesso de endpoints aos domínios da VPC:

- AuthorizeVpcEndpointAccess
- ListVpcEndpointAccess
- ListVpcEndpointsForDomain
- RevokeVpcEndpointAccess

Autenticação SAML para painéis OpenSearch

A autenticação SAML para OpenSearch painéis permite que você use seu provedor de identidade existente para oferecer login único (SSO) para painéis em domínios do Amazon OpenSearch Service executados no Elasticsearch 6.7 ou posterior. OpenSearch Para usar autenticação do SAML, é necessário habilitar o controle de acesso refinado.

Em vez de se autenticar por meio do Amazon Cognito ou do banco de dados interno de usuários, a autenticação SAML OpenSearch para painéis permite que você use provedores de identidade terceirizados para fazer login nos painéis, gerenciar o controle de acesso refinado, pesquisar seus dados e criar visualizações. OpenSearch O serviço oferece suporte a provedores que usam o padrão SAML 2.0, como Okta, Keycloak, Active Directory Federation Services (ADFS), Auth0 e. Centro de Identidade do AWS IAM

A autenticação SAML para painéis serve apenas para acessar OpenSearch painéis por meio de um navegador da web. Suas credenciais SAML não permitem que você faça solicitações HTTP diretas para os painéis OpenSearch ou painéis. APIs

Visão geral da configuração do SAML

Esta documentação pressupõe que você tenha um provedor de identidade existente e alguma familiaridade com ele. Não podemos fornecer etapas de configuração detalhadas para seu provedor exato, somente para seu domínio OpenSearch de serviço.

O fluxo de login do OpenSearch Dashboards pode assumir uma das duas formas:

- Provedor de serviço (SP) iniciado: você navega até Dashboards (por exemplo, `https://my-domain.us-east-1.es.amazonaws.com/_dashboards`), a qual redireciona você para a tela de login. Após você fazer login, o provedor de identidade redirecionará você para o Dashboards.
- Provedor de identidade (IdP) iniciado: você navega até seu provedor de identidade, faz login e escolhe OpenSearch Painéis em um diretório de aplicativos.

OpenSearch O serviço fornece dois logons únicos URLs, iniciados pelo SP e iniciados pelo IDP, mas você só precisa daquele que corresponda ao fluxo de login do Dashboards desejado. OpenSearch

Independentemente do tipo de autenticação utilizado, o objetivo é fazer login por meio do provedor de identidade e receber uma declaração SAML que contenha seu nome de usuário (obrigatório) e quaisquer funções de backend (opcional, mas recomendado). Esta informação permite que o controle de acesso refinado atribua permissões a usuários do SAML. Em provedores de identidade externos, as funções de backend são normalmente chamadas de “funções” ou “grupos”

Considerações

Considere o seguinte ao configurar a autenticação SAML:

- Devido ao tamanho do arquivo de metadados de IdP, é altamente recomendável usar o console da AWS para configurar a autenticação SAML.
- Os domínios oferecem suporte a apenas um método de autenticação do Dashboards por vez. Se você tiver a autenticação do Amazon Cognito para OpenSearch painéis ativada, deverá desativá-la antes de habilitar a autenticação SAML.
- Se você usa um network load balancer com SAML, primeiro deve criar um endpoint personalizado. Para obter mais informações, consulte ???.
- As Políticas de Controle de Serviços (SCP) não serão aplicáveis nem avaliadas no caso de identidades que não sejam do IAM (como SAML no OpenSearch Amazon Serverless e SAML e autorização básica de usuário interno para o Amazon Service). OpenSearch

Autenticação SAML para domínios de VPC

O SAML não requer comunicação direta entre o seu provedor de identidade e seu provedor de serviços. Portanto, mesmo que seu OpenSearch domínio esteja hospedado em uma VPC privada, você ainda poderá usar o SAML, desde que seu navegador possa se comunicar com seu OpenSearch cluster e seu provedor de identidade. O seu navegador atua essencialmente como

intermediário entre o seu provedor de identidade e seu provedor de serviços. Para um diagrama útil que explica o fluxo de autenticação SAML, consulte a Documentação do Okta.

Modificar a política de acesso ao domínio

Antes de configurar a autenticação SAML, é necessário atualizar a política de acesso ao domínio para permitir que os usuários de SAML acessem o domínio. Caso contrário, haverá erros de acesso negado.

Recomendamos a seguinte política de acesso ao domínio, que fornece acesso total aos sub-recursos (`/*`) no domínio:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Para tornar a política mais restritiva, você pode adicionar a ela uma condição de endereço IP. Essa condição limita o acesso somente ao intervalo de endereços IP ou à sub-rede especificada. Por exemplo, a política a seguir permite acesso apenas da sub-rede 192.0.2.0/24:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

> **Note**
>
> Uma política de acesso a domínio aberta exige que o controle de acesso refinado seja habilitado no domínio, caso contrário, você verá o seguinte erro:
> `To protect domains with public access, a restrictive policy or fine-grained access control is required.`
> Se você tiver um usuário-mestre ou interno configurado com uma senha forte, manter a política aberta ao usar um controle de acesso refinado pode ser aceitável do ponto de vista da segurança. Para saber mais, consulte ???.

Configurar a autenticação iniciada por SP ou IdP

Essas etapas explicam como habilitar a autenticação SAML com autenticação iniciada por SP ou IdP para painéis. OpenSearch Para visualizar a etapa extra necessária para habilitar ambos, consulte Configurar a autenticação iniciada tanto por SP quanto por IdP.

Etapa 1: Habilitar a autenticação SAML

É possível habilitar a autenticação SAML durante a criação do domínio ou escolhendo Ações, Editar configuração de segurança em um domínio existente. As etapas a seguir variam um pouco, dependendo de qual delas você escolher.

Na configuração do domínio, em Autenticação SAML para OpenSearch Dashboards/Kibana, selecione Habilitar autenticação SAML.

Etapa 2: Configurar seu provedor de identidade

Conclua as etapas a seguir, dependendo de quando você está configurando a autenticação SAML.

Se estiver criando um novo domínio

Se você estiver criando um novo domínio, o OpenSearch Service ainda não poderá gerar um ID de entidade do provedor de serviços ou SSO URLs. Seu provedor de identidade requer esses valores para habilitar adequadamente a autenticação SAML, mas eles apenas podem ser gerados depois da criação do domínio. Para solucionar essa interdependência durante a criação do domínio, forneça valores temporários na sua configuração de IdP para gerar os metadados necessários e depois atualizá-los quando o domínio estiver ativo.

Se você estiver usando um endpoint personalizado, poderá inferir qual URLs será. Por exemplo, se o endpoint personalizado for `www.custom-endpoint.com`, o ID de entidade do provedor de serviços será `www.custom-endpoint.com`, o URL de SSO iniciado pelo IdP será `www.custom-endpoint.com/_dashboards/_opendistro/_security/saml/acs/idpinitiated` e o URL de SSO iniciado pelo SP será `www.custom-endpoint.com/_dashboards/_opendistro/_security/saml/acs`. É possível usar os valores para configurar seu provedor de identidade antes de criar o domínio. Consulte a próxima seção para ver exemplos.

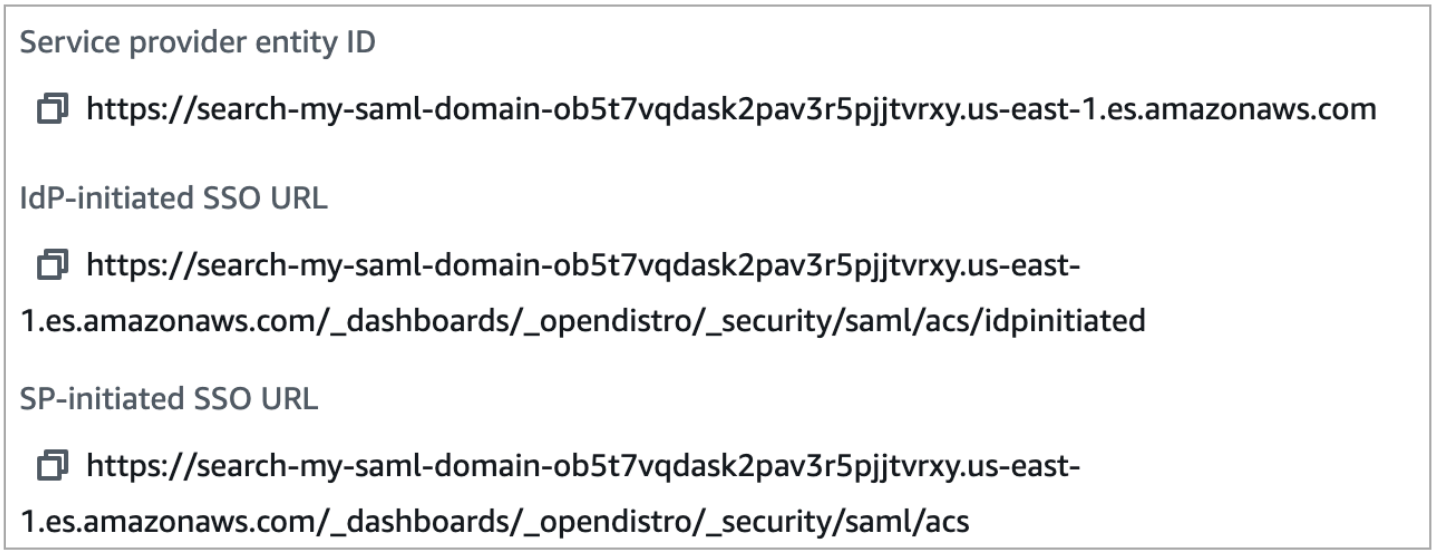
> **Note**
>
> Você não pode fazer login com um endpoint de pilha dupla porque o FQDN de uma solicitação HTTP é diferente do FQDN de uma solicitação SAML. Um OpenSearch administrador precisará configurar um endpoint personalizado e definir o valor CNAME como endpoint de pilha dupla se você quiser entrar usando um endpoint de pilha dupla.

Se você não estiver usando um endpoint personalizado, poderá inserir valores temporários no seu IdP para gerar os metadados necessários e atualizá-los posteriormente quando o domínio estiver ativo.

Por exemplo, no Okta, você pode inserir `https://temp-endpoint.amazonaws.com` nos campos URL de login único e URI do público - ID da entidade SP, o que permite gerar os metadados. Depois que o domínio estiver ativo, você poderá recuperar os valores corretos do OpenSearch Serviço e atualizá-los no Okta. Para instruções, consulte the section called “Etapa 6: atualize seu IdP URLs”.

Se estiver editando um domínio existente

Se você estiver habilitando a autenticação SAML em um domínio existente, copie o ID da entidade do provedor de serviços e um dos URLs SSO. Para orientação sobre qual URL usar, consulte the section called “Visão geral da configuração do SAML”.



Use os valores para configurar seu provedor de identidade. Essa é a parte mais complexa do processo e, infelizmente, a terminologia e as etapas variam muito de acordo com o provedor. Consulte a documentação do seu provedor.

No Okta, por exemplo, você deve criar uma aplicação web SAML 2.0. Para URL de acesso único, especifique o URL de SSO. Para URI do público (ID da entidade do SP), especifique o ID da entidade do provedor de serviços.

Em vez de usuários e funções de backend, o Okta tem usuários e grupos. Em Instruções de atributo de grupo, recomendamos adicionar `role` ao campo Nome e a expressão regular `.+` ao campo Filtro. Esta instrução diz ao provedor de identidade do Okta para incluir todos os grupos de usuários sob o campo `role` da asserção SAML após a autenticação de um usuário.

No IAM Identity Center, você especifica o ID da entidade SP como o público do Application SAML. Você também precisa especificar o mapeamento dos seguintes atributos: `Subject=${user:subject}:format=unspecified` e `Role=${user:groups}:format=uri`.

No Auth0, você cria uma aplicação Web regular e, em seguida, habilita o complemento SAML 2.0. No Keycloak, você cria um cliente.

Etapa 3: Importar metadados do IdP

Aopis você configurar o provedor de identidade, ele gera um arquivo de metadados IdP. Esse arquivo XML contém informações sobre o provedor, como um certificado TLS, endpoints de acesso único e o ID de entidade do provedor de identidade.

Copie o conteúdo do arquivo de metadados do IdP e cole-o no campo Metadados do IdP no console de serviço. OpenSearch Alternativamente, escolha Importar de arquivo XML e carregue o arquivo. O arquivo de metadados deve ser semelhante ao seguinte:

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

Etapa 4: Configurar campos SAML

Depois de inserir seus metadados do IdP, configure os seguintes campos adicionais no OpenSearch console de serviço:

- ID da entidade do IdP: copie o valor da propriedade `entityID` do seu arquivo de metadados e cole-o nesse campo. Muitos provedores de identidade também exibem esse valor como parte de um resumo pós-configuração. Alguns provedores chamam isso de “emissor”.

- Nome de usuário principal do SAML e função de back-end principal do SAML — A função de and/or back-end do usuário que você especifica recebe permissões completas para o cluster, equivalentes a um novo usuário mestre, mas só pode usar essas permissões nos painéis. OpenSearch

  No Okta, por exemplo, você pode ter um usuário `jdoe` que pertence ao grupo `admins`. Se você adicionar `jdoe` ao nome de usuário primário do SAML, somente esse usuário receberá permissões completas. Se você adicionar `admins` ao campo Perfil de backend primário SAML, qualquer usuário que pertença ao grupo `admins` receberá permissões completas.

  > **Note**
  >
  > O conteúdo da asserção SAML deve corresponder exatamente às strings que você usa para o nome de usuário primário do SAML e o perfil primário SAML. Alguns provedores de identidade adicionam um prefixo antes de seus nomes de usuário, o que pode causar uma hard-to-diagnose incompatibilidade. Na interface do usuário do provedor de identidade, talvez você veja `jdoe`, mas a asserção SAML poderá conter `auth0|jdoe`. Use sempre a string da asserção SAML.

Muitos provedores de identidade permitem que você visualize uma declaração de exemplo durante o processo de configuração, e ferramentas como o SAML-tracer podem ajudar a examinar e solucionar problemas de conteúdo de asserções reais. As asserções são semelhantes a:

```
<?xml version="1.0" encoding="UTF-8"?>
<saml2:Assertion ID="id67229299299259351343340162"
 IssueInstant="2020-09-22T22:03:08.633Z" Version="2.0"
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
  <saml2:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">idp-issuer</
saml2:Issuer>
  <saml2:Subject>
    <saml2:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">username</saml2:NameID>
    <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
      <saml2:SubjectConfirmationData NotOnOrAfter="2020-09-22T22:08:08.816Z"
 Recipient="domain-endpoint/_dashboards/_opendistro/_security/saml/acs"/>
    </saml2:SubjectConfirmation>
  </saml2:Subject>
  <saml2:Conditions NotBefore="2020-09-22T21:58:08.816Z"
 NotOnOrAfter="2020-09-22T22:08:08.816Z">
```

```
    <saml2:AudienceRestriction>
      <saml2:Audience>domain-endpoint</saml2:Audience>
    </saml2:AudienceRestriction>
  </saml2:Conditions>
  <saml2:AuthnStatement AuthnInstant="2020-09-22T19:54:37.274Z">
    <saml2:AuthnContext>

 <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport<
saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
  </saml2:AuthnStatement>
  <saml2:AttributeStatement>
    <saml2:Attribute Name="role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:unspecified">
      <saml2:AttributeValue
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:type="xs:string">GroupName Match Matches regex ".+" (case-sensitive)
      </saml2:AttributeValue>
    </saml2:Attribute>
  </saml2:AttributeStatement>
</saml2:Assertion>
```

Etapa 5: (Opcional) Configurar definições adicionais

Em Configurações adicionais, defina os seguintes campos opcionais:

- Chave do requerente: você pode deixar esse campo vazio para usar o elemento `NameID` da asserção SAML como nome de usuário. Se sua asserção não usar este elemento padrão e, em vez disso, incluir o nome de usuário como um atributo personalizado, especifique esse atributo aqui.
- Chave de perfis: se quiser usar funções de backend (recomendadas), especifique um atributo da afirmação nesse campo, como `role` ou `group`. Esta é outra situação em que ferramentas como o SAML tracer podem ajudar.
- Tempo de ativação da sessão — Por padrão, o OpenSearch Dashboards desconecta os usuários após 24 horas. Você pode configurar esse valor como qualquer número entre 60 e 1.440 (24 horas) especificando um novo valor.

Quando estiver satisfeito com sua configuração, salve o domínio.

Etapa 6: atualize seu IdP URLs

Se você ativou a autenticação SAML ao criar um domínio, precisará especificar o temporário URLs no seu IdP para gerar o arquivo de metadados XML. Depois que o status do domínio mudar para`Active`, você poderá obter o IdP correto URLs e modificar seu IdP.

Para recuperar o URLs, selecione o domínio e escolha Ações, Editar configuração de segurança. Em Autenticação SAML para OpenSearch dashboards/Kibana, você pode encontrar o ID de entidade do provedor de serviços e o SSO corretos. URLs Copie os valores e use-os para configurar seu provedor de identidade, substituindo o temporário URLs que você forneceu na etapa 2.

Etapa 7: mapear usuários SAML para perfis

Quando o status do seu domínio estiver Ativo e seu IdP estiver configurado corretamente, navegue até OpenSearch Painéis.

- Se você escolheu o URL iniciado pelo SP, navegue até *`domain-endpoint`*`/_dashboards`. Para fazer login diretamente em um inquilino específico, você pode anexar `?security_tenant=`*`tenant-name`* ao URL.
- Se você escolheu o URL iniciado pelo IsP, navegue até o diretório de aplicações do provedor de identidade.

Em ambos os casos, faça login como usuário primário do SAML ou um usuário que pertence à função de backend primário do SAML. Para continuar o exemplo da etapa 7, faça login como `jdoe` ou como um membro do grupo `admins`.

Depois que os OpenSearch painéis forem carregados, escolha Segurança, Funções. Em seguida, mapeie as funções para permitir que outros usuários acessem os OpenSearch painéis.

Por exemplo, você pode mapear o seu colega confiável `jroe` nas funções `all_access` e `security_manager`. Você também pode mapear a função de backend `analysts` nas funções `readall` e `opensearch_dashboards_user`.

Se você preferir usar a API em vez de OpenSearch painéis, veja o seguinte exemplo de solicitação:

```
PATCH _plugins/_security/api/rolesmapping
[
  {
    "op": "add", "path": "/security_manager", "value": { "users": ["master-user",
 "jdoe", "jroe"], "backend_roles": ["admins"] }
```

```
  },
  {
    "op": "add", "path": "/all_access", "value": { "users": ["master-user", "jdoe",
 "jroe"], "backend_roles": ["admins"] }
  },
  {
    "op": "add", "path": "/readall", "value": { "backend_roles": ["analysts"] }
  },
  {
    "op": "add", "path": "/opensearch_dashboards_user", "value": { "backend_roles":
 ["analysts"] }
  }
]
```

Configurar a autenticação iniciada por SP ou IdP

Caso pretenda configurar a autenticação iniciada pelo SP e pelo IdP, você deverá fazê-lo via provedor de identidade. Por exemplo, no Okta, você pode executar as seguintes etapas:

1. Em sua aplicação SAML, vá para Geral, Configurações SAML.
2. Para o URL de autenticação única, forneça o URL de SSO iniciado por IdP. Por exemplo, .`https://search-domain-hash/_dashboards/_opendistro/_security/saml/acs/idpinitiated`
3. Ative Permitir que este aplicativo solicite outro SSO URLs.
4. Em SSO solicitável URLs, adicione um ou mais SSO iniciados pelo SP. URLs Por exemplo, .`https://search-domain-hash/_dashboards/_opendistro/_security/saml/acs`

Configurar a autenticação SAML (AWS CLI)

O AWS CLI comando a seguir ativa a autenticação SAML para OpenSearch painéis em um domínio existente:

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --advanced-security-options '{"SAMLOptions":{"Enabled":true,"MasterUserName":"my-
idp-user","MasterBackendRole":"my-idp-group-or-role","Idp":{"EntityId":"entity-
id","MetadataContent":"metadata-content-with-quotes-escaped"},"RolesKey":"optional-
roles-key","SessionTimeoutMinutes":180,"SubjectKey":"optional-subject-key"}}'
```

Você deve “escapar” todas as aspas e caracteres de nova linha no XML de metadados. Por exemplo, use `<KeyDescriptor use=\"signing\">\n` em vez de `<KeyDescriptor use="signing">` e uma quebra de linha. Para obter informações detalhadas sobre como usar a AWS CLI, consulte a Referência de comandos da AWS CLI.

Configurar a autenticação SAML (API de configuração)

A solicitação a seguir para a API de configuração permite a autenticação SAML para OpenSearch painéis em um domínio existente:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "SAMLOptions": {
      "Enabled": true,
      "MasterUserName": "my-idp-user",
      "MasterBackendRole": "my-idp-group-or-role",
      "Idp": {
        "EntityId": "entity-id",
        "MetadataContent": "metadata-content-with-quotes-escaped"
      },
      "RolesKey": "optional-roles-key",
      "SessionTimeoutMinutes": 180,
      "SubjectKey": "optional-subject-key"
    }
  }
}
```

Você deve “escapar” todas as aspas e caracteres de nova linha no XML de metadados. Por exemplo, use `<KeyDescriptor use=\"signing\">\n` em vez de `<KeyDescriptor use="signing">` e uma quebra de linha. Para obter informações detalhadas sobre como usar a API de configuração, consulte a referência da API de OpenSearch serviço.

Solução de problemas de SAML

Erro	Detalhes
Sua solicitação: '*/some/path* ' não é permitida.	Verifique se você forneceu o URL de SSO correto (etapa 3) ao seu provedor de identidade.

Erro	Detalhes
Forneça um documento de metadados do provedor de identidades válido para habilitar o SAML.	O arquivo de metadados do IdP não atende ao padrão SAML 2.0. Verifique se há erros usando uma ferramenta de validação.
As opções de configuração do SAML não estão visíveis no console.	Atualize para o software de serviço mais recente.
Erro de configuração do SAML: algo errado ocorreu ao tentar recuperar a configuração do SAML. Verifique suas configurações.	Esse erro genérico pode ocorrer por vários motivos. • Verifique se você forneceu ao provedor de identidade o ID de entidade do provedor de serviços e o URL de SSO corretos. • Gere novamente o arquivo de metadados do IdP e verifique o ID de entidade do IdP. Adicione quaisquer metadados atualizados no console do AWS . • Verifique se sua política de acesso ao domínio permite acesso aos OpenSearch painéis e. `_plugins/_security/*` Em geral, recomenda mos uma política de acesso aberto para domínios que usam controle de acesso refinado. • Consulte a documentação do provedor de identidad e para obter as etapas de configuração do SAML.

Erro	Detalhes
Função ausente: nenhuma função disponível para este usuário, entre em contato com o administrador do sistema..	Você autenticou com êxito, mas o nome de usuário e quaisquer funções de backend da asserção SAML não são mapeados em nenhuma função e, portanto, não têm permissões. Esses mapeamentos diferenci am maiúsculas de minúsculas. O administrador do sistema pode verificar o conteúdo da asserção SAML usando uma ferramenta como o SAML-tracer e depois verificar o mapeamento de perfis usando a seguinte solicitação: `GET _plugins/_security/api/rolesmapping`
Seu navegador redireciona ou recebe continuamente erros HTTP 500 ao tentar acessar OpenSearch os painéis.	Esses erros podem ocorrer se sua asserção SAML contiver um grande número de funções totalizando aproximadamente 1.500 caracteres. Por exemplo, se você passar 80 funções com tamanho médio de 20 caracteres, o limite de tamanho para cookies em seu navegador da Web poderá ser excedido. A partir da OpenSearch versão 2.7, a asserção SAML oferece suporte a funções de até 5.000 caracteres.
Não é possível sair do ADFS.	O ADFS exige que todas as solicitações de logout sejam assinadas, o que o OpenSearch Serviço não suporta. Remova `<SingleLogoutService />` do arquivo de metadados do IdP para forçar o OpenSearch Serviço a usar seu próprio mecanismo de logout interno.
`Could not find entity descriptor for __PATH__.`	O ID da entidade do IdP fornecido nos metadados XML to OpenSearch Service é diferente daquele na resposta SAML. Para corrigir isso, verifique se eles correspondem. Ative logs de erros do aplicativo CW no seu domínio para encontrar a mensagem de erro para depurar o problema de integração do SAML.

Erro	Detalhes
`Signature validation failed. SAML response rejected.`	OpenSearch O serviço não consegue verificar a assinatura na resposta SAML usando o certificado do IdP fornecido no XML de metadados. Você pode ter cometido um erro ou seu IdP alterou o certificado. Atualize o certificado mais recente do seu IdP no XML de metadados fornecido ao OpenSearch Serviço por meio do. Console de gerenciamento da AWS
`__PATH__ is not a valid audience for this response.`	O campo de público na resposta do SAML não corresponde ao endpoint do domínio. Para corrigir esse erro, atualize o campo “SP audience” para corresponder ao endpoint do seu domínio. Se você ativou endpoints personalizados, o campo de público deve corresponder ao seu endpoint personalizado. Ative logs de erros do aplicativo CW no seu domínio para encontrar a mensagem de erro para depurar o problema de integração do SAML.
Seu navegador recebe um erro HTTP 400 na resposta ao `Invalid Request Id`.	Esse erro geralmente ocorre se você configurou o URL iniciado pelo IdP com o formato *<DashboardsURL>* `/_opendistro/_security/saml/acs`. Em vez disso, configure o URL com o formato *<DashboardsURL>* `/_opendistro/_security/saml/acs/idpinitiated` .

Erro	Detalhes
A resposta foi recebida às __PATH__ em vez de __PATH__.	O campo de destino na resposta do SAML não corresponde a um dos seguintes formatos de URL: • *<DashboardsURL>* /_opendistro/_security/saml/acs • *<DashboardsURL>* /_opendistro/_security/saml/acs/idpinitiated . Dependendo do fluxo de login que você usa (iniciado pelo SP ou iniciado pelo IDP), insira um campo de destino que corresponda a um dos. OpenSearch URLs
A resposta tem um InResponseTo atributo, enquanto InResponseTo não era esperado.	Você está usando o URL iniciado por IdP para um fluxo de login iniciado por SP. Em vez disso, use o URL iniciado por SP.

Desabilitação da autenticação SAML

Para desativar a autenticação SAML para OpenSearch painéis (console)

1. Escolha o domínio, Ações, e Editar configuração de segurança.
2. Desmarque Habilitar autenticação SAML.
3. Escolha Salvar alterações.
4. Após o processamento do domínio, verifique o mapeamento de função de controle de acesso refinado com a seguinte solicitação:

   ```
   GET _plugins/_security/api/rolesmapping
   ```

   A desativação da autenticação SAML para painéis não remove os mapeamentos do nome de usuário and/or principal do SAML e da função de back-end principal do SAML. Se você quiser remover esses mapeamentos, faça login no Dashboards usando o banco de dados interno de usuários (se habilitado) ou use a API para removê-los:

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "users": [
    "master-user"
  ]
}
```

Suporte confiável de propagação de identidade do IAM Identity Center para OpenSearch

Agora você pode usar os diretores AWS do IAM Identity Center configurados centralmente (usuários e grupos) por meio do Trusted Identity Propagation para acessar os domínios do Amazon OpenSearch Service por meio de aplicativos de serviço. OpenSearch Para habilitar o suporte do IAM Identity Center OpenSearch, você precisará habilitar o uso do IAM Identity Center. Para saber mais sobre como fazer isso, consulte O que é o IAM Identity Center? . Consulte Como associar OpenSearch domínio como fonte de dados em OpenSearch aplicativos? para obter detalhes.

Você pode configurar o IAM Identity Center usando o console de OpenSearch serviço, o AWS Command Line Interface (AWS CLI) ou AWS SDKs o.

> **Note**
>
> As entidades principais do IAM Identity Center não são compatíveis com Dashboards (no mesmo local do cluster). Eles só são suportados por meio de uma interface de OpenSearch usuário centralizada (painéis).

Considerações

Antes de usar o IAM Identity Center com o Amazon OpenSearch Service, você deve considerar o seguinte:

- O IAM Identity está habilitado na conta.
- O IAM Identity Center está disponível na sua região.
- A versão do OpenSearch domínio é 1.3 ou posterior.
- O controle de acesso refinado está habilitado no domínio.

- O domínio está na mesma região que a instância do IAM Identity Center.
- O domínio e o OpenSearch aplicativo pertencem à mesma AWS conta.

Modificar a política de acesso ao domínio

Antes de configurar o IAM Identity Center, você deve atualizar a política de acesso ao domínio ou as permissões da função do IAM configurada nos OpenSearch aplicativos para o Trusted Identity Propagation.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/OpenSearchRole"
            },
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/example-domain/
*"
        }
    ]
}
```

Configurar autenticação e autorização do IAM Identity Center (console)

Você pode habilitar a autenticação e a autorização do IAM Identity Center durante o processo de criação do domínio ou atualizando um domínio existente. As etapas de configuração variam um pouco, dependendo de qual opção você escolher.

As etapas a seguir explicam como configurar um domínio existente para autenticação e autorização do IAM Identity Center no console do Amazon OpenSearch Service:

1. Em Configuração do domínio, navegue até Configuração de segurança, escolha Editar e vá até a seção Autenticação do IAM Identity Center e selecione Habilitar acesso de API autenticado com o IAM Identity Center.

2. Selecione a tecla SubjectKey and Roles da seguinte forma.

   - Chave de assunto - escolha uma das UserId (padrão) UserName e E-mail para usar o atributo correspondente como principal acessando o domínio.
   - Chave de funções - escolha uma das GroupId (padrão) e GroupName use os valores de atributos correspondentes como função de back-end fine-grained-access-controlpara todos os grupos associados ao diretor do IAM Identity Center.

Depois de fazer as alterações, salve o seu domínio.

Habilitar o controle de acesso refinado

Depois de habilitar a opção IAM Identity Center em seu OpenSearch domínio, você pode configurar o acesso aos principais do IAM Identity Center criando um mapeamento de funções para a função de back-end. O valor da função de back-end para o principal é baseado na associação ao grupo principal do IAM Identity Center e na RolesKey configuração de GroupId ou GroupName.

> **Note**
>
> O Amazon OpenSearch Service pode oferecer suporte a até 100 grupos para um único usuário. Se você tentar usar mais do que o número permitido de instâncias, ocorrerá uma inconsistência no processamento da fine-grained-access-control autorização e receberá uma mensagem de erro 403.

Configurar autenticação e autorização do IAM Identity Center (CLI)

```
  aws opensearch update-domain-config \
      --domain-name my-domain \
      --identity-center-options '{"EnabledAPIAccess": true,
 "IdentityCenterInstanceARN": "instance arn",  "SubjectKey": "UserId/UserName/
UserEmail" , "RolesKey": "GroupId/GroupName"}'
```

Desabilitar a autenticação do IAM Identity Center no domínio

Para desativar o IAM Identity Center em seu OpenSearch domínio:

1. Escolha o domínio, Ações, e Editar configuração de segurança.
2. Desmarque a Habilitar acesso de API autenticado com o IAM Identity Center.
3. Escolha Salvar alterações.
4. Depois que o processamento do domínio for concluído, remova os mapeamentos de função adicionados aos diretores do IAM Identity Center

Para desabilitar o IAM Identity Center usando a CLI, você pode usar o que o procedimento a seguir

```
aws opensearch update-domain-config \
    --domain-name my-domain \
    --identity-center-options '{"EnabledAPIAccess": false}'
```

Configurando a autenticação do Amazon Cognito para painéis OpenSearch

Você pode autenticar e proteger sua instalação padrão de OpenSearch painéis do Amazon OpenSearch Service usando o Amazon Cognito. A autenticação do Amazon Cognito é opcional e está disponível somente para domínios que usam o Elasticsearch OpenSearch 5.1 ou posterior. Se você não configurar a autenticação do Amazon Cognito, ainda poderá proteger o Dashboards usando uma política de acesso baseada em IP e um servidor de proxy, autenticação básica HTTP ou SAML.

Grande parte do processo de autenticação ocorre no Amazon Cognito, mas esta seção oferece diretrizes e requisitos para configurar os recursos do Amazon Cognito para trabalhar com domínios de serviço. OpenSearch Preços padrão aplicam-se a todos os recursos do Amazon Cognito.

> **Tip**
>
> Na primeira vez que você configura um domínio para usar a autenticação do Amazon Cognito para OpenSearch painéis, recomendamos usar o console. Os recursos do Amazon Cognito são extremamente personalizáveis, e o console pode ajudar você a identificar e compreender os recursos que são importantes para você.

Tópicos

- Pré-requisitos
- Configuração de um domínio para uso da autenticação do Amazon Cognito
- Como permitir a função autenticada
- Configuração de provedores de identidade
- (Opcional) Configuração de acesso granular
- (Opcional) Personalização da página de login
- (Opcional) Configuração da segurança avançada
- Teste
- Cotas
- Problemas de configuração comuns
- Desabilitando a autenticação do Amazon Cognito para painéis OpenSearch
- Excluindo domínios que usam a autenticação do Amazon Cognito para painéis OpenSearch

Pré-requisitos

Antes de configurar a autenticação do Amazon Cognito para OpenSearch painéis, você deve cumprir vários pré-requisitos. O console OpenSearch de serviço ajuda a simplificar a criação desses recursos, mas entender a finalidade de cada recurso ajuda na configuração e na solução de problemas. A autenticação do Amazon Cognito para Dashboards requer os seguintes recursos:

- Conjunto de usuários do Amazon Cognito
- Grupo de identidades do Amazon Cognito
- Função do IAM com a política `AmazonOpenSearchServiceCognitoAccess` anexada (`CognitoAccessForAmazonOpenSearch`)

> **Note**
>
> Os grupos de usuários e identidades devem estar na mesma Região da AWS. Você pode usar o mesmo grupo de usuários, grupo de identidades e função do IAM para adicionar a autenticação do Amazon Cognito para painéis a vários OpenSearch domínios de serviço. Para saber mais, consulte the section called “Cotas”.

Sobre o grupo de usuários

Os grupos de usuários têm dois recursos principais: criar e gerenciar um diretório de usuários e permitir que os usuários se cadastrem e façam login. Para obter instruções para criar um grupo de usuários, consulte Introdução a grupos de usuários no Guia do desenvolvedor do Amazon Cognito.

Ao criar um grupo de usuários para usar com o OpenSearch Serviço, considere o seguinte:

- Seu grupo de usuários do Amazon Cognito deve ter um nome de domínio. OpenSearch O serviço usa esse nome de domínio para redirecionar os usuários para uma página de login para acessar os painéis. Além de um nome de domínio, o grupo de usuários não exige qualquer configuração não padrão.
- Você deve especificar os atributos padrão necessários do grupo, como nome, data de nascimento, endereço de e-mail e número de telefone. Você não pode alterar esses atributos depois de criar o grupo de usuários. Portanto, escolha os atributos importantes para você neste momento.
- Ao criar seu grupo de usuários, escolha se os usuários podem criar suas próprias contas, a segurança mínima da senha para contas e se deseja habilitar a autenticação multifator. Se você planeja usar um provedor de identidade externo, essas configurações são não têm qualquer consequência. Tecnicamente, você pode habilitar o grupo de usuários como um provedor de identidade e habilitar um provedor de identidade externo, mas a maioria das pessoas prefere um ou o outro.

O pool de usuários IDs assume a forma de*region*_*ID*. Se você planeja usar a AWS CLI ou um AWS SDK para configurar o OpenSearch serviço, anote o ID.

Sobre o grupo de identidades

Os grupos de identidades permitem que você atribua funções temporárias e de privilégio limitado a usuários depois que eles fazem login. Para obter instruções para criar um banco de identidades, consulte Visão geral do console do banco de identidades no Guia do usuário do Amazon Cognito. Ao criar um grupo de identidades para usar com o OpenSearch Service, considere o seguinte:

- Se você usar o console do Amazon Cognito, deverá marcar a caixa de seleção EPermitir acesso a identidades não autenticadas para criar o grupo de identidades. Depois de criar o grupo de identidades e configurar o domínio do OpenSearch Serviço, o Amazon Cognito desativa essa configuração.

- Você não precisa adicionar provedores de identidade externos ao grupo de identidades. Quando você configura o OpenSearch Serviço para usar a autenticação do Amazon Cognito, ele configura o grupo de identidades para usar o grupo de usuários que você acabou de criar.
- Depois de criar o grupo de identidades, você deve escolher as funções do IAM autenticadas e não autenticadas. Essas funções especificam as políticas de acesso que os usuários têm antes e depois de fazer login. Se você usar o console do Amazon Cognito ele poderá criar essas funções para você. Depois de criar a função autenticada, anote o nome do recurso da Amazon (ARN), que tem o formato de `arn:aws:iam::123456789012:role/Cognito_identitypoolnameAuth_Role`.

O pool de identidade IDs assume a forma de`region:ID-ID-ID-ID-ID`. Se você planeja usar a AWS CLI ou um AWS SDK para configurar o OpenSearch serviço, anote o ID.

Sobre a CognitoAccessForAmazonOpenSearch função

OpenSearch O serviço precisa de permissões para configurar os grupos de usuários e identidades do Amazon Cognito e usá-los para autenticação. Você pode usar`AmazonOpenSearchServiceCognitoAccess`, que é uma política AWS gerenciada, para essa finalidade. `AmazonESCognitoAccess`é uma política antiga que foi substituída por `AmazonOpenSearchServiceCognitoAccess` quando o serviço foi renomeado para Amazon OpenSearch Service. Ambas as políticas fornecem as permissões mínimas do Amazon Cognito necessárias para habilitar a autenticação do Amazon Cognito. Para obter detalhes sobre a política, consulte AmazonOpenSearchServiceCognitoAccesso Guia de referência de políticas AWS gerenciadas.

Se você usa o console para criar ou configurar seu domínio de OpenSearch serviço, ele cria uma função do IAM para você e anexa a `AmazonOpenSearchServiceCognitoAccess` política (ou a `AmazonESCognitoAccess` política, se for um domínio do Elasticsearch) à função. O nome padrão desta função é `CognitoAccessForAmazonOpenSearch`.

As políticas de permissões de função `AmazonOpenSearchServiceCognitoAccess` e `AmazonESCognitoAccess` ambas permitem que o OpenSearch Serviço conclua as seguintes ações em todos os grupos de identidades e usuários:

- Ação: `cognito-idp:DescribeUserPool`
- Ação: `cognito-idp:CreateUserPoolClient`
- Ação: `cognito-idp:DeleteUserPoolClient`

- Ação: `cognito-idp:UpdateUserPoolClient`
- Ação: `cognito-idp:DescribeUserPoolClient`
- Ação: `cognito-idp:AdminInitiateAuth`
- Ação: `cognito-idp:AdminUserGlobalSignOut`
- Ação: `cognito-idp:ListUserPoolClients`
- Ação: `cognito-identity:DescribeIdentityPool`
- Ação: `cognito-identity:SetIdentityPoolRoles`
- Ação: `cognito-identity:GetIdentityPoolRoles`

Se você usar o AWS CLI ou um dos AWS SDKs, deverá criar sua própria função, anexar a política e especificar o ARN para essa função ao configurar seu domínio de OpenSearch serviço. A função deve ter a seguinte relação de confiança:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "opensearchservice.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Para obter instruções, consulte Criar uma função para delegar permissões a um AWS serviço e Adicionar e remover permissões de identidade do IAM no Guia do usuário do IAM.

Configuração de um domínio para uso da autenticação do Amazon Cognito

Depois de concluir os pré-requisitos, você pode configurar um domínio de OpenSearch serviço para usar o Amazon Cognito for Dashboards.

> **Note**
>
> O Amazon Cognito não está disponível em todos. Regiões da AWS Para obter uma lista das regiões compatíveis, consulte Pontos de serviço para o Amazon Cognito. Você não precisa usar a mesma região para o Amazon Cognito que você usa para OpenSearch o Service.

Configuração da autenticação do Amazon Cognito (console)

Como ele cria o perfil `CognitoAccessForAmazonOpenSearch` para você, o console propicia a mais simples experiência de configuração. Além das permissões padrão do OpenSearch Serviço, você precisa do seguinte conjunto de permissões para usar o console e criar um domínio que usa a autenticação do Amazon Cognito para OpenSearch painéis.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools",
        "iam:CreateRole",
        "iam:AttachRolePolicy"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
    }
  ]
}
```

Para obter instruções sobre como adicionar permissões a uma identidade (usuário, grupo de usuários ou função), consulte Adicionar permissões de identidade do IAM (console).

Se `CognitoAccessForAmazonOpenSearch` já existir, você precisará de um número menor de permissões:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
    }
  ]
}
```

Para configurar a autenticação do Amazon Cognito para Dashboards (console)

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa/.
2. Em Domínios, selecione o domínio que deseja configurar.
3. Escolha Ações, Editar configuração de segurança.
4. Selecione Habilitar autenticação do Amazon Cognito.
5. Em Região, selecione Região da AWS aquela que contém seu grupo de usuários e grupo de identidades do Amazon Cognito.

6. Em Grupo de usuários do Cognito, selecione um grupo de usuários ou crie um. Para saber mais, consulte the section called “Sobre o grupo de usuários”.
7. Em Grupo de identidades do Cognito, selecione um grupo de identidades ou crie um. Para saber mais, consulte the section called “Sobre o grupo de identidades”.

   > **Note**
   >
   > Os links Criar grupo de usuários e Criar grupo de identidades direcionam você para o console do Amazon Cognito e exigem que você crie esses recursos manualmente. O processo não é automático. Para saber mais, consulte the section called “Pré-requisitos”.

8. Em Nome da função do IAM, use o valor padrão de `CognitoAccessForAmazonOpenSearch` (recomendado) ou insira um novo nome. Para saber mais, consulte the section called “Sobre a CognitoAccessForAmazonOpenSearch função”.
9. Escolha Salvar alterações.

Depois que o seu domínio concluir o processamento, consulte the section called “Como permitir a função autenticada” e the section called “Configuração de provedores de identidade” para ver as etapas de configuração adicionais.

Configuração da autenticação do Amazon Cognito (AWS CLI)

Use o `--cognito-options` parâmetro para configurar seu domínio OpenSearch de serviço. A sintaxe a seguir é usada pelos comandos `create-domain` e `update-domain-config`:

```
--cognito-options Enabled=true,UserPoolId="user-pool-id",IdentityPoolId="identity-pool-id",RoleArn="arn:aws:iam::123456789012:role/CognitoAccessForAmazonOpenSearch"
```

Exemplo

O exemplo a seguir cria um domínio na região `us-east-1` que habilita a autenticação do Amazon Cognito para o Dashboards usando a função `CognitoAccessForAmazonOpenSearch` e fornece acesso ao domínio para `Cognito_Auth_Role`:

```
aws opensearch create-domain --domain-name my-domain --
region us-east-1 --access-policies '{ "Version": "2012-10-17",
        "Statement":[{"Effect":"Allow","Principal":{"AWS":
 ["arn:aws:iam::123456789012:role/
Cognito_Auth_Role"]},"Action":"es:ESHttp*","Resource":"arn:aws:es:us-
```

```
east-1:123456789012:domain/*" }]}' --engine-version "OpenSearch_1.0"
 --cluster-config InstanceType=m4.xlarge.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeSize=10 --cognito-options
 Enabled=true,UserPoolId="us-east-1_123456789",IdentityPoolId="us-
east-1:12345678-1234-1234-1234-123456789012",RoleArn="arn:aws:iam::123456789012:role/
CognitoAccessForAmazonOpenSearch"
```

Depois que o seu domínio concluir o processamento, consulte the section called “Como permitir a função autenticada” e the section called “Configuração de provedores de identidade” para ver as etapas de configuração adicionais.

Configurando a autenticação do Amazon Cognito ()AWS SDKs

O AWS SDKs (exceto Android e iOS SDKs) suporta todas as operações definidas na Amazon OpenSearch Service API Reference, incluindo o `CognitoOptions` parâmetro para as `UpdateDomainConfig` operações `CreateDomain` e. Para obter mais informações sobre como instalar e usar o AWS SDKs, consulte Kits AWS de desenvolvimento de software.

Depois que o seu domínio concluir o processamento, consulte the section called “Como permitir a função autenticada” e the section called “Configuração de provedores de identidade” para ver as etapas de configuração adicionais.

Como permitir a função autenticada

Por padrão, a função autenticada do IAM que você configurou seguindo as diretrizes em the section called “Sobre o grupo de identidades” não tem os privilégios necessários para acessar OpenSearch os painéis. Você deve fornecer permissões adicionais à função.

> **Note**
>
> Se tiver configurado o controle de acesso detalhado e usar uma política de acesso “aberta” ou baseada em IP, poderá ignorar esta etapa.

Você pode incluir essas permissões em uma política baseada em identidade, mas, a menos que queira que os usuários autenticados tenham acesso a todos os domínios do OpenSearch Serviço, uma política baseada em recursos anexada a um único domínio é a melhor abordagem.

Para o `Principal`, especifique o ARN da função autenticada do Cognito que você configurou com as diretrizes em the section called “Sobre o grupo de identidades”.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":[
               "arn:aws:iam::111122223333:role/Cognito_identitypoolname/
Auth_Role"
            ]
         },
         "Action":[
            "es:ESHttp*"
         ],
         "Resource":"arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
      }
   ]
}
```

Para obter instruções sobre como adicionar uma política baseada em recursos a um domínio OpenSearch de serviço, consulte. the section called “Configuração de políticas de acesso”

Configuração de provedores de identidade

Quando você configura um domínio para usar a autenticação do Amazon Cognito para painéis, o OpenSearch Service adiciona um cliente de aplicativo ao grupo de usuários e adiciona o grupo de usuários ao grupo de identidades como um provedor de autenticação.

> **Warning**
>
> Não renomeie nem exclua o cliente do aplicativo.

Dependendo de como você configurou o grupo de usuários, talvez precise criar contas de usuário manualmente ou os usuários podem ser capazes de criar suas próprias contas. Se essas configurações forem aceitáveis, você não precisará realizar ações adicionais. No entanto, muitas pessoas preferem usar provedores de identidade externos.

Para habilitar um provedor de identidade SAML 2.0, você deve fornecer um documento de metadados do SAML. Para habilitar provedores de identidades sociais, como o Login with Amazon, o Facebook e o Google, você deve ter um ID e um segredo do aplicativo a partir desses provedores. Você pode habilitar qualquer combinação de provedores de identidade.

A maneira mais fácil de configurar seu grupo de usuários é usar o console do Amazon Cognito. Para obter instruções, consulte Login de grupo de usuários com provedores de identidades de terceiros e Configurações específicas da aplicação com cliente da aplicação no Guia do Desenvolvedor da Amazon Cognito.

(Opcional) Configuração de acesso granular

Você deve ter notado que as configurações padrão do grupo de identidades atribuem a cada usuário que faz login na mesma função do IAM (`Cognito_`*`identitypool`*`Auth_Role`), o que significa que todos os usuários podem acessar os mesmos AWS recursos. Para usar o controle de acesso refinado com o Amazon Cognito, por exemplo, se desejar que os analistas da sua organização tenham acesso somente leitura a vários índices, mas que os desenvolvedores tenham acesso de gravação a todos os índices, você terá duas opções:

- Criar grupos de usuários e configurar seu provedor de identidade para escolher a função do IAM com base no token de autenticação do usuário (recomendado).
- Configurar o provedor de identidade para escolher a função do IAM com base em uma ou mais regras.

Para obter um passo a passo que inclui controle de acesso refinado, consulte the section called “Tutorial: Controle de acesso minucioso com autenticação Cognito”.

> **Important**
>
> Assim como a função padrão, o Amazon Cognito deve fazer parte da relação de confiança de cada função adicional. Para obter mais detalhes, consulte Criar perfis para mapeamento de perfis no Guia do desenvolvedor do Amazon Cognito.

Grupos de usuários e tokens

Quando você cria um grupo de usuários, escolhe uma função do IAM para os membros do grupo. Para obter informações sobre a criação de grupos, consulte Adicionar grupos a um grupo de usuários no Guia do desenvolvedor do Amazon Cognito.

Depois de criar um ou mais grupos de usuários, você pode configurar o provedor de autenticação para atribuir aos usuários suas funções de grupo em vez da função padrão do grupo de identidades. Selecione Escolher função do token e, em seguida, escolha Usar função autenticada padrão ou NEGAR para especificar como o pool de identidades lidará com os usuários que não fazem parte do grupo.

Regras

As regras são essencialmente uma série de instruções `if` que o Amazon Cognito avalia em sequência. Por exemplo, se um endereço de e-mail do usuário contiver `@corporate`, o Amazon Cognito atribuirá `Role_A` a esse usuário. Se um endereço de e-mail do usuário contém `@subsidiary`, esse usuário é atribuído a `Role_B`. Caso contrário, ele atribui ao usuário a função autenticada padrão.

Para saber mais, consulte Usar mapeamento baseado em regras para atribuir perfis aos usuários no Guia do desenvolvedor do Amazon Cognito.

(Opcional) Personalização da página de login

Você pode usar o console do Amazon Cognito para carregar de um logo personalizado e fazer alterações de CSS na página de login. Para ver as instruções e uma lista completa das propriedades de CSS, consulte Personalizar a marca da interface hospedada (clássica) no Guia do desenvolvedor do Amazon Cognito.

(Opcional) Configuração da segurança avançada

Os grupos de usuários do Amazon Cognito oferecem suporte a recursos de segurança avançada, como a autenticação multifator, a verificação de credenciais comprometidas e a autenticação adaptável. Para saber mais, consulte Usar os recursos de segurança dos grupos de usuários do Amazon Cognito no Guia do desenvolvedor do Amazon Cognito.

Teste

Depois que você estiver satisfeito com sua configuração, verifique se a experiência do usuário atende às suas expectativas.

Para acessar os OpenSearch painéis

1. Vá para `https://opensearch-domain/_dashboards` em um navegador da Web. Para fazer login diretamente em um inquilino específico, anexe `?security_tenant=tenant-name` ao URL.
2. Faça login usando suas credenciais preferenciais.
3. Depois que os OpenSearch painéis forem carregados, configure pelo menos um padrão de índice. O Dashboards usa esses padrões para identificar quais índices você deseja analisar. Digite *, escolha Próxima etapa e, em seguida, Criar padrão de índice.
4. Para pesquisar ou explorar seus dados, escolha Descobrir.

Se qualquer etapa desse processo falhar, consulte the section called “Problemas de configuração comuns” para obter informações sobre soluções de problemas.

Cotas

O Amazon Cognito tem limites flexíveis em muitos dos seus recursos. Se você quiser habilitar a autenticação de painéis para um grande número de domínios de OpenSearch serviço, revise as cotas no Amazon Cognito e solicite aumentos de limite conforme necessário.

Cada domínio OpenSearch de serviço adiciona um cliente de aplicativo ao grupo de usuários, o que adiciona um provedor de autenticação ao grupo de identidades. Se você habilitar a autenticação de OpenSearch painéis para mais de 10 domínios, poderá encontrar o limite “máximo de provedores de grupos de usuários do Amazon Cognito por grupo de identidades”. Se você exceder um limite, qualquer domínio de OpenSearch serviço que você tentar configurar para usar a autenticação do Amazon Cognito para painéis poderá ficar preso em um estado de configuração de Processamento.

Problemas de configuração comuns

As tabelas a seguir listam os problemas de configuração comuns e as soluções.

Configurando o serviço OpenSearch

Problema	Solução
`OpenSearch Service can't create the role` (console)	Você não tem as permissões corretas do IAM. Adicione as permissões especificadas em the section called "Configuração da autenticação do Amazon Cognito (console)".
`User is not authorized to perform: iam:PassRole on resource CognitoAccessForAmazonOpenSearch` (console)	Você não tem `iam:PassRole` permissões para a CognitoAccessForAmazonOpenSearchfunção. Anexe a política a seguir à sua conta: JSON `{ "Version":"2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "iam:PassRole" ], "Resource": "arn:aws:iam:: 123456789012:role/service-role/CognitoAccessForAmazonOpenSearch " } ] }` Como alternativa, você pode anexar a política `IAMFullAccess` .
`User is not authorized to perform: cognito-identity:ListIdentityPools on resource`	Você não tem permissões de leitura para o Amazon Cognito. Anexe a política `AmazonCognitoReadOnly` à sua conta.

Problema	Solução
`An error occurred (ValidationException) when calling the CreateDomain operation: OpenSearch Service must be allowed to use the passed role`	OpenSearch O serviço não está especificado na relação de confiança da `CognitoAccessForAmazonOpenSearch` função. Verifique se a sua função usa a relação de confiança especificada em the section called “Sobre a CognitoAccessForAmazonOpenSearch função”. Como alternativa, use o console para configurar a autenticação do Amazon Cognito. O console cria uma função para você.
`An error occurred (ValidationException) when calling the CreateDomain operation: User is not authorized to perform: cognito-idp: action on resource: user pool`	A função especificada em `--cognito-options` não tem permissões para acessar o Amazon Cognito. Verifique se a função tem a `AmazonOpenSearchServiceCognitoAccess` política AWS gerenciada anexada. Como alternativa, use o console para configurar a autenticação do Amazon Cognito. O console cria uma função para você.
`An error occurred (ValidationException) when calling the CreateDomain operation: User pool does not exist`	OpenSearch O serviço não consegue encontrar o grupo de usuários. Confirme se você criou um e se tem o ID correto. Para encontrar o ID, você pode usar o console do Amazon Cognito ou o seguinte comando: AWS CLI `aws cognito-idp list-user-pools --max-results 60 --region region`
`An error occurred (ValidationException) when calling the CreateDomain operation: IdentityPool not found`	OpenSearch O serviço não consegue encontrar o pool de identidades. Confirme se você criou um e se tem o ID correto. Para encontrar o ID, você pode usar o console do Amazon Cognito ou o seguinte comando: AWS CLI `aws cognito-identity list-identity-pools --max-results 60 --region region`

Problema	Solução
`An error occurred (ValidationException) when calling the CreateDomain operation : Domain needs to be specified for user pool`	O grupo de usuários não tem um nome de domínio. Você pode configurar um usando o console do Amazon Cognito ou o seguinte comando da AWS CLI : `aws cognito-idp create-user-pool-domain --domain name --user-pool-id id`

Acessando OpenSearch painéis

Problema	Solução
A página de login não mostra meus provedores de identidade preferenciais.	Verifique se você habilitou o provedor de identidade para o cliente do aplicativo OpenSearch Service, conforme especificado emthe section called “Configuração de provedores de identidade”.
A página de login não parece estar associada à minha organização.	Consulte the section called “(Opcional) Personalização da página de login”.
Minhas credenciais de login não funcionam.	Verifique se você configurou o provedor de identidade conforme especificado em the section called “Configuração de provedores de identidade”. Se você usa o grupo de usuários como seu provedor de identidade, verifique se a conta existe no console do Amazon Cognito.
OpenSearch Os painéis não carregam nem um pouco ou não funcionam corretamente.	A função autenticada do Amazon Cognito precisa de permissões `es:ESHttp*` para o domínio (`/*`) a fim de acessar e usar o Dashboards. Verifique se você adicionou uma política de acesso conforme especificado em the section called “Como permitir a função autenticada”.
Quando eu saio dos OpenSearch Painéis de uma guia, as guias	Quando você sai de uma sessão do OpenSearch Dashboards enquanto usa a autenticação do Amazon

Problema	Solução
restantes exibem uma mensagem informando que o token de atualizaç ão foi revogado.	Cognito OpenSearch , o Service executa AdminUser GlobalSignOutuma operação que desconecta você de todas as sessões OpenSearch ativas do Dashboards.
`Invalid identity pool configuration. Check assigned IAM roles for this pool.`	O Amazon Cognito não tem permissões para assumir a função do IAM em nome do usuário autenticado. Modifique a relação de confiança da função para incluir: JSON `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": { "Federated": "cognito-identity.amazonaws.com" }, "Action": "sts:AssumeRoleWithWebIdentity", "Condition": { "StringEquals": { "cognito-identity.amazonaws.com:aud": " identity-pool-id " }, "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated" } } }] }`
`Token is not from a supported provider of this identity pool.`	Este erro incomum pode ocorrer quando você remover o cliente do aplicativo a partir do grupo de usuários. Tente abrir o Dashboards em uma nova sessão do navegador.

Desabilitando a autenticação do Amazon Cognito para painéis OpenSearch

Use o procedimento a seguir para desabilitar a autenticação do Amazon Cognito para Dashboards.

Para desabilitar a autenticação do Amazon Cognito para Dashboards (console)

1. Abra o console do Amazon OpenSearch Service.
2. Em Domínios, escolha o domínio que deseja configurar.
3. Escolha Ações, Editar configuração de segurança.
4. Desmarque a opção Habilitar autenticação do Amazon Cognito.
5. Escolha Salvar alterações.

> **Important**
>
> Se você não precisar mais do grupo de usuários e do grupo de identidades do Amazon Cognito, exclua-os. Caso contrário, você continuará a ser cobrado.

Excluindo domínios que usam a autenticação do Amazon Cognito para painéis OpenSearch

Para evitar que domínios que usam a autenticação do Amazon Cognito para painéis fiquem presos em um estado de configuração de processamento, OpenSearch exclua os domínios do Serviço antes de excluir seus grupos de usuários e identidades associados do Amazon Cognito.

Usando funções vinculadas a serviços para o Amazon Service OpenSearch

O Amazon OpenSearch Service usa funções vinculadas ao serviço AWS Identity and Access Management (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente ao OpenSearch Serviço. As funções vinculadas ao serviço são predefinidas pelo OpenSearch Serviço e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

Uma função vinculada ao serviço facilita a configuração do OpenSearch Serviço porque você não precisa adicionar manualmente as permissões necessárias. OpenSearch O serviço define as

permissões de suas funções vinculadas ao serviço e, a menos que definido de outra forma, somente o OpenSearch serviço pode assumir suas funções. As permissões definidas incluem a política de confiança e a política de permissões, que não pode ser anexada a nenhuma outra entidade do IAM. Para atualizações das políticas de funções e permissões vinculadas ao serviço, consulte Histórico de documentos do Amazon OpenSearch Service.

Para obter informações sobre outros serviços que oferecem suporte a funções vinculadas a serviços, consulte AWS Serviços que funcionam com IAM e procure os serviços que têm Sim na coluna Funções vinculadas ao serviço. Escolha um Sim com um link para visualizar a documentação do perfil vinculado a esse serviço.

Tópicos

- Uso de perfis vinculados ao serviço para criar domínios de VPC e fontes de dados de consulta direta
- Usando funções vinculadas a serviços para criar OpenSearch coleções sem servidor
- Usando funções vinculadas a serviços para criar pipelines de ingestão OpenSearch

Uso de perfis vinculados ao serviço para criar domínios de VPC e fontes de dados de consulta direta

O Amazon OpenSearch Service usa funções vinculadas ao serviço AWS Identity and Access Management (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente ao OpenSearch Serviço. As funções vinculadas ao serviço são predefinidas pelo OpenSearch Serviço e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

OpenSearch O serviço usa a função vinculada ao serviço chamada AWSServiceRoleForAmazonOpenSearchService, que fornece as permissões mínimas do Amazon EC2 e do Elastic Load Balancing necessárias para que a função permita o acesso à VPC para um domínio ou uma fonte de dados de consulta direta.

Função legada do Elasticsearch

O Amazon OpenSearch Service usa uma função vinculada ao serviço chamada. `AWSServiceRoleForAmazonOpenSearchService` Suas contas também podem conter uma função vinculada ao serviço herdada chamada

`AWSServiceRoleForAmazonElasticsearchService`, que funciona com os endpoints obsoletos da API Elasticsearch.

Se a função legada do Elasticsearch não existir em sua conta, o OpenSearch Service cria automaticamente uma nova função OpenSearch vinculada ao serviço na primeira vez que você cria um domínio. OpenSearch Caso contrário, sua conta continuará usando a função Elasticsearch. Para que essa criação automática seja bem-sucedida, você precisa ter permissões para a ação `iam:CreateServiceLinkedRole`.

Permissões

O perfil vinculado ao serviço `AWSServiceRoleForAmazonOpenSearchService` confia nos seguintes serviços para aceitar o perfil:

- `opensearchservice.amazonaws.com`

A política de permissões de função nomeada `AmazonOpenSearchServiceRolePolicy`permite que o OpenSearch Serviço conclua as seguintes ações nos recursos especificados:

- Ação: `acm:DescribeCertificate` em `*`
- Ação: `cloudwatch:PutMetricData` em `*`
- Ação: `ec2:CreateNetworkInterface` em `*`
- Ação: `ec2:DeleteNetworkInterface` em `*`
- Ação: `ec2:DescribeNetworkInterfaces` em `*`
- Ação: `ec2:ModifyNetworkInterfaceAttribute` em `*`
- Ação: `ec2:DescribeSecurityGroups` em `*`
- Ação: `ec2:DescribeSubnets` em `*`
- Ação: `ec2:DescribeVpcs` em `*`
- Ação: `ec2:CreateTags` em todas as interfaces de rede e endpoints da VPC
- Ação: `ec2:DescribeTags` em `*`
- Ação: `ec2:CreateVpcEndpoint` em todos os grupos de segurança VPCs, sub-redes e tabelas de rotas, bem como em todos os VPC endpoints quando a solicitação contém a tag `OpenSearchManaged=true`

- Ação: `ec2:ModifyVpcEndpoint` em todos os grupos de segurança VPCs, sub-redes e tabelas de rotas, bem como em todos os VPC endpoints quando a solicitação contém a tag `OpenSearchManaged=true`
- Ação: `ec2:DeleteVpcEndpoints` em todos os endpoints quando a solicitação contiver a tag `OpenSearchManaged=true`
- Ação: `ec2:AssignIpv6Addresses` em *
- Ação: `ec2:UnAssignIpv6Addresses` em *
- Ação: `elasticloadbalancing:AddListenerCertificates` em *
- Ação: `elasticloadbalancing:RemoveListenerCertificates` em *

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua um perfil vinculado a serviço. Para saber mais, consulte Permissões de função vinculada a serviços no Guia do Usuário do IAM.

Criando uma função vinculada ao serviço

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você cria um domínio habilitado para VPC ou uma fonte de dados de consulta direta usando o Console de gerenciamento da AWS, o OpenSearch Service cria a função vinculada ao serviço para você. Para que essa criação automática seja bem-sucedida, você precisa ter permissões para a ação `iam:CreateServiceLinkedRole`.

Também é possível usar o console do IAM, a CLI do IAM ou a API do IAM para criar manualmente uma função vinculada ao serviço. Para saber mais, consulte Criação de uma função vinculada ao serviço no Manual do usuário do IAM.

Edição da função vinculada ao serviço

OpenSearch O serviço não permite que você edite a função `AWSServiceRoleForAmazonOpenSearchService` vinculada ao serviço. Depois que você criar um perfil vinculado ao serviço, não poderá alterar o nome do perfil, pois várias entidades podem fazer referência ao perfil. No entanto, você poderá editar a descrição do perfil usando o IAM. Para saber mais, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Excluindo uma função vinculada ao serviço

Se você não precisar mais usar um recurso ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Dessa forma, você não tem uma entidade não utilizada que não seja

monitorada ativamente ou mantida. No entanto, você deve limpar suo perfil vinculado ao serviço para excluí-la manualmente.

Limpar a função vinculada ao serviço

Antes de você poder usar o IAM para excluir uma função vinculada ao serviço, você deve primeiro confirmar que a função não tem sessões ativas e remover quaisquer recursos usados pela função.

Para verificar se a função vinculada ao serviço tem uma sessão ativa no console do IAM

1. Faça login no Console de gerenciamento da AWS e abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação do console do IAM, escolha Funções. A seguir, selecione o nome (não a caixa de seleção) da função `AWSServiceRoleForAmazonOpenSearchService`.
3. Na página Resumo para a função selecionada, escolha a guia Consultor de Acesso.
4. Na guia Consultor de Acesso, revise a atividade recente para a função vinculada ao serviço.

> **Note**
>
> Se você não tiver certeza se o OpenSearch Serviço está usando a `AWSServiceRoleForAmazonOpenSearchService` função, tente excluir a função. Se o serviço estiver usando a função, a exclusão falhará e você poderá visualizar os recursos usando a função. Se a função estiver sendo usada, você deverá aguardar o término da sessão antes de excluir a função e and/or excluir os recursos usando a função. Não é possível revogar a sessão de uma função vinculada a um serviço.

Exclusão manual de uma função vinculada ao serviço

Exclua funções vinculadas ao serviço do console, da API ou da CLI AWS do IAM. Para obter instruções, consulte Exclusão de uma função vinculada ao serviço no Manual do usuário do IAM.

Usando funções vinculadas a serviços para criar OpenSearch coleções sem servidor

OpenSearch O Serverless usa funções vinculadas a serviços AWS Identity and Access Management (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente ao OpenSearch Serviço. As funções vinculadas ao serviço são predefinidas pelo OpenSearch

Serviço e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

OpenSearch O Serverless usa a função vinculada ao serviço chamada AWSServiceRoleForAmazonOpenSearchServerless, que fornece as permissões necessárias para que a função publique métricas relacionadas à tecnologia sem servidor CloudWatch em sua conta. A política de permissões de função associada à AWSService RoleForAmazonOpenSearchServerless é nomeada`AmazonOpenSearchServerlessServiceRolePolicy`. Para obter mais informações sobre a política, consulte o Guia AmazonOpenSearchServerlessServiceRolePolicyde referência de políticas AWS gerenciadas.

Permissões de função vinculadas a serviços para Serverless OpenSearch

OpenSearch O Serverless usa a função vinculada ao serviço chamada AWSServiceRoleForAmazonOpenSearchServerless, que permite que o OpenSearch Serverless chame serviços em seu nome. AWS

A função AWSService RoleForAmazonOpenSearchServerless vinculada ao serviço confia nos seguintes serviços para assumir a função:

- `observability.aoss.amazonaws.com`

A política de permissões de função nomeada `AmazonOpenSearchServerlessServiceRolePolicy` permite que o OpenSearch Serverless conclua as seguintes ações nos recursos especificados:

- Ação: `cloudwatch:PutMetricData` em todos os recursos da AWS .

> **Note**
>
> A política inclui a chave de condição`{"StringEquals": {"cloudwatch:namespace": "AWS/AOSS"}}`, o que significa que a função vinculada ao serviço só pode enviar dados métricos para o `AWS/AOSS` CloudWatch namespace.

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua um perfil vinculado a serviço. Para obter mais informações, consulte Permissões de perfil vinculado a serviços no Guia do usuário do IAM.

Criando a função vinculada ao serviço para Serverless OpenSearch

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você cria uma coleção OpenSearch Serverless na, na ou na AWS API Console de gerenciamento da AWS AWS CLI, a OpenSearch Serverless cria a função vinculada ao serviço para você.

> **Note**
>
> Na primeira vez que você criar uma coleção, deverá receber a atribuição de `iam:CreateServiceLinkedRole` em uma política baseada em identidade.

Se excluir esse perfil vinculado ao serviço e precisar criá-lo novamente, será possível usar esse mesmo processo para recriar o perfil em sua conta. Quando você cria uma coleção OpenSearch Serverless, OpenSearch Serverless cria a função vinculada ao serviço para você novamente.

Você também pode usar o console do IAM para criar uma função vinculada ao serviço com o caso de uso do Amazon OpenSearch Serverless. Na AWS CLI ou na AWS API, crie uma função vinculada ao serviço com o nome do `observability.aoss.amazonaws.com` serviço:

```
aws iam create-service-linked-role --aws-service-name
 "observability.aoss.amazonaws.com"
```

Para saber mais, consulte Criar um perfil vinculado a serviço no Guia do usuário do IAM. Se você excluir essa função vinculada ao serviço, será possível usar esse mesmo processo para criar a função novamente.

Editando a função vinculada ao serviço para Serverless OpenSearch

OpenSearch O Serverless não permite que você edite a função vinculada ao AWSService RoleForAmazonOpenSearchServerless serviço. Depois que você criar um perfil vinculado ao serviço, não poderá alterar o nome do perfil, pois várias entidades podem fazer referência ao perfil. No entanto, você poderá editar a descrição do perfil usando o IAM. Para saber mais, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Excluindo a função vinculada ao serviço para Serverless OpenSearch

Se você não precisar mais usar um atributo ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Isso evita que você tenha uma entidade não utilizada que não seja

ativamente monitorada ou mantida. No entanto, você deve limpar os recursos de seu perfil vinculado ao serviço antes de excluí-lo manualmente.

Para excluir o AWSServiceRoleForAmazonOpenSearchServerless, você deve primeiro excluir todas as coleções OpenSearch sem servidor em seu. Conta da AWS

> **Note**
>
> Se o OpenSearch Serverless estiver usando a função quando você tentar excluir os recursos, a exclusão poderá falhar. Se isso acontecer, espere alguns minutos e tente a operação novamente.

Como excluir manualmente o perfil vinculado ao serviço usando o IAM

Use o console do IAM AWS CLI, o ou a AWS API para excluir a função AWSService RoleForAmazonOpenSearchServerless vinculada ao serviço. Para saber mais, consulte Excluir um perfil vinculado ao serviço no Guia do usuário do IAM.

Regiões suportadas para funções vinculadas a serviços OpenSearch sem servidor

OpenSearch O Serverless oferece suporte ao uso da função AWSService RoleForAmazonOpenSearchServerless vinculada ao serviço em todas as regiões em que OpenSearch o Serverless está disponível. Para obter uma lista das regiões suportadas, consulte os endpoints e cotas do Amazon OpenSearch Serverless no. Referência geral da AWS

Usando funções vinculadas a serviços para criar pipelines de ingestão OpenSearch

O Amazon OpenSearch Ingestion usa funções AWS Identity and Access Management vinculadas a serviços (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente à OpenSearch ingestão. As funções vinculadas ao serviço são predefinidas pelo OpenSearch Inestion e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

OpenSearch A ingestão usa a função vinculada ao serviço chamada AWSServiceRoleForAmazonOpenSearchIngestionService, exceto quando você usa uma VPC autogerenciada. Nesse caso, ela usa a função vinculada ao serviço chamada. AWSServiceRoleForOpensearchIngestionSelfManagedVpce A política anexada fornece as

permissões necessárias para que a função crie uma nuvem privada virtual (VPC) entre sua conta e a OpenSearch Ingestão e publique CloudWatch métricas em sua conta.

Permissões

O perfil vinculado ao serviço `AWSServiceRoleForAmazonOpenSearchIngestionService` confia nos seguintes serviços para aceitar o perfil:

- `osis.amazonaws.com`

A política de permissões de função chamada `AmazonOpenSearchIngestionServiceRolePolicy` permite que o OpenSearch Inestion conclua as seguintes ações nos recursos especificados:

- Ação: `cloudwatch:PutMetricData` em `cloudwatch:namespace": "AWS/OSIS"`
- Ação: `ec2:CreateTags` em `arn:aws:ec2:*:*:network-interface/*`
- Ação: `ec2:CreateVpcEndpoint` em `*`
- Ação: `ec2:DeleteVpcEndpoints` em `*`
- Ação: `ec2:DescribeSecurityGroups` em `*`
- Ação: `ec2:DescribeSubnets` em `*`
- Ação: `ec2:DescribeVpcEndpoints` em `*`
- Ação: `ec2:ModifyVpcEndpoint` em `*`

O perfil vinculado ao serviço `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` confia nos seguintes serviços para aceitar o perfil:

- `self-managed-vpce.osis.amazonaws.com`

A política de permissões de função chamada `OpenSearchIngestionSelfManagedVpcePolicy` permite que o OpenSearch Inestion conclua as seguintes ações nos recursos especificados:

- Ação: `ec2:DescribeSubnets` em `*`
- Ação: `ec2:DescribeSecurityGroups` em `*`
- Ação: `ec2:DescribeVpcEndpoints` em `*`
- Ação: `cloudwatch:PutMetricData` em `cloudwatch:namespace": "AWS/OSIS"`

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua um perfil vinculado a serviço. Para obter mais informações, consulte Permissões de perfil vinculado a serviços no Guia do usuário do IAM.

Criação da função vinculada ao serviço para ingestão OpenSearch

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você cria um pipeline de OpenSearch ingestão na Console de gerenciamento da AWS, na ou na AWS API AWS CLI, a OpenSearch ingestão cria a função vinculada ao serviço para você.

Se excluir esse perfil vinculado ao serviço e precisar criá-lo novamente, será possível usar esse mesmo processo para recriar o perfil em sua conta. Quando você cria um pipeline OpenSearch de ingestão, o OpenSearch Inestion cria a função vinculada ao serviço para você novamente.

Editando a função vinculada ao serviço para ingestão OpenSearch

OpenSearch A ingestão não permite que você edite a função vinculada ao `AWSServiceRoleForAmazonOpenSearchIngestionService` serviço. Depois que criar um perfil vinculado ao serviço, você não poderá alterar o nome do perfil, pois várias entidades podem fazer referência a ele. No entanto, será possível editar a descrição do perfil usando o IAM. Para saber mais, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Excluindo a função vinculada ao serviço para ingestão OpenSearch

Se você não precisar mais usar um atributo ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ativamente ou mantida. No entanto, você deve limpar os recursos de sua função vinculada ao serviço antes de exclui-la manualmente.

Limpar um perfil vinculado ao serviço

Antes de usar o IAM para excluir um perfil vinculado ao serviço, você deverá excluir qualquer recurso usado pelo perfil.

> **Note**
>
> Se o OpenSearch Inestion estiver usando a função quando você tentar excluir os recursos, a exclusão poderá falhar. Se isso acontecer, espere alguns minutos e tente a operação novamente.

Para excluir os recursos de OpenSearch ingestão usados pela função **`AWSServiceRoleForAmazonOpenSearchIngestionService`** ou **`AWSServiceRoleForOpensearchIngestionSelfManagedVpce`**

1. Navegue até o console do Amazon OpenSearch Service e escolha Ingestão.
2. Exclua todos os pipelines. Para instruções, consulte the section called “Exclusão de pipelines”.

Exclua a função vinculada ao serviço para ingestão OpenSearch

Você pode usar o console OpenSearch de ingestão para excluir uma função vinculada ao serviço.

Para excluir uma função vinculada ao serviço (console)

1. Navegue até o console do IAM.
2. Escolha Funções e pesquise a AWSServiceRoleForOpensearchIngestionSelfManagedVpcefunção AWSServiceRoleForAmazonOpenSearchIngestionServiceou.
3. Selecione a função e escolha Excluir.

Código de exemplo para o Amazon OpenSearch Service

Este capítulo contém um exemplo de código comum para trabalhar com o Amazon OpenSearch Service: assinatura de solicitações HTTP em várias linguagens de programação, compactação de corpos de solicitações HTTP e uso do AWS SDKs para criar domínios.

Tópicos

- Compatibilidade com clientes Elasticsearch
- Compressão de solicitações HTTP no Amazon OpenSearch Service
- Usando o AWS SDKs para interagir com o Amazon OpenSearch Service

Compatibilidade com clientes Elasticsearch

As versões mais recentes dos clientes Elasticsearch podem incluir verificações de licença ou versão que interrompem artificialmente a compatibilidade. A tabela a seguir inclui recomendações sobre quais versões desses clientes devem ser usadas para melhor compatibilidade com o OpenSearch Serviço.

> **Important**
>
> Essas versões do cliente estão desatualizadas e não estão atualizadas com as dependências mais recentes, incluindo o Log4j. É altamente recomendável usar as OpenSearch versões dos clientes sempre que possível.

Cliente	Versão recomendada
Cliente REST de baixo nível Java	7.13.4
Cliente REST de alto nível Java	7.13.4
Cliente Elasticsearch Python	7.13.4
Cliente Elasticsearch Ruby	7.13.3
Cliente Elasticsearch Node.js	7.13.0

Compressão de solicitações HTTP no Amazon OpenSearch Service

Você pode compactar solicitações e respostas HTTP nos domínios do Amazon OpenSearch Service usando a compressão gzip. A compactação gzip pode ajudar a reduzir o tamanho de seus documentos e reduzir a utilização e a latência da largura de banda, levando a velocidades de transferência aprimoradas.

A compactação Gzip é compatível com todos os domínios que executam o Elasticsearch 6.0 OpenSearch ou posterior. Alguns OpenSearch clientes têm suporte integrado para compressão gzip e muitas linguagens de programação têm bibliotecas que simplificam o processo.

Habilitação da compactação gzip

Não deve ser confundido com OpenSearch configurações semelhantes, `http_compression.enabled` é específico do OpenSearch Serviço e ativa ou desativa a compactação gzip em um domínio. Domínios em execução OpenSearch ou Elasticsearch 7. x têm a compactação gzip ativada por padrão, enquanto os domínios que executam o Elasticsearch 6. x desabilite-o por padrão.

Para habilitar a compactação gzip, envie a seguinte solicitação:

```
PUT _cluster/settings
{
  "persistent" : {
    "http_compression.enabled": true
  }
}
```

As solicitações para `_cluster/settings` devem ser descompactadas, então você talvez precise usar um cliente separado ou uma solicitação HTTP padrão para atualizar as configurações do cluster.

Para confirmar que você habilitou a compactação gzip com êxito, envie a seguinte solicitação:

```
GET _cluster/settings?include_defaults=true
```

Verifique se você vê a seguinte configuração na resposta:

```
...
```

```
"http_compression": {
  "enabled": "true"
}
...
```

Cabeçalhos obrigatórios

Ao incluir um corpo de solicitação compactado com gzip, mantenha o cabeçalho `Content-Type: application/json` padrão e adicione o cabeçalho `Content-Encoding: gzip`. Para aceitar uma resposta compactada por gzip, adicione o cabeçalho `Accept-Encoding: gzip` também. Se um OpenSearch cliente oferecer suporte à compressão gzip, provavelmente incluirá esses cabeçalhos automaticamente.

Código de exemplo (Python 3)

O exemplo a seguir usa opensearch-py para executar a compactação e enviar a solicitação. Esse código assina a solicitação usando suas credenciais do IAM.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3

host = '' # e.g. my-test-domain.us-east-1.es.amazonaws.com
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Create the client.
search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    http_compress = True, # enables gzip compression for request bodies
    connection_class = RequestsHttpConnection
)

document = {
  "title": "Moneyball",
```

```
  "director": "Bennett Miller",
  "year": "2011"
}

# Send the request.
print(search.index(index='movies', id='1', body=document, refresh=True))

# print(search.index(index='movies', doc_type='_doc', id='1', body=document,
 refresh=True))
```

Alternativamente, você pode especificar os cabeçalhos apropriados, compactar o corpo da solicitação e usar uma biblioteca HTTP padrão como Requests. Este código assina a solicitação usando credenciais básicas HTTP, às quais seu domínio pode oferecer suporte se você usa o controle de acesso refinado.

```
import requests
import gzip
import json

base_url = '' # The domain with https:// and a trailing slash. For example, https://my-
test-domain.us-east-1.es.amazonaws.com/
auth = ('master-user', 'master-user-password') # For testing only. Don't store
 credentials in code.

headers = {'Accept-Encoding': 'gzip', 'Content-Type': 'application/json',
           'Content-Encoding': 'gzip'}

document = {
  "title": "Moneyball",
  "director": "Bennett Miller",
  "year": "2011"
}

# Compress the document.
compressed_document = gzip.compress(json.dumps(document).encode())

# Send the request.
path = 'movies/_doc?refresh=true'
url = base_url + path
response = requests.post(url, auth=auth, headers=headers, data=compressed_document)
print(response.status_code)
print(response.text)
```

Usando o AWS SDKs para interagir com o Amazon OpenSearch Service

Esta seção inclui exemplos de como usar o AWS SDKs para interagir com a API de configuração do Amazon OpenSearch Service. Esses exemplos de código mostram como criar, atualizar e excluir domínios OpenSearch de serviço.

Java

Esta seção inclui exemplos das versões 1 e 2 do AWS SDK para Java.

Version 2

Este exemplo usa o OpenSearchClientBuilderconstrutor da versão 2 do AWS SDK para Java para criar um OpenSearch domínio, atualizar sua configuração e excluí-lo. Remova os comentários das chamadas para `waitForDomainProcessing` (e comente as chamadas para `deleteDomain`) para permitir que o domínio fique online e seja utilizável.

```
package com.example.samples;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.opensearch.OpenSearchClient;
import software.amazon.awssdk.services.opensearch.model.ClusterConfig;
import software.amazon.awssdk.services.opensearch.model.EBSOptions;
import software.amazon.awssdk.services.opensearch.model.CognitoOptions;
import software.amazon.awssdk.services.opensearch.model.NodeToNodeEncryptionOptions;
import software.amazon.awssdk.services.opensearch.model.CreateDomainRequest;
import software.amazon.awssdk.services.opensearch.model.CreateDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainRequest;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainResponse;
import software.amazon.awssdk.services.opensearch.model.OpenSearchException;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;

/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
```

```
 * and delete Amazon OpenSearch Service domains.
 */

public class OpenSearchSample {

    public static void main(String[] args) {

     String domainName = "my-test-domain";

     // Build the client using the default credentials chain.
        // You can use the CLI and run `aws configure` to set access key, secret
        // key, and default region.

        OpenSearchClient client = OpenSearchClient.builder()
          // Unnecessary, but lets you use a region different than your default.
          .region(Region.US_EAST_1)
          // Unnecessary, but if desired, you can use a different provider chain.
          .credentialsProvider(DefaultCredentialsProvider.create())
                     .build();

        // Create a new domain, update its configuration, and delete it.
        createDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        updateDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        deleteDomain(client, domainName);
    }

    /**
     * Creates an Amazon OpenSearch Service domain with the specified options.
     * Some options require other Amazon Web Services resources, such as an Amazon
 Cognito user pool
     * and identity pool, whereas others require just an instance type or instance
     * count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain you want to create
     */

    public static void createDomain(OpenSearchClient client, String domainName) {

     // Create the request and set the desired configuration options
```

```
        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .dedicatedMasterEnabled(true)
                    .dedicatedMasterCount(3)
                    // Small, inexpensive instance types for testing. Not
 recommended for production.
                    .dedicatedMasterType("t2.small.search")
                    .instanceType("t2.small.search")
                    .instanceCount(5)
                    .build();

            // Many instance types require EBS storage.
            EBSOptions ebsOptions = EBSOptions.builder()
                    .ebsEnabled(true)
                    .volumeSize(10)
                    .volumeType("gp2")
                    .build();

            NodeToNodeEncryptionOptions encryptionOptions =
 NodeToNodeEncryptionOptions.builder()
                    .enabled(true)
                    .build();

            CreateDomainRequest createRequest = CreateDomainRequest.builder()
                    .domainName(domainName)
                    .engineVersion("OpenSearch_1.0")
                    .clusterConfig(clusterConfig)
                    .ebsOptions(ebsOptions)
                    .nodeToNodeEncryptionOptions(encryptionOptions)
                    // You can uncomment this line and add your account ID, a
 username, and the
                    // domain name to add an access policy.
                    // .accessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")
                    .build();

            // Make the request.
            System.out.println("Sending domain creation request...");
            CreateDomainResponse createResponse =
 client.createDomain(createRequest);
```

```
            System.out.println("Domain status:
"+createResponse.domainStatus().toString());
            System.out.println("Domain ID:
"+createResponse.domainStatus().domainId());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain to update
     */

    public static void updateDomain(OpenSearchClient client, String domainName) {

     // Updates the domain to use three data instances instead of five.
        // You can uncomment the Cognito line and fill in the strings to enable
Cognito
        // authentication for OpenSearch Dashboards.

        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .instanceCount(5)
                    .build();

            CognitoOptions cognitoOptions = CognitoOptions.builder()
                    .enabled(true)
                    .userPoolId("user-pool-id")
                    .identityPoolId("identity-pool-id")
                    .roleArn("role-arn")
                    .build();
```

```
            UpdateDomainConfigRequest updateRequest =
UpdateDomainConfigRequest.builder()
                    .domainName(domainName)
                    .clusterConfig(clusterConfig)
                    //.cognitoOptions(cognitoOptions)
                    .build();

            System.out.println("Sending domain update request...");
            UpdateDomainConfigResponse updateResponse =
client.updateDomainConfig(updateRequest);
            System.out.println("Domain config:
"+updateResponse.domainConfig().toString());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
     * several minutes.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain that you want to delete
     */

    public static void deleteDomain(OpenSearchClient client, String domainName) {

        try {

            DeleteDomainRequest deleteRequest = DeleteDomainRequest.builder()
                    .domainName(domainName)
                    .build();

            System.out.println("Sending domain deletion request...");
            DeleteDomainResponse deleteResponse =
client.deleteDomain(deleteRequest);
            System.out.println("Domain status: "+deleteResponse.toString());
```

```
        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
     * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
     * take a similar amount of time. This method checks every 15 seconds and
finishes only when
     * the domain's processing status changes to false.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain that you want to check
     */

    public static void waitForDomainProcessing(OpenSearchClient client, String
domainName) {
        // Create a new request to check the domain status.
        DescribeDomainRequest describeRequest = DescribeDomainRequest.builder()
                .domainName(domainName)
                .build();

        // Every 15 seconds, check whether the domain is processing.
        DescribeDomainResponse describeResponse =
client.describeDomain(describeRequest);
        while (describeResponse.domainStatus().processing()) {
            try {
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
                describeResponse = client.describeDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        // Once we exit that loop, the domain is available
```

```
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description: "+describeResponse.toString());
    }
}
```

Version 1

Este exemplo usa o AWSElasticsearchClientBuilderconstrutor da versão 1 do AWS SDK para Java para criar um domínio legado do Elasticsearch, atualizar sua configuração e excluí-lo. Remova os comentários das chamadas para `waitForDomainProcessing` (e comente as chamadas para `deleteDomain`) para permitir que o domínio fique online e seja utilizável.

```
package com.amazonaws.samples;

import java.util.concurrent.TimeUnit;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.elasticsearch.AWSElasticsearch;
import com.amazonaws.services.elasticsearch.AWSElasticsearchClientBuilder;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainResult;
import
 com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.EBSOptions;
import com.amazonaws.services.elasticsearch.model.ElasticsearchClusterConfig;
import com.amazonaws.services.elasticsearch.model.ResourceNotFoundException;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigRequest;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigResult;
import com.amazonaws.services.elasticsearch.model.VolumeType;

/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
 * and delete Amazon OpenSearch Service domains.
 */

public class OpenSearchSample {
```

```
   public static void main(String[] args) {

       final String domainName = "my-test-domain";

       // Build the client using the default credentials chain.
       // You can use the CLI and run `aws configure` to set access key, secret
       // key, and default region.
       final AWSElasticsearch client = AWSElasticsearchClientBuilder
               .standard()
               // Unnecessary, but lets you use a region different than your
default.
               .withRegion(Regions.US_WEST_2)
               // Unnecessary, but if desired, you can use a different provider
chain.
               .withCredentials(new DefaultAWSCredentialsProviderChain())
               .build();

       // Create a new domain, update its configuration, and delete it.
       createDomain(client, domainName);
       // waitForDomainProcessing(client, domainName);
       updateDomain(client, domainName);
       // waitForDomainProcessing(client, domainName);
       deleteDomain(client, domainName);
   }

   /**
    * Creates an Amazon OpenSearch Service domain with the specified options.
    * Some options require other Amazon Web Services resources, such as an Amazon
Cognito user pool
    * and identity pool, whereas others require just an instance type or instance
    * count.
    *
    * @param client
    *            The client to use for the requests to Amazon OpenSearch Service
    * @param domainName
    *            The name of the domain you want to create
    */
   private static void createDomain(final AWSElasticsearch client, final String
domainName) {

       // Create the request and set the desired configuration options
       CreateElasticsearchDomainRequest createRequest = new
CreateElasticsearchDomainRequest()
```

```
                .withDomainName(domainName)
                .withElasticsearchVersion("7.10")
                .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                        .withDedicatedMasterEnabled(true)
                        .withDedicatedMasterCount(3)
                        // Small, inexpensive instance types for testing. Not
 recommended for production
                        // domains.
                        .withDedicatedMasterType("t2.small.elasticsearch")
                        .withInstanceType("t2.small.elasticsearch")
                        .withInstanceCount(5))
                // Many instance types require EBS storage.
                .withEBSOptions(new EBSOptions()
                        .withEBSEnabled(true)
                        .withVolumeSize(10)
                        .withVolumeType(VolumeType.Gp2));
                // You can uncomment this line and add your account ID, a username,
 and the
                // domain name to add an access policy.
                // .withAccessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")


        // Make the request.
        System.out.println("Sending domain creation request...");
        CreateElasticsearchDomainResult createResponse =
 client.createElasticsearchDomain(createRequest);
        System.out.println("Domain creation response from Amazon OpenSearch
 Service:");
        System.out.println(createResponse.getDomainStatus().toString());
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
 such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
```

```
     *            The name of the domain to update
     */
    private static void updateDomain(final AWSElasticsearch client, final String
domainName) {
        try {
            // Updates the domain to use three data instances instead of five.
            // You can uncomment the Cognito lines and fill in the strings to enable
Cognito
            // authentication for OpenSearch Dashboards.
            final UpdateElasticsearchDomainConfigRequest updateRequest = new
UpdateElasticsearchDomainConfigRequest()
                    .withDomainName(domainName)
                    // .withCognitoOptions(new CognitoOptions()
                            // .withEnabled(true)
                            // .withUserPoolId("user-pool-id")
                            // .withIdentityPoolId("identity-pool-id")
                            // .withRoleArn("role-arn")
                    .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                            .withInstanceCount(3));

            System.out.println("Sending domain update request...");
            final UpdateElasticsearchDomainConfigResult updateResponse = client
                    .updateElasticsearchDomainConfig(updateRequest);
            System.out.println("Domain update response from Amazon OpenSearch
Service:");
            System.out.println(updateResponse.toString());
        } catch (ResourceNotFoundException e) {
            System.out.println("Domain not found. Please check the domain name.");
        }
    }

    /**
     * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
     * several minutes.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain that you want to delete
     */
    private static void deleteDomain(final AWSElasticsearch client, final String
domainName) {
        try {
```

```
            final DeleteElasticsearchDomainRequest deleteRequest = new
DeleteElasticsearchDomainRequest()
                    .withDomainName(domainName);

            System.out.println("Sending domain deletion request...");
            final DeleteElasticsearchDomainResult deleteResponse =
client.deleteElasticsearchDomain(deleteRequest);
            System.out.println("Domain deletion response from Amazon OpenSearch
Service:");
            System.out.println(deleteResponse.toString());
        } catch (ResourceNotFoundException e) {
            System.out.println("Domain not found. Please check the domain name.");
        }
    }

    /**
     * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
     * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
     * take a similar amount of time. This method checks every 15 seconds and
finishes only when
     * the domain's processing status changes to false.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain that you want to check
     */
    private static void waitForDomainProcessing(final AWSElasticsearch client, final
String domainName) {
        // Create a new request to check the domain status.
        final DescribeElasticsearchDomainRequest describeRequest = new
DescribeElasticsearchDomainRequest()
                .withDomainName(domainName);

        // Every 15 seconds, check whether the domain is processing.
        DescribeElasticsearchDomainResult describeResponse =
client.describeElasticsearchDomain(describeRequest);
        while (describeResponse.getDomainStatus().isProcessing()) {
            try {
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
```

```
                describeResponse =
 client.describeElasticsearchDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        // Once we exit that loop, the domain is available
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description response from Amazon OpenSearch
 Service:");
        System.out.println(describeResponse.toString());
    }
}
```

Python

Este exemplo usa o cliente Python de OpenSearchServicebaixo nível AWS SDK para Python (Boto) do para criar um domínio, atualizar sua configuração e excluí-lo.

```
import boto3
import botocore
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-west-2'
)

client = boto3.client('opensearch', config=my_config)

domainName = 'my-test-domain'  # The name of the domain


def createDomain(client, domainName):
    """Creates an Amazon OpenSearch Service domain with the specified options."""
```

```
    response = client.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_1.0',
        ClusterConfig={
            'InstanceType': 't2.small.search',
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies="{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-west-2:123456789012:domain/my-test-domain/*
\"}]}",
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    print("Creating domain...")
    print(response)


def updateDomain(client, domainName):
    """Updates the domain to use three data nodes instead of five."""
    try:
        response = client.update_domain_config(
            DomainName=domainName,
            ClusterConfig={
                'InstanceCount': 3
            }
        )
        print('Sending domain update request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
```

```
            raise error


def deleteDomain(client, domainName):
    """Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes."""
    try:
        response = client.delete_domain(
            DomainName=domainName
        )
        print('Sending domain deletion request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
            raise error


def waitForDomainProcessing(client, domainName):
    """Waits for the domain to finish processing changes."""
    try:
        response = client.describe_domain(
            DomainName=domainName
        )
        # Every 15 seconds, check whether the domain is processing.
        while response["DomainStatus"]["Processing"] == True:
            print('Domain still processing...')
            time.sleep(15)
            response = client.describe_domain(
                DomainName=domainName)

        # Once we exit the loop, the domain is available.
        print('Amazon OpenSearch Service has finished processing changes for your
 domain.')
        print('Domain description:')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
            raise error
```

```
def main():
    """Create a new domain, update its configuration, and delete it."""
    createDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    updateDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    deleteDomain(client, domainName)
```

Nó

Este exemplo usa a versão 3 do SDK JavaScript no OpenSearch cliente Node.js para criar um domínio, atualizar sua configuração e excluí-lo.

```
var {
    OpenSearchClient,
    CreateDomainCommand,
    DescribeDomainCommand,
    UpdateDomainConfigCommand,
    DeleteDomainCommand
} = require("@aws-sdk/client-opensearch");
var sleep = require('sleep');

var client = new OpenSearchClient();

var domainName = 'my-test-domain'

// Create a new domain, update its configuration, and delete it.
createDomain(client, domainName)
waitForDomainProcessing(client, domainName)
updateDomain(client, domainName)
waitForDomainProcessing(client, domainName)
deleteDomain(client, domainName)

async function createDomain(client, domainName) {
    // Creates an Amazon OpenSearch Service domain with the specified options.
    var command = new CreateDomainCommand({
        DomainName: domainName,
        EngineVersion: 'OpenSearch_1.0',
        ClusterConfig: {
        'InstanceType': 't2.small.search',
        'InstanceCount': 5,
```

```
        'DedicatedMasterEnabled': 'True',
        'DedicatedMasterType': 't2.small.search',
        'DedicatedMasterCount': 3
        },
        EBSOptions:{
            'EBSEnabled': 'True',
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies: "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/my-test-domain/*
\"}]}",
         NodeToNodeEncryptionOptions:{
            'Enabled': 'True'
        }
    });
    const response = await client.send(command);
    console.log("Creating domain...");
    console.log(response);
}

async function updateDomain(client, domainName) {
    // Updates the domain to use three data nodes instead of five.
    var command = new UpdateDomainConfigCommand({
        DomainName: domainName,
        ClusterConfig: {
        'InstanceCount': 3
        }
    });
    const response = await client.send(command);
    console.log('Sending domain update request...');
    console.log(response);
}

async function deleteDomain(client, domainName) {
    // Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes.
    var command = new DeleteDomainCommand({
        DomainName: domainName
    });
    const response = await client.send(command);
    console.log('Sending domain deletion request...');
    console.log(response);
```

```
}

async function waitForDomainProcessing(client, domainName) {
    // Waits for the domain to finish processing changes.
    try {
        var command = new DescribeDomainCommand({
            DomainName: domainName
        });
        var response = await client.send(command);

        while (response.DomainStatus.Processing == true) {
            console.log('Domain still processing...')
            await sleep(15000) // Wait for 15 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        // Once we exit the loop, the domain is available.
        console.log('Amazon OpenSearch Service has finished processing changes for your
 domain.');
        console.log('Domain description:');
        console.log(response);

    } catch (error) {
        if (error.name === 'ResourceNotFoundException') {
            console.log('Domain not found. Please check the domain name.');
            }
    };
}
```

Indexação de dados no Amazon Service OpenSearch

Como o Amazon OpenSearch Service usa uma API REST, existem vários métodos para indexar documentos. Você pode usar os clientes padrão, como curl, ou qualquer linguagem de programação que possa enviar solicitações HTTP. Para simplificar ainda mais o processo de interação com ele, o OpenSearch Service tem clientes para várias linguagens de programação. Os usuários avançados podem ir diretamente para the section called “Carregando dados de streaming no OpenSearch Serviço”.

É altamente recomendável que você use o Amazon OpenSearch Ingestion para ingerir dados, que é um coletor de dados totalmente gerenciado criado dentro do Service. OpenSearch Para obter mais informações, consulte Amazon OpenSearch Ingestion.

Para uma introdução à indexação, consulte a OpenSearchdocumentação.

Restrições de nomenclatura para índices

OpenSearch Os índices de serviço têm as seguintes restrições de nomenclatura:

- Todas as letras devem estar em minúscula.
- Os nomes de índice não podem começar com `_` ou `-`.
- Os nomes de índice não podem conter espaços, vírgulas, `:`, `"`, `*`, `+`, `/`, `\`, `|`, `?`, `#`, `>` ou `<`.

Não inclua informações confidenciais nos nomes de índice, tipo ou ID do documento. OpenSearch O serviço usa esses nomes em seus Identificadores Uniformes de Recursos (URIs). Servidores e aplicativos geralmente registram solicitações HTTP, o que pode levar à exposição desnecessária de dados se URIs contiverem informações confidenciais:

```
2018-10-03T23:39:43 198.51.100.14 200 "GET https://opensearch-domain/dr-jane-doe/flu-patients-2018/202-555-0100/ HTTP/1.1"
```

Mesmo se não tivesse permissões para visualizar o documento JSON associado, você poderia inferir por essa linha de log falsa que um dos pacientes do Dr. Doe cujo número de telefone é 202-555-0100 contraiu gripe em 2018.

Se o OpenSearch Serviço detectar um endereço IP real ou percebido em um nome de índice (por exemplo,`my-index-12.34.56.78.91`), ele mascara o endereço IP. Uma chamada para `_cat/indices` produz a seguinte resposta:

```
green open my-index-x.x.x.x.91    soY19tBERoKo71WcEScidw 5 1 0 0   2kb  1kb
```

Para evitar confusões desnecessárias, evite incluir endereços IP em nomes de índices.

Redução do tamanho da resposta

As respostas do `_index` e `_bulk` APIs contêm muitas informações. Essas informações podem ser úteis para solução de problemas de solicitações ou para implementar a lógica de tentativas repetidas, mas pode usar consideravelmente a banda larga. Neste exemplo, indexar um documento de 32 bytes resulta em uma resposta de 339 bytes (incluindo cabeçalhos):

```
PUT opensearch-domain/more-movies/_doc/1
{"title": "Back to the Future"}
```

Resposta

```
{
  "_index": "more-movies",
  "_type": "_doc",
  "_id": "1",
  "_version": 4,
  "result": "updated",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 3,
  "_primary_term": 1
}
```

Esse tamanho de resposta pode parecer insignificante, mas se você indexar 1.000.000 documentos por dia — aproximadamente 11,5 documentos por segundo —, 339 bytes por resposta representam 10,17 GB de tráfego de download por mês.

Se os custos de transferência de dados forem uma preocupação, use o `filter_path` parâmetro para reduzir o tamanho da resposta do OpenSearch serviço, mas tenha cuidado para não filtrar os campos necessários para identificar ou repetir solicitações com falha. Esses campos variam de acordo com o cliente. O `filter_path` parâmetro funciona para todos os OpenSearch serviços REST APIs, mas é especialmente útil quando você chama com frequência, como o `_index` e `_bulk` APIs: APIs

```
PUT opensearch-domain/more-movies/_doc/1?filter_path=result,_shards.total
{"title": "Back to the Future"}
```

Resposta

```
{
  "result": "updated",
  "_shards": {
    "total": 2
  }
}
```

Em vez de incluir campos, você pode excluir campos com um prefixo `-`. `filter_path` também oferece suporte a curingas:

```
POST opensearch-domain/_bulk?filter_path=-took,-items.index._*
{ "index": { "_index": "more-movies", "_id": "1" } }
{"title": "Back to the Future"}
{ "index": { "_index": "more-movies", "_id": "2" } }
{"title": "Spirited Away"}
```

Resposta

```
{
  "errors": false,
  "items": [
    {
      "index": {
        "result": "updated",
        "status": 200
      }
    },
    {
      "index": {
```

```
        "result": "updated",
        "status": 200
      }
    }
  ]
}
```

Codecs de índice

Os codecs de índice determinam como os campos armazenados em um índice são compactados e armazenados no disco. O codec de índice é controlado pela configuração estática `index.codec`, que especifica o algoritmo de compactação. Essa configuração afeta o tamanho do fragmento de índice e o desempenho da operação.

Para obter uma lista dos codecs compatíveis e suas características de desempenho, consulte Codecs compatíveis na documentação. OpenSearch

Ao escolher um codec de índice, considere o seguinte:

- Para evitar os desafios de alterar a configuração do codec de um índice existente, teste uma workload representativa em um ambiente que não seja de produção antes de usar uma nova configuração de codec. Para mais informações, consulte Alterar um codec de índice.
- Você não pode usar os codecs de compressão Zstandard (`"index.codec": "zstd"` ou `"index.codec": "zstd_no_dict"`) e para índices k-NN ou Security Analytics.

Carregando dados de streaming no Amazon OpenSearch Service

Você pode usar o OpenSearch Inestion para carregar diretamente dados de streaming em seu domínio do Amazon OpenSearch Service, sem precisar usar soluções de terceiros. Para enviar dados para o OpenSearch Ingestion, você configura seus produtores de dados e o serviço entrega automaticamente os dados ao domínio ou à coleção que você especificar. Para começar a usar o OpenSearch Ingestion, consultethe section called “Tutorial: Ingestão de dados em uma coleção”.

Você ainda pode usar outras fontes para carregar dados de streaming, como Amazon Data Firehose e Amazon CloudWatch Logs, que têm suporte integrado para OpenSearch o Service. Outras, como Amazon S3, Amazon Kinesis Data Streams e Amazon DynamoDB, usam funções do AWS Lambda como manipuladores de eventos. As funções do Lambda respondem a novos dados processando e transmitindo-os para seu domínio.

> **Note**
>
> O Lambda oferece suporte a várias linguagens de programação populares e está disponível na maioria das Regiões da AWS. Para saber mais, consulte Conceitos básicos do Lambda no Guia do desenvolvedor AWS Lambda e Endpoints de serviço da AWS na Referência geral da AWS.

Carregando dados de streaming do OpenSearch Ingestion

Você pode usar o Amazon OpenSearch Ingestion para carregar dados em um domínio OpenSearch de serviço. Você configura seus produtores de dados para enviar dados para OpenSearch Ingestão, e ele entrega automaticamente os dados para a coleção que você especificar. Você também pode configurar a OpenSearch ingestão para transformar seus dados antes de entregá-los. Para obter mais informações, consulte OpenSearch Ingestão da Amazon.

Carregamento de dados de transmissão do Amazon S3

Você pode usar o Lambda para enviar dados para seu domínio de OpenSearch serviço do Amazon S3. Os novos dados recebidos em um bucket do S3 acionam uma notificação de evento para o Lambda, que executa seu código personalizado para realizar a indexação.

Esse método de streaming de dados é extremamente flexível. Você pode indexar metadados de objeto, ou se o objeto for texto simples, analisar e indexar alguns elementos do corpo do objeto. Esta seção inclui alguns códigos de exemplo Python simples que usam expressões regulares para analisar um arquivo de log e indexar as correspondências.

Pré-requisitos

Para continuar, você deve ter os recursos a seguir.

Pré-requisito	Description
Bucket do Amazon S3.	Para saber mais, consulte Criar seu primeiro bucket do S3 no Manual do usuário do Amazon Simple Storage Service. O bucket deve residir na mesma região do seu domínio OpenSearch de serviço.

Pré-requisito	Description
OpenSearch Domínio do serviço	O destino dos dados depois que a função do Lambda os processa. Para saber mais, consulte the section called “Criação OpenSearch de domínios de serviço”.

Criar o pacote de implantação do Lambda

Os pacotes de implantação são arquivos ZIP ou JAR que contêm o código e as dependências. Esta seção inclui código de exemplo Python. Para outras linguagens de programação, consulte Pacotes de implantação do Lambda no Guia do desenvolvedor do AWS Lambda .

1. Crie um diretório. Neste exemplo, usamos o nome `s3-to-opensearch`.
2. Crie um arquivo no diretório chamado `sample.py`:

```
import boto3
import re
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-s3-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype

headers = { "Content-Type": "application/json" }

s3 = boto3.client('s3')

# Regular expressions used to parse some simple log lines
ip_pattern = re.compile('(\d+\.\d+\.\d+\.\d+)')
time_pattern = re.compile('\[(\d+\/\w\w\w\/\d\d\d\d:\d\d:\d\d:\d\d\s-\d\d\d\d)\]')
message_pattern = re.compile('\"(.+)\"')
```

```
# Lambda execution starts here
def handler(event, context):
    for record in event['Records']:

        # Get the bucket name and key for the new file
        bucket = record['s3']['bucket']['name']
        key = record['s3']['object']['key']

        # Get, read, and split the file into lines
        obj = s3.get_object(Bucket=bucket, Key=key)
        body = obj['Body'].read()
        lines = body.splitlines()

        # Match the regular expressions to each line and index the JSON
        for line in lines:
            line = line.decode("utf-8")
            ip = ip_pattern.search(line).group(1)
            timestamp = time_pattern.search(line).group(1)
            message = message_pattern.search(line).group(1)

            document = { "ip": ip, "timestamp": timestamp, "message": message }
            r = requests.post(url, auth=awsauth, json=document, headers=headers)
```

Edite as variáveis de `region` e `host`.

3. Se ainda não o fez, instale o pip. Em seguida, instale as dependências em um novo diretório `package`:

```
cd s3-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Como todos os ambientes de execução do Lambda têm o Boto3 instalado, você não precisa incluí-lo no pacote de implantação.

4. Empacote o código do aplicativo e as dependências:

```
cd package
zip -r ../lambda.zip .

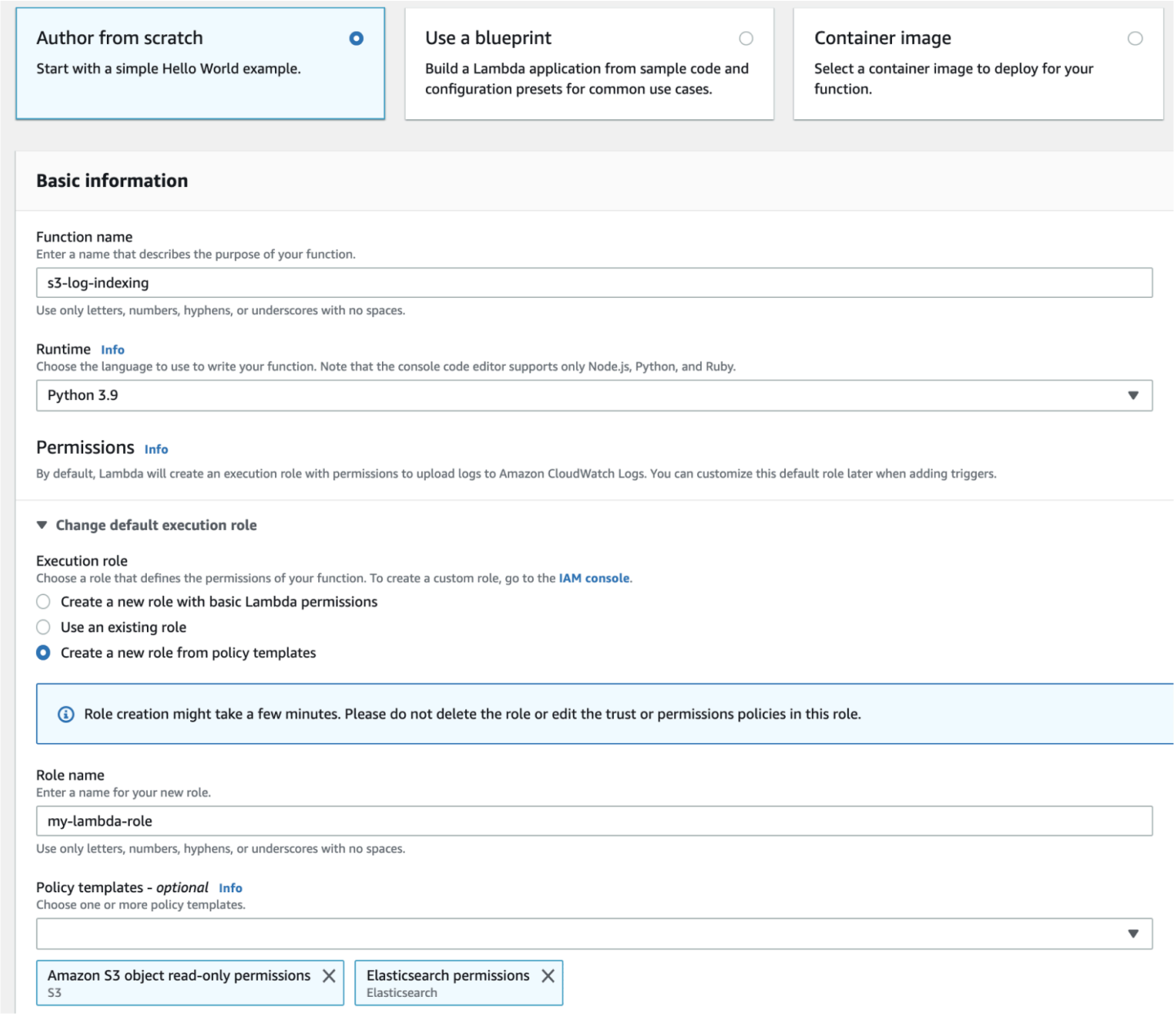
cd ..
```

```
zip -g lambda.zip sample.py
```

Criar a função do Lambda

Depois de criar o pacote de implantação, você poderá criar a função do Lambda. Ao criar uma função, escolha um nome, o runtime (por exemplo, Python 3.8) e a função do IAM. A função do IAM define as permissões para a função. Para obter instruções detalhadas, consulte Criar uma função Lambda com o console no Guia do desenvolvedor do AWS Lambda .

Esse exemplo pressupõe que você está usando o console. Escolha Python 3.9 e uma função que tenha permissões de leitura do S3 e permissões de gravação do OpenSearch Serviço, conforme mostrado na captura de tela a seguir:



Depois de criar a função, você deverá adicionar um gatilho. Neste exemplo, queremos que o código seja executado sempre que um arquivo de log chegue em um bucket do S3:

1. Escolha Adicionar acionador e selecione S3.
2. Escolha o bucket.
3. Em Tipo de evento, selecione PUT.
4. Em Prefixo, digite `logs/`.
5. Em Sufixo, digite `.log`.
6. Confirme o aviso de invocação recursiva e escolha Adicionar.

Por fim, você pode carregar o pacote de implantação:

1. Escolha Carregar de e arquivo .zip e siga os prompts para carregar do pacote de implantação.
2. Depois que o carregamento terminar, edite as Configurações de runtime e altere o Manipulador para `sample.handler`. Essa configuração informa ao Lambda o arquivo (`sample.py`) e o método (`handler`) que deverão ser executados depois de um acionador.

Nesse ponto, você tem um conjunto completo de recursos: um bucket para arquivos de log, uma função que é executada sempre que um arquivo de log é adicionado ao bucket, código que executa a análise e a indexação e um domínio de OpenSearch serviço para pesquisa e visualização.

Teste da função do Lambda

Após criar a função, você poderá testá-la carregando de um arquivo no bucket do Amazon S3. Crie um arquivo chamado `sample.log` usando as seguintes linhas de log de exemplo:

```
12.345.678.90 - [10/Oct/2000:13:55:36 -0700] "PUT /some-file.jpg"
12.345.678.91 - [10/Oct/2000:14:56:14 -0700] "GET /some-file.jpg"
```

Carregue o arquivo na pasta `logs` do bucket do S3. Para obter instruções, consulte Carregar um objeto para o seu bucket no Manual do usuário do Amazon Simple Storage Service.

Em seguida, use o console de OpenSearch serviço ou os OpenSearch painéis para verificar se o `lambda-s3-index` índice contém dois documentos. Você também pode fazer uma solicitação de pesquisa padrão:

```
GET https://domain-name/lambda-s3-index/_search?pretty
{
  "hits" : {
    "total" : 2,
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
        "_id" : "vTYXaWIBJWV_TTkEuSDg",
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.91",
          "message" : "GET /some-file.jpg",
```

```
          "timestamp" : "10/Oct/2000:14:56:14 -0700"
        }
      },
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
        "_id" : "vjYmaWIBJWV_TTkEuCAB",
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.90",
          "message" : "PUT /some-file.jpg",
          "timestamp" : "10/Oct/2000:13:55:36 -0700"
        }
      }
    ]
  }
}
```

Carregamento dados de transmissão do Amazon Kinesis Data Streams

Você pode carregar dados de streaming do Kinesis Data OpenSearch Streams para o Service. Os novos dados recebidos no fluxo de dados acionam uma notificação de evento para o Lambda, o qual executa seu código personalizado para realizar a indexação. Esta seção inclui um código de exemplo Python simples.

Pré-requisitos

Para continuar, você deve ter os recursos a seguir.

Pré-requisito	Description
Amazon Kinesis Data Streams	A fonte do evento para a função do Lambda. Para saber mais, consulte Kinesis Data Streams.
OpenSearch Domínio do serviço	O destino dos dados depois que a função do Lambda os processa. Para saber mais, consulte the section called “Criação OpenSearch de domínios de serviço”.
Perfil do IAM	Essa função deve ter permissões básicas OpenSearch de Service, Kinesis e Lambda, como as seguintes:

Pré-requisito	Description

	JSON
	
	{
	"Version":"2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Action": [
	"es:ESHttpPost",
	"es:ESHttpPut",
	"logs:CreateLogGroup",
	"logs:CreateLogStream",
	"logs:PutLogEvents",
	"kinesis:GetShardIterator",
	"kinesis:GetRecords",
	"kinesis:DescribeStream",
	"kinesis:ListStreams"
	],
	"Resource": "*"
	}
	]
	}
	
	A função deve ter a seguinte relação de confiança:
	
	JSON
	
	{
	"Version":"2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Principal": {
	"Service": "lambda.amazonaws.com"
	},
	"Action": "sts:AssumeRole"
	}
	]

Pré-requisito	Description
	<pre>}</pre> Para saber mais, consulte Criação de funções do IAM no Manual do usuário do IAM.

Criar a função do Lambda

Siga as instruções no the section called “Criar o pacote de implantação do Lambda”, mas crie um diretório chamado `kinesis-to-opensearch` e use o seguinte código para `sample.py`:

```
import base64
import boto3
import json
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-kine-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        id = record['eventID']
        timestamp = record['kinesis']['approximateArrivalTimestamp']

        # Kinesis data is base64-encoded, so decode here
        message = base64.b64decode(record['kinesis']['data'])
```

```
        # Create the JSON document
        document = { "id": id, "timestamp": timestamp, "message": message }
        # Index the document
        r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return 'Processed ' + str(count) + ' items.'
```

Edite as variáveis de `region` e `host`.

Caso ainda não tenha feito, instale o pip. Em seguida, use os seguintes comandos para instalar as dependências:

```
cd kinesis-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Depois siga as instruções em the section called “Criar a função do Lambda”, mas especifique a função do IAM por the section called “Pré-requisitos” e as seguintes configurações do gatilho:

- Fluxo do Kinesis: o fluxo do Kinesis
- Tamanho do lote: 100
- Posição inicial: redução horizontal

Para saber mais, consulte O que é o Amazon Kinesis Data Streams? no Guia do desenvolvedor do Amazon Kinesis Data Streams.

Nesse ponto, você tem um conjunto completo de recursos: um stream de dados do Kinesis, uma função que é executada depois que o stream recebe novos dados e indexa esses dados, e um domínio de OpenSearch serviço para pesquisa e visualização.

Testar a função do Lambda

Depois de criar a função, você poderá testá-la adicionando um novo registro ao streaming de dados usando a AWS CLI:

```
aws kinesis put-record --stream-name test --data "My test data." --partition-key
 partitionKey1 --region us-west-1
```

Em seguida, use o console de OpenSearch serviço ou os OpenSearch painéis para verificar se `lambda-kine-index` contém um documento. Você também pode usar a seguinte solicitação:

```
GET https://domain-name/lambda-kine-index/_search
{
  "hits" : [
    {
      "_index": "lambda-kine-index",
      "_type": "_doc",
      "_id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42",
      "_score": 1,
      "_source": {
        "timestamp": 1523648740.051,
        "message": "My test data.",
        "id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42"
      }
    }
  ]
}
```

Carregamento de dados de transmissão do Amazon DynamoDB

Você pode usar AWS Lambda para enviar dados para seu domínio de OpenSearch serviço do Amazon DynamoDB. Os novos dados recebidos na tabela do banco de dados acionam uma notificação de evento para o Lambda, que executa seu código personalizado para realizar a indexação.

Pré-requisitos

Para continuar, você deve ter os recursos a seguir.

Pré-requisito	Description
Tabela DynamoDB	A tabela contém os dados de origem. Para saber mais, consulte Operações básicas nas tabelas do DynamoDB no Guia do desenvolvedor do Amazon DynamoDB.

Pré-requisito	Description
	A tabela deve residir na mesma região do seu domínio OpenSearch de serviço e ter um stream definido como Nova imagem. Para saber mais, consulte Como habilitar um stream.
OpenSearch Domínio do serviço	O destino dos dados depois que a função do Lambda os processa. Para saber mais, consulte the section called “Criação OpenSearch de domínios de serviço”.

Pré-requisito	Description
perfil do IAM	Essa função deve ter permissões básicas OpenSearch de execução de Service, DynamoDB e Lambda, como as seguintes: JSON `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Action": ["es:ESHttpPost", "es:ESHttpPut", "dynamodb:DescribeStream", "dynamodb:GetRecords", "dynamodb:GetShardIterator", "dynamodb:ListStreams", "logs:CreateLogGroup", "logs:CreateLogStream", "logs:PutLogEvents"], "Resource": "*" }] }`  A função deve ter a seguinte relação de confiança:  JSON  `{ "Version":"2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" },`

Pré-requisito	Description

	<pre>"Action": "sts:AssumeRole"
	}
	]
	}</pre>
	Para saber mais, consulte Criação de funções do IAM no Manual do usuário do IAM.

Criar a função do Lambda

Siga as instruções no the section called “Criar o pacote de implantação do Lambda”, mas crie um diretório chamado `ddb-to-opensearch` e use o seguinte código para `sample.py`:

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-east-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        # Get the primary key for use as the OpenSearch ID
        id = record['dynamodb']['Keys']['id']['S']

        if record['eventName'] == 'REMOVE':
            r = requests.delete(url + id, auth=awsauth)
```

```
        else:
            document = record['dynamodb']['NewImage']
            r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return str(count) + ' records processed.'
```

Edite as variáveis de `region` e `host`.

Caso ainda não tenha feito, instale o pip. Em seguida, use os seguintes comandos para instalar as dependências:

```
cd ddb-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Depois siga as instruções em the section called “Criar a função do Lambda”, mas especifique a função do IAM por the section called “Pré-requisitos” e as seguintes configurações do gatilho:

- Tabela: a tabela do DynamoDB
- Tamanho do lote: 100
- Posição inicial: redução horizontal

Para saber mais, consulte Processar novos itens com o DynamoDB Streams e o Lambda no Guia do desenvolvedor do Amazon DynamoDB.

Neste momento, você tem um conjunto completo de recursos: uma tabela do DynamoDB para seus dados de origem, um stream de alterações na tabela do DynamoDB, uma função que é executada após a alteração dos dados de origem e indexa essas alterações e um domínio de serviço para pesquisa e visualização. OpenSearch

Testar a função do Lambda

Depois de criar a função, você poderá testá-la adicionando um novo item à tabela do DynamoDB usando a AWS CLI:

```
aws dynamodb put-item --table-name test --item '{"director": {"S": "Kevin
 Costner"},"id": {"S": "00001"},"title": {"S": "The Postman"}}' --region us-west-1
```

Em seguida, use o console de OpenSearch serviço ou os OpenSearch painéis para verificar se `lambda-index` contém um documento. Você também pode usar a seguinte solicitação:

```
GET https://domain-name/lambda-index/_doc/00001
{
    "_index": "lambda-index",
    "_type": "_doc",
    "_id": "00001",
    "_version": 1,
    "found": true,
    "_source": {
        "director": {
            "S": "Kevin Costner"
        },
        "id": {
            "S": "00001"
        },
        "title": {
            "S": "The Postman"
        }
    }
}
```

Carregamento de dados de transmissão do Amazon Data Firehose

O Firehose oferece suporte OpenSearch ao serviço como destino de entrega. Para obter instruções sobre como carregar dados de streaming no OpenSearch Serviço, consulte Criação de um stream de entrega do Kinesis Data Firehose OpenSearch e Escolha o serviço para seu destino no Guia do desenvolvedor do Amazon Data Firehose.

Antes de carregar dados no OpenSearch Serviço, talvez seja necessário realizar transformações nos dados. Para saber mais sobre como usar funções do Lambda para executar essa tarefa, consulte Transformação de dados do Amazon Kinesis Data Firehose no mesmo guia.

Ao configurar um stream de entrega, o Firehose apresenta uma função IAM de “um clique” que fornece o acesso aos recursos necessários para enviar dados ao OpenSearch Serviço, fazer backup de dados no Amazon S3 e transformar dados usando o Lambda. Em virtude da complexidade envolvida na criação manual de uma função como essa, é recomendável usar a função fornecida.

Carregando dados de streaming da Amazon CloudWatch

Você pode carregar dados de streaming do CloudWatch Logs para seu domínio do OpenSearch Serviço usando uma assinatura do CloudWatch Logs. Para obter informações sobre as CloudWatch assinaturas da Amazon, consulte Processamento em tempo real de dados de log com assinaturas. Para obter informações de configuração, consulte Streaming de dados de CloudWatch registros para o Amazon OpenSearch Service no Amazon CloudWatch Developer Guide.

Carregando dados de streaming de AWS IoT

Você pode enviar dados AWS IoT usando regras. Para saber mais, consulte a OpenSearchação no Guia do AWS IoT desenvolvedor.

Carregamento de dados no Amazon OpenSearch Service com o Logstash

A versão de código aberto do Logstash (Logstash OSS) fornece uma maneira conveniente de usar a API em massa para fazer upload de dados em seu domínio do Amazon Service. OpenSearch O serviço oferece suporte a todos os plug-ins de entrada padrão do Logstash, incluindo o plug-in de entrada Amazon S3. OpenSearch O serviço oferece suporte ao plug-in logstash-output-opensearchde saída, que oferece suporte à autenticação básica e às credenciais do IAM. O plug-in funciona com a versão 8.1 e inferior do Logstash OSS.

Configuração

A configuração do Logstash varia de acordo com o tipo de autenticação utilizada pelo seu domínio.

Não importa que método de autenticação você use, é necessário definir `ecs_compatibility`como `disabled` na seção de saída do arquivo de configuração. O Logstash 8.0 introduziu uma mudança revolucionária em que todos os plug-ins são executados no modo de compatibilidade com ECS por padrão. Você deve substituir o valor padrão para manter o comportamento herdado.

Configuração do controle de acesso refinado

Se seu domínio OpenSearch de serviço usa controle de acesso refinado com autenticação básica HTTP, a configuração é semelhante a qualquer outro cluster. OpenSearch Este arquivo de configuração de exemplo obtém a entrada da versão de código aberto do Filebeat (Filebeat OSS):

```
input {
  beats  {
    port => 5044
  }
}

output {
  opensearch {
    hosts       => "https://domain-endpoint:443"
    user        => "my-username"
    password    => "my-password"
    index       => "logstash-logs-%{+YYYY.MM.dd}"
    ecs_compatibility => disabled
    ssl_certificate_verification => false
  }
}
```

A configuração varia de acordo com a aplicações Beats e o caso de uso, mas sua configuração do Filebeat OSS pode ser semelhante a esta:

```
filebeat.inputs:
- type: log
  enabled: true
  paths:
    - /path/to/logs/dir/*.log
filebeat.config.modules:
  path: ${path.config}/modules.d/*.yml
  reload.enabled: false
setup.ilm.enabled: false
setup.ilm.check_exists: false
setup.template.settings:
  index.number_of_shards: 1
output.logstash:
  hosts: ["logstash-host:5044"]
```

Configuração do IAM

Se seu domínio usa uma política de acesso ao domínio baseada em IAM ou um controle de acesso refinado com um usuário principal, você deve assinar todas as solicitações ao OpenSearch Serviço usando as credenciais do IAM. A política baseada em identidade a seguir concede todas as solicitações HTTP para os sub-recursos do seu domínio.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Para configurar o Logstash, altere o seu arquivo de configuração para usar o plug-in para a saída. Este arquivo de configuração de exemplo obtém a entrada de arquivos em um bucket do S3:

```
input {
  s3 {
    bucket => "amzn-s3-demo-"
    region => "us-east-1"
  }
}

output {
  opensearch {
    hosts => ["domain-endpoint:443"]
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      }
      index  => "logstash-logs-%{+YYYY.MM.dd}"
      ecs_compatibility => disabled
  }
}
```

Se você não quiser fornecer as suas credenciais do IAM no arquivo de configuração, poderá exportá-las (ou executar o `aws configure`):

```
export AWS_ACCESS_KEY_ID="your-access-key"
export AWS_SECRET_ACCESS_KEY="your-secret-key"
export AWS_SESSION_TOKEN="your-session-token"
```

Se seu domínio OpenSearch de serviço estiver em uma VPC, a máquina Logstash OSS deverá ser capaz de se conectar à VPC e ter acesso ao domínio por meio dos grupos de segurança da VPC. Para obter mais informações, consulte the section called “Sobre políticas de acesso em domínios da VPC”.

Pesquisando dados no Amazon OpenSearch Service

Há vários métodos comuns para pesquisar documentos no Amazon OpenSearch Service, incluindo pesquisas de URI e pesquisas de corpos de solicitações. OpenSearch O serviço oferece funcionalidades adicionais que melhoram a experiência de pesquisa, como pacotes personalizados, suporte a SQL e pesquisa assíncrona. Para obter uma referência abrangente da API de OpenSearch pesquisa, consulte a OpenSearchdocumentação.

> **Note**
>
> Os exemplos de solicitações a seguir funcionam com OpenSearch APIs. Algumas solicitações podem não funcionar com versões mais antigas do Elasticsearch.

Tópicos

- Pesquisas de URI
- Pesquisas de corpo da solicitação
- Paginação de resultados da pesquisa
- Dashboards Query Language
- Importação e gerenciamento de pacotes no Amazon Service OpenSearch
- Consultando seus dados do Amazon OpenSearch Service com SQL
- Pesquisa entre clusters no Amazon Service OpenSearch
- Aprendendo a se classificar no Amazon OpenSearch Service
- Pesquisa assíncrona no Amazon Service OpenSearch
- Pesquisa pontual no Amazon OpenSearch Service
- Pesquisa de agentes no Amazon Service OpenSearch
- Pesquisa semântica no Amazon Service OpenSearch
- Pesquisa simultânea de segmentos no Amazon Service OpenSearch
- Geração de consultas em linguagem natural no Amazon OpenSearch Service

Pesquisas de URI

As pesquisas do URI (Universal Resource Identifier, Identificador de recurso universal) são a forma mais simples de pesquisa. Em uma pesquisa do URI, você especifica a consulta como um parâmetro de solicitação HTTP.

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/_search?q=house
```

Uma resposta de exemplo pode ser a seguinte:

```
{
  "took": 25,
  "timed_out": false,
  "_shards": {
    "total": 10,
    "successful": 10,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 85,
      "relation": "eq",
    },
    "max_score": 6.6137657,
    "hits": [
      {
        "_index": "movies",
        "_type": "movie",
        "_id": "tt0077975",
        "_score": 6.6137657,
        "_source": {
          "directors": [
            "John Landis"
          ],
          "release_date": "1978-07-27T00:00:00Z",
          "rating": 7.5,
          "genres": [
            "Comedy",
            "Romance"
          ],
```

```
          "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTY2OTQxNTc1OF5BMl5BanBnXkFtZTYwNjA3NjI5._V1_SX400_.jpg",
          "plot": "At a 1962 College, Dean Vernon Wormer is determined to expel the
 entire Delta Tau Chi Fraternity, but those troublemakers have other plans for him.",
          "title": "Animal House",
          "rank": 527,
          "running_time_secs": 6540,
          "actors": [
            "John Belushi",
            "Karen Allen",
            "Tom Hulce"
          ],
          "year": 1978,
          "id": "tt0077975"
        }
      },
      ...
    ]
  }
}
```

Por padrão, essa consulta pesquisa todos os campos de todos os índices do termo casa. Para restringir a pesquisa, especifique um índice (`movies`) e um campo de documento (`title`) no URI:

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?q=title:house
```

Você pode incluir parâmetros adicionais na solicitação, mas os parâmetros compatíveis fornecem apenas um pequeno subconjunto das opções de OpenSearch pesquisa. A solicitação a seguir retorna 20 resultados (em vez do padrão de 10) e classifica por ano (em vez de por `_score`):

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?
q=title:house&size=20&sort=year:desc
```

Pesquisas de corpo da solicitação

Para realizar pesquisas mais complexas, use o corpo de solicitação HTTP e o idioma específico do domínio (DSL) do OpenSearch para consultas. A consulta DSL permite especificar o intervalo completo de opções de pesquisa do OpenSearch .

> **Note**
>
> Você não pode incluir caracteres especiais Unicode em um valor de campo de texto, ou o valor será analisado como vários valores separados pelo caractere especial. Essa análise incorreta pode levar à filtragem não intencional de documentos e potencialmente comprometer o controle sobre seu acesso. Para obter mais informações, consulte Uma nota sobre caracteres especiais Unicode em campos de texto na OpenSearch documentação.

A consulta `match` a seguir é semelhante ao exemplo de pesquisa final do URI:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "sort": {
    "year": {
      "order": "desc"
    }
  },
  "query": {
    "query_string": {
      "default_field": "title",
      "query": "house"
    }
  }
}
```

> **Note**
>
> A API `_search` aceita HTTP`GET` e `POST` para pesquisas de corpo de solicitação, mas nem todos os clientes HTTP suportam a adição de um corpo de solicitação a uma solicitação `GET`. `POST` é a escolha mais universal.

Em muitos casos, você pode pesquisar vários campos, mas não todos os campos. Use a consulta `multi_match`:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
```

```
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title", "plot", "actors", "directors"]
    }
  }
}
```

Impulsão de campos

Você pode melhorar a relevância de pesquisa “aumentando” determinados campos. Boosts são multiplicadores que ponderam os resultados em um campo maior do que os correspondentes em outros campos. No exemplo a seguir, uma correspondência a john no campo `title` influencia `_score` duas vezes mais que uma correspondência no campo `plot` e quatro vezes mais que uma correspondência nos campos `actors` ou `directors`. O resultado é que filmes como John Wick e John Carter estão próximos do topo dos resultados de busca, e filmes estrelados por John Travolta estão quase no fim.

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "john",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  }
}
```

Destaques de resultados da pesquisa

A `highlight` opção diz OpenSearch para retornar um objeto adicional dentro da `hits` matriz se a consulta corresponder a um ou mais campos:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
```

```
    }
  },
  "highlight": {
    "fields": {
      "plot": {}
    }
  }
}
```

Se a consulta corresponder ao conteúdo do campo `plot`, um resultado pode ser semelhante ao seguinte:

```
{
  "_index": "movies",
  "_type": "movie",
  "_id": "tt0091541",
  "_score": 11.276199,
  "_source": {
    "directors": [
      "Richard Benjamin"
    ],
    "release_date": "1986-03-26T00:00:00Z",
    "rating": 6,
    "genres": [
      "Comedy",
      "Music"
    ],
    "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTIzODEzODE2OF5BMl5BanBnXkFtZTcwNjQ3ODcyMQ@@._V1_SX400_.jpg",
    "plot": "A young couple struggles to repair a hopelessly dilapidated house.",
    "title": "The Money Pit",
    "rank": 4095,
    "running_time_secs": 5460,
    "actors": [
      "Tom Hanks",
      "Shelley Long",
      "Alexander Godunov"
    ],
    "year": 1986,
    "id": "tt0091541"
  },
  "highlight": {
    "plot": [
```

```
      "A young couple struggles to repair a hopelessly dilapidated <em>house</em>."
    ]
  }
}
```

Por padrão, OpenSearch divide a string correspondente em tags, fornece até 100 caracteres de contexto em torno da correspondência e divide o conteúdo em frases identificando sinais de pontuação, espaços, tabulações e quebras de linha. Todas estas configurações são personalizáveis:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  },
  "highlight": {
    "fields": {
      "plot": {}
    },
    "pre_tags": "<strong>",
    "post_tags": "</strong>",
    "fragment_size": 200,
    "boundary_chars": ".,!? "
  }
}
```

API de contagem

Se você não estiver interessado no conteúdo de seus documentos e quiser apenas saber o número de correspondências, poderá usar a API `_count` em vez da API `_search`. A solicitação a seguir usa a consulta `query_string` para identificar comédias românticas:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_count
{
  "query": {
    "query_string": {
      "default_field": "genres",
      "query": "romance AND comedy"
    }
```

```
  }
}
```

Uma resposta de exemplo pode ser a seguinte:

```
{
  "count": 564,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  }
}
```

Paginação de resultados da pesquisa

Se precisar exibir um grande número de resultados de pesquisa, você poderá implementar a paginação usando vários métodos diferentes.

Ponto de tempo

O atributo point in time (PIT – um ponto no tempo) é um tipo de pesquisa que permite executar consultas diferentes em um conjunto de dados fixo no tempo. Esse é o método de paginação preferido em OpenSearch, especialmente para paginação profunda. Você pode usar o PIT com a versão 2.5 e posterior do OpenSearch Service. Para ter mais informações sobre o PIT, consulte ???.

Os parâmetros **`from`** e **`size`**.

A maneira mais simples de paginar é com os parâmetros `from` e `size`. A seguinte solicitação retorna resultados de 20 a 39 da lista indexada zero de resultados da pesquisa:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "from": 20,
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
```

```
    }
  }
}
```

Para obter mais informações sobre paginação de pesquisa, consulte Resultados de paginação na documentação. OpenSearch

Dashboards Query Language

Você pode usar a Dashboards Query Language (DQL) para pesquisar dados e visualizações em painéis. OpenSearch A DQL usa quatro tipos de consulta principais:termos, boleana, data e intervalo e campo aninhado.

Consulta de termos

Uma consulta de termos exige que você especifique o termo que está procurando.

Para executar uma consulta de termos, insira o seguinte:

```
host:www.example.com
```

Consulta booleana

É possível usar os operadores booleanos `AND`, `or` e `not` para combinar várias consultas.

Para executar uma consulta booleana, cole o seguinte:

```
host.keyword:www.example.com and response.keyword:200
```

Consulta de data e intervalo

Você pode usar uma consulta de data e intervalo para encontrar uma data antes ou depois da consulta.

- > indica uma pesquisa por uma data posterior à data especificada.
- < indica uma pesquisa por uma data anterior à data especificada.

```
@timestamp > "2020-12-14T09:35:33"
```

Consulta de campo aninhado

Se você tiver um documento com campos aninhados, será necessário especificar quais partes do documento você deseja recuperar. Veja a seguir um exemplo de documento que contém campos aninhados:

```
{"NBA players":[
    {"player-name": "Lebron James",
      "player-position": "Power forward",
      "points-per-game": "30.3"
    },
    {"player-name": "Kevin Durant",
      "player-position": "Power forward",
      "points-per-game": "27.1"
    },
    {"player-name": "Anthony Davis",
      "player-position": "Power forward",
      "points-per-game": "23.2"
    },
    {"player-name": "Giannis Antetokounmpo",
      "player-position": "Power forward",
      "points-per-game":"29.9"
    }
  ]
}
```

Para recuperar um campo específico usando DQL, cole o seguinte:

```
NBA players: {player-name: Lebron James}
```

Para recuperar vários objetos do documento aninhado, cole o seguinte:

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo}
```

Para pesquisar em um intervalo, cole o seguinte:

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo and < 30}
```

Se o documento tiver um objeto aninhado em outro objeto, você ainda poderá recuperar dados especificando todos os níveis. Para fazer isso, cole o seguinte:

```
Top-Power-forwards.NBA players: {player-name:Lebron James}
```

Importação e gerenciamento de pacotes no Amazon Service OpenSearch

O Amazon OpenSearch Service permite que você faça upload de arquivos de dicionário personalizados, como palavras irrelevantes e sinônimos, e associe plug-ins ao seu domínio. Esses plug-ins podem ser pré-empacotados, personalizados ou de terceiros, o que dá a você flexibilidade para estender a funcionalidade do domínio. O termo genérico para todos esses tipos de arquivos é pacotes.

- Os arquivos de dicionário ajudam a refinar os resultados da pesquisa instruindo OpenSearch a ignorar palavras comuns de alta frequência ou a tratar termos semelhantes, como “creme congelado”, “gelato” e “sorvete”, como equivalentes. Eles também podem melhorar a redução à raiz, como no plug-in de análise japonês (kuromoji).
- Os plug-ins pré-empacotados fornecem funcionalidade integrada, como o plug-in Amazon Personalize, para gerar resultados de pesquisa personalizados. Esses plug-ins usam o tipo de pacote `ZIP-PLUGIN`. Para saber mais, consulte the section called “Plug-ins por versão do mecanismo”.
- Plug-ins personalizados e de terceiros permitem adicionar atributos personalizados ou integrar com sistemas externos, o que oferece ainda mais flexibilidade ao domínio. Como acontece com os plug-ins pré-empacotados, você carrega os plug-ins personalizados como pacotes da `ZIP-PLUGIN`. Para plug-ins de terceiros, é necessário também importar a licença do plug-in e os arquivos de configuração como pacotes separados e associá-los ao domínio.

  Para saber mais, consulte os seguintes tópicos:

  - the section called “Plugins personalizados”
  - the section called “Plug-ins de terceiros”

> **Note**
>
> Você pode associar até 20 plug-ins a um único domínio. Esse limite inclui todos os tipos de plug-ins: opcionais, de terceiros e personalizados.

Tópicos

- Permissões obrigatórias
- Carregar pacotes para o Amazon S3
- Importação e associação de pacotes
- Usando pacotes com OpenSearch
- Atualização de pacotes
- Atualizar manualmente os índices com um novo dicionário
- Dissociação e remoção de pacotes
- Gerenciando plug-ins personalizados no Amazon OpenSearch Service
- Instalação de plug-ins de terceiros no Amazon OpenSearch Service

Permissões obrigatórias

Usuários sem acesso de administrador precisam de determinadas ações AWS Identity and Access Management (IAM) para gerenciar pacotes:

- `es:CreatePackage`: criar um pacote
- `es:DeletePackage`: excluir um pacote
- `es:AssociatePackage`: associar um pacote a um domínio
- `es:DissociatePackage`: desassociar um pacote de um domínio

Você também precisa de permissões no caminho do bucket do Amazon S3 ou no objeto em que o pacote personalizado reside.

Conceda todas as permissões no IAM, e não na política de acesso ao domínio. Para saber mais, consulte the section called “Gerenciamento de Identidade e Acesso”.

Carregar pacotes para o Amazon S3

Esta seção aborda como carregar pacotes de dicionários personalizados, já que os pacotes de plug-ins pré-empacotados já estão pré-instalados. Antes de associar um dicionário customizado ao seu domínio, você deverá carregá-lo em um bucket do Amazon S3. Para saber mais, consulte Carregar objetos no Manual do usuário do Amazon Simple Storage Service. Plug-ins compatíveis não precisam ser carregados.

Se seu dicionário contiver informações confidenciais, especifique a criptografia do lado do servidor com chaves gerenciadas pelo S3 ao fazer o upload. OpenSearch O serviço não pode acessar arquivos no S3 que você protege usando uma AWS KMS chave.

Depois de carregar o arquivo, anote o caminho do S3. O formato do caminho é `s3://amzn-s3-demo-bucket/file-path/file-name`.

Você pode usar o seguinte arquivo de sinônimos para fazer testes. Salve-o como `synonyms.txt`.

```
danish, croissant, pastry
ice cream, gelato, frozen custard
sneaker, tennis shoe, running shoe
basketball shoe, hightop
```

Certos dicionários, como dicionários Hunspell, usam vários arquivos e exigem seus próprios diretórios no sistema de arquivos. No momento, o OpenSearch Service oferece suporte apenas a dicionários de arquivo único.

Importação e associação de pacotes

O console é a maneira mais simples de importar um dicionário personalizado para o OpenSearch Service. Quando você importa um dicionário do Amazon S3, o OpenSearch Service armazena sua própria cópia do pacote e criptografa automaticamente essa cópia usando AES-256 com chaves gerenciadas pelo serviço. OpenSearch

Os plug-ins opcionais já estão pré-instalados no OpenSearch Service, então você não precisa carregá-los sozinho, mas precisa associar um plug-in a um domínio. Os plug-ins disponíveis estão listados na tela Pacotes, no console.

Importar e associar um pacote a um domínio

1. No console do Amazon OpenSearch Service, escolha Pacotes.
2. Escolha Import package (Importar pacote).
3. Dê um nome descritivo ao pacote.
4. Forneça o caminho do S3 até o arquivo e selecione Import (Importar).
5. Retorne à tela Pacotes.
6. Quando o status do pacote estiver Disponível, selecione-o.
7. Escolha Associar a um domínio.

8. Selecione um domínio e escolha Avançar. Revise os pacotes e escolha Associar.
9. No painel de navegação, escolha o domínio vá para a guia Pacotes.
10. Se o pacote for um dicionário personalizado, anote o ID quando o pacote se tornar Disponível. Use `analyzers/id` como caminho do arquivo em solicitações para OpenSearch.

Usando pacotes com OpenSearch

Esta seção aborda como usar os dois tipos de pacotes: dicionários personalizados e plug-ins opcionais.

Uso de dicionários customizados

Depois de associar um arquivo a um domínio, será possível usá-lo em parâmetros como `synonyms_path`, `stopwords_path` e `user_dictionary` ao criar tokenizadores e filtros de token. O parâmetro exato varia de acordo com o objeto. Vários objetos oferecem suporte a `synonyms_path` e `stopwords_path`, mas `user_dictionary` é exclusivo para o plug-in kuromoji.

Para o plug-in de análise IK (chinês), você pode carregar um arquivo de dicionário personalizado como um pacote personalizado e associá-lo a um domínio, e o plug-in o seleciona automaticamente sem exigir um parâmetro `user_dictionary`. Se seu arquivo for um arquivo de sinônimos, use o parâmetro `synonyms_path`.

O seguinte exemplo adiciona um arquivo de sinônimo a um novo índice:

```
PUT my-index
{
  "settings": {
    "index": {
      "analysis": {
        "analyzer": {
          "my_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["my_filter"]
          }
        },
        "filter": {
          "my_filter": {
            "type": "synonym",
```

```
            "synonyms_path": "analyzers/F111111111",
            "updateable": true
          }
        }
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "standard",
        "search_analyzer": "my_analyzer"
      }
    }
  }
}
```

Esta solicitação cria um analisador personalizado para o índice que utiliza o tokenizer padrão e um filtro de token de sinônimo.

- Os tokenizers quebram fluxos de caracteres em tokens (normalmente palavras) de acordo com algum conjunto de regras. O exemplo mais simples é o tokenizer de espaço em branco, que divide os caracteres anteriores em um token cada vez que encontra um caractere de espaço em branco. Um exemplo mais complexo é o tokenizer padrão, que usa um conjunto de regras com base na gramática para trabalhar em vários idiomas.
- Os filtros de token adicionam, modificam ou excluem tokens. Por exemplo, um filtro de token de sinônimo adiciona tokens quando encontra uma palavra na lista de sinônimos. O filtro de token de palavras irrelevantes remove tokens quando encontra uma palavra na lista de palavras irrelevantes.

Essa solicitação também adiciona um campo de texto (`description`) ao mapeamento e solicita OpenSearch o uso do novo analisador como analisador de pesquisa. Você pode ver que ele ainda usa o analisador padrão como seu analisador de índices.

Finalmente, observe a linha `"updateable": true` no filtro de token. Este campo aplica-se somente a analisadores de pesquisas, e não a analisadores de índices, e será crítico se você desejar atualizar o analisador de pesquisas automaticamente.

Para fazer testes, adicione alguns documentos ao índice:

```
POST _bulk
{ "index": { "_index": "my-index", "_id": "1" } }
{ "description": "ice cream" }
{ "index": { "_index": "my-index", "_id": "2" } }
{ "description": "croissant" }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "description": "tennis shoe" }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "description": "hightop" }
```

Depois, pesquise-os usando um sinônimo:

```
GET my-index/_search
{
  "query": {
    "match": {
      "description": "gelato"
    }
  }
}
```

Nesse caso, OpenSearch retorna a seguinte resposta:

```
{
  "hits": {
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 0.99463606,
    "hits": [{
      "_index": "my-index",
      "_type": "_doc",
      "_id": "1",
      "_score": 0.99463606,
      "_source": {
        "description": "ice cream"
      }
    }]
  }
}
```

> **Tip**
>
> Arquivos de dicionário usam espaço de heap Java proporcional ao seu tamanho. Por exemplo, um arquivo de dicionário de 2 GiB pode consumir 2 GiB de espaço de heap em um nó. Ao usar arquivos grandes, verifique se os nós têm espaço de heap suficiente para acomodá-los. Monitore a métrica `JVMMemoryPressure` e dimensione o cluster conforme necessário.

Usar plug-ins pré-empacotados

OpenSearch O serviço permite que você associe OpenSearch plug-ins opcionais pré-instalados para usar com seu domínio. Um pacote de plug-in pré-empacotado é compatível com uma OpenSearch versão específica e só pode ser associado a domínios com essa versão. A lista de pacotes disponíveis para seu domínio inclui todos os plug-ins compatíveis com a versão do seu domínio. Depois de associar um plug-in a um domínio, um processo de instalação no domínio é iniciado. Em seguida, você pode referenciar e usar o plug-in ao fazer solicitações ao OpenSearch Serviço.

Associar e dissociar um plug-in requer uma implantação. blue/green Para obter mais informações, consulte the section called “Mudanças que geralmente causam blue/green implantações”.

Plug-ins opcionais incluem analisadores de idioma e resultados de pesquisa personalizados. Por exemplo, o plug-in Amazon Personalize Search Ranking usa machine learning para personalizar os resultados da pesquisa para seus clientes. Para obter mais informações sobre esse plug-in, consulte Personalização dos resultados da pesquisa de OpenSearch. Para obter uma lista de todos os plug-ins compatíveis, consulte the section called “Plug-ins por versão do mecanismo”.

Plug-in Sudachi

Quando você reassocia um arquivo de dicionário do plug-in Sudachi, ele não reflete imediatamente no domínio. O dicionário é atualizado quando a próxima blue/green implantação é executada no domínio como parte de uma alteração de configuração ou outra atualização. Como alternativa, você pode criar um novo pacote com os dados atualizados, criar um novo índice usando esse novo pacote, reindexar o índice existente ao novo e, em seguida, excluir o índice antigo. Se preferir usar a abordagem de reindexação, use um alias de índice para que não haja interrupções no tráfego.

Além disso, o plug-in Sudachi suporta apenas dicionários binários do Sudachi, que você pode carregar com a operação da API. CreatePackage Para obter informações sobre o dicionário do

sistema pré-construído e o processo para compilar dicionários do usuário, consulte a documentação do Sudachi.

> **Note**
>
> Ao fazer o upload de arquivos de dicionário binário para o Amazon S3, você deve definir o tipo de conteúdo do objeto do S3 como. `binary/octet-stream` O uso `application/octet-stream` fará com que a importação do pacote falhe.

O exemplo a seguir demonstra como usar os dicionários do sistema e do usuário com o tokenizador do Sudachi. Você deve carregar esses dicionários como pacotes personalizados com tipo `TXT-DICTIONARY` e fornecer o pacote IDs nas configurações adicionais.

```
PUT sudachi_sample
{
  "settings": {
    "index": {
      "analysis": {
        "tokenizer": {
          "sudachi_tokenizer": {
            "type": "sudachi_tokenizer",
            "additional_settings": "{\"systemDict\": \"<system-dictionary-package-id>\",\"userDict\": [\"<user-dictionary-package-id>\"]}"
        }
        },
        "analyzer": {
          "sudachi_analyzer": {
            "filter": ["my_searchfilter"],
            "tokenizer": "sudachi_tokenizer",
            "type": "custom"
          }
        },
        "filter":{
          "my_searchfilter": {
            "type": "sudachi_split",
            "mode": "search"
          }
        }
      }
    }
  }
```

```
}
```

Atualização de pacotes

Esta seção aborda apenas como atualizar um pacote de dicionário personalizado, porque os pacotes de plug-ins pré-empacotados já foram atualizados para você. O upload de uma nova versão de um dicionário para o Amazon S3 não atualiza automaticamente o pacote no Amazon OpenSearch Service. OpenSearch O serviço armazena sua própria cópia do arquivo, portanto, se você fizer upload de uma nova versão para o S3, deverá atualizá-la manualmente.

Cada um dos seus domínios associados armazena sua própria cópia do arquivo também. Para manter o comportamento de pesquisa previsível, os domínios continuarão a usar a versão atual do pacote até que você os atualize explicitamente. Para atualizar um pacote personalizado, modifique o arquivo Amazon S3 Control, atualize o pacote no OpenSearch Serviço e, em seguida, aplique a atualização.

Console

1. No console de OpenSearch serviço, escolha Pacotes.
2. Escolha um pacote e, em seguida, Atualizar.
3. Forneça um novo caminho do S3 para o arquivo e escolha Atualizar pacote.
4. Retorne à tela Pacotes.
5. Quando o status do pacote mudar para Disponível, selecione-o. Em seguida, escolha um ou mais domínios associados, Aplicar atualização e confirme. Aguarde até que o status da associação mude para Ativo.
6. As próximas etapas variam dependendo de como você configurou os índices:

   - Se seu domínio está executando OpenSearch o Elasticsearch 7.8 ou posterior e usa apenas analisadores de pesquisa com o campo atualizável definido como verdadeiro, você não precisa realizar nenhuma ação adicional. OpenSearch O serviço atualiza automaticamente seus índices usando a API _plugins/_refresh_search_analyzers.
   - Se o seu domínio estiver executando o Elasticsearch 7.7 ou anterior, usa analisadores de índices ou não usa o campo `updateable`, consulte the section called “Atualizar manualmente os índices com um novo dicionário”.

Embora o console seja o método mais simples, você também pode usar a API de configuração AWS CLI SDKs, ou para atualizar pacotes OpenSearch de serviços. Para obter mais informações, consulte a Referência de AWS CLI Comandos e a Referência da API do Amazon OpenSearch Service.

AWS SDK

Em vez de atualizar manualmente um pacote no console, você pode usar o SDKs para automatizar o processo de atualização. O exemplo de script Python a seguir carrega um novo arquivo de pacote no Amazon S3, atualiza o pacote no OpenSearch Service e aplica o novo pacote ao domínio especificado. Depois de confirmar que a atualização foi bem-sucedida, ele faz uma chamada de amostra para OpenSearch demonstrar que os novos sinônimos foram aplicados.

Você deve fornecer valores para `host`, `region`, `file_name`, `bucket_name`, `s3_key`, `package_id`, `domain_name` e `query`.

```
from requests_aws4auth import AWS4Auth
import boto3
import requests
import time
import json
import sys

host = ''  # The OpenSearch domain endpoint with https:// and a trailing slash. For
 example, https://my-test-domain.us-east-1.es.amazonaws.com/
region = ''  # For example, us-east-1
file_name = ''  # The path to the file to upload
bucket_name = ''  # The name of the S3 bucket to upload to
s3_key = ''  # The name of the S3 key (file name) to upload to
package_id = ''  # The unique identifier of the OpenSearch package to update
domain_name = ''  # The domain to associate the package with
query = ''  # A test query to confirm the package has been successfully updated

service = 'es'
credentials = boto3.Session().get_credentials()
client = boto3.client('opensearch')
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def upload_to_s3(file_name, bucket_name, s3_key):
    """Uploads file to S3"""
    s3 = boto3.client('s3')
    try:
```

```
        s3.upload_file(file_name, bucket_name, s3_key)
        print('Upload successful')
        return True
    except FileNotFoundError:
        sys.exit('File not found. Make sure you specified the correct file path.')


def update_package(package_id, bucket_name, s3_key):
    """Updates the package in OpenSearch Service"""
    print(package_id, bucket_name, s3_key)
    response = client.update_package(
        PackageID=package_id,
        PackageSource={
            'S3BucketName': bucket_name,
            'S3Key': s3_key
        }
    )
    print(response)


def associate_package(package_id, domain_name):
    """Associates the package to the domain"""
    response = client.associate_package(
        PackageID=package_id, DomainName=domain_name)
    print(response)
    print('Associating...')


def wait_for_update(domain_name, package_id):
    """Waits for the package to be updated"""
    response = client.list_packages_for_domain(DomainName=domain_name)
    package_details = response['DomainPackageDetailsList']
    for package in package_details:
        if package['PackageID'] == package_id:
            status = package['DomainPackageStatus']
            if status == 'ACTIVE':
                print('Association successful.')
                return
            elif status == 'ASSOCIATION_FAILED':
                sys.exit('Association failed. Please try again.')
            else:
                time.sleep(10)  # Wait 10 seconds before rechecking the status
                wait_for_update(domain_name, package_id)
```

```
def sample_search(query):
    """Makes a sample search call to OpenSearch"""
    path = '_search'
    params = {'q': query}
    url = host + path
    response = requests.get(url, params=params, auth=awsauth)
    print('Searching for ' + '"' + query + '"')
    print(response.text)
```

> **Note**
>
> Se você receber um erro de “pacote não encontrado” ao executar o script usando o AWS CLI, provavelmente significa que o Boto3 está usando a região especificada em ~/.aws/config, que não é a região em que seu bucket do S3 está. Execute `aws configure` e especifique a região correta ou adicione explicitamente a região ao cliente:
>
> ```
> client = boto3.client('opensearch', region_name='us-east-1')
> ```

Atualizar manualmente os índices com um novo dicionário

As atualizações manuais de índice se aplicam somente aos dicionários personalizados, não aos plug-ins pré-empacotados. Para usar um dicionário atualizado, será necessário atualizar manualmente seus índices se você atender a qualquer uma das seguintes condições:

- Seu domínio executa o Elasticsearch 7.7 ou anterior.
- Você usa pacotes personalizados como analisadores de índices.
- Você usa pacotes personalizados como analisadores de pesquisas, mas não inclui o campo atualizável.

Para atualizar os analisadores com os novos arquivos de pacote, você tem duas opções:

- Feche e abra todos os índices que deseja atualizar:

```
POST my-index/_close
POST my-index/_open
```

- Reindexe os índices. Primeiro, crie um índice que use o arquivo de sinônimos atualizado (ou um arquivo inteiramente novo). Observe que apenas o UTF-8 é compatível.

```
PUT my-new-index
{
  "settings": {
    "index": {
      "analysis": {
        "analyzer": {
          "synonym_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["synonym_filter"]
          }
        },
        "filter": {
          "synonym_filter": {
            "type": "synonym",
            "synonyms_path": "analyzers/F222222222"
          }
        }
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "synonym_analyzer"
      }
    }
  }
}
```

Depois reindexe o índice antigo para o novo:

```
POST _reindex
{
  "source": {
    "index": "my-index"
  },
  "dest": {
```

```
    "index": "my-new-index"
  }
}
```

Se você atualiza analisadores de índices com frequência, use aliases de índices para manter um caminho consistente para o índice mais recente:

```
POST _aliases
{
  "actions": [
    {
      "remove": {
        "index": "my-index",
        "alias": "latest-index"
      }
    },
    {
      "add": {
        "index": "my-new-index",
        "alias": "latest-index"
      }
    }
  ]
}
```

Se não precisar do índice antigo, exclua-o:

```
DELETE my-index
```

Dissociação e remoção de pacotes

Desassociar um pacote, seja um dicionário personalizado ou um plug-in pré-empacotado, de um domínio significa que você não poderá mais usar esse pacote ao criar novos índices. Depois que um pacote é dissociado, os índices existentes que estavam usando o pacote não podem mais usá-lo. Você deve remover o pacote de qualquer índice antes de poder dissociá-lo, caso contrário, a dissociação falhará.

O console é a maneira mais simples de dissociar um pacote de um domínio e removê-lo do OpenSearch Serviço. Remover um pacote do OpenSearch Serviço não o remove de sua localização original no Amazon S3.

Desassociar um pacote de um domínio

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Domínios.
3. Escolha o domínio e navegue até a guia Pacotes.
4. Escolha um pacote, Ações e Dissociar. Confirme sua escolha.
5. Aguarde até que o pacote desapareça da lista. Talvez seja necessário atualizar o navegador.
6. Se desejar usar o pacote com outros domínios, pare aqui. Para continuar com a remoção do pacote (se for um dicionário customizado), escolha Pacotes no painel de navegação.
7. Selecione o pacote e Excluir.

Como alternativa, use a API de configuração AWS CLI SDKs, ou para dissociar e remover pacotes. Para obter mais informações, consulte a Referência de AWS CLI Comandos e a Referência da API do Amazon OpenSearch Service.

Gerenciando plug-ins personalizados no Amazon OpenSearch Service

Usando plug-ins personalizados para o OpenSearch Serviço, você pode estender a OpenSearch funcionalidade em áreas como análise de linguagem, filtragem personalizada, classificação e muito mais, possibilitando a criação de experiências de pesquisa personalizadas. Os plug-ins personalizados para OpenSearch podem ser desenvolvidos estendendo a `org.opensearch.plugins.Plugin` classe e, em seguida, empacotando-a em um `.zip` arquivo.

As seguintes extensões de plug-in são atualmente suportadas pelo Amazon OpenSearch Service:

- AnalysisPlugin— amplia a funcionalidade de análise adicionando, por exemplo, analisadores personalizados, tokenizadores de caracteres ou filtros para processamento de texto.
- SearchPlugin— aprimora os recursos de pesquisa com tipos de consulta personalizados, algoritmos de similaridade, opções de sugestão e agregações.
- MapperPlugin— Permite criar tipos de campo personalizados e suas configurações de mapeamento no OpenSearch, permitindo definir como diferentes tipos de dados devem ser armazenados e indexados.

- ScriptPlugin— Permite adicionar recursos de script personalizados a OpenSearch, por exemplo, scripts personalizados para operações como pontuação, classificação e transformações de valores de campo durante a pesquisa ou indexação.

Você pode usar o console do OpenSearch Service ou os comandos de API existentes para pacotes personalizados para carregar e associar o plug-in ao cluster do Amazon OpenSearch Service. Você também pode usar o DescribePackagescomando para descrever todos os pacotes em sua conta e visualizar detalhes, como detalhes da OpenSearch versão e do erro. OpenSearch O serviço valida o pacote de plug-ins quanto à compatibilidade de versões, vulnerabilidades de segurança e operações permitidas do plug-in. Para saber mais sobre pacotes personalizados, consulte the section called “Pacotes”.

OpenSearch versão e Região da AWS suporte

Os plug-ins personalizados são compatíveis com domínios OpenSearch de serviço que executam a OpenSearch versão 2.15 da seguinte forma: Regiões da AWS

- Leste dos EUA (Ohio) (us-east-2)
- Leste dos EUA (Norte da Virgínia) (us-east-1)
- Oeste dos EUA (Oregon) (us-west-2)
- Ásia-Pacífico (Mumbai) (ap-south-1)
- Ásia-Pacífico (Seul) (ap-northeast-2)
- Ásia-Pacífico (Singapura) (ap-southeast-1)
- Ásia-Pacífico (Sydney) (ap-southeast-2)
- Ásia Pacific (Tóquio) (ap-northeast-1)
- Canadá (Central) (ca-central-1)
- Europa (Frankfurt) (eu-central-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- Europa (Paris) (eu-west-3)
- América do Sul (São Paulo) (sa-east-1)

> **Note**
>
> Os plug-ins personalizados contêm código desenvolvido pelo usuário. Os problemas, incluindo violações de SLA, causados por código desenvolvido pelo usuário não são elegíveis a créditos de SLA. Para obter mais informações, consulte Amazon OpenSearch Service - Service Level Agreement.

Tópicos

- Cotas de plug-ins
- Pré-requisitos
- Solução de problemas
- Instalar um plug-in personalizado usando o console
- Gerenciando plug-ins personalizados usando o AWS CLI
- AWS KMS Integração de pacotes personalizados do Amazon OpenSearch Service

Cotas de plug-ins

- Você pode criar até 25 plugins personalizados por conta, por região.
- O tamanho máximo de um plug-in sem compactação é 1 GB.
- O número máximo de servidores que podem ser associados a um único domínio é 20. Essa cota se aplica a todos os tipos de plug-in combinados: opcionais, de terceiros e personalizados.
- Os plug-ins personalizados são compatíveis com domínios que executam a OpenSearch versão 2.15 ou posterior.
- O arquivo `descriptor.properties` do plug-in deve ser compatível com uma versão de mecanismo semelhante à 2.15.0 ou qualquer versão 2.x.x, na qual a versão do patch é definida como zero.

Pré-requisitos

Antes de instalar um plug-in personalizado e associá-lo a um domínio, certifique-se de que os seguintes requisitos são atendidos:

- A versão do mecanismo compatível com o plug-in no arquivo `descriptor.properties` deve ser semelhante à `2.15.0` ou `2.x.0`. Ou seja, a versão do patch deve ser zero.

- Os seguintes atributos devem estar habilitados no domínio:
  - Node-to-node criptografia
  - Criptografia em repouso
  - `EnforceHTTPS` está definido como 'verdadeiro'

    Veja também opensearch-https-requiredno Guia do AWS Config desenvolvedor.
  - Os clientes devem ser compatíveis com a Policy-Min-TLS-1-2-PFS-2023-10. É possível especificar essa compatibilidade usando o comando a seguir. Substitua *`placeholder value`* o por suas próprias informações:

    ```
    aws opensearch update-domain-config \
        --domain-name domain-name \
        --domain-endpoint-options '{"TLSSecurityPolicy":"Policy-Min-TLS-1-2-
    PFS-2023-10" }'
    ```

    Para obter mais informações, consulte DomainEndpointOptionsa Amazon OpenSearch Service API Reference.

Solução de problemas

Se o sistema retornar o erro `PluginValidationFailureReason : The provided plugin could not be loaded`, consulte the section called “Ocorre falha na instalação do plug-in personalizado devido a incompatibilidade entre versões” para obter informações de solução de problemas.

Instalar um plug-in personalizado usando o console

Para associar um plug-in de terceiros a um domínio, primeiro importe a licença e a configuração do plug-in como pacotes.

Para instalar um plug-in personalizado

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, selecione Pacotes.
3. Escolha Importar pacote.
4. Em Nome, insira um nome exclusivo e fácil de identificar para o plug-in.

5. (Opcional) Em Descrição, forneça todos os detalhes úteis sobre o pacote ou sua finalidade.
6. Em Tipo de pacote, escolha Plug-in.
7. Em Origem do pacote, insira o caminho ou navegue até o arquivo ZIP do plug-in no Amazon S3.
8. Para a versão do OpenSearch mecanismo, escolha a versão compatível com OpenSearch o plug-in.
9. Em Criptografia de pacote, escolha se a chave de criptografia do pacote será personalizada ou não. Por padrão, o OpenSearch Service criptografa o pacote do plug-in com um Chave pertencente à AWS. Também é possível usar uma chave gerenciada pelo cliente.
10. Escolha Importar.

Depois de importar o pacote do plug-in, associe-o a um domínio. Para instruções, consulte the section called “Importar e associar um pacote a um domínio”.

Gerenciando plug-ins personalizados usando o AWS CLI

Você pode usar o AWS CLI para gerenciar várias tarefas personalizadas do plug-in.

Tarefas

- Instalando um plug-in personalizado usando o AWS CLI
- Atualizando um plug-in personalizado usando o AWS CLI
- Crie ou atualize um plug-in personalizado com uma AWS KMS chave de segurança
- Atualizando um domínio OpenSearch de serviço com plug-ins personalizados para uma versão posterior do OpenSearch uso do AWS CLI
- Desinstalar e visualizar o status de desassociação de um plug-in personalizado

Instalando um plug-in personalizado usando o AWS CLI

Antes de começar

Antes de associar um plug-in personalizado ao domínio, você deve carregá-lo em um bucket do Amazon Simple Storage Service (Amazon S3). O bucket deve estar localizado no mesmo Região da AWS local em que você pretende usar o plug-in. Para mais informações sobre carregar um objeto em um buckets do S3, consulte Fazer upload de objetos no Guia do usuário do Amazon Simple Storage Service.

Se o pacote contiver informações sensíveis, especifique a criptografia no lado do servidor com chaves gerenciadas pelo S3 ao carregá-lo. Depois de carregar o arquivo, anote o caminho do S3. O formato do caminho é `s3://amzn-s3-demo-bucket/file-path/file-name`.

> **Note**
>
> Opcionalmente, você pode proteger um plug-in personalizado ao criar o plug-in especificando uma chave AWS Key Management Service (AWS KMS). Para mais informações, consulte the section called “Crie ou atualize um plug-in personalizado com uma AWS KMS chave de segurança”.

Para instalar um plug-in personalizado usando o AWS CLI

1. Crie um novo pacote para seu plug-in personalizado executando o seguinte comando create-package e garantindo que os seguintes requisitos sejam atendidos:

   - O bucket e a localização da chave devem apontar para o arquivo `.zip` do plug-in em um bucket do S3 na conta em que você executa os comandos.
   - O bucket do S3 deve estar na mesma região em que o pacote está sendo criado.
   - Somente arquivos `.zip` são compatíveis com pacotes `ZIP-PLUGIN`.
   - O conteúdo do arquivo `.zip` deve seguir a estrutura de diretórios conforme esperado pelo plug-in.
   - O valor fornecido em `--engine-version` deve estar no formato `OpenSearch_{MAJOR}.{MINOR}`. Por exemplo: **`OpenSearch_2.17`**.

   Substitua *`placeholder values`* o por suas próprias informações:

   ```
   aws opensearch create-package \
       --package-name package-name \
       --region region \
       --package-type ZIP-PLUGIN \
       --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
       --engine-version opensearch-version
   ```

2. (Opcional) Visualize o status da operação `create-package`, incluindo quaisquer descobertas de validação e vulnerabilidade de segurança, usando o comando describe-packages. Substitua *`placeholder values`* o por suas próprias informações:

```
aws opensearch describe-packages \
    --region region  \
    --filters '[{"Name": "PackageType","Value": ["ZIP-PLUGIN"]}, {"Name":
 "PackageName","Value": ["package-name"]}]'
```

O comando retorna informações semelhantes às seguintes:

```
{
    "PackageDetailsList": [{
        "PackageID": "pkg-identifier",
        "PackageName": "package-name",
        "PackageType": "ZIP-PLUGIN",
        "PackageStatus": "VALIDATION_FAILED",
        "CreatedAt": "2024-11-11T13:07:18.297000-08:00",
        "LastUpdatedAt": "2024-11-11T13:10:13.843000-08:00",
        "ErrorDetails": {
            "ErrorType": "",
            "ErrorMessage": "PluginValidationFailureReason : Dependency Scan
 reported 3 vulnerabilities for the plugin: CVE-2022-23307, CVE-2019-17571,
 CVE-2022-23305"
        },
        "EngineVersion": "OpenSearch_2.15",
        "AllowListedUserList": [],
        "PackageOwner": "OWNER-XXXX"
    }]
}
```

> **Note**
>
> Durante a `create-package` operação, o Amazon OpenSearch Service verifica o `ZIP-PLUGIN` valor da compatibilidade de versões, extensões de plug-in suportadas e vulnerabilidades de segurança. As vulnerabilidades de segurança são examinadas usando o serviço Amazon Inspector. Os resultados dessas verificações são mostrados no campo `ErrorDetails` da resposta da API.

3. Use o comando associate-package para associar o plug-in ao domínio de OpenSearch serviço de sua escolha usando o ID do pacote criado na etapa anterior.

> **Tip**
>
> Se você tiver vários plug-ins, poderá usar o comando associate-packages para associar vários pacotes a um domínio em uma única operação.

Substitua *placeholder values* o por suas próprias informações:

```
aws opensearch associate-package \
    --domain-name domain-name \
    --region region \
    --package-id package-id
```

> **Note**
>
> O plug-in é instalado e desinstalado usando um processo de implantação azul/verde.

4. (Opcional) Use o list-packages-for-domaincomando para visualizar o status da associação. O status da associação se altera à medida que o fluxo de trabalho avança de `ASSOCIATING` para `ACTIVE`. O status da associação passa para ATIVO após a conclusão da instalação do plug-in, e ele está pronto para uso.

   Substitua *placeholder values* por suas próprias informações.

   ```
   aws opensearch list-packages-for-domain \
       --region region \
       --domain-name domain-name
   ```

Atualizando um plug-in personalizado usando o AWS CLI

Use o comando update-package para fazer alterações em um plug-in.

> **Note**
>
> Opcionalmente, você pode proteger um plug-in personalizado ao atualizar o plug-in especificando uma chave AWS Key Management Service (AWS KMS). Para mais

informações, consulte the section called “Crie ou atualize um plug-in personalizado com uma AWS KMS chave de segurança”.

Para atualizar um plug-in personalizado usando o AWS CLI

- Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch update-package \
    --region region \
    --package-id package-id \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
    --package-description description
```

Depois de atualizar um pacote, você pode usar o comando associate-package ou associate-packages para aplicar atualizações de pacote em um domínio.

> **Note**
>
> Você pode auditar, criar, atualizar, associar e desassociar operações no plug-in usando AWS CloudTrail. Para obter mais informações, consulte Monitorando chamadas OpenSearch de API do Amazon Service com AWS CloudTrail.

Crie ou atualize um plug-in personalizado com uma AWS KMS chave de segurança

Opcionalmente, você pode proteger um plug-in personalizado ao atualizá-lo ou atualizá-lo especificando uma chave do AWS KMS . Para fazer isso, defina `PackageEncryptionOptions` como `true` e especifique o nome do recurso da Amazon (ARN) da chave, como mostrado nos exemplos a seguir.

Exemplo: criar um plug-in personalizado com AWS KMS chave de segurança

```
aws opensearch create-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
```

```
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

Exemplo: atualizar um plug-in personalizado com a AWS KMS chave de segurança

```
aws opensearch update-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

Important

Se a AWS KMS chave especificada for desativada ou excluída, ela poderá deixar o cluster associado inoperacional.

Para obter mais informações sobre AWS KMS integração com pacotes personalizados,the section called “AWS KMS Integração de pacotes personalizados do Amazon OpenSearch Service”.

Atualizando um domínio OpenSearch de serviço com plug-ins personalizados para uma versão posterior do OpenSearch uso do AWS CLI

Quando você precisar atualizar um domínio OpenSearch de serviço que usa plug-ins personalizados para uma versão posterior do OpenSearch, conclua os processos a seguir.

Para atualizar um domínio OpenSearch de serviço com plug-ins personalizados para uma versão posterior do OpenSearch uso do AWS CLI

1. Use o comando create-package para criar um novo pacote para seu plug-in especificando a nova versão. OpenSearch

   O nome do pacote deve ser o mesmo para o plug-in de todas as versões do mecanismo. A alteração do nome do pacote faz com que o processo de atualização do domínio falhe durante a blue/green implantação.
2. Atualize o domínio para a versão acima seguindo as etapas em the section called “Atualização de domínios”.

   Durante esse processo, o Amazon OpenSearch Service desassocia a versão anterior do pacote de plug-ins e instala a nova versão usando uma implantação. blue/green

Desinstalar e visualizar o status de desassociação de um plug-in personalizado

Para desinstalar o plug-in de qualquer domínio, use o comando dissociate-package. A execução desse comando também remove as configurações ou pacotes de licença relacionados. Em seguida, você pode usar o list-packages-for-domaincomando para visualizar o status da dissociação.

> **Tip**
>
> Você também pode usar o comando dissociate-packages para desinstalar vários plug-ins de um domínio em uma única operação.

Para desinstalar e visualizar o status de desassociação de um plug-in personalizado

1. Desabilite o plug-in em todos os índices. Faça isso antes de desassociar o pacote de plug-in.

   Se você tentar desinstalar um plug-in antes de desativá-lo de todos os índices, o processo de blue/green implantação permanecerá preso no `Processing` estado.
2. Execute o comando a seguir para desinstalar o plug-in. Substitua *`placeholder values`* por suas próprias informações.

```
aws opensearch dissociate-package \
    --region region \
    --package-id plugin-package-id \
```

```
    --domain-name domain name
```

3. (Opcional) Execute o list-packages-for-domaincomando para ver o status da dissociação.

AWS KMS Integração de pacotes personalizados do Amazon OpenSearch Service

Os pacotes personalizados do Amazon OpenSearch Service fornecem criptografia por padrão para proteger seus `ZIP-PLUGIN` pacotes em repouso usando Chaves gerenciadas pela AWS.

- Chaves pertencentes à AWS— Os pacotes personalizados do Amazon OpenSearch Service usam essas chaves por padrão para criptografar automaticamente seus `ZIP-PLUGIN` pacotes. Você não pode visualizar, gerenciar, usar Chaves pertencentes à AWS ou auditar seu uso. No entanto, você não precisa realizar nenhuma ação nem alterar nenhum programa para proteger as chaves que criptografam seus dados. Para saber mais, consulte Chaves pertencentes à AWS no Guia do desenvolvedor do AWS Key Management Service .
- Chaves gerenciadas pelo cliente — Você pode adicionar uma segunda camada de criptografia sobre a existente Chaves pertencentes à AWS escolhendo uma chave gerenciada pelo cliente ao criar seu pacote `ZIP-PLUGIN` personalizado.

  Os pacotes personalizados do Amazon OpenSearch Service oferecem suporte ao uso de uma chave simétrica gerenciada pelo cliente que você cria, possui e gerencia para adicionar uma segunda camada de criptografia sobre a criptografia existente AWS . Como você tem controle total dessa camada de criptografia, pode realizar as seguintes tarefas:

  - Estabelecer e manter as políticas de chaves
  - Estabelecer e manter AWS Identity and Access Management (IAM) políticas e subsídios
  - Habilitar e desabilitar políticas de chave
  - Alternar o material de criptografia de chaves
  - Adicionar tags.
  - Criar aliases de chaves
  - Programar a exclusão de chaves

Para saber mais, consulte Chaves mestras do cliente (CMKs) no AWS Key Management Service Guia do desenvolvedor.

> **Note**
>
> Os pacotes personalizados do Amazon OpenSearch Service habilitam automaticamente a criptografia em repouso Chaves pertencentes à AWS , usando gratuitamente. No entanto, AWS KMS cobranças se aplicam quando você usa uma chave gerenciada pelo cliente. Para saber mais sobre preços, consulte Preços do AWS Key Management Service.

Como o OpenSearch serviço de pacotes personalizados do Amazon Service usa subsídios em AWS KMS

OpenSearch Os pacotes personalizados de serviços exigem uma concessão para usar sua chave gerenciada pelo cliente.

Quando você cria um `ZIP-PLUGIN` pacote criptografado com uma chave gerenciada pelo cliente, o OpenSearch serviço de pacotes personalizados do Amazon Service cria uma concessão em seu nome enviando uma CreateGrantsolicitação para AWS KMS. As concessões concedem ao OpenSearch Serviço acesso a uma AWS KMS chave em sua conta. AWS KMS As concessões criadas pelos pacotes personalizados do OpenSearch Service têm uma restrição que permite operações somente quando a solicitação inclui um contexto de criptografia com seu ID de pacote personalizado.

Os pacotes personalizados do Amazon OpenSearch Service exigem que a concessão use sua chave gerenciada pelo cliente para as seguintes operações internas:

Operation	Description
`DescribeKey`	Envia `DescribeKey` solicitações AWS KMS para verificar se o ID simétrico da chave gerenciada pelo cliente inserido ao criar o pacote de plug-ins é válido.
`GenerateDataKeyWithoutPlaintext`	Envia `GenerateDataKeyWithoutPlaintext` solicitações AWS KMS para gerar chaves de dados criptografadas pela chave gerenciada pelo cliente.
`GenerateDataKey`	Envia `GenerateDataKey` solicitações AWS KMS para gerar chaves de dados para criptografar o pacote ao copiá-lo internamente.

Operation	Description
`Decrypt`	Envia `Decrypt` solicitações AWS KMS para descriptografar as chaves de dados criptografadas para que elas possam ser usadas para descriptografar seus dados.

É possível revogar o acesso à concessão ou remover o acesso do serviço à chave gerenciada pelo cliente a qualquer momento. Se você fizer isso, o OpenSearch Serviço não poderá acessar nenhum dado criptografado pela chave gerenciada pelo cliente, o que afeta as operações que dependem desses dados. Por exemplo, se você tentar associar um pacote de plug-in que o OpenSearch Serviço não pode acessar, a operação retornará um `AccessDeniedException` erro.

Criar uma chave gerenciada pelo cliente

Você pode criar uma chave simétrica gerenciada pelo cliente usando o Console de gerenciamento da AWS ou o. AWS KMS APIs

Para criar uma chave simétrica gerenciada pelo cliente

- Para criar uma chave do KMS, siga as etapas em Criar uma chave do KMS no Guia do desenvolvedor do AWS Key Management Service .

Política de chave

As políticas de chaves controlam o acesso à chave gerenciada pelo cliente. Cada chave gerenciada pelo cliente deve ter exatamente uma política de chaves, que contém declarações que determinam quem pode usar a chave e como pode usá-la. Ao criar a chave gerenciada pelo cliente, é possível especificar uma política de chave. Para saber mais, consulte Políticas de chaves no AWS KMS no Guia do desenvolvedor do AWS Key Management Service .

Para usar a chave gerenciada pelo cliente com os recursos, você deve permitir as seguintes operações de API na política de chave:

- `kms:CreateGrant`: adiciona uma concessão a uma chave gerenciada pelo cliente. Concede acesso de controle a uma AWS KMS chave especificada, permitindo o acesso às operações de concessão exigidas pelos pacotes personalizados do OpenSearch Serviço. Para saber mais sobre o uso de concessões, consulte o Guia do desenvolvedor do AWS KMS.

  Isso permite que o OpenSearch Serviço faça o seguinte:

  - Chame `GenerateDataKeyWithoutPlainText` para gerar uma chave de dados criptografada e armazená-la para validações adicionais.
  - Chame `GenerateDataKey` para copiar o pacote do plug-in internamente.
  - Chame `Decrypt` para acessar o pacote do plug-in internamente.
  - Configure uma entidade principal aposentada para permitir que o serviço para `RetireGrant`.
- `kms:DescribeKey`— Fornece os detalhes da chave gerenciada pelo cliente para permitir que o OpenSearch Serviço valide a chave.
- `kms:GenerateDataKey`,`kms:GenerateDataKeyWithoutPlaintext`, `kms:Decrypt` — Fornece acesso aos pacotes personalizados do OpenSearch Serviço para usar essas operações na concessão.

Veja a seguir exemplos de declarações de política que você pode adicionar aos pacotes personalizados do OpenSearch Service:

```
"Statement" : [
  {
    "Sid" : "Allow access to principals authorized to use OpenSearch Service custom
 packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:CreateGrant",
      "kms:GenerateDataKey",
      "kms:GenerateDataKeyWithoutPlaintext",
      "kms:Decrypt"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      },
      "StringEquals" : {
        "kms:EncryptionContext:packageId": "Id of the package"
      }
    }
  },
```

```
  {
    "Sid" : "Allow access to principals authorized to use Amazon OpenSearch Service
 custom packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:DescribeKey"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      }
    }
  }
]
```

Para saber mais sobre a especificação de permissões em uma política, consulte Políticas de chave no AWS KMS no Guia do desenvolvedor do AWS Key Management Service .

Para obter mais informações sobre como solucionar problemas de acesso por chave, consulte Solução de problemas de AWS KMS permissões no Guia do AWS Key Management Service desenvolvedor.

Especifique uma chave gerenciada pelo cliente para pacotes personalizados do Amazon OpenSearch Service

Você pode especificar uma chave gerenciada pelo cliente como uma segunda camada de criptografia para pacotes `ZIP-PLUGIN`.

Ao criar um pacote de plug-in, você pode especificar a chave de dados inserindo um ID de AWS KMS chave, que os pacotes personalizados do OpenSearch Serviço usam para criptografar o pacote de plug-in.

AWS KMS ID da chave — Um identificador de chave para uma chave gerenciada pelo AWS KMS cliente. Insira uma ID de chave, um ARN de chave, um nome de alias ou um ARN de alias.

Contexto de criptografia de pacotes personalizados do Amazon OpenSearch Service

Um contexto de criptografia é um conjunto opcional de pares de chave/valor que pode conter informações contextuais adicionais sobre os dados.

AWS KMS usa o contexto de criptografia como dados autenticados adicionais para oferecer suporte à criptografia autenticada. Quando você inclui um contexto de criptografia em uma solicitação para criptografar dados, AWS KMS vincula o contexto de criptografia aos dados criptografados. Para descriptografar os dados, você inclui o mesmo contexto de criptografia na solicitação.

Contexto de criptografia de pacotes personalizados do Amazon OpenSearch Service

Os pacotes personalizados do Amazon OpenSearch Service usam o mesmo contexto de criptografia em todas as operações AWS KMS criptográficas, onde a chave está `packageId` e o valor é o `package-id` do seu pacote de plug-in.

Usar o contexto de criptografia para monitoramento

Ao usar uma chave simétrica gerenciada pelo cliente para criptografar o pacote de plug-in, você também pode usar o contexto de criptografia em registros e logs de auditoria para identificar como a chave gerenciada pelo cliente está sendo usada. O contexto de criptografia também aparece nos registros gerados pelo AWS CloudTrail ou Amazon CloudWatch Logs.

Usar o contexto de criptografia para controlar o acesso à chave gerenciada pelo cliente

Você pode usar o contexto de criptografia nas políticas de chave e políticas do IAM como condições para controlar o acesso à sua chave simétrica gerenciada pelo cliente. Você também pode usar restrições no contexto de criptografia em uma concessão.

OpenSearch Os pacotes personalizados de serviços usam uma restrição de contexto de criptografia nas concessões para controlar o acesso à chave gerenciada pelo cliente em sua conta ou região. A restrição da concessão exige que as operações permitidas pela concessão usem o contexto de criptografia especificado.

Veja a seguir exemplos de declarações de políticas de chave para conceder acesso a uma chave gerenciada pelo cliente para um contexto de criptografia específico. A condição nesta declaração de política exige que as concessões tenham uma restrição de contexto de criptografia que especifique o contexto de criptografia.

```
{
```

```
    "Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action": "kms:DescribeKey",
    "Resource": "*"
},
{
    "Sid": "Enable OpenSearch Service custom packages to use the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action" : [
         "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals" : {
            "kms:EncryptionContext:packageId": "ID of the package"
         }
    }
}
```

Monitorando suas chaves de criptografia para serviços de pacotes OpenSearch personalizados

Ao usar uma chave gerenciada pelo AWS KMS cliente com seus recursos de OpenSearch serviço de pacotes personalizados de serviços, você pode usar CloudTrail ou CloudWatch Logs para rastrear solicitações enviadas por pacotes OpenSearch personalizados AWS KMS.

Saiba mais

Os recursos a seguir fornecem mais informações sobre a criptografia de dados em repouso.

- Para obter mais informações sobre conceitos AWS KMS básicos, consulte AWS KMS keyso Guia do AWS Key Management Service desenvolvedor.
- Para obter mais informações sobre as melhores práticas de segurança AWS KMS, consulte o guia de orientação AWS prescritiva para obter AWS Key Management Service as melhores práticas.

Instalação de plug-ins de terceiros no Amazon OpenSearch Service

O Amazon OpenSearch Service oferece suporte a plug-ins de terceiros de parceiros selecionados. Esses plug-ins podem aprimorar sua OpenSearch configuração com recursos adicionais, como analisadores personalizados, tokenizadores ou recursos de criptografia. Siga as instruções específicas de instalação e configuração fornecidas pelos desenvolvedores terceirizados para garantir a integração adequada com seu domínio OpenSearch de serviço.

> **Note**
>
> Você deve conseguir e manter licenças válidas diretamente dos desenvolvedores terceirizados. Alguns provedores podem não habilitar todos os plug-ins Regiões da AWS, portanto, verifique a disponibilidade com o provedor do plug-in.

Os seguintes plug-ins de terceiros estão disponíveis para uso com o OpenSearch Serviço:

- Plugin de criptografia Portal26 (Titanium-lockbox): usa criptografia certificada pelo NIST FIPS 140-2 para criptografar dados à medida que são indexados. Inclui compatibilidade com Bring Your Own Key (BYOK), que permite que você gerencie suas chaves de criptografia para aumentar a segurança. O plugin é fornecido pelo Portal26 e requer a OpenSearch versão 2.15 ou superior.
- Name Match (RNI): verifica a correspondência de nomes, organizações, endereços e datas em mais de 24 idiomas, o que aumenta a segurança e a conformidade. O plugin é fornecido pela Babel Street e requer a OpenSearch versão 2.15 ou superior.

Tópicos

- Pré-requisitos
- Instalar plug-ins de terceiros
- Próximas etapas

Pré-requisitos

Antes de instalar um plug-ins de terceiros, realize as seguintes etapas:

- Obteve a configuração do plug-in e os arquivos de licença e os carregou em um bucket do Amazon S3. O bucket deve estar no Região da AWS mesmo domínio.

- Um plug-in de terceiros é um tipo de plug-in personalizado. Certifique-se de que o domínio atenda aos pré-requisitos para plug-ins personalizados.

Instalar plug-ins de terceiros

Para associar um plug-in de terceiros a um domínio de OpenSearch serviço, primeiro você deve fazer upload de três pacotes separados: o pacote de licença, o pacote de configuração e o pacote de plug-in.

- O pacote de licença inclui as informações de licenciamento ou os metadados associados ao plug-in, no formato .json ou .xml.
- O pacote de configuração contém os arquivos de configuração do plug-in e os ativos e as configurações de apoio. Esses arquivos definem como o plug-in se comporta ou se integra. OpenSearch
- O pacote do plug-in contém o binário do plug-in compilado, que é o código executável OpenSearch executado. Esse é o núcleo da funcionalidade do plug-in.

Depois de carregar ambos os pacotes, você poderá associar o plug-in e a licença a um domínio compatível.

Console

Para associar um plug-in de terceiros a um domínio, primeiro importe a licença e a configuração do plug-in como pacotes.

Para instalar um plug-in de terceiros

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, selecione Pacotes.
3. Primeiro, importe o pacote de licença. Escolha Importar pacote.
4. Em Tipo de pacote, escolha Licença.
5. Em Origem do pacote, insira o caminho para o arquivo de licença JSON ou XML no Amazon S3.
6. Escolha Importar. O pacote aparece na guia Licenças da página Pacotes.
7. Agora, importe a configuração do plugin. Escolha Importar pacote novamente.
8. Em Tipo de pacote, escolha Configuração.

9. Em Origem do pacote, insira o caminho até o arquivo ZIP de configuração do plug-in no Amazon S3.
10. Escolha Importar.
11. Por fim, importe o plug-in em si. Escolha Importar pacote.
12. Em Tipo de pacote, escolha Plug-in.
13. Em Origem do pacote, insira o caminho até o arquivo ZIP do plug-in no Amazon S3.
14. Selecione a versão OpenSearch do mecanismo compatível com o plug-in.
15. Escolha Importar.

Para associar um plug-in de terceiros a um domínio

1. Agora, associe a licença e a configuração do plug-in ao domínio. No painel de navegação à esquerda, selecione Domínios.
2. Escolha o nome do domínio para abrir a configuração de cluster.
3. Navegue até a guia Plug-ins.
4. Escolha Associar pacotes e selecione os pacotes de plug-in, licença e configuração que você acabou de importar.
5. Escolha Selecionar.
6. Escolha Próximo. Revise os pacotes a serem associados e escolha Associar.

CLI

Primeiro, use o comando create-package para criar um novo pacote contendo a licença do plug-in. A `S3Key` deve apontar para um arquivo .json ou .xml no Amazon S3 que inclua o texto ou os metadados da licença.

```
aws opensearch create-package \
  --package-name plugin-license-package \
  --package-type PACKAGE-LICENSE \
  --package-source S3BucketName=my-bucket,S3Key=licenses/my-plugin-license.json
```

Use o comando create-package novamente para criar um pacote contendo a configuração do plug-in. A `S3Key` deve apontar para um arquivo .zip no Amazon S3 que siga a estrutura de diretórios esperada pelo plug-in.

```
aws opensearch create-package \
  --package-name plugin-config-package \
  --package-type PACKAGE-CONFIG \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Use o comando create-package novamente para criar um pacote contendo o plug-in em si. A `S3Key` deve apontar para o arquivo .zip do plug-in no Amazon S3.

```
aws opensearch create-package \
  --package-name plugin-package \
  --package-type ZIP-PLUGIN \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Por fim, use o comando associate-package para vincular o plug-in, a licença e a configuração do parceiro a um domínio compatível especificando o pacote para cada um. IDs Especifique o ID do plug-in como um pré-requisito para os outros pacotes, o que significa que ele deve ser associado ao domínio antes dos outros pacotes.

```
aws opensearch associate-packages \
  --domain-name my-domain \
  --package-list '[{"PackageID": "plugin-package-id"},{"PackageID": "license-package-id","PrerequisitePackageIDList":["plugin-package-id"]},{"PackageID":"config-package-id","PrerequisitePackageIDList":["plugin-package-id"]}]'
```

Próximas etapas

Quando a associação for concluída, você poderá habilitar o plug-in em índices específicos ou configurá-lo conforme necessário segundo seus requisitos. Para aplicar a funcionalidade do plug-in de terceiros a índices específicos, modifique as configurações de índice durante a criação do índice ou atualize os índices existentes. Por exemplo, se o plug-in de terceiros incluir um analisador personalizado, o referencie nas configurações de índice.

Para aplicar os atributos do plug-in de modo consistente entre vários índices, use modelos de índice que incluem as configurações do plug-in. Sempre consulte a documentação do plug-in para entender como configurar seus recursos para sua OpenSearch configuração.

Consultando seus dados do Amazon OpenSearch Service com SQL

Você pode usar o SQL para consultar seu Amazon OpenSearch Service, em vez de usar a DSL de OpenSearch consulta baseada em JSON. Consultar com SQL é útil se você já está familiarizado com a linguagem ou se deseja integrar seu domínio a uma aplicação que usa SQL. O suporte a SQL está disponível em domínios que executam o Elasticsearch 6.5 OpenSearch ou superior.

> **Note**
>
> Esta documentação descreve a compatibilidade de versões entre o OpenSearch Service e várias versões do plug-in SQL, bem como o driver JDBC e ODBC. Consulte a OpenSearchdocumentação de código aberto para obter informações sobre a sintaxe de consultas básicas e complexas, funções, consultas de metadados e funções agregadas.

Use a tabela a seguir para encontrar a versão do plug-in SQL compatível com cada uma OpenSearch das versões do Elasticsearch.

OpenSearch

OpenSearch versão	Versão do plug-in SQL	Recursos notáveis
2.19.0	2.19.0.0	
2.18.0	2.18.0.0	
2.17.0	2.17.0.0	
2.15.0	2.15.0.0	
2.13.0	2.13.0.0	
2.11.0	2.11.0.0	Adicionar suporte para linguagem e consultas PPL
2.9.0	2.9.0.0	Adicione o conector Spark e suporte à tabela e às funções PromQL

OpenSearch versão	Versão do plug-in SQL	Recursos notáveis
2.7.0	2.7.0.0	Adicionar API `datasource`
2.5.0	2.5.0.0	
2.3.0	2.3.0.0	Adicione funções de data e hora `maketime` e `makedate`
1.3.0	1.3.0.0	Suporta tamanho limite de consulta padrão e cláusula IN para selecionar em uma lista de valores
1.2.0	1.2.0.0	Adicionar novo protocolo para o formato de resposta de visualização
1.1.0	1.1.0.0	Ofereça suporte à função de correspondência como um filtro no SQL e PPL
1.0.0	1.0.0.0	Suporte à consulta de um fluxo de dados

Open Distro for Elasticsearch

Versão do Elasticsearch	Versão do plug-in SQL	Recursos notáveis
7.10	1.13.0	NULL FIRST e LAST para funções de janela, função CAST (), comandos SHOW e DESCRIBE
7.9	1.11.0	Adicione date/time funções adicionais, palavra-chave ORDER BY
7.8	1.9.0	
7.7	1.8.0	
7.3	1.3.0	Operadores de strings e numéricos diversos
7.1	1.1.0	

Chamada de exemplo

Para consultar seus dados usando o SQL, envie solicitações HTTP para `_sql` usando o seguinte formato:

```
POST domain-endpoint/_plugins/_sql
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

> **Note**
>
> Se seu domínio estiver executando o Elasticsearch em vez de OpenSearch, o formato é. `_opendistro/_sql`

Notas e diferenças

As chamadas para `_plugins/_sql` incluem nomes de índice no corpo da solicitação, portanto, elas têm as mesmas considerações da política de acesso das operações bulk, mget, e msearch. Como sempre, siga o princípio do privilégio mínimo ao conceder permissões para operações de API.

Para obter considerações de segurança sobre o uso de SQL com o controle de acesso refinado, consulte the section called “Controle de acesso refinado”.

O plug-in OpenSearch SQL inclui muitas configurações ajustáveis. No OpenSearch Serviço, use o `_cluster/settings` caminho, não o caminho das configurações do plug-in (`_plugins/_query/settings`):

```
PUT _cluster/settings
{
  "transient" : {
    "plugins.sql.enabled" : true
  }
}
```

Para domínios herdados do Elasticsearch, substitua `plugins` por `opendistro`:

```
PUT _cluster/settings
```

```
{
  "transient" : {
    "opendistro.sql.enabled" : true
  }
}
```

SQL Workbench

O SQL Workbench é uma interface de usuário de OpenSearch painéis que permite executar consultas SQL sob demanda, traduzir SQL em seu equivalente REST e visualizar e salvar resultados como texto, JSON, JDBC ou CSV. Para saber mais, consulte Query Workbench.

SQL CLI

O SQL CLI é uma aplicação Python autônoma que você pode executar com o comando `opensearchsql`. Para obter as etapas de instalação, configuração e uso, consulte SQL CLI.

Driver JDBC

O driver Java Database Connectivity (JDBC) permite integrar domínios OpenSearch de serviço com seus aplicativos favoritos de business intelligence (BI). Para baixar o driver, clique aqui. Para obter mais informações, consulte o GitHubrepositório.

Driver ODBC

O driver de conectividade do banco de dados aberta (ODBC) é um driver ODBC somente leitura para Windows e macOS que permite conectar aplicativos de business intelligence e visualização de dados, como o Microsoft Excel, ao plug-in SQL.

Para obter informações sobre a instalação do driver, consulte o repositório SQL em GitHub.

Pesquisa entre clusters no Amazon Service OpenSearch

A pesquisa entre clusters no Amazon OpenSearch Service permite realizar consultas e agregações em vários domínios conectados. Muitas vezes, faz mais sentido usar vários domínios menores em vez de um único domínio grande, principalmente quando você está executando diferentes tipos de workloads.

Os domínios específicos de workloads permitem executar as seguintes tarefas:

- Otimizar cada domínio escolhendo tipos de instância para workloads específicas.
- Estabelecer limites de isolamento de falhas entre workloads. Isso significa que, se uma de suas workloads falhar, a falha estará contida nesse domínio específico e não afetará as outras workloads.
- Dimensionar mais facilmente entre domínios.

A pesquisa entre clusters é compatível com OpenSearch painéis, para que você possa criar visualizações e painéis em todos os seus domínios. Você paga taxas padrão AWS de transferência de dados pelos resultados de pesquisa transferidos entre domínios.

> **Note**
>
> O código aberto OpenSearch também tem documentação para pesquisa entre clusters. A configuração difere significativamente para clusters de código aberto em comparação com os domínios gerenciados do Amazon OpenSearch Service. Mais notavelmente, em OpenSearch Service, você configura conexões entre clusters usando o cURL Console de gerenciamento da AWS em vez de por meio de cURL. Além disso, o serviço gerenciado usa AWS Identity and Access Management (IAM) para autenticação entre clusters, além do controle de acesso refinado. Portanto, recomendamos usar essa documentação, em vez da OpenSearch documentação de código aberto, para configurar a pesquisa entre clusters para seus domínios.

Tópicos

- Limitações
- Pré-requisitos da pesquisa entre clusters
- Preços da pesquisa entre clusters
- Configuração de uma conexão
- Remoção de uma conexão
- Configuração da segurança e demonstração de exemplo
- OpenSearch Painéis

Limitações

A pesquisa entre clusters tem várias limitações importantes:

- Você não pode conectar um domínio do Elasticsearch a um OpenSearch domínio.
- Você não pode se conectar a OpenSearch/Elasticsearch clusters autogerenciados.
- Para conectar domínios entre regiões, ambos os domínios devem estar no Elasticsearch 7.10 ou posterior ou. OpenSearch
- Um domínio pode ter um máximo de 20 conexões de saída. Da mesma forma, um domínio pode ter um máximo de 20 conexões de entrada. Em outras palavras, um domínio pode se conectar a um máximo de 20 outros domínios.
- O domínio de origem deve estar na mesma versão ou em uma versão superior à do domínio de destino. Se você configurar uma conexão bidirecional entre dois domínios e quiser atualizar um ou ambos, primeiro exclua uma das conexões.
- Não é possível usar dicionários personalizados ou o SQL com a pesquisa entre clusters.
- Você não pode usar CloudFormation para conectar domínios.
- Não é possível usar a pesquisa entre clusters em instâncias M3 ou expansíveis (T2 e T3).

Pré-requisitos da pesquisa entre clusters

Antes de configurar a pesquisa entre clusters, verifique se os domínios atendem aos seguintes requisitos:

- Dois OpenSearch domínios ou domínios do Elasticsearch na versão 6.7 ou posterior
- Controle de acesso refinado habilitado
- Node-to-node criptografia ativada
- Se um dos domínios estiver em uma VPC, os domínios devem estar conectados via VPC Peering ou Transit Gateway, e os grupos de segurança devem permitir o tráfego entre eles.

Preços da pesquisa entre clusters

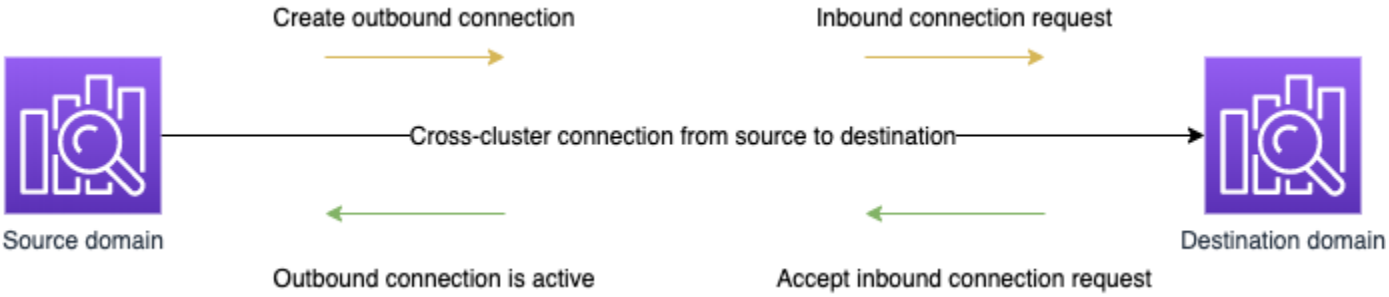
Não há custo adicional para a pesquisa entre domínios.

Configuração de uma conexão

O domínio de “origem” refere-se ao domínio no qual uma solicitação de pesquisa entre clusters se origina. Ou seja, o domínio de origem é aquele para o qual você envia a solicitação de pesquisa inicial.

O domínio de “destino” é aquele que o domínio de origem consulta.

Uma conexão entre clusters é unidirecional do domínio de origem para o domínio de destino. Isso significa que o domínio de destino não pode consultar o domínio de origem. No entanto, você pode configurar outra conexão na direção oposta.



O domínio de origem cria uma conexão de “saída” com o domínio de destino. O domínio de destino recebe uma solicitação de conexão de “entrada” do domínio de origem.

Como configurar uma conexão

1. No painel do domínio, escolha um domínio e escolha a guia Conexões.
2. Na seção Conexões de saída, escolha Solicitar.
3. Em Alias de conexão, insira um nome para a conexão.
4. Escolha entre se conectar a um domínio em sua Conta da AWS região ou em outra conta ou região.
   - Para se conectar a um cluster em sua Conta da AWS região, selecione o domínio no menu suspenso e escolha Solicitar.
   - Para se conectar a um cluster em outra região Conta da AWS ou região, selecione o ARN do domínio remoto e escolha Solicitar. Para conectar domínios entre regiões, ambos os domínios devem estar executando a versão 7.10 ou posterior do Elasticsearch ou. OpenSearch
5. Para ignorar clusters indisponíveis para consultas de cluster, selecione Ignorar indisponíveis. Essa configuração garante que suas consultas entre clusters retornem resultados parciais, apesar de falhas em um ou mais clusters remotos.
6. A pesquisa entre clusters primeiro valida a solicitação de conexão para ter certeza de que os pré-requisitos são atendidos. Se os domínios forem considerados incompatíveis, a solicitação de conexão entrará no estado `Validation failed`.
7. Depois que a solicitação de conexão é validada com êxito, ela é enviada para o domínio de destino, onde precisa ser aprovada. Até que essa aprovação aconteça, a conexão permanecerá

em um estado `Pending acceptance`. Quando a solicitação de conexão é aceita no domínio de destino, o estado muda para `Active` e o domínio de destino torna-se disponível para consultas.

- A página de domínio mostra os detalhes gerais da integridade do domínio e da instância do domínio de destino. Os proprietários de domínios têm a flexibilidade de criar, visualizar, remover e monitorar conexões de saída e de entrada de seus domínios.

Depois de estabelecida a conexão, todo o tráfego que flui entre os nós dos domínios conectados é criptografado. Se você conectar um domínio da VPC a um domínio que não seja de VPC e o domínio de não VPC for um endpoint público que pode receber tráfego da Internet, o tráfego entre clusters entre os domínios ainda será criptografado e seguro.

Remoção de uma conexão

A remoção de uma conexão interrompe qualquer operação entre clusters em seus índices.

1. No painel do domínio, selecione a guia Conexões.
2. Selecione as conexões de domínio a serem removidas e escolha Excluir. Em seguida, confirme a exclusão.

Você pode executar essas etapas no domínio de origem ou de destino para remover a conexão. Depois que a conexão é removida, ela permanece visível com o status `Deleted` por um período de 15 dias.

Não é possível excluir um domínio com conexões ativas entre clusters. Para excluir um domínio, primeiro remova todas as conexões de entrada e saída desse domínio. Isso garante que você leve em consideração os usuários de domínio entre clusters antes de excluir o domínio.

Configuração da segurança e demonstração de exemplo

1. Envie uma solicitação de pesquisa entre clusters para o domínio de origem.
2. O domínio de origem avalia essa solicitação em relação à política de acesso ao domínio. Como a pesquisa entre clusters requer controle de acesso refinado, recomendamos uma política de acesso aberto no domínio de origem.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/src-domain/*"
    }
  ]
}
```

> **Note**
>
> Se você incluir índices remotos no caminho, deverá codificar em URL o URI no ARN do domínio. Por exemplo, use `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst%3Aremote_index` em vez de `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst:remote_index`.

Se você optar por usar uma política de acesso restritiva além do controle de acesso refinado, sua política deverá permitir o acesso, no mínimo, ao `es:ESHttpGet`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": [
          "arn:aws:iam::111122223333:user/test-user"
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/src-domain/*"
    }
  ]
}
```

3. O controle de acesso refinado no domínio de origem avalia a solicitação:

   - A solicitação é assinada com credenciais básicas do IAM ou HTTP válidas?
   - Em caso afirmativo, o usuário tem permissão para realizar a pesquisa e acessar os dados?

   Se a solicitação pesquisar dados somente no domínio de destino (por exemplo, `dest-alias:dest-index/_search`), você só precisará de permissões no domínio de destino.

   Se a solicitação pesquisar dados nos dois domínios (por exemplo, `source-index,dest-alias:dest-index/_search`), você precisará de permissões nos dois domínios.

   No controle de acesso refinado, os usuários devem ter a permissão `indices:admin/shards/search_shards` além das permissões `read` ou `search` padrão para os índices relevantes.

4. O domínio de origem passa a solicitação para o domínio de destino. O domínio de destino avalia essa solicitação em relação à política de acesso ao domínio. Você deve incluir a permissão `es:ESCrossClusterGet` no domínio de destino:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain"
```

```
    }
  ]
}
```

Verifique se a permissão `es:ESCrossClusterGet` é aplicada a `/dst-domain` e não a `/dst-domain/*`.

No entanto, essa política mínima só permite pesquisas entre clusters. Para executar outras operações, como indexar documentos e executar pesquisas padrão, você precisa de permissões adicionais. Recomendamos a seguinte política no domínio de destino:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/dst-domain"
    }
  ]
}
```

> **Note**
>
> Todas as solicitações de pesquisa entre clusters entre domínios são criptografadas em trânsito por padrão como parte da node-to-node criptografia.

5. O domínio de destino executa a pesquisa e retorna os resultados para o domínio de origem.
6. O domínio de origem combina seus próprios resultados (se houver) com os resultados do domínio de destino e os retorna para você.
7. Recomendamos o Postman para solicitações de teste:

   - No domínio de destino, indexe um documento:

   ```
   POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1

   {
     "Dracula": "Bram Stoker"
   }
   ```

   - Para consultar esse índice do domínio de origem, inclua o alias de conexão do domínio de destino dentro da consulta.

   ```
   GET https://src-domain.us-east-1.es.amazonaws.com/<connection_alias>:books/
   _search

   {
       ...
     "hits": [
       {
         "_index": "source-destination:books",
         "_type": "_doc",
         "_id": "1",
         "_score": 1,
         "_source": {
           "Dracula": "Bram Stoker"
         }
       }
     ]
   }
   ```

Você pode encontrar o alias de conexão na guia Conexões no painel do domínio.

- Se você configurar uma conexão entre `domain-a -> domain-b` com alias de conexão `cluster_b` e `domain-a -> domain-c` com alias de conexão `cluster_c`, `domain-a`, pesquise `domain-b` e `domain-c` da seguinte forma:

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
    "match": {
      "user": "domino"
    }
  }
}
```

Resposta

```
{
  "took": 150,
  "timed_out": false,
  "_shards": {
    "total": 3,
    "successful": 3,
    "failed": 0,
    "skipped": 0
  },
  "_clusters": {
    "total": 3,
    "successful": 3,
    "skipped": 0
  },
  "hits": {
    "total": 3,
    "max_score": 1,
    "hits": [
      {
        "_index": "local_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 1,
        "_source": {
```

```
          "user": "domino",
          "message": "This is message 1",
          "likes": 0
        }
      },
      {
        "_index": "cluster_b:b_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 2,
        "_source": {
          "user": "domino",
          "message": "This is message 2",
          "likes": 0
        }
      },
      {
        "_index": "cluster_c:c_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 3,
        "_source": {
          "user": "domino",
          "message": "This is message 3",
          "likes": 0
        }
      }
    ]
  }
}
```

Se você não optou por ignorar clusters indisponíveis na sua configuração de conexão, todos os clusters de destino na sua pesquisa precisam estar disponíveis para que sua solicitação de pesquisa seja executada com êxito. Caso contrário, toda a solicitação falhará — mesmo que um dos domínios não esteja disponível, nenhum resultado da pesquisa será retornado.

OpenSearch Painéis

Você pode visualizar dados de vários domínios conectados da mesma maneira que de um único domínio, exceto que você deve acessar os índices remotos usando `connection-alias:index`. Portanto, o padrão de índice deve corresponder a `connection-alias:index`.

Aprendendo a se classificar no Amazon OpenSearch Service

OpenSearch usa uma estrutura de classificação probabilística chamada BM-25 para calcular as pontuações de relevância. Se uma palavra-chave distintiva aparece com mais frequência em um documento, o BM-25 atribui uma pontuação de relevância maior a esse documento. Esse framework, no entanto, não leva em conta o comportamento do usuário, como dados de cliques, o que pode melhorar ainda mais a relevância.

O Learning to Rank é um plug-in de código aberto que permite que você use machine learning e dados comportamentais para ajustar a relevância de documentos. Ele usa modelos das bibliotecas XGBoost e Ranklib para reavaliar os resultados da pesquisa. O plugin Elasticsearch LTR foi desenvolvido inicialmente pela OpenSource Connections, com contribuições significativas da Fundação Wikimedia, Snagajob Engineering, Bonsai e Yelp Engineering. A OpenSearch versão do plug-in é derivada do plug-in Elasticsearch LTR.

Aprender a classificar requer o Elasticsearch 7.7 OpenSearch ou posterior. Para usar o plug-in Learning to Rank, é necessário ter permissões de administrador completas. Para saber mais, consulte the section called “Modificação do usuário primário”.

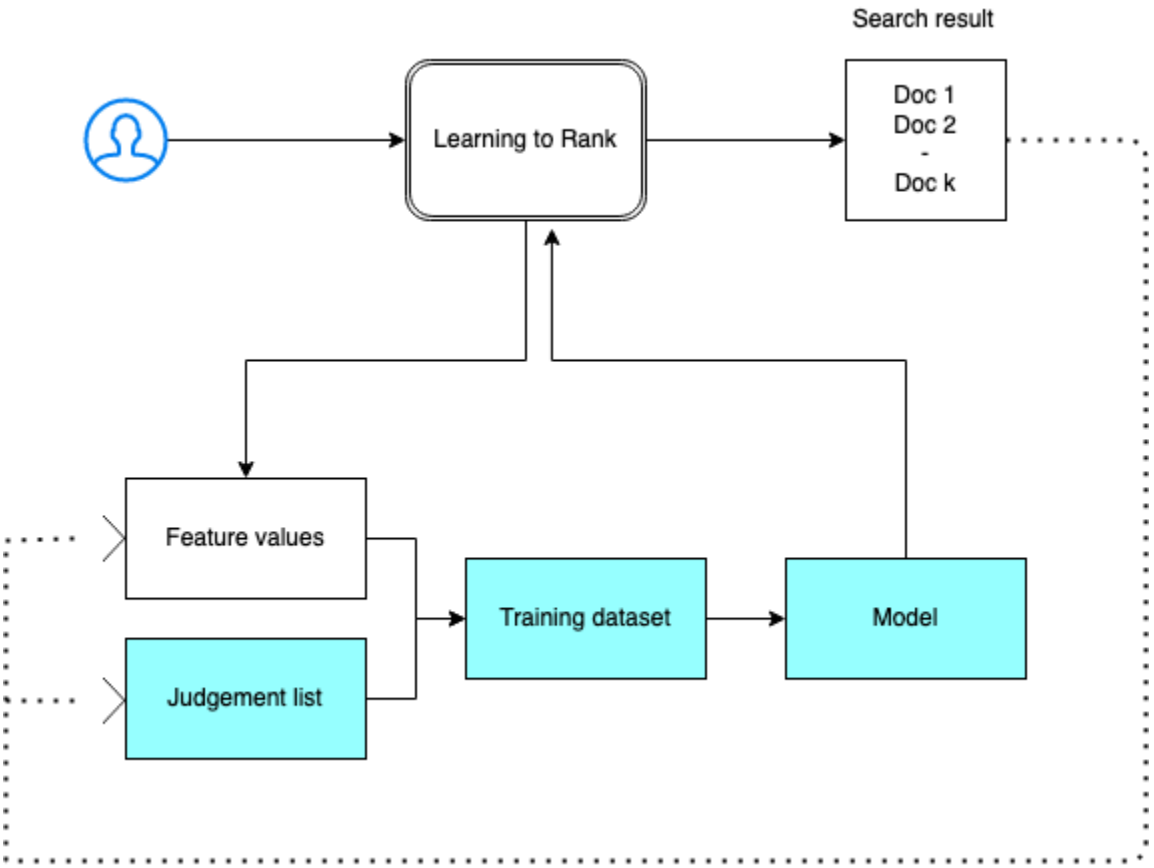
> **Note**
>
> Esta documentação fornece uma visão geral do plug-in Learning to Rank e ajuda você a começar a usá-lo. A documentação completa, incluindo etapas detalhadas e descrições da API, está disponível na documentação do Learning to Rank.

Tópicos

- Conceitos básicos do Learning to Rank
- API do Learning to Rank

Conceitos básicos do Learning to Rank

Você precisa fornecer uma lista de julgamentos, preparar um conjunto de dados de treinamento e treinar o modelo fora do Amazon OpenSearch Service. As peças em azul ocorrem fora do OpenSearch Serviço:



Etapa 1: Inicializar o plug-in

Para inicializar o plug-in Learning to Rank, envie a seguinte solicitação para seu domínio OpenSearch de serviço:

```
PUT _ltr
```

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : ".ltrstore"
}
```

Este comando cria um índice `.ltrstore` oculto que armazena informações de metadados, como conjuntos de recursos e modelos.

Etapa 2: Criar uma lista de julgamento

> **Note**
>
> Você deve executar essa etapa fora do OpenSearch Serviço.

Uma lista de julgamentos é uma coleção de exemplos com os quais um modelo de machine learning aprende. Sua lista de julgamento deve incluir palavras-chave que são importantes para você e um conjunto de documentos classificados para cada palavra-chave.

Neste exemplo, temos uma lista de julgamento para um conjunto de dados de filmes. Um grau 4 indica uma combinação perfeita. Um grau 0 indica a pior correspondência.

Grau	Palavra-chave	ID do documento	Nome do filme
4	rambo	7555	Rambo
3	rambo	1370	Rambo III
3	rambo	1369	Rambo: First Blood Part II
3	rambo	1368	First Blood

Prepare sua lista de julgamento no seguinte formato:

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood

where qid:1 represents "rambo"
```

Para um exemplo mais completo de uma lista de julgamento, consulte Julgamentos de filmes.

Você pode criar essa lista de julgamentos manualmente com a ajuda de anotadores humanos ou inferi-la programaticamente a partir de dados analíticos.

Etapa 3: Construir um conjunto de recursos

Um recurso é um campo que corresponde à relevância de um documento — por exemplo, `title`, `overview`, `popularity score`(número de visualizações) e assim por diante.

Crie um conjunto de recursos com um modelo do Mustache para cada recurso. Para saber mais sobre os recursos, consulte Como trabalhar com recursos.

Neste exemplo, construímos um conjunto de recursos `movie_features` com os campos `title` e `overview`:

```
POST _ltr/_featureset/movie_features
{
  "featureset" : {
      "name" : "movie_features",
      "features" : [
        {
          "name" : "1",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "title" : "{{keywords}}"
            }
          }
        },
        {
          "name" : "2",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "overview" : "{{keywords}}"
            }
          }
        }
      ]
    }
}
```

Se você consultar o índice `.ltrstore` original, obterá seu conjunto de recursos de volta:

```
GET _ltr/_featureset
```

Etapa 4: Registrar os valores dos recursos

Os valores dos recursos são as pontuações de relevância calculadas pelo BM-25 para cada recurso.

Combine o conjunto de recursos e a lista de julgamento para registrar os valores dos recursos. Para saber mais sobre recursos de registro em log do, consulte Pontuações de recursos de registro.

Neste exemplo, a consulta `bool` recupera os documentos classificados com o filtro e, em seguida, seleciona o conjunto de recursos com a consulta `sltr`. A consulta `ltr_log` combina os documentos e os recursos para registrar os valores dos recursos correspondentes:

```
POST tmdb/_search
{
  "_source": {
    "includes": [
      "title",
      "overview"
    ]
  },
  "query": {
    "bool": {
      "filter": [
        {
          "terms": {
            "_id": [
              "7555",
              "1370",
              "1369",
              "1368"
            ]
          }
        },
        {
          "sltr": {
            "_name": "logged_featureset",
            "featureset": "movie_features",
            "params": {
              "keywords": "rambo"
            }
```

```
          }
        }
      ]
    }
  },
  "ext": {
    "ltr_log": {
      "log_specs": {
        "name": "log_entry1",
        "named_query": "logged_featureset"
      }
    }
  }
}
```

Uma resposta de exemplo pode ser a seguinte:

```
{
  "took" : 7,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : 0.0,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1368",
        "_score" : 0.0,
        "_source" : {
          "overview" : "When former Green Beret John Rambo is harassed by local law
 enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
 rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
 sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
```

```
          "title" : "First Blood"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1"
                },
                {
                  "name" : "2",
                  "value" : 10.558305
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 0.0,
        "_source" : {
          "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
          "title" : "Rambo"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1",
                  "value" : 11.2569065
                },
                {
                  "name" : "2",
```

```
                  "value" : 9.936821
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1369",
        "_score" : 0.0,
        "_source" : {
          "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
          "title" : "Rambo: First Blood Part II"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1",
                  "value" : 6.334839
                },
                {
                  "name" : "2",
                  "value" : 10.558305
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      },
      {
        "_index" : "tmdb",
```

```
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 0.0,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
 find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
 his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
 when Trautman is captured.",
          "title" : "Rambo III"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1",
                  "value" : 9.425955
                },
                {
                  "name" : "2",
                  "value" : 11.262714
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      }
    ]
  }
}
```

No exemplo anterior, o primeiro recurso não tem um valor de recurso porque a palavra-chave “rambo” não aparece no campo título do documento com um ID igual a 1368. Este é um valor de recurso ausente nos dados de treinamento.

Etapa 5: Criar um conjunto de dados de treinamento

> **Note**
>
> Você deve executar essa etapa fora do OpenSearch Serviço.

A próxima etapa é combinar a lista de julgamento e os valores de recursos para criar um conjunto de dados de treinamento. Se a lista de julgamento original é semelhante a:

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood
```

Converta-a para o conjunto de dados de treinamento final, que é semelhante a:

```
4 qid:1 1:12.318474 2:10.573917 # 7555 rambo
3 qid:1 1:10.357875 2:11.950391 # 1370 rambo
3 qid:1 1:7.010513 2:11.220095 # 1369 rambo
3 qid:1 1:0.0 2:11.220095 # 1368 rambo
```

Você pode executar esta etapa manualmente ou escrever um programa para automatizá-la.

Etapa 6: Escolher um algoritmo e construir o modelo

> **Note**
>
> Você deve executar essa etapa fora do OpenSearch Serviço.

Com o conjunto de dados de treinamento pronto, a próxima etapa é usar XGBoost nossas bibliotecas Ranklib para criar um modelo. XGBoost e as bibliotecas Ranklib permitem criar modelos populares, como LambdaMart, Random Forests e assim por diante.

Para ver as etapas de uso XGBoost e o Ranklib para criar o modelo, consulte a RankLibdocumentação XGBooste, respectivamente. Para usar a Amazon SageMaker para criar o XGBoost modelo, consulte XGBoostAlgoritmo.

Etapa 7: Implantar o modelo

Depois de criar o modelo, implante-o no plug-in Learning to Rank. Para saber mais sobre como implantar um modelo, consulte Carregar um modelo treinado.

Neste exemplo, construímos um modelo `my_ranklib_model` usando a biblioteca Ranklib:

```
POST _ltr/_featureset/movie_features/_createmodel?pretty
{
  "model": {
    "name": "my_ranklib_model",
    "model": {
      "type": "model/ranklib",
      "definition": """## LambdaMART
## No. of trees = 10
## No. of leaves = 10
## No. of threshold candidates = 256
## Learning rate = 0.1
## Stop early = 100

<ensemble>
   <tree id="1" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-2.0</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-2.0</output>
               </split>
               <split pos="right">
                  <output>-2.0</output>
               </split>
            </split>
         </split>
         <split pos="right">
```

```
            <output>2.0</output>
         </split>
      </split>
   </tree>
   <tree id="2" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.67031991481781</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.67031991481781</output>
               </split>
               <split pos="right">
                  <output>-1.6703200340270996</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.6703201532363892</output>
         </split>
      </split>
   </tree>
   <tree id="3" weight="0.1">
      <split>
         <feature>2</feature>
         <threshold>10.573917</threshold>
         <split pos="left">
            <output>1.479954481124878</output>
         </split>
         <split pos="right">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
```

```
                  <output>-1.4799546003341675</output>
               </split>
               <split pos="right">
                  <output>-1.479954481124878</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.479954481124878</output>
            </split>
         </split>
      </split>
   </tree>
   <tree id="4" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.3569872379302979</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.3569872379302979</output>
               </split>
               <split pos="right">
                  <output>-1.3569872379302979</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.3569873571395874</output>
         </split>
      </split>
   </tree>
   <tree id="5" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
```

```
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.2721362113952637</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.2721363306045532</output>
               </split>
               <split pos="right">
                  <output>-1.2721363306045532</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.2721362113952637</output>
         </split>
      </split>
   </tree>
   <tree id="6" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.2110036611557007</output>
               </split>
               <split pos="right">
                  <output>-1.2110036611557007</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.2110037803649902</output>
            </split>
         </split>
         <split pos="right">
            <output>1.2110037803649902</output>
         </split>
```

```
      </split>
   </tree>
   <tree id="7" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.165616512298584</output>
               </split>
               <split pos="right">
                  <output>-1.165616512298584</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.165616512298584</output>
            </split>
         </split>
         <split pos="right">
            <output>1.165616512298584</output>
         </split>
      </split>
   </tree>
   <tree id="8" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.131177544593811</output>
               </split>
               <split pos="right">
                  <output>-1.131177544593811</output>
               </split>
```

```
            </split>
            <split pos="right">
               <output>-1.131177544593811</output>
            </split>
         </split>
         <split pos="right">
            <output>1.131177544593811</output>
         </split>
      </split>
   </tree>
   <tree id="9" weight="0.1">
      <split>
         <feature>2</feature>
         <threshold>10.573917</threshold>
         <split pos="left">
            <output>1.1046180725097656</output>
         </split>
         <split pos="right">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.1046180725097656</output>
               </split>
               <split pos="right">
                  <output>-1.1046180725097656</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.1046180725097656</output>
            </split>
         </split>
      </split>
   </tree>
   <tree id="10" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
```

```
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.0838804244995117</output>
               </split>
               <split pos="right">
                  <output>-1.0838804244995117</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.0838804244995117</output>
            </split>
         </split>
         <split pos="right">
            <output>1.0838804244995117</output>
         </split>
      </split>
   </tree>
</ensemble>
"""
    }
  }
}
```

Para ver o modelo, envie a seguinte solicitação:

```
GET _ltr/_model/my_ranklib_model
```

Etapa 8: Pesquisar com Learning to Rank

Após a implantação do modelo, você estará pronto para pesquisar.

Execute a consulta `sltr` com os recursos que você está usando e o nome do modelo que deseja executar:

```
POST tmdb/_search
{
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
```

```
      "query": "rambo",
      "fields": ["title", "overview"]
    }
  },
  "rescore": {
    "query": {
      "rescore_query": {
        "sltr": {
          "params": {
            "keywords": "rambo"
          },
          "model": "my_ranklib_model"
        }
      }
    }
  }
}
```

Com o Learning to Rank, você vê “Rambo” como o primeiro resultado porque nós atribuímos a ele a nota mais alta na lista de julgamento:

```
{
  "took" : 12,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 7,
      "relation" : "eq"
    },
    "max_score" : 13.096414,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 13.096414,
        "_source" : {
```

```
        "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
        "title" : "Rambo"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1370",
      "_score" : 11.17245,
      "_source" : {
        "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
        "title" : "Rambo III"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1368",
      "_score" : 10.442155,
      "_source" : {
        "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
        "title" : "First Blood"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1369",
      "_score" : 10.442155,
      "_source" : {
        "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
```

```
          "title" : "Rambo: First Blood Part II"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "31362",
        "_score" : 7.424202,
        "_source" : {
          "overview" : "It is 1985, and a small, tranquil Florida town is being rocked
by a wave of vicious serial murders and bank robberies. Particularly sickening to the
authorities is the gratuitous use of violence by two “Rambo” like killers who dress
themselves in military garb. Based on actual events taken from FBI files, the movie
depicts the Bureau’s efforts to track down these renegades.",
          "title" : "In the Line of Duty: The F.B.I. Murders"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "13258",
        "_score" : 6.43182,
        "_source" : {
          "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
his family's stifling home life when he encounters Lee Carter (Will Poulter), the
school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
to make cinematic history by filming his own action-packed video epic. Together, these
two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
– and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
own!""",
          "title" : "Son of Rambow"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "61410",
        "_score" : 3.9719706,
        "_source" : {
          "overview" : "It's South Africa 1990. Two major events are about to happen:
The release of Nelson Mandela and, more importantly, it's Spud Milton's first year
at an elite boys only private boarding school. John Milton is a boy from an ordinary
background who wins a scholarship to a private school in Kwazulu-Natal, South Africa.
Surrounded by boys with nicknames like Gecko, Rambo, Rain Man and Mad Dog, Spud has
```

```
 his hands full trying to adapt to his new home. Along the way Spud takes his first
 tentative steps along the path to manhood. (The path it seems could be a rather long
 road). Spud is an only child. He is cursed with parents from well beyond the lunatic
 fringe and a senile granny. His dad is a fervent anti-communist who is paranoid that
 the family domestic worker is running a shebeen from her room at the back of the
 family home. His mom is a free spirit and a teenager's worst nightmare, whether it's
 shopping for Spud's underwear in the local supermarket",
          "title" : "Spud"
        }
      }
    ]
  }
}
```

Se você pesquisar sem usar o plug-in Learning to Rank, OpenSearch retornará resultados diferentes:

```
POST tmdb/_search
{
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
      "query": "Rambo",
      "fields": ["title", "overview"]
    }
  }
}
```

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 5,
      "relation" : "eq"
    },
```

```
    "max_score" : 11.262714,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 11.262714,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
          "title" : "Rambo III"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 11.2569065,
        "_source" : {
          "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
          "title" : "Rambo"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1368",
        "_score" : 10.558305,
        "_source" : {
          "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
          "title" : "First Blood"
        }
      },
      {
        "_index" : "tmdb",
```

```
        "_type" : "movie",
        "_id" : "1369",
        "_score" : 10.558305,
        "_source" : {
          "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
 secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
 rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
 life, avenge the death of a woman and bring corrupt officials to justice.",
          "title" : "Rambo: First Blood Part II"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "13258",
        "_score" : 6.4600153,
        "_source" : {
          "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
 his family's stifling home life when he encounters Lee Carter (Will Poulter), the
 school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
 to make cinematic history by filming his own action-packed video epic. Together, these
 two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
 – and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
 own!""",
          "title" : "Son of Rambow"
        }
      }
    ]
  }
}
```

Com base no quanto bem você acha que o modelo está funcionando, ajuste a lista de julgamento e os recursos. Em seguida, repita as etapas 2 a 8 para melhorar os resultados da classificação ao longo do tempo.

API do Learning to Rank

Use as operações do Learning to Rank para trabalhar programaticamente com conjuntos de recursos e modelos.

Criar armazenamento

Cria um índice `.ltrstore` oculto que armazena informações de metadados, como conjuntos de recursos e modelos.

```
PUT _ltr
```

Excluir armazenamento

Exclui o índice `.ltrstore` oculto e redefine o plug-in.

```
DELETE _ltr
```

Criar conjunto de recursos

Cria um conjunto de recursos.

```
POST _ltr/_featureset/<name_of_features>
```

Excluir conjunto de recursos

Exclui um conjunto de recursos.

```
DELETE _ltr/_featureset/<name_of_feature_set>
```

Obter conjunto de recursos

Recupera um conjunto de recursos.

```
GET _ltr/_featureset/<name_of_feature_set>
```

Criar modelo

Cria um modelo.

```
POST _ltr/_featureset/<name_of_feature_set>/_createmodel
```

Excluir modelo

Exclui um modelo.

```
DELETE _ltr/_model/<name_of_model>
```

Obter modelo

Recupera um modelo.

```
GET _ltr/_model/<name_of_model>
```

Obter estatísticas

Fornece informações sobre como o plug-in está se comportando.

```
GET _ltr/_stats
```

Também é possível usar filtros para recuperar uma única estatística:

```
GET _ltr/_stats/<stat>
```

Além disso, é possível limitar as informações a um único nó no cluster:

```
GET _ltr/_stats/<stat>/nodes/<nodeId>

{
  "_nodes" : {
    "total" : 1,
    "successful" : 1,
    "failed" : 0
  },
  "cluster_name" : "873043598401:ltr-77",
  "stores" : {
    ".ltrstore" : {
      "model_count" : 1,
      "featureset_count" : 1,
      "feature_count" : 2,
      "status" : "green"
    }
  },
  "status" : "green",
  "nodes" : {
    "DjelK-_ZSfyzstO5dhGGQA" : {
      "cache" : {
```

```
        "feature" : {
          "eviction_count" : 0,
          "miss_count" : 0,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "featureset" : {
          "eviction_count" : 2,
          "miss_count" : 2,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "model" : {
          "eviction_count" : 2,
          "miss_count" : 3,
          "entry_count" : 1,
          "memory_usage_in_bytes" : 3204,
          "hit_count" : 1
        }
      },
      "request_total_count" : 6,
      "request_error_count" : 0
    }
  }
}
```

As estatísticas são fornecidas em dois níveis, nó e cluster, conforme especificado nas seguintes tabelas:

Estatísticas em nível de nó

Nome do campo	Description
request_total_count	Contagem total de solicitações de classific ação.
request_error_count	Contagem total de solicitações malsucedidas.
cache	Estatísticas em todos os caches (recursos, conjuntos de recursos, modelos). Um acerto de cache ocorre quando um usuário consulta

Nome do campo	Description
	o plug-in e o modelo já está carregado na memória.
cache.eviction_count	Número de remoções de cache.
cache.hit_count	Número de acertos de cache.
cache.miss_count	Número de perdas no cache. Uma perda de cache ocorre quando um usuário consulta o plug-in e o modelo ainda não está carregado na memória.
cache.entry_count	Número de entradas no cache.
cache.memory_usage_in_bytes	Memória total usada em bytes.
cache.cache_capacity_reached	Indica se o limite de cache foi atingido.

Estatísticas em nível de cluster

Nome do campo	Description
stores	Indica onde os conjuntos de recursos e metadados de modelos são armazenados. (O padrão é “.ltrstore”. Caso contrário, é prefixado com “.ltrstore_”, mais um nome fornecido pelo usuário).
stores.status	O status do índice.
stores.feature_sets	Número de conjuntos de recursos.
stores.features_count	Número de recursos.
stores.model_count	Número de modelos.
status	O status do plug-in baseado no status dos índices da feature store (vermelho, amarelo

Nome do campo	Description
	ou verde) e estado do disjuntor (aberto ou fechado).
cache.cache_capacity_reached	Indica se o limite de cache foi atingido.

Get cache stats (Obter estatísticas de cache)

Retorna estatísticas sobre o uso do cache e da memória.

```
GET _ltr/_cachestats

{
    "_nodes": {
        "total": 2,
        "successful": 2,
        "failed": 0
    },
    "cluster_name": "opensearch-cluster",
    "all": {
        "total": {
            "ram": 612,
            "count": 1
        },
        "features": {
            "ram": 0,
            "count": 0
        },
        "featuresets": {
            "ram": 612,
            "count": 1
        },
        "models": {
            "ram": 0,
            "count": 0
        }
    },
    "stores": {
        ".ltrstore": {
            "total": {
                "ram": 612,
```

```
                "count": 1
            },
            "features": {
                "ram": 0,
                "count": 0
            },
            "featuresets": {
                "ram": 612,
                "count": 1
            },
            "models": {
                "ram": 0,
                "count": 0
            }
        }
    },
    "nodes": {
        "ejF6uutERF20wOFNOXB61A": {
            "name": "opensearch1",
            "hostname": "172.18.0.4",
            "stats": {
                "total": {
                    "ram": 612,
                    "count": 1
                },
                "features": {
                    "ram": 0,
                    "count": 0
                },
                "featuresets": {
                    "ram": 612,
                    "count": 1
                },
                "models": {
                    "ram": 0,
                    "count": 0
                }
            }
        },
        "Z2RZNWRLSveVcz2c6lHf5A": {
            "name": "opensearch2",
            "hostname": "172.18.0.2",
            "stats": {
                ...
```

```
            }
        }
    }
}
```

Clear cache (Limpar cache)

Limpa o cache do plug-in. Use esta opção para atualizar o modelo.

```
POST _ltr/_clearcache
```

Pesquisa assíncrona no Amazon Service OpenSearch

Com a pesquisa assíncrona do Amazon OpenSearch Service, você pode enviar uma consulta de pesquisa que é executada em segundo plano, monitorar o progresso da solicitação e recuperar os resultados em um estágio posterior. Você pode recuperar resultados parciais à medida que eles se tornam disponíveis antes da conclusão da pesquisa. Após a conclusão da pesquisa, salve os resultados para recuperação e análise posteriores.

A pesquisa assíncrona requer OpenSearch 1.0 ou posterior, ou Elasticsearch 7.10 ou posterior.

Esta documentação contém uma breve visão geral da pesquisa assíncrona. Ele também discute as limitações do uso da pesquisa assíncrona com um domínio gerenciado do Amazon OpenSearch Service em vez de um cluster de código aberto. OpenSearch Para obter a documentação completa da pesquisa assíncrona, incluindo configurações disponíveis, permissões e uma referência completa da API, consulte Pesquisa assíncrona na documentação. OpenSearch

Exemplo de chamada de pesquisa

Para executar uma pesquisa assíncrona, envie solicitações HTTP ao `_plugins/_asynchronous_search` usando o seguinte formato:

```
POST opensearch-domain/_plugins/_asynchronous_search
```

> **Note**
>
> Se você estiver usando o Elasticsearch 7.10 em vez de uma OpenSearch versão, `_plugins` substitua por `_opendistro` em todas as solicitações de pesquisa assíncrona.

Também é possível especificar as seguintes opções de pesquisa assíncrona:

Opções	Description	Valor padrão	Obrigatório
`wait_for_completion_timeout`	Especifica o tempo que você planeja esperar pelos resultados. Você pode ver os resultados obtidos dentro deste tempo, assim como em uma pesquisa normal. Você pode consultar os resultados restantes com base em um ID. O valor máximo é 300 segundos.	1 segundo	Não
`keep_on_completion`	Especifica se você deseja salvar os resultados no cluster após a conclusão da pesquisa. Você poderá examinar os resultados armazenados mais tarde.	`false`	Não
`keep_alive`	Especifica por quanto tempo o resultado é salvo no cluster. Por exemplo, 2d significa que os resultados são armazenados no cluster por 48 horas. Os resultados da pesquisa salvos serão excluídos após esse período ou se a pesquisa for cancelada. Observe que isso inclui o runtime da consulta. Se a consulta ultrapassar este tempo, o processo cancelará a consulta automaticamente.	12 horas	Não

Exemplo de solicitação

```
POST _plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=1ms&keep_on_completion=true&request_cache=false
{
  "aggs": {
    "city": {
      "terms": {
        "field": "city",
        "size": 10
      }
    }
```

```
  }
}
```

> **Note**
>
> Todos os parâmetros de solicitação aplicáveis a uma consulta `_search` padrão são aceitos. Se você estiver usando o Elasticsearch 7.10 em vez de uma OpenSearch versão, substitua por. `_plugins` `_opendistro`

Permissões da pesquisa assíncrona

A pesquisa assíncrona oferece suporte ao controle de acesso refinado. Para obter detalhes sobre combinação e correspondência de permissões para se adequar ao seu caso de uso, consulte Segurança da pesquisa assíncrona.

Para domínios com controle de acesso refinado habilitado, você precisa das seguintes permissões mínimas para uma função:

```
# Allows users to use all asynchronous search functionality
asynchronous_search_full_access:
  reserved: true
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/*'
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - 'indices:data/read/search*'

# Allows users to read stored asynchronous search results
asynchronous_search_read_access:
  reserved: true
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/get'
```

Para domínios com controle de acesso refinado desabilitado, use o acesso do IAM e a chave secreta para assinar todas as solicitações. Você pode acessar os resultados com o ID da pesquisa assíncrona.

Configurações da pesquisa assíncrona

OpenSearch permite que você altere todas as configurações de pesquisa assíncrona disponíveis usando a API. `_cluster/settings` Em OpenSearch Serviço, você só pode alterar as seguintes configurações:

- `plugins.asynchronous_search.node_concurrent_running_searches`
- `plugins.asynchronous_search.persist_search_failures`

Pesquisa entre clusters

Você pode executar uma pesquisa assíncrona em clusters com as seguintes limitações secundárias:

- Você pode executar uma pesquisa assíncrona somente no domínio de origem.
- Não é possível minimizar round trips de rede como parte de uma consulta de pesquisa entre clusters.

Se você configurar uma conexão entre `domain-a -> domain-b` com alias de conexão `cluster_b` e `domain-a -> domain-c` com alias de conexão `cluster_c`, `domain-a`, pesquise assincronamente `domain-b` e `domain-c` da seguinte forma:

```
POST https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=500ms&keep_on_completion=true&request_cache=false
{
  "size": 0,
  "_source": {
    "excludes": []
  },
  "aggs": {
    "2": {
      "terms": {
        "field": "clientip",
        "size": 50,
        "order": {
          "_count": "desc"
        }
      }
    }
  },
```

```
  "stored_fields": [
    "*"
  ],
  "script_fields": {},
  "docvalue_fields": [
    "@timestamp"
  ],
  "query": {
    "bool": {
      "must": [
        {
          "query_string": {
            "query": "status:404",
            "analyze_wildcard": true,
            "default_field": "*"
          }
        },
        {
          "range": {
            "@timestamp": {
              "gte": 1483747200000,
              "lte": 1488326400000,
              "format": "epoch_millis"
            }
          }
        }
      ],
      "filter": [],
      "should": [],
      "must_not": []
    }
  }
}
```

Resposta

```
{
  "id" :
 "Fm9pYzJyVG91U19xb0hIQUJnMHJfRFEAAAAAAAknghQ1OWVBczNZQjVEa2dMYTBXaTdEagAAAAAAAAAB",
  "state" : "RUNNING",
  "start_time_in_millis" : 1609329314796,
  "expiration_time_in_millis" : 1609761314796
}
```

Para obter mais informações, consulte the section called “Pesquisa entre clusters”.

UltraWarm

As pesquisas assíncronas com UltraWarm índices continuam funcionando. Para obter mais informações, consulte the section called “UltraWarm armazenamento”.

> **Note**
>
> Você pode monitorar estatísticas de pesquisa assíncrona em. CloudWatch Para obter uma lista completa de métricas, consulte the section called “Métricas de pesquisa assíncrona”.

Pesquisa pontual no Amazon OpenSearch Service

Point in Time (PIT) é um tipo de pesquisa que permite executar diversas consultas em um conjunto de dados que é fixo no tempo. Normalmente, quando você executa a mesma consulta no mesmo índice em momentos diferentes, recebe resultados diferentes porque os documentos são constantemente indexados, atualizados e excluídos. Com o PIT, você pode consultar um estado constante do seu conjunto de dados.

O principal uso da pesquisa PIT é combiná-la com a funcionalidade `search_after`. Esse é o método de paginação preferido OpenSearch, especialmente para paginação profunda, porque opera em um conjunto de dados congelado no tempo, não está vinculado a uma consulta e oferece suporte à paginação consistente para frente e para trás. Você pode usar o PIT com um domínio executando a OpenSearch versão 2.5.

> **Note**
>
> Este tópico fornece uma visão geral do PIT e algumas coisas a considerar ao usá-lo em um domínio gerenciado do Amazon OpenSearch Service em vez de em um OpenSearch cluster autogerenciado. Para obter a documentação completa do PIT, incluindo uma referência abrangente da API, consulte Point in Time na OpenSearch documentação de código aberto.

Considerações

Considere o seguinte ao configurar suas pesquisas com o PIT:

- Se você estiver atualizando do domínio executando a OpenSearch versão 2.3 e precisar de um controle de acesso refinado nas ações do PIT, precisará adicionar manualmente essas ações e funções.
- Não há resiliência para o PIT. A reinicialização do nó, o encerramento do nó, as implantações em azul/verde e a reinicialização do OpenSearch processo fazem com que todos os dados do PIT sejam perdidos.
- Se um fragmento for realocado durante a blue/green implantação, somente segmentos de dados ativos serão transferidos para o novo nó. Segmentos de fragmentos mantidos pelo PIT (tanto exclusivos quanto aqueles compartilhados com dados ativos) permanecem no nó antigo.
- Atualmente, as pesquisas com PIT não funcionam com a pesquisa assíncrona.

Criar um PIT

Para executar uma consulta PIT, envie solicitações HTTP para `_search/point_in_time` usando o seguinte formato:

```
POST opensearch-domain/my-index/_search/point_in_time?keep_alive=time
```

Você pode especificar as seguintes opções de PIT:

Opções	Description	Valor padrão	Obrigatório
`keep_alive`	A quantidade de tempo para manutenção do PIT. Toda vez que você acessa um PIT com uma solicitação de pesquisa, a vida útil do PIT é estendida pela quantidade de tempo igual ao parâmetro `keep_alive` . Esse parâmetro de consulta é obrigatório quando você cria um PIT, mas é opcional em uma solicitação de pesquisa.		Sim
`preference`	Uma string que especifica o nó ou o fragmento usado para realizar a pesquisa.	Aleatório	Não

Opções	Description	Valor padrão	Obrigatório
`routing`	Uma string que especifica o roteamento de solicitações de pesquisa para um fragmento específico.	O documento é `_id`	Não
`expand_wildcards`	Uma string que especifica o tipo de índice que pode corresponder ao padrão curinga. É compatível com valores separados por vírgulas. Os valores válidos são os seguintes: • `all`: combine qualquer índice ou fluxo de dados, inclusive os ocultos. • `open`: combine índices abertos e não ocultos ou fluxos de dados não ocultos. • `closed`: combine índices fechados e não ocultos ou fluxos de dados não ocultos. • `hidden`: combine índices ou fluxos de dados ocultos. Deve ser combinado com aberto, fechado ou aberto e fechado. • `none`: nenhum padrão curinga é aceito.	`open`	Não
`allow_partial_pit_creation`	Um booleano que especifica se um PIT deve ser criado com falhas parciais.	`true`	Não

Exemplo de resposta

```
{
    "pit_id":
 "o463QQEPbXktaW5kZXgtMDAwMDAxFnNOWU43ckt3U3IyaFVpbGE1UWEtMncAFjFyeXBsRGJmVFM2RTB6eVg1aVVqQncAA
    "_shards": {
        "total": 1,
        "successful": 1,
        "skipped": 0,
        "failed": 0
    },
    "creation_time": 1658146050064
```

```
}
```

Ao criar um PIT, você recebe um PIT ID na resposta. Esse é o ID que você usa para realizar pesquisas com o PIT.

Permissões pontuais

O PIT é compatível com o controle de acesso detalhado. Se você estiver atualizando para um domínio da OpenSearch versão 2.5 e precisar de um controle de acesso refinado, precisará criar funções manualmente com as seguintes permissões:

```
# Allows users to use all point in time search search functionality
point_in_time_full_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
        - "indices:data/read/point_in_time/readall"
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"


# Allows users to use point in time search search functionality for specific index
# All type operations like list all PITs, delete all PITs are not supported in this
 case

point_in_time_index_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - 'my-index-1'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"
```

Para domínios com a OpenSearch versão 2.5 e superior, você pode usar a `point_in_time_full_access` função integrada. Para obter mais informações, consulte Modelo de segurança na OpenSearch documentação.

Configurações do PIT

OpenSearch permite que você altere todas as configurações de PIT disponíveis usando a `_cluster/settings` API. No momento, no OpenSearch Serviço, você não pode modificar as configurações.

Pesquisa entre clusters

Você pode criar PITs, pesquisar com PIT IDs PITs, listar e excluir PITs em clusters com as seguintes pequenas limitações:

- Você pode listar tudo e excluir tudo PITs somente no domínio de origem.
- Não é possível minimizar round trips de rede como parte de uma consulta de pesquisa entre clusters.

Para obter mais informações, consulte the section called “Pesquisa entre clusters”.

UltraWarm

As pesquisas PIT com UltraWarm índices continuam funcionando. Para obter mais informações, consulte the section called “UltraWarm armazenamento”.

> **Note**
>
> Você pode monitorar as estatísticas de pesquisa do PIT em CloudWatch. Para obter uma lista completa de métricas, consulte the section called “Métricas pontuais”.

Pesquisa de agentes no Amazon Service OpenSearch

A partir da OpenSearch versão 3.3, a busca por agentes permite um processo baseado em IA que usa um agente autônomo para executar pesquisas complexas em seu nome.

O Agentic Search apresenta um sistema de agente inteligente que entende a intenção do usuário, organiza as ferramentas certas e gera consultas otimizadas. Ele fornece resumos transparentes de suas decisões por meio de uma interface de linguagem natural.

Com o OpenSearch Serviço, você pode configurar conectores de IA para Serviços da AWS serviços externos. Usando o console, você também pode criar um modelo de ML com um CloudFormation modelo que pode ser usado para criar seu agente. Para obter mais informações, consulte Configurando o Agentic Search com Bedrock Claude.

Para obter a documentação e step-by-step a implementação completas, consulte Agentic Search na OpenSearch documentação.

Pesquisa semântica no Amazon Service OpenSearch

A partir da versão 2.19, você pode usar o enriquecimento semântico automático para implementar a pesquisa semântica com o mínimo esforço de configuração. Esse recurso gera automaticamente uma codificação esparsa para seus campos de texto, eliminando a necessidade de configuração manual do pipeline de ingestão. Para obter detalhes, consulte Enriquecimento semântico automático na OpenSearch documentação.

A partir da OpenSearch versão 2.9, você pode usar a pesquisa semântica para ajudá-lo a entender as consultas de pesquisa e melhorar a relevância da pesquisa. Você pode usar a pesquisa semântica de duas maneiras: com a pesquisa neural e com a pesquisa k-Nearest Neighbor (k-NN).

Com o OpenSearch Serviço, você pode configurar conectores de IA para Serviços da AWS serviços externos. Usando o console, você também pode criar um modelo de ML com um CloudFormation modelo. Para obter mais informações, consulte the section called “CloudFormation integrações de modelos”.

Para obter a documentação completa da pesquisa semântica, incluindo um step-by-step guia para usar a pesquisa semântica, consulte Pesquisa semântica na documentação de código aberto. OpenSearch

Pesquisa simultânea de segmentos no Amazon Service OpenSearch

A partir da OpenSearch versão 2.17, a pesquisa simultânea de segmentos usa uma nova configuração para controlar o comportamento da pesquisa simultânea.

- Os novos domínios criados com a versão 2.17 têm a pesquisa simultânea de segmentos padrão definida no modo automático por padrão em nós 2xl ou acima.
- Os domínios existentes que estão sendo atualizados para a versão 2.17 terão a pesquisa simultânea de segmentos padrão definida como automática com base no tipo de instância para todos os nós 2xl ou acima, e se a utilização de CPU do cluster em geral for abaixo de 45% na última semana.
- Para saber mais, consulte Pesquisa simultânea de segmentos versão 2.17.

A partir da OpenSearch versão 2.13, você pode usar a pesquisa simultânea de segmentos para ajudá-lo a pesquisar segmentos em paralelo durante a fase de consulta. Para obter a documentação completa da pesquisa simultânea de segmentos, consulte Pesquisa simultânea de segmentos na documentação de código OpenSearch aberto. Para obter informações sobre CloudWatch métricas da Amazon relacionadas à pesquisa simultânea de segmentos, consulte Métricas e UltraWarm métricas de instância.

Há algumas limitações adicionais que se aplicam quando você usa a pesquisa de segmentos atual com o Amazon OpenSearch Service:

- Por padrão, o OpenSearch Serviço usa uma contagem de 2 fatias com o mecanismo de contagem máxima de fatias.

Geração de consultas em linguagem natural no Amazon OpenSearch Service

O recurso de geração de consultas em linguagem natural no Amazon OpenSearch Service permite que você consulte seus dados de log operacionais e de segurança por meio de linguagem natural. OpenSearch é uma opção ideal para explorar dados de registro porque é um mecanismo de pesquisa e análise de registros altamente escalável e eficiente, e agora você pode usar linguagem natural para explorar esses registros. Esse recurso permite identificar problemas sem depender da OpenSearch Piped Processing Language (PPL) ou ter que pesquisar definições de dados ao criar suas consultas. Você pode usar o recurso de geração de consultas de linguagem natural em domínios OpenSearch de serviço com a versão 2.13 e posterior. O controle de acesso refinado deve estar habilitado.

Esse recurso foi criado com o OpenSearch Assistant Toolkit. Se quiser criar atributos semelhantes que se conectem aos seus modelos grandes de linguagem, você poderá usar o kit de ferramentas para configurar seus próprios atendentes e ferramentas.

Pré-requisitos

Antes de usar o atributo de geração de consultas em linguagem natural, o domínio deve ter o seguinte:

- Versão 2.13 ou posterior.
- Software de serviço R20240520-P4 ou superior.
- Controle de acesso refinado habilitado. Para saber mais, consulte the section called “Habilitar o controle de acesso detalhado”.

Introdução

A geração de consultas em linguagem natural é habilitada por padrão em todos os domínios criados com a versão 2.13 ou posterior que tenha controle de acesso refinado habilitado.

Para outros domínios, habilite-a selecionando Habilitar os recursos de geração de consultas em linguagem natural e o Amazon Q Developer.

Depois de habilitá-lo, navegue até a página de registros em OpenSearch painéis. Escolha Explorador de eventos e faça uma pergunta com o assistente de consulta.

Configurar permissões do

Se você habilitar a geração de consultas de linguagem natural em um domínio de OpenSearch serviço preexistente, a função query_assistant_access pode não estar definida no domínio. Os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices warm usando o controle de aceso detalhado. Para criar manualmente o perfil query_assistant_access, realize as seguintes etapas:

1. Em OpenSearch Painéis, acesse Segurança e escolha Funções.
2. Escolha Criar perfil e configure as seguintes permissões de cluster:
   - `cluster:admin/opensearch/ml/config/get`
   - `cluster:admin/opensearch/ml/execute`

- `cluster:admin/opensearch/ml/predict`
- `cluster:admin/opensearch/ppl`

3. Nomeie o perfil query_assistant_access.
4. Selecione Criar perfil. O perfil query_assistant_access agora está disponível.

> **Note**
>
> Você também deve ter as permissões de índice `indices:admin/mappings/get` e `read` para os índices com os quais deseja usar perguntas em linguagem natural.

Automação de configuração

O Flow Framework é um OpenSearch plug-in que fornece uma maneira de automatizar OpenSearch configurações para casos de uso, como geração de consultas e bate-papo conversacional. Como o plug-in rastreia os recursos que habilitam o atributo de geração de consultas em linguagem natural, o índice da estrutura de fluxo armazena um modelo para cada domínio que usa o assistente de consulta.

O Flow Framework permite que você selecione um conjunto de modelos predefinidos ou crie suas próprias automações para conectores, ferramentas, agentes e outros componentes de aprendizado de máquina que se preparam OpenSearch como back-end para modelos generativos.

Pesquisa vetorial

A pesquisa vetorial no Amazon OpenSearch Service permite que você pesquise conteúdo semanticamente semelhante usando incorporações de aprendizado de máquina em vez da correspondência tradicional de palavras-chave. A pesquisa vetorial converte os dados (texto, imagens, áudio etc.) em vetores numéricos de alta dimensão (incorporações) que capturam o significado semântico do conteúdo. Ao realizar uma pesquisa, OpenSearch compara a representação vetorial da sua consulta com os vetores armazenados para encontrar os itens mais semelhantes.

A pesquisa vetorial inclui os seguintes componentes principais.

Campos vetoriais

OpenSearch suporta o tipo de `knn_vector` campo para armazenar vetores densos com dimensões configuráveis (até 16.000).

Métodos de pesquisa

- k-NN (vizinhos mais próximos de k): encontra os k vetores mais semelhantes
- k-NN aproximado: usa algoritmos como o HNSW (Hierarchical Navigable Small World) para pesquisas mais rápidas em grandes conjuntos de dados

Métricas de distância

É compatível com vários cálculos de similaridade, incluindo:

- Distância euclidiana
- Similaridade de cossenos
- Produto escalar

Casos de uso comuns

A pesquisa vetorial aceita os seguintes casos de uso comuns.

- Pesquisa semântica: encontrar documentos com significado semelhante, não apenas palavras-chave correspondentes
- Sistemas de recomendação: sugerir produtos, conteúdos ou usuários semelhantes
- Pesquisa de imagens: encontrar imagens visualmente semelhantes

- Detecção de anomalias: identificar valores discrepantes em padrões de dados
- RAG (Retrieval Augmented Generation): aprimorar as respostas do LLM com contexto relevante

Integração com machine learning

OpenSearch se integra aos seguintes serviços e modelos de aprendizado de máquina:

- Amazon Bedrock: para gerar incorporações usando modelos básicos
- Amazon SageMaker AI: para implantação de modelo de ML personalizado
- Modelos Hugging Face: modelos de incorporação pré-treinados
- Modelos personalizados: seus próprios modelos de incorporação treinados

Com a pesquisa vetorial, você pode criar aplicativos sofisticados baseados em IA que entendem o contexto e o significado, indo muito além dos recursos tradicionais de correspondência de texto.

Importação dos vetores do Amazon S3 para o Serverless OpenSearch

O Amazon S3 Vectors disponibiliza o primeiro armazenamento de objetos em nuvem com compatibilidade nativa com armazenamento e consulta de vetores. O S3 Vectors oferece armazenamento vetorial econômico, elástico e durável que pode ser consultado com base em significado semântico e similaridade. Ele oferece tempos de resposta a consulta de menos de um segundo e custos até 90% mais baixos para carregar, armazenar e consultar vetores.

O Amazon S3 Vectors introduz buckets vetoriais do S3, que você pode usar para armazenar, acessar e consultar dados vetoriais sem provisionar nenhuma infraestrutura. Dentro de um bucket vetorial, você pode organizar os dados vetoriais dentro de índice vetoriais. O bucket vetorial pode ter vários índices vetoriais, e cada índice pode conter milhões de vetores. Para saber mais, consulte Trabalhar com o Amazon S3 Vectors e buckets de vetores no Guia do usuário do Amazon S3.

Cada vetor consiste em:

- Uma chave exclusiva
- Dados vetoriais
- Metadados opcionais no formato JSON

Os índices vetoriais são compatíveis com as funções de distância euclidiana e cosseno para operações de pesquisa de similaridade.

> **Note**
>
> A principal vantagem dos buckets vetoriais é a capacidade de armazenar grandes conjuntos de dados a um custo extremamente baixo e, ao mesmo tempo, oferecer acesso direto à API para operações vetoriais.

Para saber mais sobre os buckets do Amazon S3 Vectors, consulte Trabalhar com o S3 Vectors e buckets de vetores no Guia do usuário do Amazon S3. Para obter mais informações sobre a integração com o OpenSearch Service além do que está descrito neste tópico, consulte Usando vetores do S3 com o Service OpenSearch

Você pode usar o S3 Vectors com o Amazon OpenSearch Service para reduzir o custo do armazenamento vetorial quando as consultas são menos frequentes e, em seguida, mover rapidamente esses conjuntos de dados à medida que as demandas aumentam ou para OpenSearch aprimorar os recursos de pesquisa.

OpenSearch O serviço se integra aos vetores do Amazon S3 para fornecer desempenho e funcionalidade aprimorados, além do que os buckets vetoriais do Amazon S3 oferecem sozinhos. Considere essa integração quando precisar de:

- Maior throughput de queries
- Latência de pesquisa abaixo de um segundo
- Recursos avançados de analytics como agregações
- Pesquisa híbrida combinando texto e dados vetoriais

Essa integração é particularmente útil quando várias aplicações consomem os mesmos dados vetoriais com requisitos de performance diferentes. Você pode fazer com que alguns aplicativos interajam diretamente com os buckets vetoriais do Amazon S3 para casos de uso econômicos, enquanto outros aproveitam a OpenSearch integração para operações de desempenho crítico.

Arquiteturas de integração

A integração usa o Amazon OpenSearch Ingestion (OSI) como o pipeline de dados entre os índices vetoriais do Amazon S3 e as coleções vetoriais do Amazon OpenSearch Serverless. OpenSearch A

ingestão exporta automaticamente os dados vetoriais do seu índice vetorial especificado e os ingere em coleções vetoriais OpenSearch sem servidor para operações de pesquisa de alto desempenho.

> **Note**
>
> Após a exportação, os dados ainda estarão presentes no índice vetorial do S3. Você tem duas cópias dos dados.

Cada índice vetorial é mapeado para um índice correspondente na coleção OpenSearch Serverless. A integração:

- Preserva as dimensões vetoriais
- Retém metadados
- Otimiza a estrutura de dados para os OpenSearch recursos de pesquisa vetorial

Após a configuração, o OpenSearch Inestion inicia o processo de exportação de dados consumindo vetores do índice vetorial especificado usando a API do Amazon ListVectors S3. O serviço processa vetores em paralelo para otimizar a velocidade de ingestão, respeitando os limites de escalabilidade do Ingestion e do Amazon Serverless OpenSearch . OpenSearch

Durante a ingestão, o serviço:

- Transforma os dados vetoriais para corresponder ao formato esperado para OpenSearch o Serviço
- Preserva informações essenciais, incluindo valores vetoriais, metadados e métricas de distância
- Lida com cenários de falha por meio de mecanismos inteligentes de repetição
- Coloca registros problemáticos em um bucket do Amazon S3 usado como fila de mensagens não entregues para análise posterior

A integração lida com grandes conjuntos de dados de forma eficiente, sendo que a performance depende das dimensões vetoriais, do tamanho do conjunto de dados e dos limites de escalabilidade configurados. O OSI pode escalar até 16 trabalhadores por pipeline, enquanto o OpenSearch Serverless ajusta automaticamente a capacidade com base nas demandas de ingestão. Por padrão, OpenSearch aumenta a Unidade `maxSearch` OpenSearch Computacional (OCU) no lado OpenSearch sem servidor para 100.

> **Note**
>
> A integração prioriza a eficiência de custos por meio de:
>
> - Desligamento automático do pipeline após a conclusão da exportação
> - OpenSearch Dimensionamento de coleções sem servidor
> - Pay-per-use modelo de recursos

Permissões obrigatórias do IAM

A integração exige uma configuração cuidadosa das permissões do IAM para permitir a comunicação segura entre os serviços. OpenSearch A ingestão precisa de permissões para ler os índices vetoriais do Amazon S3, gravar OpenSearch nas coleções de vetores do Service e gerenciar as políticas de segurança associadas.

Ao habilitar a integração usando o procedimento apresentado posteriormente neste tópico, você pode escolher uma das seguintes opções para gerenciamento de permissões:

- Permitir que o sistema crie automaticamente um perfil de serviço com as permissões necessárias
- Fornecer um perfil existente que atenda aos requisitos

O perfil criado automaticamente inclui políticas para:

- Acessando o índice vetorial do Amazon S3 APIs
- Gerenciando operações OpenSearch de coleta de serviços
- Lidar com operações de fila de mensagens não entregues para tentativas de ingestão malsucedidas

Se você optar por especificar um perfil existente, verifique se ele tem as seguintes permissões do IAM:

(Obrigatório): permissões de pipeline de dados entre OpenSearch Ingestion e OpenSearch Serverless

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowAPIs",
            "Effect": "Allow",
            "Action": [ "aoss:APIAccessAll", "aoss:BatchGetCollection" ],
            "Resource": [ "arn:aws:aoss:*:111122223333:collection/collection-
id" ]
        },
        {
            "Sid": "allowSecurityPolicy",
            "Effect": "Allow",
            "Action": [
                "aoss:CreateSecurityPolicy",
                "aoss:UpdateSecurityPolicy",
                "aoss:GetSecurityPolicy"
            ],
            "Resource": "*",
            "Condition":{
               "StringLike":{
                  "aoss:collection": [ "collection-name" ]
                },
               "StringEquals": {
                  "aws:ResourceAccount": [ "111122223333" ]
               }
            }
        }
    ]
}
```

(Obrigatório): Permissões de ingestão de dados entre a OpenSearch ingestão e a fila de mensagens mortas do Amazon S3

JSON

```
{
    "Version":"2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "s3Access",
            "Effect": "Allow",
            "Action": [
              "s3:PutObject"
            ],
            "Resource": [ "arn:aws:s3:::bucket/*" ]
        }
    ]
}
```

(Obrigatório): Permissões de ingestão de dados entre a OpenSearch ingestão e os vetores do Amazon S3

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowS3VectorIndexAccess",
            "Effect": "Allow",
            "Action": [
               "s3vectors:ListVectors",
               "s3vectors:GetVectors"
            ],
            "Resource": [
                "arn:aws:s3vectors:us-east-1:111122223333:bucket/bucket-name/
index/index-name"
            ]
        }
    ]
}
```

(Obrigatório se a AWS KMS criptografia estiver habilitada): Permissões de decodificação para comunicação entre a OpenSearch ingestão e os vetores do Amazon S3

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "allowS3VectorDecryptionOfCustomManagedKey",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [ "arn:aws:kms:us-east-1:111122223333:key/key-id" ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3vectors.us-east-1.amazonaws.com",
                    "kms:EncryptionContext:aws:s3vectors:arn": [
                        "arn:aws:s3vectors:us-east-1:111122223333:bucket/example-
bucket",
                        "arn:aws:s3vectors:us-east-1:111122223333:bucket/example-
bucket/index/example-index"
                        ]
                 }
             }
        }
    ]
}
```

Configurando a integração do Amazon S3 Vectors com OpenSearch

Use o procedimento a seguir para configurar a integração do Amazon S3 Vectors com o Serverless. OpenSearch

> **Note**
>
> Se você iniciou o processo de configuração da integração a partir do console do Amazon S3 escolhendo a opção Exportar OpenSearch para na página Vector buckets, algumas das etapas do procedimento a seguir não são aplicáveis, conforme observado no procedimento.

Para configurar a integração do Amazon S3 Vectors com o Serverless OpenSearch

1. Abra a página Importar índice vetorial do S3 para mecanismo OpenSearch vetorial no console do Amazon OpenSearch Service. A página é exibida automaticamente se você clicar em Exportar para OpenSearch no console do Amazon S3. Se você estiver iniciando no OpenSearch console, escolha Integração no painel de navegação à esquerda e escolha Importar índice vetorial do S3.
2. Na seção Origem, se você começou no console do Amazon S3, verifique se o nome do índice vetorial e seu nome do recurso da Amazon (ARN) já estão especificados. Se você começou no OpenSearch console, insira o ARN do índice no campo ARN do índice vetorial S3.
3. Na seção Adicionar contas, escolha uma opção. Se você escolher um perfil existente, verifique se ele tem todas as permissões necessárias para integração, conforme descrito em Permissões obrigatórias do IAM.
4. (Opcional) Expanda Additional settings. Para Habilitar redundância (réplicas ativas), recomendamos deixar essa opção selecionada para ambientes de produção. Quando você cria sua primeira coleção, o OpenSearch Serverless instancia duas OCUs — uma para indexação e outra para pesquisa. Para garantir alta disponibilidade, ele também lança um conjunto de nós em espera em outra zona de disponibilidade. Para fins de desenvolvimento e teste, você pode desativar a configuração Ativar redundância para uma coleção, que elimina as duas réplicas em espera e instancia apenas duas. OCUs Por padrão, as réplicas ativas redundantes estão habilitadas, o que significa que um total de quatro OCUs são instanciadas para a primeira coleção em uma conta.

   Em Adicionar AWS KMS chave gerenciada pelo cliente para o vetor Amazon OpenSearch Serverless, escolha essa opção para criptografar dados na coleção vetorial usando uma chave gerenciada pelo cliente. Por padrão, OpenSearch usa um Chave gerenciada pela AWS.
5. Se você iniciou esse processo clicando na OpenSearch opção Exportar para no console do Amazon S3, a seção Detalhes da exportação lista as etapas OpenSearch a seguir. Quando estiver pronto, selecione Exportar.

   Se você iniciou esse processo no console de OpenSearch serviço, a seção Detalhes da importação lista as etapas a serem OpenSearch seguidas a seguir. Quando estiver pronto, selecione Importar.

   OpenSearch abre a página de histórico para exibir todos os índices vetoriais exports/imports do Amazon S3 em índices sem servidor. OpenSearch

Após a ingestão bem-sucedida, o OSI interrompe automaticamente o pipeline para evitar custos desnecessários e, ao mesmo tempo, manter os dados exportados. OpenSearch Você pode monitorar o progresso da integração por meio de CloudWatch métricas e acessar registros detalhados para solucionar problemas.

A OpenSearch coleção permanece ativa e disponível para consultas após a conclusão da ingestão inicial. Você pode realizar:

- Pesquisas de similaridade
- Agregações
- Operações de analytics

Recursos avançados de pesquisa com um mecanismo vetorial Amazon S3

O Amazon OpenSearch Service oferece a capacidade de usar o Amazon S3 como um mecanismo vetorial para índices vetoriais. Esse atributo permite descarregar dados vetoriais para o Amazon S3 enquanto mantém recursos de pesquisa vetorial em fração de segundos por um preço baixo.

Com esse recurso, OpenSearch armazena incorporações vetoriais em um índice vetorial do Amazon S3 enquanto mantém outros campos do documento no armazenamento OpenSearch do cluster. Essa arquitetura oferece os seguintes benefícios:

- Durabilidade: os dados gravados no S3 Vectors são armazenados no S3, que foi projetado para oferecer 99,999999999% de durabilidade de dados.
- Escalabilidade: descarregue grandes conjuntos de dados vetoriais para o S3 sem consumir armazenamento em cluster.
- Custo-benefício: otimize os custos de armazenamento para workloads com muitos vetores.

OpenSearch tem os seguintes requisitos para usar índices vetoriais S3:

- OpenSearch versão 2.19 ou posterior
- OpenSearch Instâncias otimizadas
- Versão de patch mais recente para seu OpenSearch lançamento

Habilitar vetores do S3

Ao criar um novo domínio ou atualizar um domínio existente, você pode escolher a opção Habilitar o S3 Vectors como mecanismo na seção Recursos avançados. Essa configuração permite OpenSearch criar um bucket vetorial S3 quando você utiliza os vetores S3 como seu mecanismo. Quando você ativa essa opção, OpenSearch configura os vetores do S3 para seu domínio da seguinte forma:

1. Criando duas novas concessões na AWS KMS chave configurada com seu domínio:
   - Uma concessão para os trabalhos de indexação em segundo plano do S3 Vectors com privilégios de descriptografia
   - Uma concessão OpenSearch para criar buckets de vetores S3 com permissões `GenerateDataKey`
2. Configurando a chave KMS usada pelo seu OpenSearch domínio como CMK para criptografia em repouso de todos os dados de índice vetorial.

Criar índices com o mecanismo vetorial do S3

Depois de configurar um domínio, você pode criar um ou mais índices k-NN com campos usando o `s3vector` como mecanismo vetorial de backend nos mapeamentos de índices. Você pode configurar diferentes campos vetoriais com diferentes tipos de mecanismos com base no seu caso de uso.

> **Important**
>
> Você só pode usar o mecanismo `s3vector` no mapeamento de uma definição de campo durante a criação do índice. Não é possível adicionar ou atualizar o mapeamento com o mecanismo `s3vector` após a criação do índice.

Aqui estão alguns exemplos que criam índices de mecanismo vetorial do S3.

Exemplo: criação de um índice k-NN com o mecanismo vetorial do S3

```
PUT my-first-s3vector-index
{
  "settings": {
    "index": {
```

```
      "knn": true
    }
  },
  "mappings": {
    "properties": {
        "my_vector_1": {
          "type": "knn_vector",
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
          "type": "float"
        }
    }
  }
}
```

Exemplo: Criação de um índice k-NN com os mecanismos vetoriais e FAISS do S3

Este exemplo destaca o fato de que é possível usar vários mecanismos vetoriais dentro do mesmo índice.

```
PUT my-vector-index
{
  "settings": {
    "index": {
      "knn": true
    }
  },
  "mappings": {
    "properties": {
        "my_vector_1": {
          "type": "knn_vector",
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
```

```
          "type": "float"
        },
        "my_vector_2": {
            "type": "knn_vector",
            "dimension": 2,
            "space_type": "cosine",
            "method": {
                "name": "hnsw",
                "engine": "faiss",
                "parameters": {
                    "ef_construction": 128,
                    "m": 24
                }
            }
        }
    }
  }
}
```

Exemplo do que não é compatível: adição do mecanismo vetorial do S3 após a criação do índice

A abordagem a seguir não é compatível e não funcionará.

```
PUT my-first-s3vector-index
{
  "settings": {
    "index": {
      "knn": true
    }
  }
}

PUT my-first-s3vector-index/_mapping
{
  "properties": {
        "my_vector_1": {
          "type": "knn_vector",
          "dimension": 2,
          "space_type": "l2",
          "method": {
            "engine": "s3vector"
          }
        },
        "price": {
```

```
          "type": "float"
        }
    }
}
```

Limitações de funcionalidade

Considere as seguintes limitações antes de usar o mecanismo `s3vector` em um índice:

Características e comportamentos não compatíveis com o mecanismo s3vector

Recurso	Comportamento
Índice do Split/Shrink/Clone	Eles APIs falham quando usados com um índice configurado com o `s3vector` mecanismo em `knn_vector` campo.
Snapshots	Os índices que usam o mecanismo `s3vector` não são compatíveis com snapshots. Para domínios gerenciados: • Os snapshots automatizados incluem apenas os índices que não usam o mecanismo `s3vector`. • Ocorre falha nas solicitações manuais de snapshots para índices do `s3vector`. **Note** Embora os snapshots não sejam compatíveis com a point-in-time recuperação, o `s3vector` engine, junto com as instâncias OpenSearch otimizadas, oferecem 11 nove de durabilidade.
UltraWarm nível	Os índices configurados com o `s3vector` mecanismo não podem migrar para UltraWarm o nível.
Replicação entre clusters	Os índices configurados com o mecanismo `s3vector` não são compatíveis com replicação entre clusters.
Proteção contra exclusão acidental	Como os snapshots não são compatíveis com índices que usam o mecanismo `s3vector`, a proteção contra exclusão acidental

Recurso	Comportamento
	não está disponível. Você ainda pode restaurar outros índices no domínio.
Pesquisa radial	As consultas com pesquisa radial não são compatíveis em campos que usam o mecanismo `s3vector`.

Indexar documentos

Depois de criar um índice com o mecanismo vetorial S3, você pode ingerir documentos usando a API padrão`_bulk`. OpenSearch descarrega automaticamente os dados vetoriais dos `knn_vector` campos usando o `s3vector` mecanismo para o índice vetorial S3 em tempo real. Os dados pertencentes a outros `knn_vector` campos ou campos que usam mecanismos diferentes serão mantidos OpenSearch em sua própria camada de armazenamento.

Para todas as solicitações em massa que são reconhecidas, OpenSearch garante que todos os dados (vetoriais e não vetoriais) sejam duráveis. Se uma solicitação receber uma confirmação negativa, não há garantia de durabilidade dos documentos dessa solicitação em massa. Você deve repetir essas solicitações, de preferência depois de excluir a solicitação anterior que falhou, usando o ID do documento para evitar documentos duplicados nesses casos raros.

Exemplo de indexação em massa

```
POST _bulk
{ "index": { "_index": "my-first-s3vector-index", "_id": "1" } }
{ "my_vector_1": [1.5, 2.5], "price": 12.2 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "2" } }
{ "my_vector_1": [2.5, 3.5], "price": 7.1 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "3" } }
{ "my_vector_1": [3.5, 4.5], "price": 12.9 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "4" } }
{ "my_vector_1": [5.5, 6.5], "price": 1.2 }
{ "index": { "_index": "my-first-s3vector-index", "_id": "5" } }
{ "my_vector_1": [4.5, 5.5], "price": 3.7 }
```

Pesquisar documentos

Você pode pesquisar seu índice usando a API `_search` padrão para executar consultas de texto, k-NN ou híbridas. Para consultas em `knn_vector` campos configurados com o `s3vector` mecanismo, descarrega OpenSearch automaticamente a consulta para o índice de vetores S3 correspondente.

> **Note**
>
> Com o `s3vector` motor, as consultas de pesquisa k-NN suportam um k valor máximo de 100. Isso significa que no máximo 100 vizinhos mais próximos podem ser retornados nos resultados da pesquisa.

Exemplo de consulta de pesquisa

```
GET my-first-s3vector-index/_search
{
  "size": 2,
  "query": {
    "knn": {
      "my_vector_1": {
        "vector": [2.5, 3.5],
        "k": 2
      }
    }
  }
}
```

Você pode executar uma pesquisa vetorial filtrada no índice OpenSearch kNN usando o mecanismo s3vector. OpenSearch aplica o filtro como pós-filtro e usa o mecanismo de sobreamostragem com base em certas heurísticas para equilibrar recuperação versus latência.

Exemplo de consulta de pesquisa com filtro:

```
GET my-index/_search
{
  "size": 10,
  "query": {
    "knn": {
```

```
      "my_vector_field": {
        "vector": [2.5, 3.5, 1.2, 4.8],
        "k": 10,
        "filter": {
          "range": {
            "price": {
              "gte": 10,
              "lte": 100
            }
          }
        }
      }
    }
  }
}
```

Parâmetros de mapeamento compatíveis

Com o mecanismo `s3vector`, o campo `knn_vector` é compatível com os seguintes parâmetros nos mapeamentos.

Parâmetros de campo vetorial

Parâmetro	Obrigatório	Description	Valores com suporte
`type`	Sim	O tipo de campo presente no documento.	`knn_vector`
`dimension`	Sim	A dimensão de cada vetor que será ingerido no índice.	>0, <=4096
`space_type`	Não	O espaço vetorial usado para calcular a distância entre vetores.	`l2`, `cosinesimil`
`method.engine`	Sim	O mecanismo de k-NN aproximado a ser usado para indexação e pesquisa.	`s3vector`
`method.name`	Não	O método de vizinho mais próximo	""

Parâmetro	Obrigatório	Description	Valores com suporte
`store`	N/D	Habilitar ou desabilitar esse parâmetro de mapeamento não é operacional, pois os dados knn_vector não são armazenados em. OpenSearch	Não suportado
`doc_values`	N/D	Habilitar ou desabilitar esse parâmetro de mapeamento não é operacional, pois os dados knn_vector não são armazenados em. OpenSearch	Não suportado

> **Important**
>
> Os tipos de campo `knn_vector` aninhados não são compatíveis com o mecanismo `s3vector`

Medição e faturamento

Para obter informações sobre medição e cobrança desse recurso, consulte os preços do Amazon OpenSearch Service.

Desabilitar o mecanismo s3vector

Antes de desabilitar o mecanismo `s3vector`, exclua todos os índices que o usam atualmente. Se você não fizer isso, qualquer tentativa de desabilitar o mecanismo será malsucedida.

Observe também que habilitar ou desabilitar o mecanismo `s3vector` aciona uma implantação azul/verde no domínio.

Para desabilitar o mecanismo `s3vector`, edite a configuração do domínio e defina `S3VectorsEngine.Enabled: false`.

Pesquisa K-Nearest Neighbor (k-NN) no Amazon Service OpenSearch

Abreviação do algoritmo associado de k-vizinhos mais próximos, o k-NN for Amazon OpenSearch Service permite pesquisar pontos em um espaço vetorial e encontrar os “vizinhos mais próximos” desses pontos por distância euclidiana ou similaridade de cosseno. Os casos de uso incluem recomendações (por exemplo, um atributo de “outras músicas que você pode gostar” em um aplicativo de música), reconhecimento de imagem e detecção de fraudes.

> **Note**
>
> Esta documentação fornece uma breve visão geral do plug-in k-NN, bem como as limitações ao usar o plug-in com o serviço gerenciado OpenSearch . Para obter uma documentação abrangente do plug-in k-NN, incluindo exemplos simples e complexos, referências de parâmetros e a referência completa da API, consulte a documentação de código OpenSearch aberto. A documentação de código aberto também aborda o ajuste de desempenho e as configurações do k-NN-specific cluster.

Conceitos básicos do k-NN

Para usar o k-NN, é necessário criar um índice com a configuração `index.knn` e adicionar um ou mais campos do tipo de dados `knn_vector`.

```
PUT my-index

{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "my_vector1": {
        "type": "knn_vector",
        "dimension": 2
      },
      "my_vector2": {
        "type": "knn_vector",
        "dimension": 4
```

```
      }
    }
  }
}
```

O tipo de dados `knn_vector` oferece suporte a uma única lista de até 10.000 flutuantes, com o número de flutuantes definido pelo parâmetro `dimension`. Depois de criar o índice, adicione alguns dados a ele.

```
POST _bulk

{ "index": { "_index": "my-index", "_id": "1" } }
{ "my_vector1": [1.5, 2.5], "price": 12.2 }
{ "index": { "_index": "my-index", "_id": "2" } }
{ "my_vector1": [2.5, 3.5], "price": 7.1 }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "my_vector1": [3.5, 4.5], "price": 12.9 }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "my_vector1": [5.5, 6.5], "price": 1.2 }
{ "index": { "_index": "my-index", "_id": "5" } }
{ "my_vector1": [4.5, 5.5], "price": 3.7 }
{ "index": { "_index": "my-index", "_id": "6" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 10.3 }
{ "index": { "_index": "my-index", "_id": "7" } }
{ "my_vector2": [2.5, 3.5, 5.6, 6.7], "price": 5.5 }
{ "index": { "_index": "my-index", "_id": "8" } }
{ "my_vector2": [4.5, 5.5, 6.7, 3.7], "price": 4.4 }
{ "index": { "_index": "my-index", "_id": "9" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 8.9 }
```

Em seguida, você poderá pesquisar os dados usando o tipo de consulta `knn`.

```
GET my-index/_search
{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
    }
```

```
  }
}
```

Nesse caso, k é o número de vizinhos a serem retornados pela consulta, mas também é necessário incluir a opção `size`. Caso contrário, você obterá k resultados para cada fragmento (e cada segmento) em vez de k resultados para toda a consulta. O k-NN oferece suporte a um valor de k máximo de 10.000.

Se você misturar a consulta `knn` com outras cláusulas, poderá receber menos do que k resultados. Neste exemplo, a cláusula `post_filter` reduz o número de resultados de 2 para 1.

```
GET my-index/_search

{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
    }
  },
  "post_filter": {
    "range": {
      "price": {
        "gte": 6,
        "lte": 10
      }
    }
  }
}
```

Se precisar lidar com um grande volume de consultas e, ao mesmo tempo, manter o desempenho ideal, você pode usar a API `_msearch` para criar uma pesquisa em massa com JSON e enviar uma única solicitação para realizar várias pesquisas:

```
GET _msearch

{ "index": "my-index"}
{ "query": { "knn": {"my_vector2":{"vector": [2, 3, 5, 6],"k":2 }} } }
{ "index": "my-index", "search_type": "dfs_query_then_fetch"}
```

```
{ "query": { "knn": {"my_vector1":{"vector": [2, 3],"k":2 }} } }
```

O vídeo a seguir demonstra como configurar pesquisas vetoriais em massa para consultas K-NN.

Diferenças, ajustes e limitações do k-NN

OpenSearch permite que você modifique todas as configurações do k-NN usando a `_cluster/settings` API. No OpenSearch Serviço, você pode alterar todas as configurações, exceto `knn.memory.circuit_breaker.enabled` `knn.circuit_breaker.triggered` e. As estatísticas k-NN são incluídas como métricas da Amazon CloudWatch .

Em particular, verifique a `KNNGraphMemoryUsage` métrica em cada nó de dados em relação à `knn.memory.circuit_breaker.limit` estatística e à RAM disponível para o tipo de instância. OpenSearch O serviço usa metade da RAM de uma instância para o heap Java (até um tamanho de heap de 32 GiB). Por padrão, o k-NN usa até 50% da metade restante, portanto, um tipo de instância com 32 GiB de RAM pode acomodar 8 GiB de gráficos (32 * 0,5 * 0,5). A performance poderá ser prejudicada se o uso da memória do gráfico exceder esse valor.

Você pode migrar um índice k-NN criado na versão 2.x ou posterior ou um armazenamento frio em um domínio com a versão 2.17 UltraWarmou posterior.

A API de limpeza de cache e as apis de aquecimento para índices k-NN são bloqueadas para índices mornos. Quando a primeira consulta é iniciada para o índice, baixa os arquivos gráficos do Amazon S3 e carrega o gráfico na memória. Da mesma forma, quando o TTL expira para os gráficos, os arquivos são automaticamente removidos da memória.

Ingestão de vetores

A ingestão de vetores ajuda você a ingerir e indexar rapidamente OpenSearch domínios e OpenSearch coleções sem servidor. O serviço examina seu domínio ou coleção e cria um pipeline de ingestão em seu nome para carregar seus dados. OpenSearch A ingestão e a indexação do seu domínio ou coleção são gerenciadas para você pela ingestão de vetores.

Você pode acelerar e otimizar o processo de indexação Aceleração por GPU para indexação vetorial ativando Otimização automática recursos. Com a ingestão de vetores, você não precisa gerenciar a infraestrutura subjacente, corrigir o software ou escalar clusters para dar suporte à indexação e ingestão de seu banco de dados vetoriais. Isso permite que você crie rapidamente seu banco de dados vetoriais para atender às suas necessidades.

Como funciona

A ingestão de vetores examina seu domínio ou coleção e seu índice. Você pode configurar manualmente seus campos de índice vetorial ou OpenSearch permitir o uso da configuração automática.

A ingestão de vetores usa a OpenSearch ingestão (OSI) como o pipeline de dados entre o Amazon S3 e. OpenSearch O serviço processa vetores em paralelo para otimizar a velocidade de ingestão, respeitando os limites de escala do OSI e. OpenSearch

OpenSearch Preços de ingestão de vetores

Em qualquer momento específico, você paga apenas pelo número de ingestão OCUs de vetores alocados a um pipeline, independentemente de haver dados fluindo pelo pipeline. OpenSearch a ingestão de vetores acomoda imediatamente suas cargas de trabalho aumentando ou diminuindo a capacidade do pipeline com base no uso.

Para obter detalhes completos sobre preços, consulte Amazon OpenSearch Service Pricing.

Pré-requisitos

Antes de usar a ingestão de vetores, verifique se você tem os seguintes recursos:

- Bucket Amazon S3 contendo seus documentos OpenSearch JSON no formato Parquet ou JSONL
- OpenSearch recurso - um domínio ou uma coleção
- OpenSearch versão `2.19` ou posterior (necessária para a integração de otimização automática)

Criar banco de dados vetoriais

Use o fluxo de trabalho de criação de tarefas de ingestão vetorial para configurar o ajuste automatizado do índice vetorial e acelerar a criação de índices em grande escala.

> **Note**
>
> O conteúdo processual desta seção está sujeito a alterações à medida que a interface do usuário é finalizada. O fluxo de trabalho pode ser atualizado em versões futuras para refletir a experiência mais recente do console.

Para criar um trabalho de ingestão de vetores

1. Na seção Detalhes da tarefa de ingestão de vetores, em Nome, insira um nome para sua tarefa de ingestão.
2. Na seção Fonte de dados, configure o seguinte:
   a. Para o URI do Amazon S3, insira a localização do bucket do Amazon S3 contendo OpenSearch seus documentos JSON de serviço.
   b. Escolha Navegar no Amazon S3 para selecionar entre os buckets disponíveis ou escolha Exibir para visualizar o conteúdo do bucket.
   c. Em Tipo de conteúdo, selecione uma das seguintes opções:
      - Vetores - Os documentos já contêm vetores e não exigem geração adicional de incorporação vetorial.
      - Texto, imagem ou áudio: os documentos contêm conteúdo como texto, imagens ou bytes de áudio que precisam ser codificados em incorporações vetoriais.
3. Na seção Permissões da fonte de dados, configure as permissões de acesso:
   a. Para Perfil do IAM, escolha uma das seguintes opções:
      - Criar uma nova função
      - Use uma função existente
   b. Em Nome da função do IAM, insira um nome para a função.
4. Na seção Destino, configure o endpoint do OpenSearch serviço:
   a. Para Endpoint, escolha Escolha uma opção para selecionar entre seus domínios ou coleções compatíveis na região atual.
   b. Escolha Avançar para continuar com o endpoint selecionado.
5. Escolha Avançar para continuar na próxima etapa ou escolha Cancelar para sair sem salvar.

Características relacionadas

A ingestão de vetores funciona com os seguintes recursos do Amazon OpenSearch Service para otimizar o desempenho do seu banco de dados vetoriais:

Aceleração por GPU para indexação vetorial

A aceleração por GPU reduz o tempo necessário para criar, atualizar e excluir índices vetoriais. Quando usado com a ingestão de vetores, você pode acelerar significativamente o processo de ingestão e indexação para bancos de dados vetoriais de grande escala.

Otimização automática

A otimização automática descobre automaticamente as melhores compensações entre latência de pesquisa, qualidade e requisitos de memória. A ingestão de vetores pode aplicar recomendações de otimização automática durante o processo de ingestão para garantir que seus índices vetoriais sejam configurados de forma ideal.

Para obter melhores resultados, considere ativar a aceleração por GPU e a otimização automática ao usar a ingestão de vetores para criar bancos de dados vetoriais em grande escala.

Exportar índice vetorial do Amazon S3 para o mecanismo vetorial OpenSearch de serviço

Uma point-in-time exportação do índice vetorial selecionado do Amazon S3 para o OpenSearch Serviço. O mecanismo vetorial OpenSearch Service fornece um armazenamento vetorial simples e escalável com funcionalidade de pesquisa avançada.

Para exportar o índice vetorial do Amazon S3 para o mecanismo vetorial OpenSearch de serviço

1. Na seção Fonte, verifique os detalhes do índice vetorial do Amazon S3:
   - Índice vetorial do Amazon S3 - O nome do seu índice de origem
   - ARN do índice vetorial Amazon S3 - O nome do recurso Amazon do seu índice
2. Na seção Acesso ao serviço, configure a autorização OpenSearch do serviço:
   a. Em Escolha um método para autorizar o OpenSearch serviço, selecione uma das seguintes opções:
      - Criar e usar um novo perfil de serviço
      - Use um perfil de serviço existente
   b. Em Nome da função de serviço, insira um nome para a função de serviço.

> **Note**
>
> O nome da função de serviço deve ter de 1 a 64 caracteres. Os caracteres válidos são a-z, A-Z, 0-9 e pontos (.).

c. Escolha Exibir detalhes da permissão para revisar as permissões necessárias.

3. Expanda Configurações adicionais - opcional para configurar opções avançadas, se necessário.
4. Na seção Detalhes da exportação, configure as seguintes opções:

   - Automatize a criação OpenSearch de coleções de vetores de OpenSearch serviços - As coleções de serviços são usadas para armazenar dados vetoriais. A capacidade computacional sem servidor é medida em Unidades de Computação OpenSearch de Serviço (OCUs); por padrão, a capacidade máxima de OCU é 50.
   - Automatize a criação de funções do IAM para acesso ao serviço — Essa função é usada pelo OpenSearch Service para ler o índice vetorial do Amazon S3 e gravar OpenSearch na coleção Service.
   - Automatize OpenSearch a criação do pipeline de ingestão de OpenSearch serviços — os pipelines de ingestão de serviços são usados para ingerir dados. Um bucket do Amazon S3 é criado como uma prática recomendada para capturar e armazenar eventos com falha em um bucket do Amazon S3 Dead Letter Queue (DLQ), permitindo fácil acesso para solução de problemas e análise.

5. Escolha Exportar para iniciar o processo de exportação ou escolha Cancelar para sair sem exportar.

Importar o namespace vetorial do Amazon S3 para o mecanismo vetorial Service OpenSearch

Analisar seus dados vetoriais com o OpenSearch Service exige uma coleta única de OpenSearch serviços e uma configuração de permissão do IAM.

Para importar o namespace vetorial do Amazon S3 para o mecanismo vetorial Service OpenSearch

1. Na seção Fonte, configure o índice vetorial do Amazon S3:

   - Para o ARN do índice vetorial do Amazon S3, insira o ARN do seu índice vetorial do Amazon S3.

> **Note**
>
> Deve estar no formato arn:aws:iam: :account-id: /*:index vector-bucket-name

2. Na seção Acesso ao serviço, configure a autorização OpenSearch do serviço:

   a. Em Escolha um método para autorizar o OpenSearch serviço, selecione uma das seguintes opções:

      - Criar e usar um novo perfil de serviço
      - Use um perfil de serviço existente

   b. Em Nome da função de serviço, insira um nome para a função de serviço.

   > **Note**
   >
   > O nome da função de serviço deve ter de 1 a 64 caracteres. Os caracteres válidos são a-z, A-Z, 0-9 e pontos (.).

   c. Escolha Exibir detalhes da permissão para revisar as permissões necessárias.

3. Expanda Configurações adicionais - opcional para configurar opções avançadas, se necessário.
4. Na seção Etapas de importação, configure as seguintes opções de automação:

   - Automatize a criação OpenSearch de coleções de vetores de OpenSearch serviços - As coleções de serviços são usadas para armazenar dados vetoriais. A capacidade computacional sem servidor é medida em Unidades de Computação OpenSearch de Serviço (OCUs); por padrão, a capacidade máxima de OCU é 50.
   - Automatize a criação de funções do IAM para acesso ao serviço — Essa função é usada pelo OpenSearch Service para ler o índice vetorial do Amazon S3 e gravar OpenSearch na coleção Service.
   - Automatize OpenSearch a criação do pipeline de ingestão de OpenSearch serviços — os pipelines de ingestão de serviços são usados para ingerir dados. Um bucket do Amazon S3 é criado como uma prática recomendada para capturar e armazenar eventos com falha em um bucket do Amazon S3 Dead Letter Queue (DLQ), permitindo fácil acesso para solução de problemas e análise.

5. Escolha Importar para iniciar o processo de importação ou escolha Cancelar para sair sem importar.

Visualize trabalhos de ingestão de vetores e histórico de importação

Os trabalhos de ingestão de vetores criam um pipeline para vetorizar conjuntos de dados, automatizar o ajuste de índices vetoriais e acelerar a criação de índices em grande escala.

Para visualizar trabalhos de ingestão de vetores

1. Na seção Trabalhos de ingestão de vetores, veja as informações resumidas:
   - Trabalhos - Número total de trabalhos de ingestão
   - Escolha Criar banco de dados vetoriais para criar um novo trabalho de ingestão.
2. Na seção Importações de vetores do Amazon S3, veja o resumo da importação:
   - Total de importações - Número de importações concluídas
   - Escolha Importar vetores do Amazon S3 para iniciar uma nova importação
3. Na tabela de trabalhos de ingestão de vetores, monitore os trabalhos ativos com as seguintes informações:
   - Nome - O nome do trabalho
   - Status - Status atual do trabalho (por exemplo, Ativo)
   - Fonte de dados - Localização da fonte (por exemplo, s3://location)
   - Destino - Destino alvo
   - Última atualização - Carimbo de data/hora da atualização mais recente
4. Use a caixa de pesquisa para encontrar trabalhos de ingestão de vetores para localizar trabalhos específicos.
5. Para gerenciar trabalhos, escolha entre as seguintes ações:
   - Escolha Excluir para remover trabalhos selecionados.
   - Escolha Criar banco de dados vetoriais para criar trabalhos adicionais.
6. Na seção Histórico de importação de vetores do Amazon S3, acompanhe os eventos de importação:
   a. Use o filtro de intervalo de datas para especificar um período para o histórico de importação.
   b. Use o menu suspenso Status para filtrar por status de importação (por exemplo, Qualquer status).

c. Use a caixa de pesquisa para encontrar importações por índice vetorial Amazon S3 na... para localizar importações específicas.

d. Veja os detalhes da importação, incluindo:

- Importação iniciada em (UTC+ 5:30) - Quando a importação começou
- Status da importação - Status atual (Em andamento, Concluído, Falha, Parcialmente concluído)
- ARN do índice vetorial Amazon S3 - Identificador do índice de origem
- OpenSearch Coleção de vetores de serviço - Coleção de destinos

7. Escolha Importar vetor do Amazon S3 para iniciar um novo processo de importação.

Otimização automática

A otimização automática é um serviço que automatiza as otimizações do índice vetorial, permitindo que os usuários equilibrem a qualidade, a velocidade e o custo da pesquisa sem exigir semanas de ajustes manuais especializados. Ele avalia as configurações do índice com base nos requisitos de latência e recall definidos pelo usuário e gera recomendações de otimização, portanto, é necessária uma experiência mínima. Normalmente, as recomendações são entregues em 30 a 60 minutos.

A configuração tradicional do índice vetorial requer experiência e experimentação significativas para alcançar o desempenho ideal. Parâmetros como `ef_construction` (que controla a qualidade de construção do índice), `m` (que determina o número de conexões gráficas), `ef_search` (que controla a pesquisa HNSW) e métodos de quantização (quantização binária (32x, 16x, 8x), quantização escalar (4x)) afetam significativamente a precisão da pesquisa e a utilização de recursos. A otimização automática usa algoritmos de otimização de hiperparâmetros para descobrir configurações de índice que são exclusivamente ideais para seu conjunto de dados dentro dos requisitos definidos de latência e recuperação.

Benefícios

A otimização automática para OpenSearch fornece os seguintes benefícios:

- Ajuste automatizado de parâmetros - Elimina a experimentação manual com algoritmo (HNSW), quantização, repontuação e parâmetros do mecanismo, economizando tempo e reduzindo a curva de aprendizado para a otimização da pesquisa vetorial.

- Otimize a velocidade de pesquisa - Por padrão, OpenSearch está configurado para desempenho na memória. A otimização automática descobre compensações favoráveis que melhoram a qualidade da pesquisa e a economia de custos, mantendo uma velocidade de pesquisa aceitável.
- Otimização de custos - reduz os custos ao encontrar opções para reduzir seus requisitos de memória de índice e, ao mesmo tempo, minimizar as compensações de qualidade e velocidade da pesquisa.
- Otimize a qualidade da pesquisa - Ofereça potencialmente um recall maior do que as configurações padrão ou descubra compensações favoráveis que proporcionam economias de custo significativas com perda mínima de recall.

A otimização automática funciona junto com outros OpenSearch recursos, como Aceleração por GPU para indexação vetorial fornecer uma otimização abrangente do desempenho para cargas de trabalho de pesquisa vetorial.

Como funciona

A otimização automática opera por meio de uma arquitetura baseada em tarefas que analisa seus dados vetoriais e fornece recomendações de otimização. Principais pontos:

- Os usuários compartilham seus conjuntos de dados no formato Parquet ou JSONL em um bucket do Amazon S3.
- Eles configuram tarefas de otimização automática sem servidor configurando seus limites aceitáveis de recuperação e latência. Limites mais flexíveis permitem que o serviço descubra otimizações de custo mais significativas.
- Otimize automaticamente as tarefas executadas em uma infraestrutura totalmente gerenciada pelo Amazon Service. OpenSearch Os trabalhos não consomem recursos em seu domínio ou coleções. Os trabalhadores trabalham paralelamente para avaliar as configurações do índice e usam a amostragem em grandes conjuntos de dados para fornecer resultados normalmente em 30 a 60 minutos.
- Cada trabalho é cobrado com uma taxa fixa previsível. Para obter informações sobre preços, consulte Amazon OpenSearch Service Pricing.

Pré-requisitos

- Formato e permissões do conjunto de dados - Você deve ter seu conjunto de dados disponível como um ou mais arquivos Parquet ou JSONL em uma pasta de bucket do Amazon S3. Por exemplo:
  - Parquet: `s3://dataset-bucket-us-east-1/dataset_folder/first_half.parquet` e `s3://dataset-bucket-us-east-1/dataset_folder/second_half.parquet`
  - JSON: `s3://dataset-bucket-us-east-1/dataset_folder/data.jsonl`

  Forneça o URI da pasta anexa (por exemplo,`s3://dataset-bucket-us-east-1/dataset_folder/`). A pasta deve conter arquivos de um único formato — não misture arquivos Parquet e JSONL na mesma pasta. Esse conjunto de dados será usado para gerar as recomendações. Certifique-se de que sua função federada tenha as seguintes permissões do Amazon S3 nesse recurso:. `"s3:Get*", "s3:List*", "s3:Describe*"`
- Especifique os metadados corretos do conjunto de dados: o conjunto de dados fornecido deve conter linhas de valores flutuantes. O nome de cada coluna e a dimensionalidade de cada vetor devem corresponder às opções fornecidas no console. Por exemplo, se o conjunto de dados contiver vetores `train_data` nomeados para cada 768 dimensão, esses valores deverão corresponder ao console de otimização automática.
- (Se estiver usando o recurso de ingestão vetorial) - Se você planeja utilizar o recurso de ingestão (seguindo recomendações de otimização automática para criar índices e ingerir dados automaticamente), você deve configurar seu OpenSearch cluster para dar permissão de otimização automática para ingerir seu conjunto de dados no cluster. OpenSearch Para OpenSearch domínios com uma política de acesso ao domínio, conceda à função recém-criada acesso por meio dessa política. Para OpenSearch domínios com controle de acesso refinado, adicione a função de pipeline como uma função de back-end. Para coleções OpenSearch sem servidor, adicione a função de pipeline à política de acesso a dados.
- Permissões do IAM — você precisa das seguintes permissões do IAM para usar a otimização automática:
  - `opensearch:SubmitAutoOptimizeJob`
  - `opensearch:GetAutoOptimizeJob`
  - `opensearch:DeleteAutoOptimizeJob`
  - `opensearch:CancelAutoOptimizeJob`
  - `opensearch:ListAutoOptimizeJobs`

> **Note**
>
> Essas são políticas baseadas em identidade. A otimização automática não oferece suporte a políticas baseadas em recursos.

- Expiração da credencial - configure sua sessão de usuário federado para ter uma expiração mínima de credencial de pelo menos 1 hora. Para conjuntos de dados muito grandes ou dimensões altas, considere aumentar a duração da expiração em até 3 horas.

Casos de uso para otimização automática

A otimização automática é particularmente valiosa nos seguintes cenários:

Otimização da configuração inicial

Ao implantar pela primeira vez aplicativos de pesquisa vetorial, determinar os parâmetros ideais de HNSW geralmente requer testes extensivos e experiência no domínio. A otimização automática elimina esse trial-and-error processo analisando seus dados e as características da carga de trabalho para recomendar configurações prontas para produção.

Esse caso de uso é ideal para equipes iniciantes na pesquisa vetorial ou para aquelas que estão migrando de outras plataformas de banco de dados vetoriais e que precisam estabelecer configurações básicas rapidamente.

Otimização de escalabilidade

À medida que seu conjunto de dados vetoriais cresce de milhares para milhões de vetores, os parâmetros que funcionaram bem inicialmente podem ficar abaixo do ideal. A otimização automática recomenda ajustes para manter o desempenho em grande escala.

Redução de custos

Os índices vetoriais podem consumir recursos significativos de computação e armazenamento, especialmente com incorporações de alta dimensão. A otimização automática identifica oportunidades de reduzir custos encontrando configurações de parâmetros mais eficientes que mantêm os níveis de desempenho necessários enquanto usam menos recursos.

Por exemplo, a otimização automática pode descobrir que seu `m` valor atual (conexão gráfica) é maior do que o necessário para seus requisitos de precisão, permitindo que você reduza o tempo de indexação e o armazenamento sem afetar a qualidade da pesquisa.

Solução de problemas de performance

Ao experimentar um desempenho de consulta lento ou alta latência nas operações de pesquisa vetorial, a otimização automática pode analisar seu conjunto de dados e identificar uma configuração mais ideal. O serviço fornece recomendações específicas para lidar com gargalos de desempenho, como ajustar a conectividade gráfica ou os parâmetros de pesquisa.

Limitações

- Disponibilidade regional — a otimização automática está disponível somente nas seguintes AWS regiões:
  - ap-south-1
  - eu-west-1
  - us-west-2
  - us-east-2
  - us-east-1
  - eu-central-1
  - ap-southeast-2
  - ap-northeast-1
  - ap-southeast-1
- Tipos de coleção - A otimização automática é suportada somente para coleções e OpenSearch domínios de pesquisa vetorial (2.19, 3.1 e 3.3).
- Suporte do motor

  Suporte do mecanismo por tipo de implantação

Mecanismo	Sem servidor	OpenSearch Gerenciado
Lucene	Não	Sim
Faiss	Sim	Sim
Nmslib	Não	Não

- Suporte a algoritmos - A otimização automática suporta somente índices vetoriais baseados em HNSW.
- Trabalhos simultâneos - Você pode executar até 10 trabalhos de otimização simultâneos por conta por região. Nenhum novo emprego pode ser aceito se o limite for atingido.
- Duração do trabalho — Os trabalhos de otimização podem levar de 15 minutos a várias horas, dependendo do tamanho do conjunto de dados, da dimensão e das métricas de desempenho necessárias.
- Recomendações - A otimização automática sugere apenas até 3 recomendações.
- Conjunto de dados
  - Formatos suportados: Parquet, JSONL
  - Armazenamento de dados: Amazon S3

Faturamento e custos

A otimização automática usa um modelo de precificação por trabalho em que você paga por cada trabalho de otimização bem-sucedido, independentemente do tamanho do conjunto de dados e das configurações de otimização. Você não será cobrado por trabalhos fracassados ou cancelados. Além disso, a otimização automática é executada em uma infraestrutura diferente da dos OpenSearch clusters gerenciados ou sem servidor, portanto, não afeta a utilização de recursos dos clusters preexistentes.

Modelo de definição de preços

Os custos de otimização automática são cobrados separadamente dos custos padrão de computação e armazenamento em domínio OpenSearch gerenciado ou OpenSearch sem servidor.

Para obter informações sobre preços, consulte Amazon OpenSearch Service Pricing.

Formatos de dados compatíveis

A otimização automática oferece suporte aos seguintes formatos de dados para conjuntos de dados vetoriais armazenados no Amazon S3:

Formato Parquet

O Parquet é um formato de armazenamento em colunas otimizado para cargas de trabalho analíticas. Cada arquivo Parquet deve conter uma coluna de matrizes flutuantes representando seus dados vetoriais.

Exemplo de estrutura de arquivo Parquet (vista como uma tabela):

```
| id  | train_data                     |
|-----|--------------------------------|
| 1   | [0.12, 0.45, 0.78, ..., 0.33] |
| 2   | [0.56, 0.89, 0.12, ..., 0.67] |
| 3   | [0.34, 0.67, 0.90, ..., 0.11] |
```

Formato JSONL

JSONL (Linhas JSON) é um formato de texto em que cada linha é um objeto JSON válido. Cada linha deve conter um campo com uma matriz flutuante representando seus dados vetoriais.

Exemplo de arquivo JSONL:

```
{"id": 1, "train_data": [0.12, 0.45, 0.78, 0.33]}
{"id": 2, "train_data": [0.56, 0.89, 0.12, 0.67]}
{"id": 3, "train_data": [0.34, 0.67, 0.90, 0.11]}
```

Conversão entre formatos

Se seus dados estiverem em um formato diferente, você poderá usar os seguintes scripts Python para convertê-los.

Converter JSON ou JSONL em Parquet

```
#!/usr/bin/env python3
import json
import pyarrow as pa
import pyarrow.parquet as pq
from pathlib import Path
from typing import Any, Dict, List


def load_json_any(path: Path) -> List[Dict[str, Any]]:
    """
    Load JSON that can be:
      - a list of objects
      - a single object
      - JSON Lines (one object per line)
    Returns list[dict].
```

```
    """
    text = path.read_text().strip()

    # Try full JSON file
    try:
        obj = json.loads(text)
        if isinstance(obj, list):
            return obj
        if isinstance(obj, dict):
            return [obj]
    except json.JSONDecodeError:
        pass

    # Fallback # JSON Lines
    records = []
    for i, line in enumerate(text.splitlines(), start=1):
        line = line.strip()
        if not line:
            continue
        try:
            rec = json.loads(line)
        except json.JSONDecodeError as e:
            raise ValueError(f"Invalid JSON on line {i}: {e}")
        if not isinstance(rec, dict):
            raise ValueError(f"Line {i} must contain a JSON object")
        records.append(rec)

    return records


def json_to_parquet(json_path: str, parquet_path: str, compression: str = "snappy"):
    """Convert ANY JSON to Parquet (schema inferred)."""
    records = load_json_any(Path(json_path))
    table = pa.Table.from_pylist(records)
    pq.write_table(table, parquet_path, compression=compression)
    print(f"Wrote {len(records)} rows to {parquet_path}")


if __name__ == "__main__":
    INPUT_JSON = "vectors.jsonl"
    OUTPUT_PARQUET = "vectors.parquet"
    json_to_parquet(INPUT_JSON, OUTPUT_PARQUET)
```

Conversão de Parquet para JSONL

```
#!/usr/bin/env python3
import json
import pyarrow.parquet as pq


def parquet_to_jsonl(parquet_path: str, jsonl_path: str):
    """Convert a Parquet file to JSONL format."""
    table = pq.read_table(parquet_path)
    rows = table.to_pylist()
    with open(jsonl_path, "w") as f:
        for row in rows:
            f.write(json.dumps(row) + "\n")
    print(f"Wrote {len(rows)} rows to {jsonl_path}")


if __name__ == "__main__":
    INPUT_PARQUET = "vectors.parquet"
    OUTPUT_JSONL = "vectors.jsonl"
    parquet_to_jsonl(INPUT_PARQUET, OUTPUT_JSONL)
```

Usando a otimização automática no console

Você pode usar o console do Amazon OpenSearch Service para criar trabalhos de ingestão de vetores, monitorar seu progresso, visualizar recomendações de otimização e criar índices com base nessas recomendações.

Pré-requisitos

Antes de usar a otimização automática no console, você deve ter o seguinte:

- Uma AWS conta ativa com acesso ao OpenSearch console.
- Uma coleção existente OpenSearch sem servidor de pesquisa vetorial de tipo ou um domínio gerenciado OpenSearch .
- Permissões do IAM para as seguintes ações:
  - `opensearch:SubmitAutoOptimizeJob`
  - `opensearch:GetAutoOptimizeJob`
  - `opensearch:DeleteAutoOptimizeJob`

- `opensearch:CancelAutoOptimizeJob`
- `opensearch:ListAutoOptimizeJobs`

> **Note**
>
> Essas são políticas baseadas em identidade. AWS não oferece suporte a políticas baseadas em recursos para otimização automática de recursos.

- Configure sua sessão de usuário federado para ter uma expiração mínima de credencial de pelo menos 1 hora. Para conjuntos de dados muito grandes ou dimensões altas, considere aumentar a duração da expiração em até 3 horas.

Criação de um trabalho de ingestão de vetores

Um trabalho de ingestão de vetores analisa seus dados vetoriais e fornece recomendações de otimização para a configuração do índice.

Para criar um trabalho de ingestão de vetores

1. Faça login no console do Amazon OpenSearch Service em Console de gerenciamento da AWS.
2. No painel de navegação, escolha Otimização automática.
3. Escolha Criar tarefa de ingestão de vetores.
4. Em Detalhes do trabalho, insira um nome para seu trabalho de ingestão de vetores. Esse nome ajuda você a identificar o trabalho no console.
5. Em Fonte de dados, configure o seguinte:

    a. Para o URI do Amazon S3, insira o URI do Amazon S3 da pasta que contém seus arquivos de dados (Parquet ou JSONL). O URI deve apontar para a pasta anexa, não para arquivos individuais. Por exemplo, se seu arquivo estiver em `s3://my-bucket/my-folder/file1.parquet` ou `s3://my-bucket/my-folder/data.jsonl`, insira `s3://my-bucket/my-folder/`.

    > **Note**
    >
    > A pasta deve conter arquivos de um único formato. Não misture arquivos Parquet e JSONL na mesma pasta.

b. Em Região, selecione a AWS região em que seu bucket do Amazon S3 está localizado. A região deve corresponder à localização do bucket.

6. Em OpenSearch domínio, selecione um domínio ou coleção existente ou escolha Criar novo para criar um.

> **Note**
>
> Você pode especificar um domínio OpenSearch gerenciado ou uma coleção OpenSearch sem servidor sem servidor.

7. Em Permissões da fonte de dados, especifique a função do IAM que tem permissões para acessar seu bucket, OpenSearch domínio ou coleção do Amazon S3. A função deve ter as permissões necessárias com base na configuração do seu domínio ou coleção:

   - Para OpenSearch domínios com uma política de acesso ao domínio, conceda acesso à função por meio dessa política.
   - Para OpenSearch domínios com controle de acesso refinado, adicione a função como uma função de back-end.
   - Para coleções OpenSearch sem servidor, adicione a função à política de acesso a dados.

8. Escolha Próximo.
9. Em Configurar índice, especifique o seguinte:

   a. Em Nome do campo, insira o nome do campo do seu conjunto de dados que contém os dados vetoriais.

   b. Para Tipo de espaço, selecione a métrica de distância usada para calcular a distância entre vetores:

      - l2 - Distância euclidiana
      - cosinesimil - Similaridade de cosseno
      - produto interno - Produto interno

   c. Em Dimensão, insira o número de valores de ponto flutuante em cada vetor.

10. Em Requisitos de desempenho, configure o seguinte:

    a. Para Recall, especifique a qualidade de pesquisa desejada como um valor decimal entre 0 e 1. Valores de recall mais altos retornam resultados mais relevantes. Por exemplo:

- 0,95 indica que, em média, 19 dos 20 vetores de documentos verdadeiros mais próximos de um vetor de consulta são retornados
- 0,9 indica 9 em 10
- 0,8 indica 8 em 10

b. Em Requisitos de latência de pesquisa, selecione sua tolerância de latência. Requisitos modestos permitem maior economia de custos por meio de métodos de compactação que diminuem os requisitos de memória.

11. Escolha Próximo.
12. Revise sua configuração e escolha Criar.

O trabalho começa a ser processado. Você pode monitorar seu progresso na tabela Vector Inestion Jobs.

Monitorando trabalhos de otimização

Você pode monitorar o status de seus trabalhos de ingestão de vetores na página de destino de otimização automática.

Para monitorar trabalhos de otimização

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, escolha Otimização automática.
3. A tabela Trabalhos de ingestão de vetores exibe todos os trabalhos com seu status atual. Atualize a página para ver as informações de status atualizadas.

> **Note**
>
> Não há mecanismo automático de atualização ou notificação. Você deve atualizar manualmente o console para ver quando um trabalho é concluído.

Entendendo os estados do status do trabalho

As tarefas de otimização automática podem ter os seguintes valores de status:

Pendente

O trabalho está na fila e aguardando o início.

Em execução

O trabalho de otimização automática é analisar ativamente seus dados e gerar recomendações.

Completed

O trabalho de otimização automática foi concluído com êxito. Todas as análises, avaliações e recomendações estão completas e disponíveis para visualização.

Failed

O trabalho encontrou um erro. Veja os detalhes do erro na página de detalhes do trabalho para determinar a causa.

Ativo

Um índice foi criado no cluster anexado e os dados foram ingeridos.

A duração do trabalho depende principalmente do tamanho do conjunto de dados e da carga de serviço atual. Os trabalhos típicos são concluídos em 15 minutos a várias horas.

Visualizar detalhes do trabalho

Você pode visualizar informações detalhadas sobre um trabalho de otimização específico, incluindo sua configuração e status.

Para ver os detalhes do trabalho

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, escolha Otimização automática.
3. Na tabela Trabalhos de ingestão de vetores, escolha o nome do trabalho.
4. A página de detalhes do trabalho exibe as seguintes informações:
   - Nome e status do trabalho
   - Configuração da fonte de dados (URI e região do Amazon S3)
   - OpenSearch domínio ou coleção

- Configuração do índice (nome do campo, tipo de espaço, dimensão)
- Requisitos de desempenho (recall e latência)
- Mensagens de erro (se o trabalho falhar)

Visualizando e entendendo os resultados

Depois que um trabalho for concluído com êxito, você poderá visualizar as recomendações de otimização.

Para visualizar os resultados da otimização

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, escolha Otimização automática.
3. Na tabela Trabalhos de ingestão de vetores, escolha um trabalho com o status Concluído.
4. A página de resultados exibe as seguintes seções:

   - Visão geral dos resultados - Mostra a recuperação estimada da qualidade da pesquisa em comparação com sua necessidade e a área ocupada pela memória do índice em comparação com a configuração mais recomendada.
   - Recomendações - Lista até três recomendações de otimização, ordenadas com a recomendação principal como a melhor opção para sua configuração. Cada recomendação inclui:
     - Parâmetros de configuração do índice
     - Parâmetros de configuração de pesquisa
     - Métricas de desempenho esperadas
     - Estimativas da pegada de memória

> **Note**
>
> Embora as recomendações sejam ordenadas pela melhor combinação, você pode selecionar qualquer recomendação que melhor se adapte ao seu caso de uso específico. Otimize automaticamente as tentativas de encontrar as correspondências mais próximas aos critérios de recall escolhidos.

Criando um índice a partir de recomendações

Depois de analisar as recomendações de otimização, você pode criar manualmente um índice usando a configuração recomendada ou criar automaticamente um índice com a recomendação selecionada.

Para criar um índice automaticamente

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, escolha Otimização automática.
3. Na tabela Trabalhos de ingestão de vetores, escolha um trabalho com o status Concluído.
4. Revise as recomendações e selecione a que você deseja usar.
5. Escolha Criar índice.
6. O sistema cria automaticamente um índice em seu cluster usando a recomendação selecionada e ingere os dados vetoriais do seu conjunto de dados.

Para criar um índice manualmente

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação, escolha Otimização automática.
3. Na tabela Trabalhos de ingestão de vetores, escolha um trabalho com o status Concluído.
4. Analise as recomendações e anote a configuração do índice e os parâmetros de configuração de pesquisa da recomendação escolhida.
5. Use a OpenSearch API ou o console para criar manualmente um índice com os parâmetros recomendados.

Características relacionadas

A otimização automática funciona em conjunto com outros recursos do Amazon OpenSearch Service para ajudar você a criar e otimizar aplicativos de pesquisa vetorial:

- Aceleração por GPU para indexação vetorial- Acelere a criação de índices vetoriais usando a aceleração por GPU para reduzir o tempo e os custos de indexação.

- Ingestão de vetores- Ingira e indexe rapidamente dados vetoriais do Amazon S3 em seu domínio ou coleção.

Aceleração por GPU para indexação vetorial

A aceleração por GPU ajuda você a criar bancos de dados vetoriais em grande escala com mais rapidez e eficiência. Você pode ativar esse recurso em OpenSearch domínios novos ou existentes e coleções sem OpenSearch servidor. Esse recurso usa aceleração por GPU para reduzir o tempo necessário para indexar dados em índices vetoriais.

Com a aceleração por GPU, você pode aumentar a velocidade de indexação vetorial em até 10 vezes a um quarto do custo de indexação.

Pré-requisitos

A aceleração por GPU é suportada em OpenSearch domínios que executam a OpenSearch versão `3.1` ou posterior e coleções sem servidor. OpenSearch Para obter mais informaçõesAtualizando domínios do Amazon OpenSearch Service, consulte UpdateDomainConfig, UpdateCollection APIse.

Como funciona

Os índices vetoriais exigem recursos computacionais significativos para criar estruturas de dados, como gráficos Hierarchical Navigable Small Worlds (HNSW). Quando você ativa a aceleração de GPU em seu domínio ou coleção, detecta OpenSearch automaticamente oportunidades para acelerar suas criações de índice e transfere as compilações de índice para instâncias de GPU. OpenSearch O serviço gerencia as instâncias da GPU em seu nome, atribuindo-as ao seu domínio ou coleção quando necessário. Isso significa que você não gerencia a utilização nem paga pelo tempo ocioso.

Você paga somente pelo processamento útil por meio de Unidades de Computação (OCU) - Aceleração vetorial. Cada OCU de aceleração vetorial é uma combinação de aproximadamente 8 GiB de memória de CPU, 2 CPUs v e 6 GiB de memória de GPU. Para obter mais informações, consulte the section called “Preços de aceleração de GPU”.

Para ativar a aceleração de GPU para seu domínio ou coleção, consulteAtivando a aceleração da GPU.

Preços de aceleração de GPU

AWS cobra quando OpenSearch detecta oportunidades de acelerar as cargas de trabalho de criação de índices do seu domínio ou coleção. Cada OCU de aceleração vetorial é uma combinação de aproximadamente 8 GiB de memória de CPU, 2 CPUs v e 6 GiB de memória de GPU.

AWS fatura a OCU com granularidade de segundo nível. No extrato da sua conta, você verá uma entrada para computação em horas de OCU.

Por exemplo, quando você usa a aceleração de GPU por uma hora para criar um índice, usando 2 vCPU e 1 GiB de memória de GPU, você recebe a cobrança de 1 OCU. Se você usar 9 GiB de memória de CPU enquanto usa a aceleração por GPU, serão cobradas 2 OCUs.

OpenSearch O Serverless adiciona mais OCUs em incrementos de 1 OCU com base na potência computacional e no armazenamento necessários para dar suporte às suas coleções. Você pode configurar um número máximo de OCUs para sua conta para controlar os custos.

> **Note**
>
> O número de OCUs provisionados a qualquer momento pode variar e não é exato. Com o tempo, o OpenSearch algoritmo usado pelo OpenSearch Serverless continuará melhorando para minimizar melhor o uso do sistema.

Para obter detalhes completos sobre preços, consulte Amazon OpenSearch Service Pricing.

Operações de aceleração e gravação de GPU

A aceleração da GPU é ativada quando a taxa de ingestão OpenSearch do vetor (MB/seg) está dentro de uma faixa. Em OpenSearch domínios, você tem a flexibilidade de configurar esse intervalo por meio de `index.knn.remote_index_build.size.min` e. `index.knn.remote_index_build.size.max` Por exemplo, com o intervalo inferior padrão de 50 MB, gravar 15.000 vetores de precisão total com 768 dimensões entre os intervalos de atualização acionará a aceleração da GPU por padrão.

Os dados são gravados com as seguintes operações de API:

- Flush
- Em massa

- Reindexar
- Índice
- Atualização
- Excluir
- Fusão forçada

A aceleração da GPU é ativada com mesclagens automáticas e manuais de segmentos.

Configurações de índice suportadas

O motor Faiss suporta aceleração por GPU.

As configurações a seguir não oferecem suporte à aceleração por GPU:

- Quantização de produtos Faiss
- Índice de arquivos invertidos (IVF)
- Biblioteca espacial não métrica
- Motor Lucene

Suportado Regiões da AWS

A aceleração por GPU está disponível no seguinte: Regiões da AWS

- Leste dos EUA (N. da Virgínia)
- Oeste dos EUA (Oregon)
- Ásia-Pacífico (Sydney)
- Ásia-Pacífico (Tóquio)
- Europa (Irlanda)

Ativando a aceleração da GPU

Você pode ativar a aceleração por GPU ao criar ou atualizar um OpenSearch domínio ou uma coleção OpenSearch sem servidor com o Console de gerenciamento da AWS,, ou SDK. AWS CLI AWS

Depois de ativar a aceleração por GPU em seu domínio ou coleção, esse recurso é ativado por padrão em todos os índices. Se você precisar desativar esse recurso no nível do índice, consulteCriação de índices vetoriais acelerados por GPU.

Console

Os procedimentos a seguir permitem a aceleração por GPU para OpenSearch domínios e coleções OpenSearch sem servidor usando o console de gerenciamento sem servidor. OpenSearch

Create new domain

Para criar um OpenSearch domínio com a aceleração por GPU ativada, consulte. Criação OpenSearch de domínios de serviço

Edit existing domain

1. Abra o console OpenSearch de gerenciamento de serviços.
2. No painel de navegação à esquerda, escolha Domínios.
3. Escolha seu nome de domínio para abrir a página de detalhes do domínio.
4. Escolha Ações e, em seguida, Editar domínio.
5. Na seção Recursos avançados, selecione Ativar aceleração de GPU. Depois que esse recurso é ativado, suas operações de indexação vetorial são aceleradas.
6. Escolha Salvar alterações.

Create new collection

Para criar uma coleção OpenSearch sem servidor com a aceleração por GPU ativada, consulte. Tutorial: Introdução ao Amazon OpenSearch Serverless Durante a criação da coleção, certifique-se de selecionar o tipo de coleta de pesquisa vetorial e ativar a aceleração por GPU na configuração da pesquisa vetorial.

Edit existing collection

1. Abra o console OpenSearch de gerenciamento de serviços.
2. No painel de navegação, escolha Coleções.
3. Escolha o nome da coleção para abrir a página de detalhes da coleção.
4. Na seção Opções de implantação, Edite a aceleração da GPU vetorial.
5. Desative ou ative a aceleração da GPU.

6. Escolha Salvar alterações.

AWS CLI

Create new domain

O AWS CLI exemplo a seguir cria um OpenSearch domínio com aceleração por GPU ativada no Leste dos EUA (Norte da Virgínia). *text*Substitua o pelo de sua própria configuração.

```
aws opensearch create-domain \
    --domain-name my-domain \
    --engine-version OpenSearch_3.1 \
    --cluster-config InstanceType=r6g.xlarge.search,\
        InstanceCount=1,\
        DedicatedMasterEnabled=true,\
        DedicatedMasterCount=3,\
        DedicatedMasterType=m6g.large.search \
    --ebs-options "EBSEnabled=true,\
        VolumeType=gp3,\
        VolumeSize=2000" \
    --encryption-at-rest-options '{"Enabled":true}' \
    --aiml-options '{"ServerlessVectorAcceleration": {"Enabled": true}}' \
    --node-to-node-encryption-options '{"Enabled":true}' \
    --domain-endpoint-options '{"EnforceHTTPS":true,\
        "TLSSecurityPolicy":"Policy-Min-TLS-1-0-2019-07"}' \
    --access-policies '{"Version": "2012-10-17",
        "Statement": [{
            "Effect": "Allow",
            "Principal": {"AWS": "*"},
            "Action": "es:*",
            "Resource": "arn:aws:es:us-east-1:123456789012:domain/my-domain/*"
        }]}' \
    --advanced-security-options '{
        "Enabled":true,
        "InternalUserDatabaseEnabled":true,
        "MasterUserOptions": {
            "MasterUserName":"USER_NAME",
            "MasterUserPassword":"PASSWORD"
        }}' \
    --region us-east-1
```

Edit existing domain

O AWS CLI exemplo a seguir ativa a aceleração por GPU para um domínio existente. OpenSearch *text*Substitua o pelo de sua própria configuração.

```
aws opensearch update-domain-config \
    --domain-name my-domain \
    --cluster-config InstanceType=r7g.16xlarge.search,InstanceCount=3 \
    --aiml-options '{"ServerlessVectorAcceleration": {"Enabled": true}}'
```

Create new collection

O AWS CLI exemplo a seguir cria uma coleção OpenSearch Serverless com aceleração por GPU ativada no Leste dos EUA (Norte da Virgínia). *text*Substitua o pelo de sua própria configuração.

```
aws opensearchserverless create-collection \
    --name "my-collection" \
    --type "VECTORSEARCH" \
    --description "My vector collection with GPU acceleration" \
    --vector-options '{"ServerlessVectorAcceleration": "ENABLED"}' \
    --region us-east-1
```

Edit existing collection

O AWS CLI exemplo a seguir ativa a aceleração por GPU para uma coleção Serverless existente OpenSearch . *text*Substitua o pelo de sua própria configuração.

```
aws opensearchserverless update-collection \
    --id 07tjusf2h91cunochc \
    --vector-options '{"ServerlessVectorAcceleration": "ENABLED"}' \
    --region us-east-1
```

Criação de índices vetoriais acelerados por GPU

Depois de ativar a aceleração por GPU em seu domínio ou coleção, crie índices vetoriais que possam aproveitar o processamento da GPU.

> **Note**
>
> Quando você cria um domínio com a aceleração por GPU ativada, a `index.knn.remote_index_build.enabled` configuração é `true` por padrão. Você não

precisa definir explicitamente essa configuração ao criar índices. Para coleções, você deve especificar explicitamente um valor para essa configuração.

Creating index with GPU-acceleration

O exemplo a seguir cria um índice vetorial otimizado para processamento de GPU. Esse índice armazena vetores de 768 dimensões (comuns para incorporação de texto).

```
PUT my-vector-index
{
  "settings": {
    "index.knn": true,
    "index.knn.remote_index_build.enabled": true
  },
  "mappings": {
    "properties": {
      "vector_field": {
        "type": "knn_vector",
        "dimension": 768
      },
      "text": {
        "type": "text"
      }
    }
  }
}
```

Principais elementos de configuração:

- `"index.knn": true`- Habilita a funcionalidade do vizinho mais próximo
- `"index.knn.remote_index_build.enabled": true`- Permite o processamento por GPU para esse índice. Quando o domínio tem a aceleração por GPU ativada, o padrão é não especificar essa configuração. `true` Para coleções, você deve especificar explicitamente um valor para essa configuração.
- `"dimension": 768`- Especifica o tamanho do vetor (ajuste com base no seu modelo de incorporação)

Creating index without GPU-acceleration

O exemplo a seguir cria um índice vetorial em que o processamento da GPU está desativado. Esse índice armazena vetores de 768 dimensões (comuns para incorporação de texto).

```
PUT my-vector-index
{
  "settings": {
    "index.knn": true,
    "index.knn.remote_index_build.enabled": false
  },
  "mappings": {
    "properties": {
      "vector_field": {
        "type": "knn_vector",
        "dimension": 768
      },
      "text": {
        "type": "text"
      }
    }
  }
}
```

Indexação de dados vetoriais e mesclagem forçada

Depois de criar um índice vetorial acelerado por GPU em seu domínio ou coleção, você pode adicionar dados vetoriais e otimizar seu índice usando operações padrão. OpenSearch A aceleração da GPU aprimora automaticamente o desempenho da indexação e as operações de mesclagem forçada, tornando mais rápido criar e manter aplicativos de pesquisa vetorial em grande escala sem exigir alterações nos fluxos de trabalho existentes.

Indexação de dados vetoriais

Indexe dados vetoriais como você faria normalmente. A aceleração da GPU se aplica automaticamente às operações de indexação e mesclagem forçada. O exemplo a seguir demonstra como adicionar documentos vetoriais ao seu índice usando a API em massa. Cada documento contém um campo vetorial com valores numéricos e conteúdo de texto associado:

```
POST _bulk
```

```
{"index": {"_index": "my-vector-index"}}
{"vector_field": [0.1, 0.2, 0.3, ...], "text": "Sample document 1"}
{"index": {"_index": "my-vector-index"}}
{"vector_field": [0.4, 0.5, 0.6, ...], "text": "Sample document 2"}
```

Operações de fusão forçada

A aceleração da GPU também se aplica às operações de mesclagem forçada, o que pode reduzir significativamente o tempo necessário para otimizar os índices vetoriais. Observe que as operações de mesclagem forçada não são suportadas em coleções. O exemplo a seguir demonstra como otimizar seu índice vetorial consolidando todos os segmentos em um único segmento:

```
POST my-vector-index/_forcemerge?max_num_segments=1
```

Práticas recomendadas

Siga estas melhores práticas para maximizar os benefícios da aceleração por GPU para suas cargas de trabalho de pesquisa vetorial:

- Aumente os clientes do índice - Para aproveitar ao máximo GPUs a criação do índice, aumente o número de clientes do índice que estão ingerindo OpenSearch dados. Isso permite uma melhor paralelização e utilização dos recursos da GPU.
- Ajuste o limite aproximado - altere a `index.knn.advanced.approximate_threshold` configuração para garantir que não ocorram construções menores de índices de segmentos, o que melhora a velocidade geral de ingestão. Um valor de 10.000 é um bom ponto de partida. Para coleções, você deve especificar explicitamente um valor para essa configuração.
- Otimize o tamanho do fragmento - tente criar fragmentos que tenham pelo menos 1 milhão de documentos. Fragmentos com menos do que esse número de documentos podem não ver os benefícios gerais da aceleração da GPU.

Usando OpenSearch painéis com o Amazon Service OpenSearch

OpenSearch O Dashboards é uma ferramenta de visualização de código aberto projetada para trabalhar com. OpenSearch O Amazon OpenSearch Service fornece uma instalação de painéis com cada domínio do OpenSearch serviço. O OpenSearch Dashboards é executado nos nós de dados quentes do domínio.

OpenSearch Os painéis são uma ferramenta de visualização para explorar e analisar dados em um único OpenSearch domínio. Por outro lado, a interface de OpenSearch usuário centralizada (também chamada de OpenSearch aplicativo) é uma interface de usuário baseada em nuvem que se conecta a vários OpenSearch domínios, coleções OpenSearch sem servidor e fontes de dados. AWS Ela inclui espaços de trabalho para casos de uso específicos, como observabilidade e analytics de segurança, e fornece uma experiência unificada em todos os conjuntos de dados. Embora o Dashboards seja vinculado aos domínios individuais, a interface de usuário centralizada permite integração e análise de dados entre domínios. Para obter mais informações, consulte OpenSearch UI.

Você pode encontrar um link para OpenSearch painéis no painel do seu domínio no console OpenSearch de serviços. Para domínios em execução OpenSearch, o URL é*domain-endpoint*`/_dashboards/`. Para domínios que executam o Elasticsearch legado, o URL é *domain-endpoint*`/_plugin/kibana`.

As consultas que usam essa instalação padrão do Dashboads têm um tempo limite de 300 segundos.

> **Note**
>
> Esta documentação discute os OpenSearch painéis no contexto do Amazon OpenSearch Service, incluindo diferentes maneiras de se conectar a ele. Para obter uma documentação abrangente, incluindo um guia de introdução, instruções para criar um painel, gerenciamento de painéis e Dashboards Query Language (DQL), consulte OpenSearch Painéis na documentação de código aberto. OpenSearch

Controlar acesso ao Dashboards

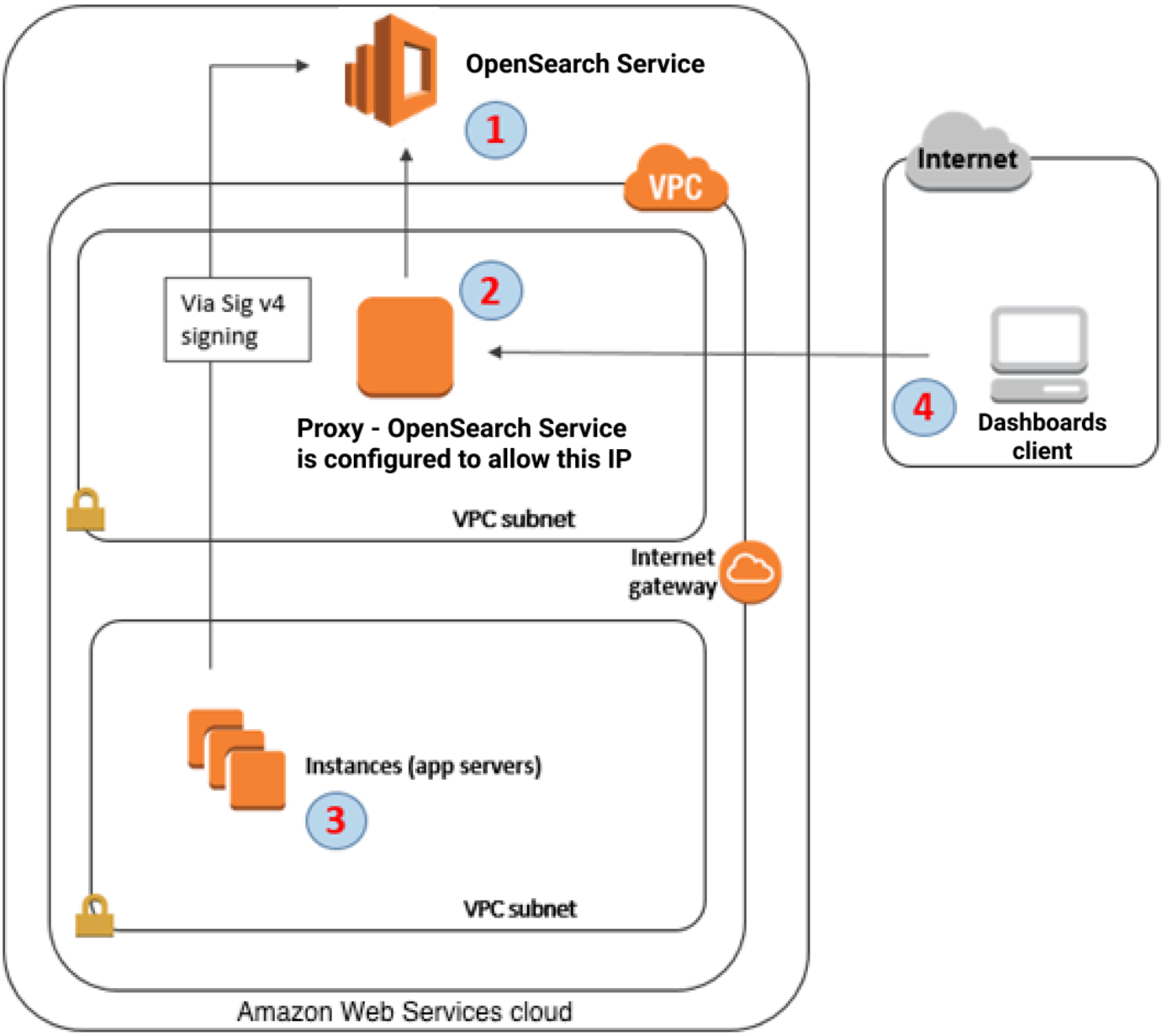
Os painéis não oferecem suporte nativo a usuários e funções do IAM, mas o OpenSearch Service oferece várias soluções para controlar o acesso aos painéis:

- Habilitar a Autenticação SAML para Dashboards.
- Usar o controle de acesso refinado com a autenticação básica HTTP.
- Configure a Autenticação do Cognito para painéis.
- Para domínios de acesso público, configure uma política de acesso baseada em IP que use ou não um servidor de proxy.
- Para domínios de acesso VPC, use uma política de acesso aberto que use ou não use um servidor de proxy e grupos de segurança para controlar o acesso. Para saber mais, consulte the section called “Sobre políticas de acesso em domínios da VPC”.

Usando um proxy para acessar o OpenSearch serviço a partir de painéis

> **Note**
>
> Esse processo só será aplicável se o domínio usar acesso público e você não quiser usar a autenticação do Cognito. Consulte the section called “Controlar acesso ao Dashboards”.

Como o Dashboards é um JavaScript aplicativo, as solicitações se originam do endereço IP do usuário. O controle de acesso baseado em IP pode ser impraticável devido ao grande número de endereços IP que você precisaria inserir em uma lista de permissões para que cada usuário tivesse acesso ao Dashboards. Uma solução alternativa é colocar um servidor proxy entre os painéis e OpenSearch o serviço. Em seguida, você pode adicionar uma política de acesso com base em IP que permite solicitações de apenas um endereço IP, o do proxy. O diagrama a seguir mostra essa configuração.



1. Esse é o seu domínio OpenSearch de serviço. O IAM fornece acesso autorizado para este domínio. Uma política de acesso adicional com base em IP fornece acesso ao servidor de proxy.
2. Este é o servidor de proxy, em execução em uma instância do Amazon EC2.
3. Outros aplicativos podem usar o processo de assinatura Signature Version 4 para enviar solicitações autenticadas ao OpenSearch Serviço.
4. Os clientes do Dashboards se conectam ao seu domínio OpenSearch de serviço por meio do proxy.

Para habilitar esse tipo de configuração, você precisa de uma política com base em recursos que especifica funções e endereços IP. Aqui está um exemplo de política:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*",
         "Principal":{
            "AWS":"arn:aws:iam::111111111111:role/allowedrole1"
         },
         "Action":[
            "es:ESHttpGet"
         ],
         "Effect":"Allow"
      },
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":"*"
         },
         "Action":"es:*",
         "Condition":{
            "IpAddress":{
               "aws:SourceIp":[
                  "203.0.113.0/24",
                  "2001:DB8:1234:5678::/64"
               ]
            }
         },
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*"
      }
   ]
}
```

Recomendamos configurar a instância do EC2 executando o servidor de proxy com um endereço IP elástico. Dessa forma, você pode substituir a instância quando necessário e ainda anexar o mesmo endereço IP público. Para saber mais, consulte Endereços IP elásticos no Guia do usuário do Amazon EC2.

Se você usar um servidor de proxy e a autenticação do Cognito, talvez seja necessário adicionar configurações do Dashboards e do Amazon Cognito para evitar erros de `redirect_mismatch`. Veja o exemplo `nginx.conf` a seguir:

```
server {
    listen 443;
    server_name $host;
    rewrite ^/$ https://$host/_plugin/_dashboards redirect;

    ssl_certificate          /etc/nginx/cert.crt;
    ssl_certificate_key      /etc/nginx/cert.key;

    ssl on;
    ssl_session_cache  builtin:1000  shared:SSL:10m;
    ssl_protocols  TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;

    location /_plugin/_dashboards {
        # Forward requests to Dashboards
        proxy_pass https://$dashboards_host/_plugin/_dashboards;

        # Handle redirects to Cognito
        proxy_redirect https://$cognito_host https://$host;

        # Update cookie domain and path
        proxy_cookie_domain $dashboards_host $host;
        proxy_cookie_path / /_plugin/_dashboards/;

        # Response buffer settings
        proxy_buffer_size 128k;
        proxy_buffers 4 256k;
        proxy_busy_buffers_size 256k;
    }

    location ~ \/(log|sign|fav|forgot|change|saml|oauth2) {
        # Forward requests to Cognito
        proxy_pass https://$cognito_host;

        # Handle redirects to Dashboards
        proxy_redirect https://$dashboards_host https://$host;

        # Update cookie domain
```

```
        proxy_cookie_domain $cognito_host $host;
    }
}
```

> **Note**
>
> (Opcional) Se você optar por provisionar um nó de coordenador dedicado, ele começará automaticamente a hospedar OpenSearch painéis. Consequentemente, a disponibilidade dos recursos dos nós de dados, como CPU e memória, aumenta. Essa maior disponibilidade dos recursos dos nós de dados pode ajudar a melhorar a resiliência geral do seu domínio.

Configurar o OpenSearch Dashboards para usar um servidor de mapas WMS

A instalação padrão do Dashboards for OpenSearch Service inclui um serviço de mapas, exceto para domínios nas regiões da Índia e da China. O serviço de mapa oferece suporte a até 10 níveis de zoom.

Independentemente da sua região, é possível configurar o Dashboards para usar um servidor diferente do Web Map Service (WMS) para coordenar visualizações de mapas. As visualizações de mapa de região oferecem suporte apenas ao serviço de mapa padrão.

Para configurar o Dashboards para usar um servidor de mapas WMS:

1. Abra o Dashboards.
2. Escolha Stack Management (Gerenciamento de pilhas).
3. Escolha Advanced Settings.
4. Localize Visualização:Tilemap:. WMSdefaults
5. Altere `enabled` para `true` e `url` para o URL de um servidor de mapas WMS válido:

```
{
  "enabled": true,
  "url": "wms-server-url",
  "options": {
    "format": "image/png",
    "transparent": true
  }
```

```
}
```

6. Escolha Salvar alterações.

Para aplicar o novo valor padrão a visualizações, talvez seja necessário recarregar o Dashboards. Se você salvou as visualizações, selecione Opções depois de abrir a visualização. Verifique se o Servidor de mapas WMS está habilitado e se o URL do WMS contém o servidor de mapas de sua preferência e selecione Aplicar alterações.

> **Note**
>
> Os serviços de mapa costumam ter taxas ou restrições de licenciamento. Você será responsável por todos esses fatores em qualquer servidor de mapas que especificar. Você pode encontrar os serviços de mapa em U.S. Geological Survey, útil para testes.

Conectando um servidor local de painéis ao serviço OpenSearch

Se você já investiu um tempo significativo na configuração de sua própria instância de Dashboards, você pode usá-la em vez (ou além) da instância de Dashboards padrão que OpenSearch o Service fornece. O procedimento a seguir destina-se a domínios que usam o controle de acesso refinado com uma política de acesso aberto.

Para conectar um servidor local do Dashboards ao Serviço OpenSearch

1. No seu domínio OpenSearch de serviço, crie um usuário com as permissões apropriadas:
   a. No Dashboards, vá para Segurança, Usuários internos e escolha Criar usuário interno.
   b. Forneça um nome de usuário e uma senha e escolha Create (Criar).
   c. Vá para Roles (Funções) e selecione uma função.
   d. Selecione Mapped users (Usuários mapeados) e escolha Manage mapping (Gerenciar mapeamento).
   e. Em Users (Usuários), adicione seu nome de usuário e escolha Map (Mapa).
2. Baixe e instale a versão apropriada do plug-in de OpenSearch segurança em sua instalação autogerenciada do Dashboards OSS.

3. No servidor local do Dashboards, abra o `config/opensearch_dashboards.yml` arquivo e adicione seu endpoint de OpenSearch serviço com o nome de usuário e a senha que você criou anteriormente:

```
opensearch.hosts: ['https://domain-endpoint']
opensearch.username: 'username'
opensearch.password: 'password'
```

Você pode usar seguinte arquivo `opensearch_dashboards.yml` de exemplo:

```
server.host: '0.0.0.0'

opensearch.hosts: ['https://domain-endpoint']

opensearchDashboards.index: ".username"

opensearch.ssl.verificationMode: none # if not using HTTPS

opensearch_security.auth.type: basicauth
opensearch_security.auth.anonymous_auth_enabled: false
opensearch_security.cookie.secure: false # set to true when using HTTPS
opensearch_security.cookie.ttl: 3600000
opensearch_security.session.ttl: 3600000
opensearch_security.session.keepalive: false
opensearch_security.multitenancy.enabled: false
opensearch_security.readonly_mode.roles: ['opensearch_dashboards_read_only']
opensearch_security.auth.unauthenticated_routes: []
opensearch_security.basicauth.login.title: 'Please log in using your username and
 password'

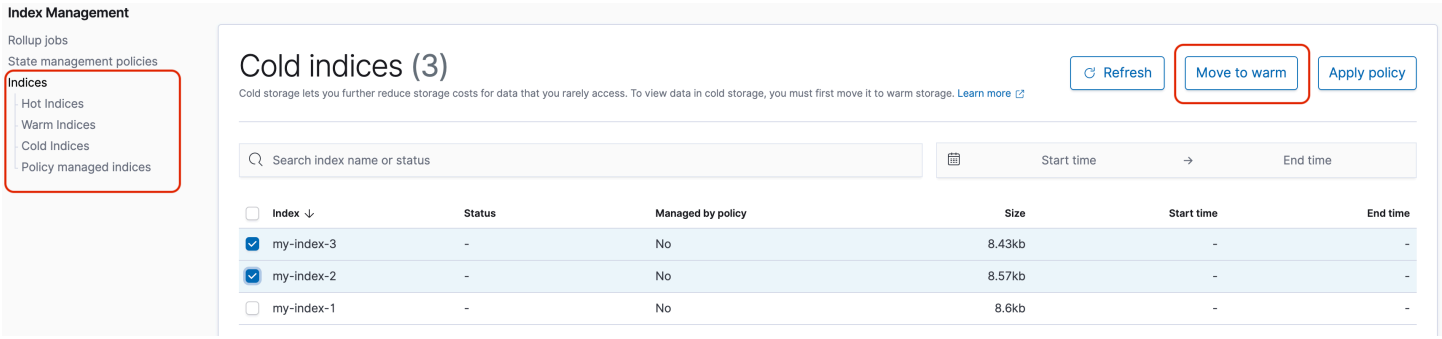
opensearch.username: 'username'
opensearch.password: 'password'
opensearch.requestHeadersWhitelist: [authorization, securitytenant,
 security_tenant]
```

Para ver seus índices OpenSearch de serviço, inicie seu servidor local de painéis, acesse Dev Tools e execute o seguinte comando:

```
GET _cat/indices
```

Gerenciar índices no Dashboards

A instalação do Dashboards em seu domínio OpenSearch de serviço fornece uma interface de usuário útil para gerenciar índices em diferentes níveis de armazenamento em seu domínio. Escolha Gerenciamento de índices no menu principal dos painéis para visualizar todos os índices em armazenamento quente e frio UltraWarm, bem como os índices gerenciados pelas políticas do Index State Management (ISM). Use o gerenciamento de índices para mover índices entre os armazenamentos mornos e frios, e para monitorar migrações entre os três níveis.



Observe que você não verá as opções de índice de quente, quente e frio, a menos que tenha o armazenamento a frio UltraWarm e/ou o armazenamento a frio ativado.

Recursos adicionais

A instalação padrão dos painéis em cada domínio do OpenSearch serviço tem alguns recursos adicionais:

- Interfaces de usuário para os vários OpenSearchplug-ins
- Locatários
- Relatórios

  Use o menu Reports (Relatórios) para gerar relatórios CSV sob demanda na página Discover (Descobrir) e relatórios PDF ou PNG de painéis ou visualizações. Os relatórios CSV têm um limite de 10.000 linhas.
- Gráficos de Gantt
- Cadernos

Usando a OpenSearch interface do usuário no Amazon OpenSearch Service

OpenSearch A UI (interface de usuário) é uma experiência de análise operacional modernizada para o Amazon OpenSearch Service que fornece uma visão unificada para você interagir com dados em várias fontes. Ao contrário dos OpenSearch painéis, que funcionam apenas com um domínio ou coleção que os hospeda, a OpenSearch interface do usuário é hospedada no Nuvem AWS. Isso possibilita que a OpenSearch interface do usuário alcance alta disponibilidade e permaneça funcional durante as atualizações do cluster, além de se conectar de forma nativa a várias fontes de dados. Para obter informações sobre OpenSearch painéis, consulte OpenSearch Painéis.

A seguir estão os principais recursos da OpenSearch interface do usuário:

- Suporte a várias fontes de dados — a OpenSearch interface do usuário pode se conectar a várias fontes de dados para criar uma visão abrangente. Isso inclui OpenSearch domínios e coleções sem servidor, bem como fontes de AWS dados integradas, como Amazon, Amazon Security Lake e CloudWatch Amazon Simple Storage Service (Amazon S3).
- Tempo de inatividade zero durante as atualizações — a OpenSearch interface do usuário está hospedada no. Nuvem AWS Isso significa que OpenSearch permanece operacional e pode recuperar dados de clusters durante os processos de atualização.
- Espaços de trabalho: os espaços selecionados para colaborações de equipe em vários fluxos de trabalho, como Observabilidade, Analytics de segurança e Pesquisa. Você pode definir as configurações de privacidade e gerenciar as permissões dos colaboradores em seu espaço de trabalho.
- Logon único — A OpenSearch interface funciona com Centro de Identidade do AWS IAM e via SAML AWS Identity and Access Management (IAM) federação para se integrar com seus provedores de identidade e criar uma experiência de login único para seus usuários finais.
- Análise baseada em Genai — A OpenSearch UI oferece suporte à geração de consultas em linguagem natural para ajudar a gerar as consultas certas para sua análise. OpenSearch A UI também trabalha com o Amazon Q Developer para fornecer o chat do Amazon Q e ajudar a gerar visualizações, resumo de alertas, insights e detectores de anomalias recomendados.
- Suporte a várias linguagens de consulta — A OpenSearch interface do usuário suporta Piped Processing Language (PPL), SQL, Lucene e Dashboards Query Language (DQL).

- Suporte entre regiões e contas — a OpenSearch interface do usuário pode utilizar o recurso de pesquisa entre clusters para se conectar com OpenSearch domínios em contas e regiões diferentes para análises e visualizações agregadas.

Para começar e criar sua primeira OpenSearch interface de usuário, siga as instruções emthe section called “Introdução”.

Para obter informações sobre os recursos mais recentes lançados para a OpenSearch interface do usuário, consultethe section called “Histórico de versões”.

Tópicos

- Histórico de lançamento da interface de usuário do Amazon OpenSearch Service
- Introdução à interface do OpenSearch usuário no Amazon OpenSearch Service
- IA agente no Amazon Service OpenSearch
- Criptografando metadados de aplicativos de OpenSearch interface de usuário com chaves gerenciadas pelo cliente
- Habilitando a federação SAML com AWS Identity and Access Management
- Gerenciar associações de fontes de dados e permissões de acesso à nuvem privada virtual
- Usando espaços de trabalho OpenSearch do Amazon Service
- Acesso a dados entre regiões e contas
- Gerenciando o acesso à OpenSearch interface do usuário a partir de um VPC endpoint
- Migração de objetos salvos dos OpenSearch painéis para a interface do usuário OpenSearch
- OpenSearch Endpoints e cotas de interface do usuário

Histórico de lançamento da interface de usuário do Amazon OpenSearch Service

A tabela a seguir lista todas as versões do suporte da Amazon OpenSearch Service para OpenSearch interface de usuário e os recursos e aprimoramentos incluídos em cada versão.

Alteração	Data de lançamento	Description
Novo atributo	2025-03-10	OpenSearch A interface do usuário agora oferece suporte ao acesso de dados entre contas aos OpenSearch domínios. Você pode configurar aplicativos em uma conta para acessar domínios em contas diferentes usando IAM ou Centro de Identidade do AWS IAM autenticação, tanto para domínios públicos quanto para VPC. Para obter mais informações, consulte the section called “Acesso a dados entre contas a OpenSearch domínios”.
Novo atributo	2025-12-29	OpenSearch A interface do usuário agora oferece suporte a chaves gerenciadas pelo cliente (CMK) para criptografar metadados do aplicativo. Esse recurso ajuda você a atender aos requisitos regulatórios e de conformidade, oferecendo controle total sobre as chaves de criptografia. Esse lançamento também aumenta o limite de tamanho de metadados para objetos salvos na OpenSearch interface do usuário, permitindo que você crie e armazene consultas complexas, visualizações extensivas e espaços de trabalho em grande escala. Para obter mais informações, consulte the section called “Criptografia e chave gerenciada pelo cliente”.
Novo atributo	2025-04-16	OpenSearch A interface do usuário agora funciona com a pesquisa entre clusters. Isso possibilita que você use a OpenSearch interface de usuário em uma Região da AWS para acessar clusters em uma região diferente. Isso é feito configurando-o como um cluster remoto conectado a um cluster na mesma região. Para saber mais, consulte the section called “Pesquisa entre clusters”.
Novo atributo	2025-03-31	O Amazon Q Developer agora está disponível ao público em geral no Amazon OpenSearch Service. Para mais informações, consulte Assistente de IA.

Alteração	Data de lançamento	Description
Novo atributo	2025-02-05	A federação Security Assertion Markup Language 2.0) (SAML) por meio de AWS Identity and Access Management (IAM) agora funciona com a interface do usuário. OpenSearch Isso possibilita a criação de uma experiência de login único (SSO) iniciada pelo provedor de identidades para os usuários finais. Para saber mais, consulte the section called “Habilitar federação SAML com o IAM”.
Nova integração	2024-12-01	A integração Zero-ETL com a Amazon CloudWatch simplifica a análise e a visualização de dados de log, reduzindo a sobrecarga técnica e os custos operacionais. Para obter mais informações, consulte New Amazon CloudWatch and Amazon OpenSearch Service lançam uma experiência de análise integrada no blog de AWS notícias.
Nova integração	2024-12-01	A integração ETL zero com o Amazon Security Lake possibilita que as organizações pesquisem, analisem e obtenham, com eficiência, insights acionáveis de seus dados de segurança. Para obter mais informações, consulte Apresentando a integração entre o Amazon OpenSearch Service e o Amazon Security Lake para simplificar a análise de segurança no blog de AWS notícias.
Lançamento inicial	2024-11-07	O lançamento público inicial da OpenSearch UI. Para obter mais informações, consulte Amazon OpenSearch Service lança a OpenSearch interface de usuário de próxima geração no blog de AWS Big Data.

Introdução à interface do OpenSearch usuário no Amazon OpenSearch Service

No Amazon OpenSearch Service, um aplicativo é uma instância da interface do OpenSearch usuário (OpenSearch UI). Cada aplicação pode ser associado a várias fontes de dados, e uma única

fonte pode ser associada a vários aplicações. Você pode criar várias aplicações para diferentes administradores usando diferentes opções de autenticação compatíveis.

Use as informações deste tópico para orientá-lo no processo de criação de um aplicativo de OpenSearch interface de usuário usando o Console de gerenciamento da AWS ou AWS CLI o.

Tópicos

- Permissões necessárias para criar aplicativos do Amazon OpenSearch Service
- Criação de um aplicativo de OpenSearch interface do usuário
- Gerenciar administradores de aplicação

Permissões necessárias para criar aplicativos do Amazon OpenSearch Service

Antes de criar uma aplicação, verifique se você recebeu as permissões necessárias para a tarefa. Entre em contato com um administrador da conta para obter ajuda, se necessário.

Permissões gerais

Para trabalhar com aplicativos no OpenSearch Serviço, você precisa das permissões mostradas na política a seguir. As permissões atendem às seguintes finalidades:

- As cinco permissões de `es:*Application` são necessárias para criar e gerenciar uma aplicação.
- As três permissões de `es:*Tags` são necessárias para adicionar, listar e remover tags da aplicação.
- As permissões de `aoss:BatchGetCollection`, `es:DescribeDomain` e `es:GetDirectQueryDataSource` são necessárias para associar fontes de dados.
- As permissões `opensearch:*DirectQuery*` `aoss:APIAccessAll`, `es:ESHttp*` e 4 são necessárias para associar fontes de dados.
- Isso `iam:CreateServiceLinkedRole` fornece permissão ao Amazon OpenSearch Service para criar uma função vinculada ao serviço (SLR) em sua conta. Essa função é usada e possibilita que o aplicativo de OpenSearch interface de usuário publique CloudWatch métricas da Amazon em sua conta. Para saber mais, consulte the section called “Permissões” no tópico Uso de perfis vinculados ao serviço para criar domínios de VPC e fontes de dados de consulta direta.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:APIAccessAll",
                "es:ESHttp*",
                "opensearch:StartDirectQuery",
                "opensearch:GetDirectQuery",
                "opensearch:CancelDirectQuery",
                "opensearch:GetDirectQueryResult",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        }
    ]
}
```

Permissões para criar uma aplicação que usa a autenticação do IAM Identity Center (opcional)

Por padrão, os aplicativos de painel são autenticados usando AWS Identity and Access Management (IAM) para gerenciar permissões para usuários AWS de recursos. No entanto, você pode optar por fornecer uma experiência de login único usando o IAM Identity Center, que permite usar seus provedores de identidade existentes para fazer login em aplicativos de OpenSearch interface do usuário. Nesse caso, você selecionará a opção Autenticação com o IAM Identity Center no procedimento mais adiante neste tópico e, em seguida, concederá aos usuários do IAM Identity Center as permissões necessárias para acessar o aplicativo de OpenSearch interface do usuário.)

Para criar uma aplicação que use a autenticação do IAM Identity Center, você precisará das permissões a seguir. Substitua *`placeholder values`* por suas próprias informações. Entre em contato com um administrador da conta para obter ajuda, se necessário.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "IDCPermissions",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources",
                "sso:CreateApplication",
                "sso:DeleteApplication",
```

```
                "sso:PutApplicationGrant",
                "sso:PutApplicationAccessScope",
                "sso:PutApplicationAuthenticationMethod",
                "sso:ListInstances",
                "sso:DescribeApplicationAssignment",
                "sso:DescribeApplication",
                "sso:CreateApplicationAssignment",
                "sso:ListApplicationAssignments",
                "sso:DeleteApplicationAssignment",
                "sso-directory:SearchGroups",
                "sso-directory:SearchUsers",
                "sso:ListDirectoryAssociations",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup",
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SLRPermission",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        },
        {
            "Sid": "PassRolePermission",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/iam-role-for-identity-
center"
        }
    ]
}
```

Criação de um aplicativo de OpenSearch interface do usuário

Crie uma aplicação que especifique o nome da aplicação, o método de autenticação e os administradores usando um dos procedimentos a seguir.

Tópicos

- Criação de um aplicativo de OpenSearch interface de usuário que usa autenticação do IAM no console
- Criação de um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação no console
- Criação de um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação usando o AWS CLI

Criação de um aplicativo de OpenSearch interface de usuário que usa autenticação do IAM no console

Para criar um aplicativo de OpenSearch interface do usuário que usa a autenticação do IAM no console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação esquerdo, escolha OpenSearch UI (painéis).
3. Selecione Criar aplicativo.
4. Em Nome da aplicação, insira um nome para a aplicação.
5. Não marque a caixa de seleção Autenticação com o IAM Identity Center. Para obter informações sobre como criar um aplicativo com autenticação por meio de Centro de Identidade do AWS IAM, consulte the section called “Criação de um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação no console” mais adiante neste tópico.
6. (Opcional) Você é automaticamente adicionado como administrador da aplicação sendo criada. Na área de gerenciamento de administradores do OpenSearch aplicativo, você pode conceder permissões de administrador a outros usuários.

> **Note**
>
> A função de administrador do aplicativo de OpenSearch interface de usuário concede permissões para editar e excluir um aplicativo de OpenSearch interface do usuário. Os administradores de aplicativos também podem criar, editar e excluir espaços de trabalho em um aplicativo de OpenSearch interface de usuário.

Para conceder permissões de administrador a outros usuários, selecione uma das opções a seguir:

- Conceder permissão do administrador a usuários específicos — No campo administradores do OpenSearch aplicativo, na lista pop-up Propriedades, selecione usuários do IAM ou

  Centro de Identidade do AWS IAM usuários e, em seguida, escolha os usuários individuais aos quais conceder permissões de administrador.
- Conceder permissão de administrador a todos os usuários: todos os usuários da sua organização ou conta recebem permissões de administrador.

7. (Opcional) Defina as configurações de criptografia. Por padrão, os metadados da OpenSearch interface do usuário são criptografados com chaves AWS próprias. Para usar sua própria chave gerenciada pelo cliente (CMK) para criptografia, consultethe section called “Criptografia e chave gerenciada pelo cliente”.
8. (Opcional) Na área Tags, aplique um ou mais name/value pares de chaves de tag ao aplicativo.

   Tags são metadados opcionais que você atribui a um recurso. As tags permitem categorizar um recurso de diferentes formas, como por finalidade, proprietário ou ambiente.
9. Escolha Criar.

Criação de um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação no console

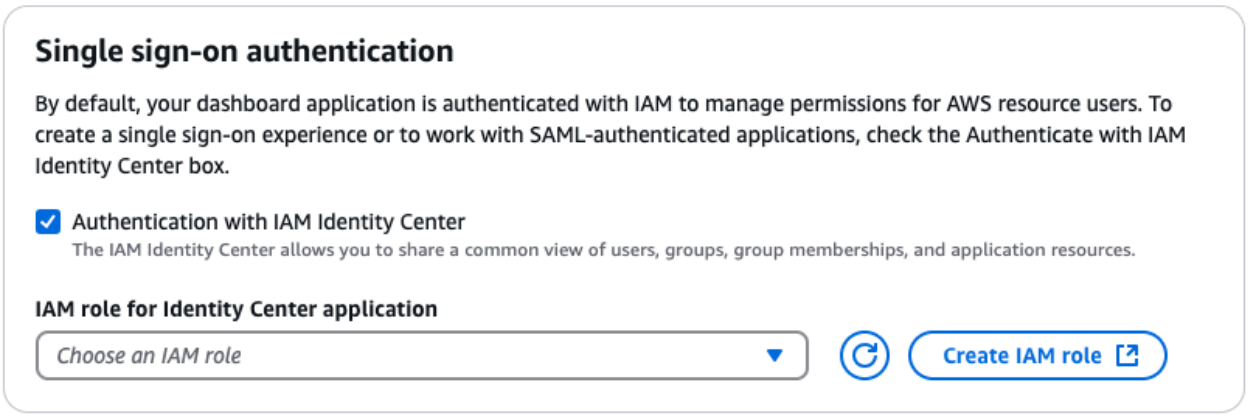
Para criar um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação, você deve ter as permissões do IAM descritas anteriormente neste tópico emthe section called “Permissões para criar uma aplicação que usa a autenticação do IAM Identity Center (opcional)”.

Para criar um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação no console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação esquerdo, escolha OpenSearch UI (painéis).
3. Selecione Criar aplicativo.

4. Em Nome da aplicação, insira um nome para a aplicação.
5. (Opcional) Para habilitar o login único para a sua organização ou conta, faça o seguinte:

   a. Marque a caixa de seleção Autenticação com o IAM Identity Center, conforme mostrado na seguinte imagem:



   b. Execute um destes procedimentos:

   - Na lista de aplicativos da função do IAM para o Identity Center, escolha uma função existente do IAM que forneça as permissões necessárias para que o IAM Identity Center acesse a OpenSearch interface do usuário e as fontes de dados associadas. Consulte as políticas no próximo tópico para ver as permissões que o perfil deve ter.
   - Crie um perfil do IAM com as permissões necessárias. Use os procedimentos a seguir no Guia do usuário do IAM com as opções especificadas para criar uma novo perfil e com a política de permissão e a política de confiança necessárias.
     - Procedimento: criar políticas do IAM (console)

       Ao seguir as etapas deste procedimento, cole a seguinte política no campo JSON do editor de políticas:

       JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "IdentityStoreOpenSearchDomainConnectivity",
            "Effect": "Allow",
            "Action": [
                "identitystore:DescribeUser",
```

```
                "identitystore:ListGroupMembershipsForMember",
                "identitystore:DescribeGroup"
            ],
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledViaLast": "es.amazonaws.com"
                }
            }
        },
        {
            "Sid": "OpenSearchDomain",
            "Effect": "Allow",
            "Action": [
                "es:ESHttp*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "OpenSearchServerless",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll"
            ],
            "Resource": "*"
        }
    ]
}
```

- Procedimento: criar um perfil usando políticas de confiança personalizadas

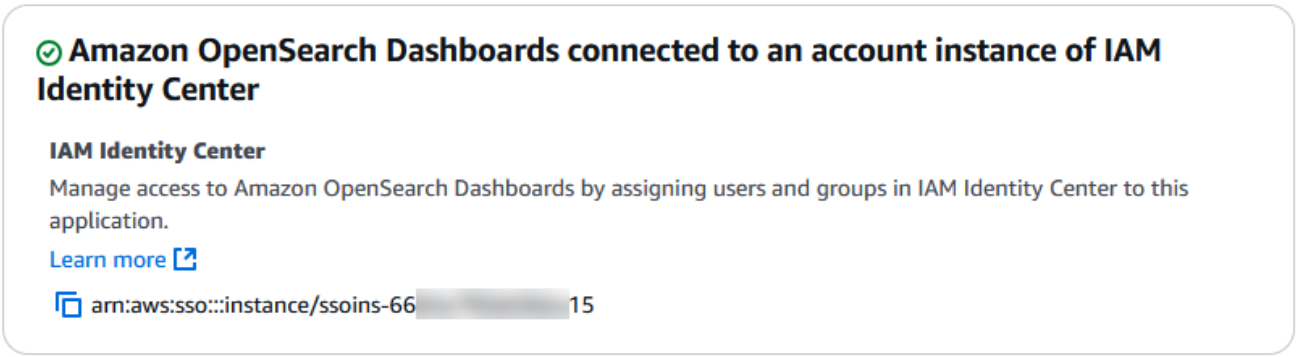
  Ao seguir as etapas desse procedimento, substitua o espaço reservado JSON na caixa Política de confiança personalizada pelo seguinte:

  > **Tip**
  >
  > Se você estiver adicionando a política de confiança a um perfil existente, adicione a política na guia Relação de confiança do perfil.

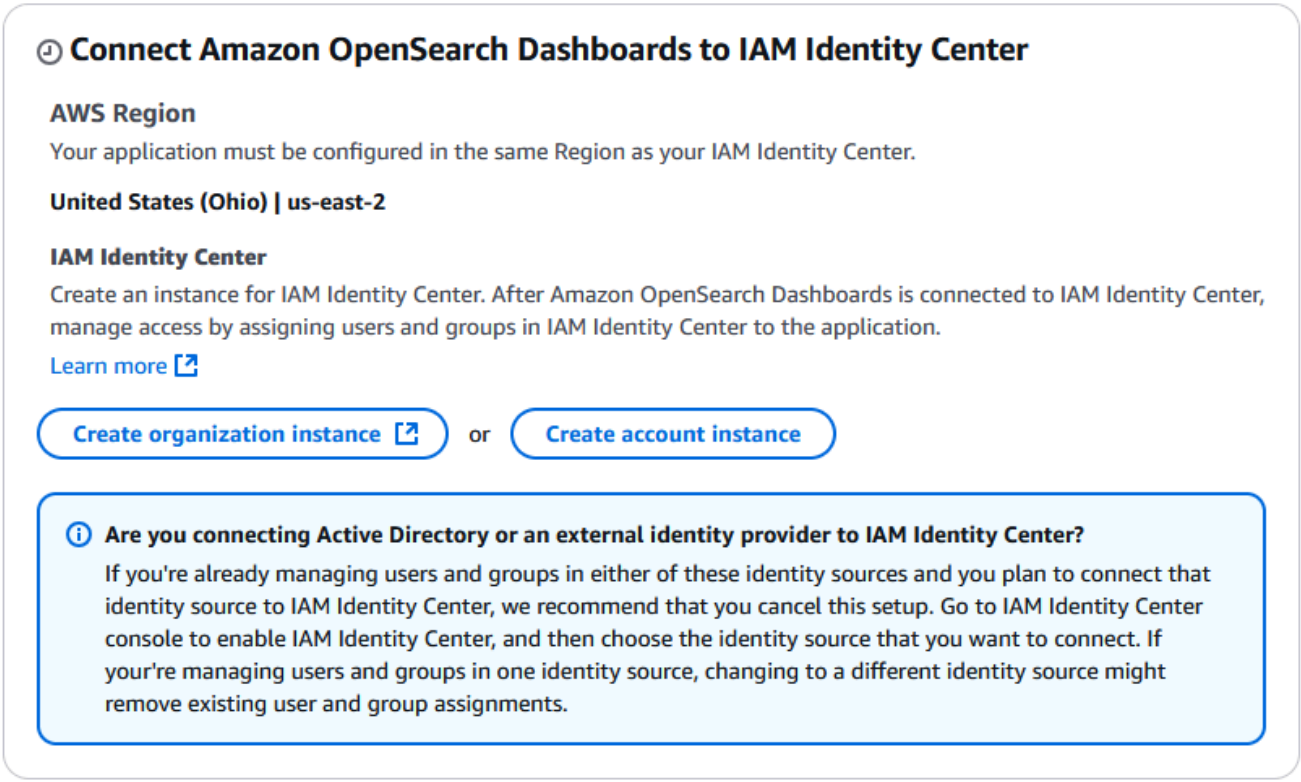
JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service":
 "application.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service":
 "application.opensearchservice.amazonaws.com"
            },
            "Action": "sts:SetContext",
            "Condition": {
                "ForAllValues:ArnEquals": {
                "sts:RequestContextProviders":
 "arn:aws:iam::123456789012:oidc-provider/portal.sso.us-
east-1.amazonaws.com/apl/application-id"
                }
            }
        }
    ]
}
```

c. Se uma instância do IAM Identity Center já tiver sido criada em sua organização ou conta, o console informa que o Amazon OpenSearch Dashboards já está conectado a uma instância organizacional do IAM Identity Center, conforme mostrado na imagem a seguir.



Se o IAM Identity Center ainda não estiver disponível em sua organização ou conta, você ou um administrador com as permissões necessárias poderão criar uma instância da organização ou instância da conta. A área Connect Amazon OpenSearch Dashboards to IAM Identity Center fornece opções para ambos, conforme mostrado na imagem a seguir:

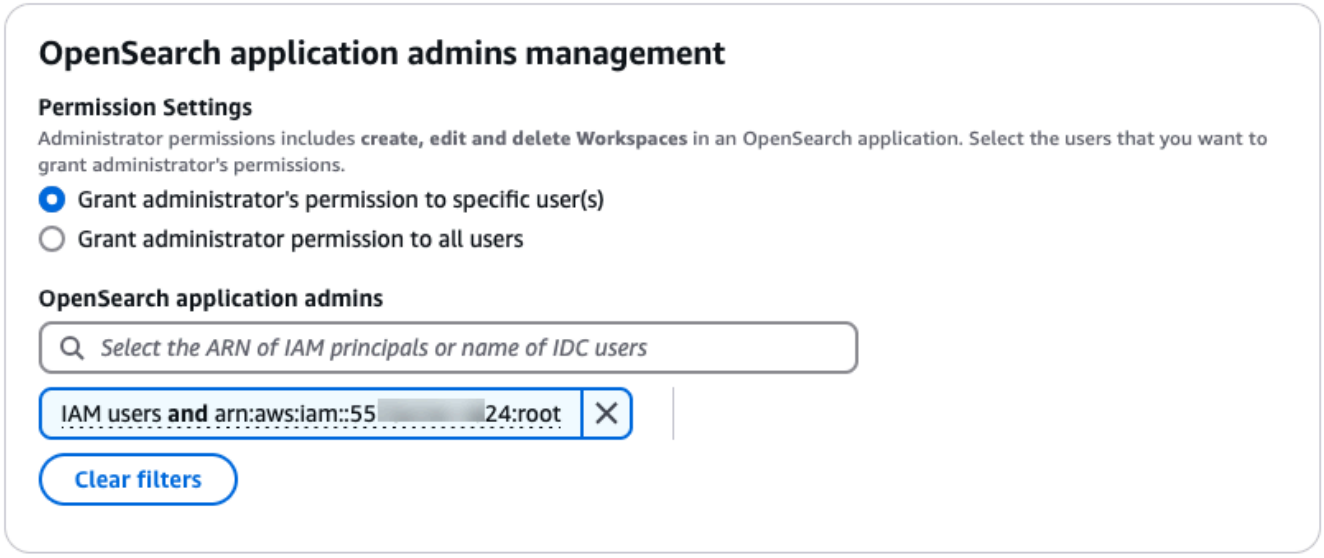


Nesse caso, você pode criar uma instância de conta no IAM Identity Center para teste ou solicitar que um administrador crie uma instância organizacional no IAM Identity Center. Para saber mais, consulte os seguintes tópicos no Manual do usuário do Centro de Identidade do AWS IAM :

> **Note**
>
> Se você quiser criar aplicativos de OpenSearch interface de usuário em uma instância organizacional Região da AWS diferente da sua do IAM Identity Center, consulte Como usar o IAM Identity Center em vários Regiões da AWS.

- Instâncias organizacionais do IAM Identity Center
- Instâncias de conta do Centro de Identidade do IAM
- Habilitar Centro de Identidade do AWS IAM

6. (Opcional) Você é automaticamente adicionado como administrador da aplicação sendo criada. Na área de gerenciamento de administradores do OpenSearch aplicativo, você pode conceder permissões de administrador a outros usuários, conforme mostrado na imagem a seguir:



> **Note**
>
> A função de administrador do aplicativo de OpenSearch interface de usuário concede permissões para editar e excluir um aplicativo de OpenSearch interface do usuário. Os administradores de aplicativos também podem criar, editar e excluir espaços de trabalho em um aplicativo de OpenSearch interface de usuário.

Para conceder permissões de administrador a outros usuários, selecione uma das opções a seguir:

- Conceder permissão do administrador a usuários específicos — No campo administradores do OpenSearch aplicativo, na lista pop-up Propriedades, selecione usuários do IAM ou

  Centro de Identidade do AWS IAM usuários e, em seguida, escolha os usuários individuais aos quais conceder permissões de administrador.
- Conceder permissão de administrador a todos os usuários: todos os usuários da sua organização ou conta recebem permissões de administrador.

7. (Opcional) Na área Tags, aplique um ou mais name/value pares de chaves de tag ao aplicativo.

   Tags são metadados opcionais que você atribui a um recurso. As tags permitem categorizar um recurso de diferentes formas, como por finalidade, proprietário ou ambiente.
8. Escolha Criar.

Criação de um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação usando o AWS CLI

Para criar um aplicativo de OpenSearch interface de usuário que usa Centro de Identidade do AWS IAM autenticação usando o AWS CLI, use o comando create-application com as seguintes opções:

- `--name`: o nome da aplicação.
- `--iam-identity-center-options`— (Opcional) A instância do IAM Identity Center e a função do IAM que OpenSearch serão usadas para autenticação e controle de acesso.

Substitua *`placeholder values`* por suas próprias informações.

```
aws opensearch create-application \
    --name application-name \
    --iam-identity-center-options "
          {
          \"enabled\":true,
          \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/sso-instance\",
          \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-
id:role/role-name\"
          }
```

```
"
```

Gerenciar administradores de aplicação

Um administrador de aplicativo de OpenSearch interface de usuário é uma função definida com permissão para editar e excluir um aplicativo de OpenSearch interface do usuário.

Por padrão, como criador de um aplicativo de OpenSearch interface de usuário, você é o primeiro administrador do aplicativo de OpenSearch interface do usuário.

Gerenciando administradores de OpenSearch interface do usuário usando o console

Você pode adicionar mais administradores a um aplicativo de OpenSearch interface de usuário no Console de gerenciamento da AWS, durante o fluxo de trabalho de criação do aplicativo ou na página Editar após a criação do aplicativo.

A função de administrador do aplicativo de OpenSearch interface de usuário concede permissões para editar e excluir um aplicativo de OpenSearch interface do usuário. Os administradores de aplicativos também podem criar, editar e excluir espaços de trabalho em um aplicativo de OpenSearch interface de usuário.

Em uma página de detalhes da aplicação, você pode pesquisar nome do recurso da Amazon (ARN) de uma entidade principal do IAM ou pesquisar o nome do usuário do IAM Identity Center.

Para gerenciar administradores de OpenSearch interface do usuário usando o console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação esquerdo, escolha OpenSearch UI (painéis).
3. Na área de OpenSearch aplicativos, escolha o nome de um aplicativo existente.
4. Selecione Edit (Editar).
5. Para conceder permissões de administrador a outros usuários, selecione uma das opções a seguir:

   - Conceder permissão do administrador a usuários específicos — No campo administradores do OpenSearch aplicativo, na lista pop-up Propriedades, selecione usuários do IAM ou

     Centro de Identidade do AWS IAM usuários e, em seguida, escolha os usuários individuais aos quais conceder permissões de administrador.

- Conceder permissão de administrador a todos os usuários: todos os usuários da sua organização ou conta recebem permissões de administrador.

6. Selecione Atualizar.

Você pode remover administradores adicionais, mas cada aplicativo de OpenSearch interface do usuário deve reter pelo menos um administrador.

Gerenciando administradores de OpenSearch interface do usuário usando o AWS CLI

Você pode criar e atualizar administradores de aplicativos de OpenSearch interface de usuário usando o. AWS CLI

Criação de administradores de OpenSearch interface do usuário usando o AWS CLI

Veja a seguir exemplos de como adicionar diretores do IAM e usuários do IAM Identity Center como administradores ao criar um aplicativo de OpenSearch interface de usuário.

Exemplo 1: criar um aplicativo de OpenSearch interface de usuário que adiciona um usuário do IAM como administrador

Execute o comando a seguir para criar um aplicativo de OpenSearch interface de usuário que adiciona um usuário do IAM como administrador. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch create-application \
    --name application-name \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Exemplo 2: Crie um aplicativo de OpenSearch interface de usuário que habilite o IAM Identity Center e adicione um ID de usuário do IAM Identity Center como administrador do aplicativo de OpenSearch interface do usuário

Execute o comando a seguir para criar um aplicativo de OpenSearch interface de usuário que habilite o IAM Identity Center e adicione um ID de usuário do IAM Identity Center como administrador do

aplicativo de OpenSearch interface do usuário. Substitua *placeholder values* por suas próprias informações.

`key`especifica o item de configuração a ser definido, como a função de administrador do aplicativo de OpenSearch interface do usuário. Os valores válidos são `opensearchDashboards.dashboardAdmin.users` e `opensearchDashboards.dashboardAdmin.groups`.

*xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx*representa o valor atribuído à chave, como o Amazon Resource Name (ARN) de um usuário do IAM.

```
aws opensearch create-application \
    --name myapplication \
    --iam-identity-center-options "
        {
        \"enabled\":true,
        \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/ssoins-instance-id\",
        \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-id:role/role-
name\"
        }
    " \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

Atualizando administradores de OpenSearch interface do usuário usando o AWS CLI

Veja a seguir exemplos de atualização dos diretores do IAM e dos usuários do IAM Identity Center designados como administradores de um aplicativo existente OpenSearch.

Exemplo 1: Adicionar um usuário do IAM como administrador de um OpenSearch aplicativo existente

Execute o comando a seguir para atualizar um aplicativo de OpenSearch interface de usuário para adicionar um usuário do IAM como administrador. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch update-application \
    --id myapplication \
    --app-configs "
```

```
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Exemplo 2: atualizar um aplicativo de OpenSearch interface do usuário para adicionar um ID de usuário do IAM Identity Center como administrador do aplicativo de OpenSearch interface do usuário

Execute o comando a seguir para atualizar um aplicativo de OpenSearch interface de usuário e adicionar um ID de usuário do IAM Identity Center como administrador do aplicativo de OpenSearch interface do usuário. Substitua *placeholder values* por suas próprias informações.

`key`especifica o item de configuração a ser definido, como a função de administrador do aplicativo de OpenSearch interface do usuário. Os valores válidos são `opensearchDashboards.dashboardAdmin.users` e `opensearchDashboards.dashboardAdmin.groups`.

*xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx*representa o valor atribuído à chave, como o Amazon Resource Name (ARN) de um usuário do IAM.

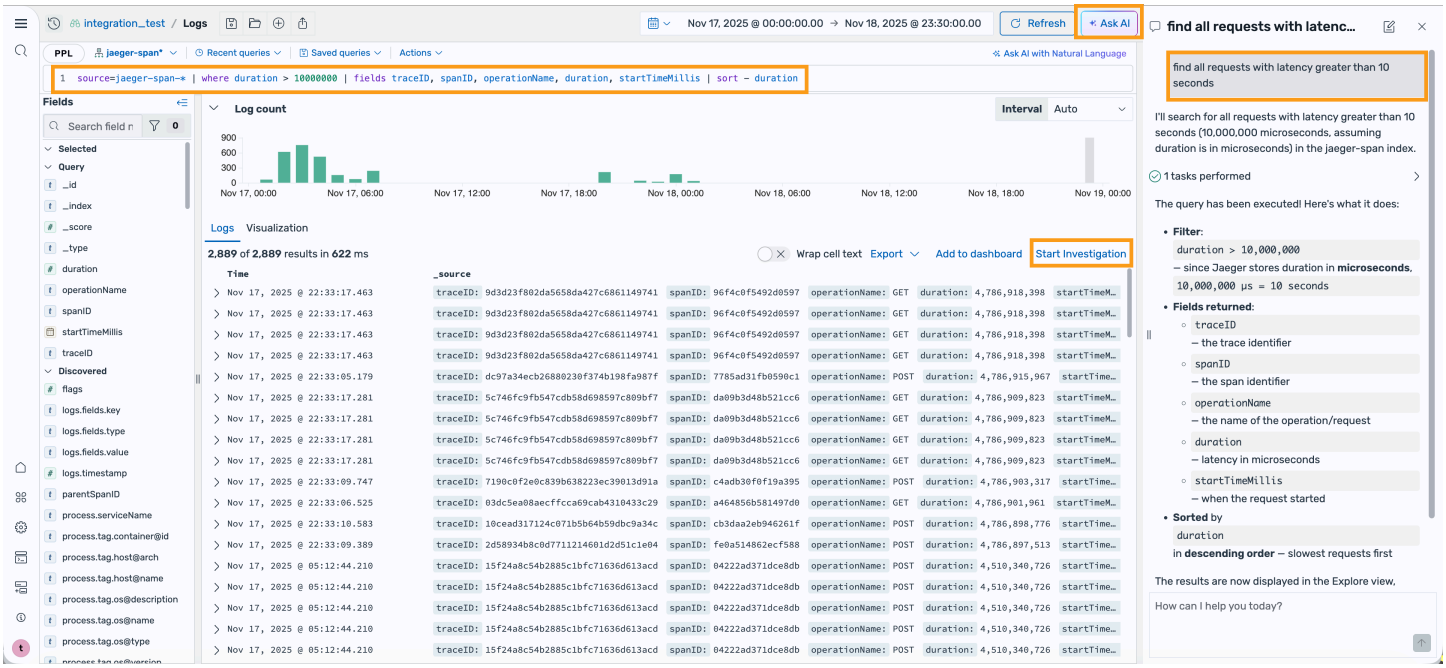
```
aws opensearch update-application \
    --id myapplication \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

IA agente no Amazon Service OpenSearch

OpenSearch A interface de usuário inclui um conjunto de recursos de IA agentes projetados para simplificar a análise operacional e acelerar as investigações de incidentes para suas equipes de engenharia e suporte.

Em cada página do seu aplicativo de OpenSearch interface de usuário, escolha o botão Perguntar à AI para abrir o Agentic Chat. O Agentic Chat entende o contexto da página que você está visualizando, usa ferramentas agentes para analisar os dados subjacentes, atualiza as consultas para exibir os resultados no Discover e responde às suas perguntas com clareza e precisão. Quando

uma análise complexa da causa raiz exige uma investigação mais profunda, você pode acionar o Agente de Investigação para planejar, executar e refletir de forma autônoma por meio de um fluxo de trabalho de várias etapas, fornecendo hipóteses estruturadas apoiadas por evidências de dados. Potencializando os dois recursos agentes está a Memória Agêntica, uma camada de memória que retém o contexto em sua conversa ou investigação, para que você possa continuar fazendo perguntas complementares com o mesmo contexto consistente em diferentes sessões da web.



Capacidades gerais

- Agentic Chat — incorporado ao recurso Ask AI em todas as páginas da OpenSearch interface do usuário, o Agentic Chat entende seu contexto atual e chama ferramentas para analisar dados para melhor responder às suas perguntas. No Discover, insira linguagem natural na interface de bate-papo para gerar e iterar consultas de Linguagem de Processamento Canalizado (PPL), para que você não precise ser um especialista em PPL para obter insights acionáveis. Você também pode iniciar conversas diretamente das visualizações, nas quais o Agentic Chat pode identificar anomalias em seus gráficos, correlacionar com os dados subjacentes e gerar análises. Quando uma análise complexa da causa raiz for necessária, use o comando `/investigate` slash para iniciar o Agente de Investigação com base na conversa em andamento ou com novas instruções.
- Agente de investigação — Um agente de pesquisa profunda orientado por objetivos que você ativa digitando `/investigate` no bate-papo ou escolhendo o botão Iniciar investigação nas páginas de recursos. Insira uma meta de investigação, como “encontrar a causa raiz desse pico de latência”. O Agente de Investigação planeja de forma autônoma usando seus dados e a meta

estabelecida, executa consultas e análises e reflete por meio de um fluxo de trabalho de várias etapas. Quando a investigação é concluída, normalmente em alguns minutos, ela gera hipóteses estruturadas classificadas por probabilidade, cada uma apoiada por evidências de dados. Ele fornece total transparência em cada etapa de seu raciocínio, para que você possa verificar e confiar nos resultados.

- Memória Agêntica — Uma camada de memória gerenciada por serviços que retém o contexto em sua conversa ou investigação, para que você possa continuar fazendo perguntas ou investigando dentro de um contexto consistente. O Agentic Memory permite que tanto o Agentic Chat quanto o Investigation Agent mantenham a continuidade durante uma sessão.

Limites de preço e uso

Os recursos de IA da agência são de uso gratuito. Os limites de uso baseados em tokens se aplicam por conta para evitar abusos.

Os recursos de IA da agência usam um modelo básico no Amazon Bedrock. Todas as configurações de segurança e governança de dados do Amazon Bedrock se aplicam aos recursos de IA da agência. Para obter mais informações, consulte Proteção de dados no Amazon Bedrock.

Segurança

Os recursos de IA agente operam dentro de sua estrutura de segurança existente e não ignoram nem substituem nenhum controle de acesso configurado. Quando você interage com o Agentic Chat ou inicia um Agente de Investigação, os recursos de IA são autenticados usando suas credenciais atuais do IAM ou do IAM Identity Center e herdam todas as permissões associadas. O assistente de IA só pode consultar fontes de dados, índices e visualizações aos quais você já tem acesso explícito por meio de suas políticas existentes de controle de acesso baseado em funções (RBAC).

Se seu OpenSearch domínio implementa um controle de acesso refinado com segurança em nível de campo ou documento, a IA respeita essas restrições e não pode revelar dados de campos ou documentos restritos que estejam fora do seu escopo de permissão. Da mesma forma, todas as consultas PPL geradas pelo Agentic Chat ou recuperações de dados realizadas durante as investigações são executadas com o contexto do usuário, garantindo que todas as políticas de segurança configuradas permaneçam totalmente aplicadas.

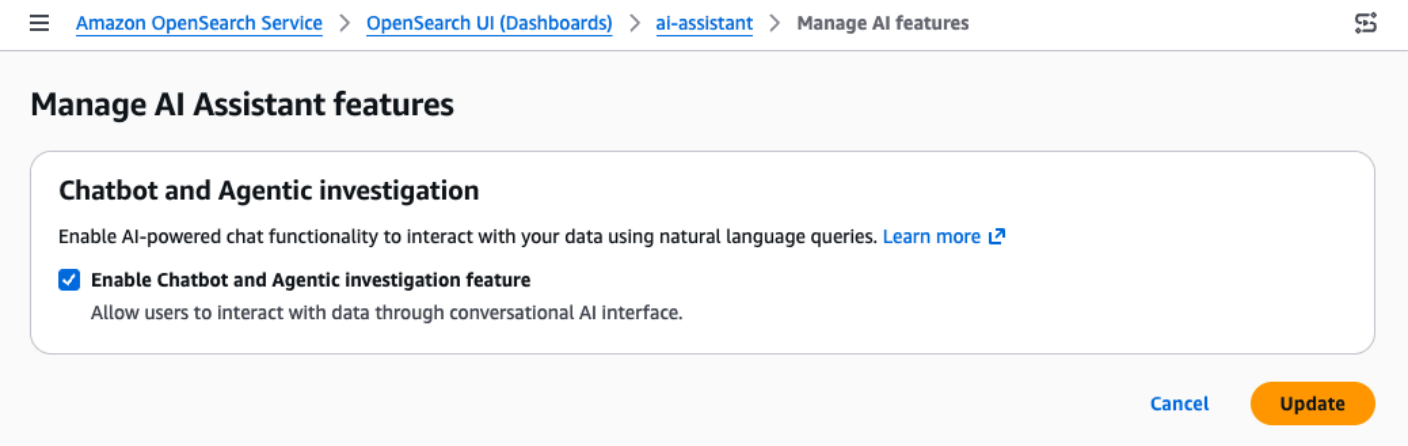
Esse modelo de herança de permissões garante que a ativação dos recursos de IA não crie novas vulnerabilidades de segurança nem exija que os administradores gerenciem estruturas de permissão separadas para interações de IA.

Ativando e desativando recursos de IA agentes

Os recursos de IA agente são habilitados automaticamente para novos aplicativos de OpenSearch interface do usuário e para aplicativos existentes, a menos que você tenha desativado explicitamente os recursos de IA em qualquer um dos seus OpenSearch domínios.

Para ativar ou desativar os recursos de IA, você pode usar a API Console de gerenciamento da AWS ou a:

- Console — Navegue até a página de detalhes do aplicativo de OpenSearch interface do usuário e atualize os recursos de IA a partir daí.



- API — Use a RegisterCapabilityAPI para ativar os recursos de IA ou use a DeregisterCapabilityAPI para desativá-los.

Disponibilidade de regiões

Os recursos de IA da agência estão disponíveis da seguinte Regiões da AWS forma:

- Ásia-Pacífico (Tóquio), ap-northeast-1
- Ásia-Pacífico (Sydney), ap-southeast-2
- Europa (Frankfurt), eu-central-1
- Europa (Estocolmo), eu-north-1
- Europa (Espanha), eu-south-2
- Europa (Irlanda), eu-west-1
- Leste dos EUA (Norte da Virgínia), us-east-1
- Leste dos EUA (Ohio), us-east-2

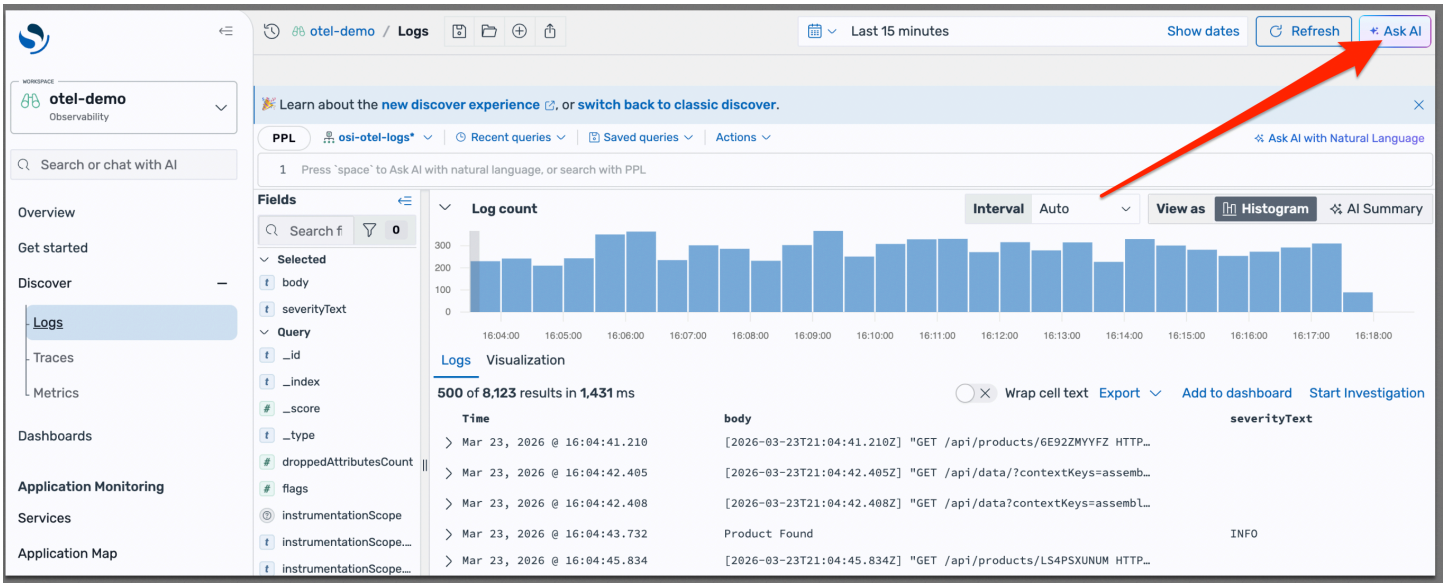
- Oeste dos EUA (Oregon), us-west-2

Tópicos

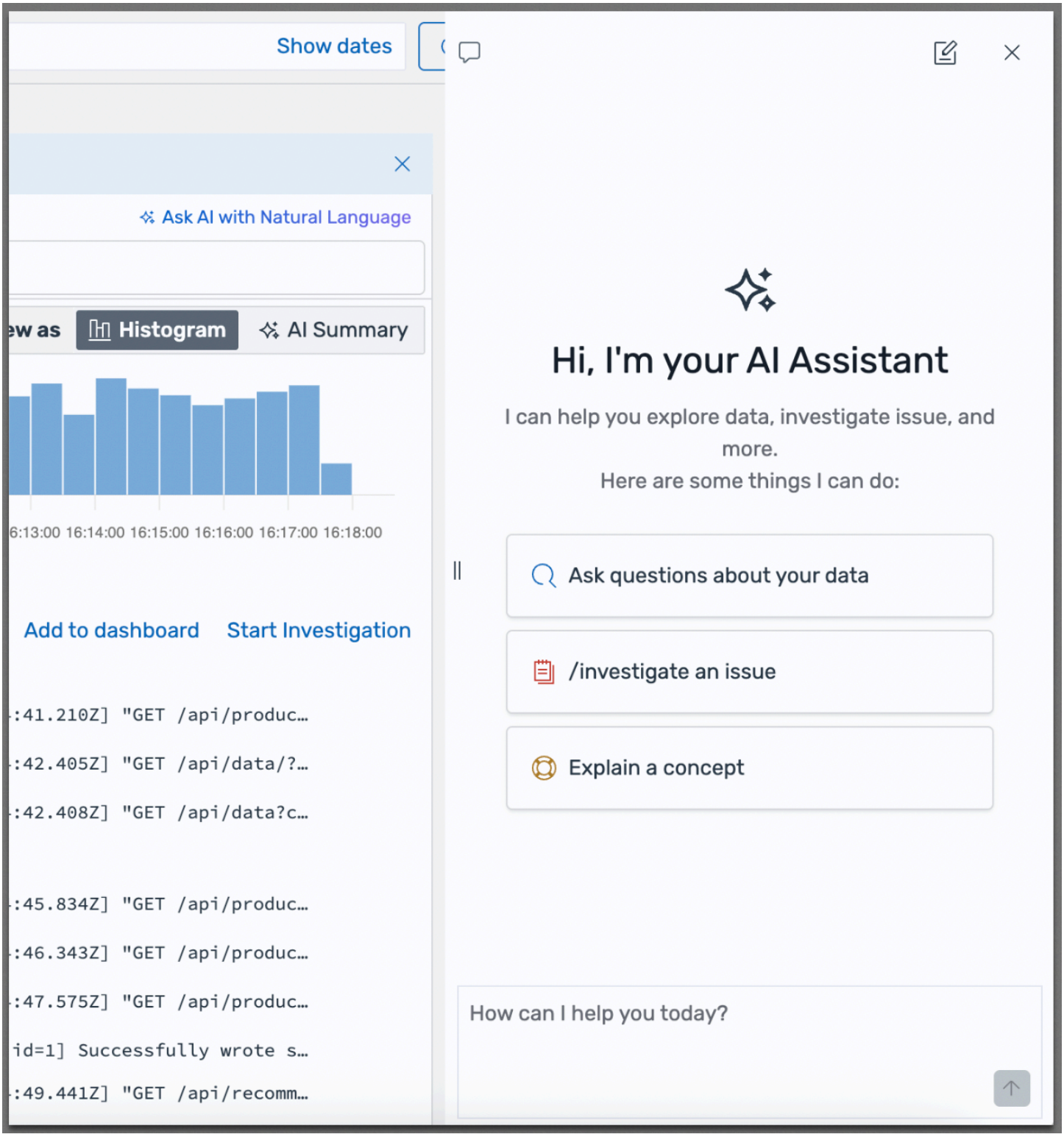
- Agentic Chat no Amazon Service OpenSearch
- Agente de investigação no Amazon OpenSearch Service
- Memória agente no Amazon Service OpenSearch

Agentic Chat no Amazon Service OpenSearch

O Agentic Chat é um assistente de IA incorporado em todas as páginas da OpenSearch interface do usuário. Escolha o botão Perguntar à IA para abrir o painel de bate-papo, onde você pode fazer perguntas sobre seus dados, gerar consultas e iniciar investigações. O Agentic Chat entende o contexto da página que você está visualizando no Discover and Investigation e usa ferramentas agentes para analisar os dados subjacentes.



Quando você abre o painel de bate-papo, o Agentic Chat apresenta opções para ajudar você a começar: faça perguntas sobre seus dados, investigue um problema ou explique um conceito. Se você iniciou uma conversa anteriormente, ela permanece visível no painel de bate-papo enquanto você navega entre as páginas, para que você possa continuar de onde parou. Como alternativa, escolha o botão Novo bate-papo no canto superior direito para iniciar uma nova conversa.



Usando o Agentic Chat com o Discover

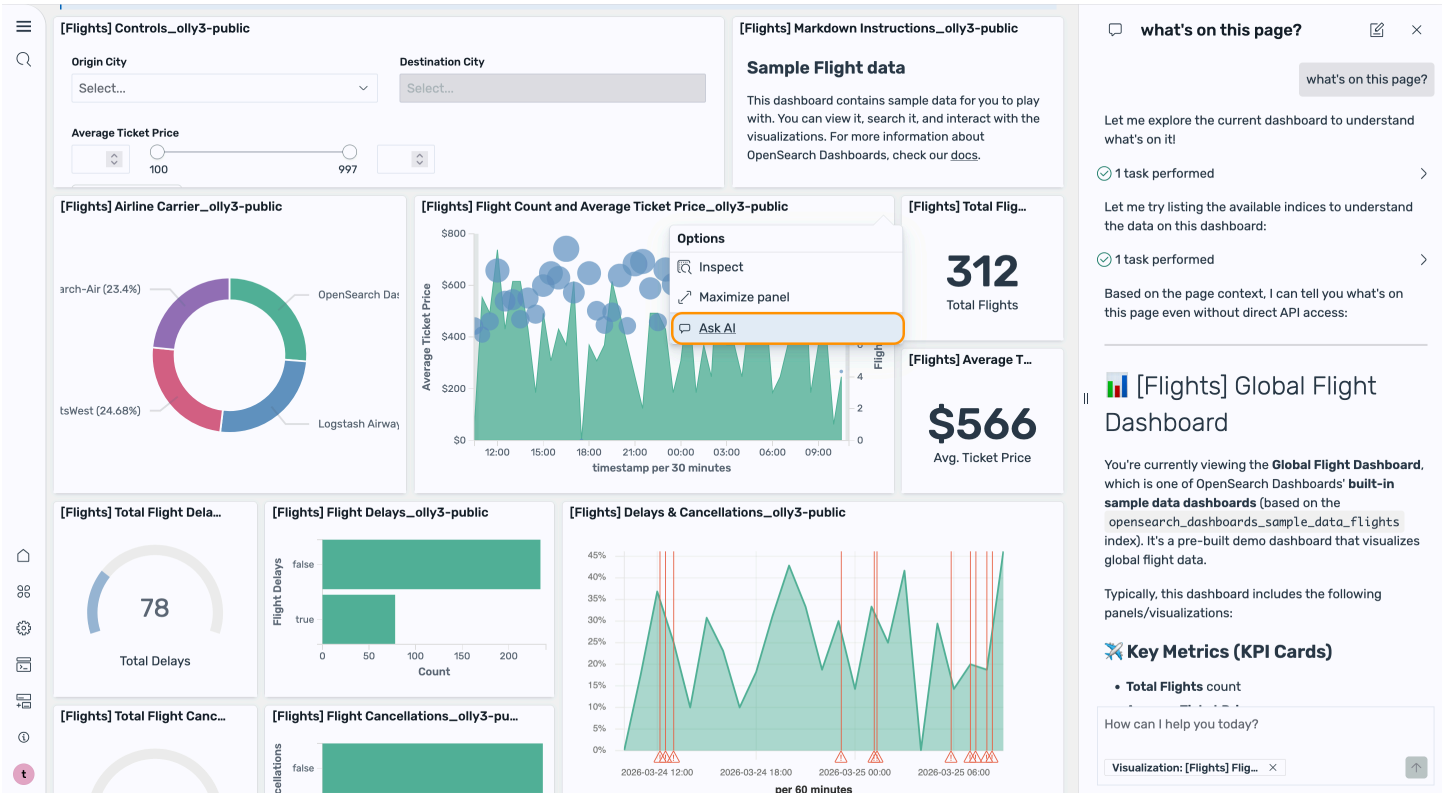
Na página Discover nos espaços de trabalho do Observability, você pode inserir linguagem natural na interface de bate-papo para gerar consultas PPL. O Agentic Chat traduz suas perguntas em PPL,

executa a consulta e exibe os resultados diretamente na visualização Discover. Você não precisa ser um especialista em PPL para obter insights acionáveis de seus dados.

Para refinar uma consulta gerada, faça perguntas complementares em linguagem natural, como “adicionar um filtro para o código de status 500". O Agentic Chat compreende o contexto da consulta atual e a modifica de acordo. Você também pode pedir para ajustar agregações, alterar intervalos de tempo ou adicionar campos adicionais aos resultados. Cada iteração atualiza a visualização Discover com os novos resultados da consulta.

Usando o Agentic Chat com visualizações

Você pode iniciar uma conversa com o Agentic Chat diretamente de uma visualização. Abra o menu de contexto em um painel de visualização e escolha Perguntar à IA. O Agentic Chat analisa a visualização, identifica anomalias em seus gráficos, correlaciona com os dados subjacentes e gera análises.



Iniciando uma investigação a partir do chat

Quando uma análise complexa da causa raiz é necessária, você pode iniciar o Agente de Investigação diretamente do Agentic Chat. Use o comando de `/investigate` barra na entrada do bate-papo ou escolha o botão Iniciar investigação nas páginas de recursos.

Para obter mais informações sobre o Agente de Investigação, consultethe section called “Agente de investigação”.

Ferramentas compatíveis

O Agentic Chat usa as seguintes ferramentas para analisar seus dados e responder perguntas. Para ver a maior up-to-date lista de ferramentas disponíveis, digite “quais ferramentas você pode usar” na interface de bate-papo.

Ferramentas de front-end

Essas ferramentas atualizam a OpenSearch interface do usuário:

- `create_investigation`— Cria um novo caderno de investigação de agentes com detalhes como metas, sintomas, índice e intervalo de tempo.
- `execute_ppl_query`— Executa consultas PPL no conjunto de dados atual e exibe os resultados na página Discover.
- `update_time_range`— Atualiza o filtro de intervalo de tempo global na página Discover atual (por exemplo, “últimas 24 horas” ou “semana passada”).

Ferramentas de back-end

Essas ferramentas interagem diretamente com OpenSearch os dados e APIs:

- `SearchIndexTool`— Pesquisa um índice usando consultas DSL.
- `MsearchTool`— Executa várias operações de pesquisa em uma única solicitação.
- `CountTool`— Retorna o número de documentos correspondentes a uma consulta.
- `ExplainTool`— Explica por que um documento corresponde ou não a uma consulta.
- `ListIndexTool`— Lista os índices no cluster com detalhes opcionais.
- `IndexMappingTool`— Recupera mapeamentos e configurações de índice.
- `GetShardsTool`— Obtém informações fragmentadas para um índice.
- `ClusterHealthTool`— Retorna informações de integridade do cluster.
- `LogPatternAnalysisTool`— Analisa padrões de registro, compara intervalos de tempo ou realiza análises de sequências de traços.
- `MetricChangeAnalysisTool`— Compara as distribuições percentuais dos campos numéricos entre dois intervalos de tempo.

- `DataDistributionTool`— Analisa as distribuições de valores de campo em um intervalo de tempo alvo, opcionalmente comparado a uma linha de base.
- `GenericOpenSearchApiTool`— Uma ferramenta flexível para chamar diretamente qualquer endpoint de OpenSearch API.

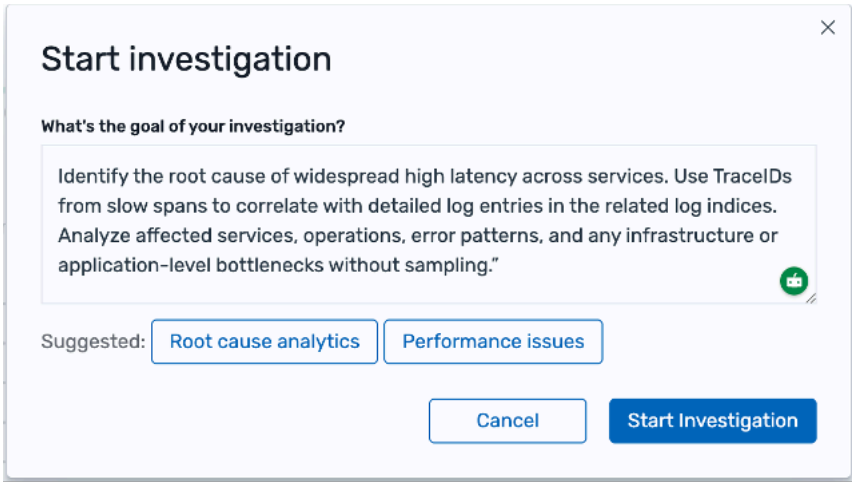
Agente de investigação no Amazon OpenSearch Service

O Agente de Investigação é um agente de pesquisa aprofundada em OpenSearch UI, orientado por objetivos, que investiga de forma autônoma questões complexas em seu nome. Ele planeja usando seus dados e a meta estabelecida, executa consultas e análises e reflete por meio de um fluxo de trabalho de várias etapas. Quando a investigação é concluída, normalmente em alguns minutos, ela gera hipóteses estruturadas classificadas por probabilidade, cada uma apoiada por evidências de dados. Ele fornece total transparência em cada etapa de seu raciocínio, para que você possa verificar e confiar nos resultados.

Iniciando uma investigação

Você pode iniciar uma investigação de duas maneiras:

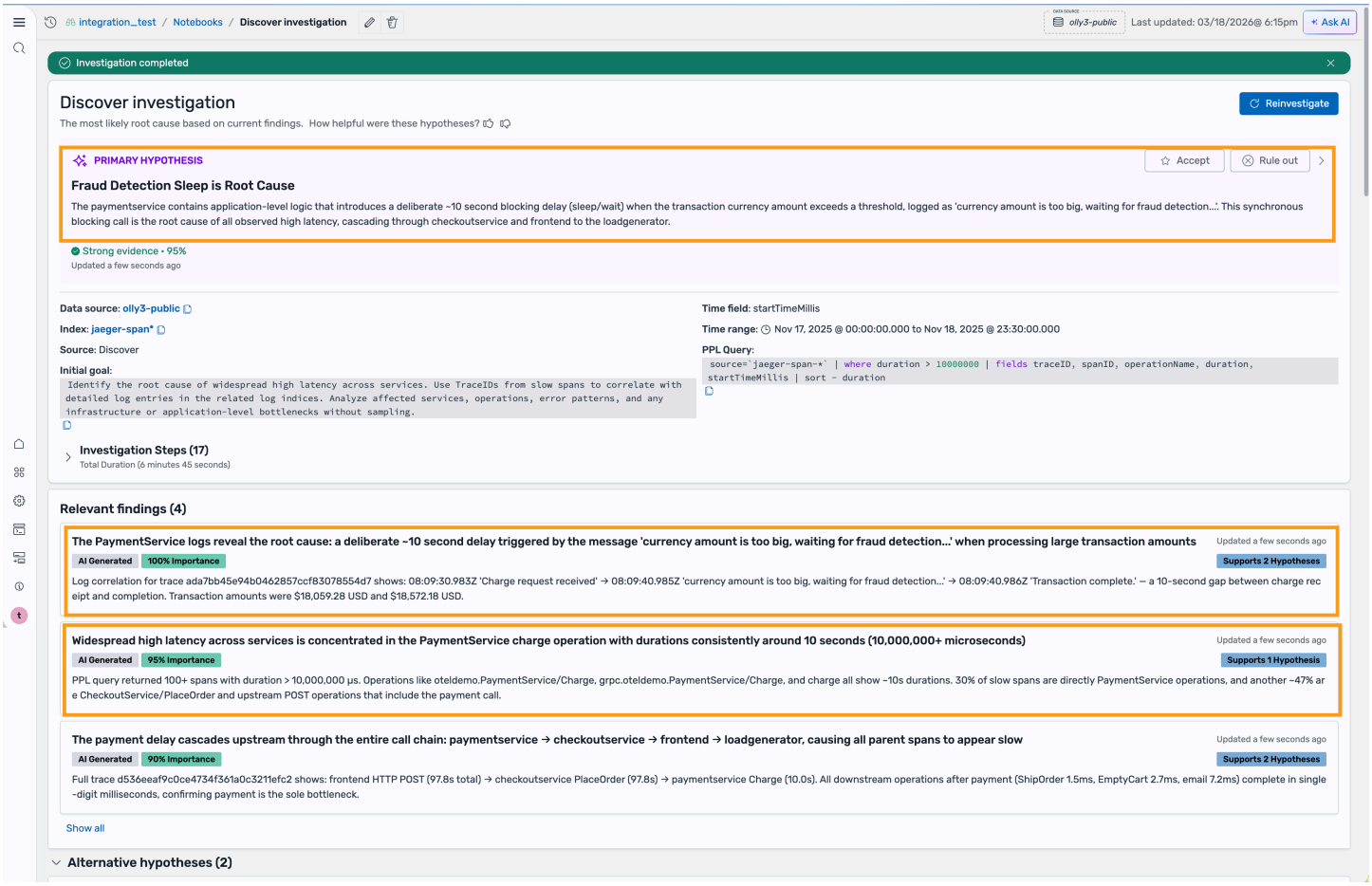
- Nas páginas de Descoberta, Visualização ou outras páginas de recursos compatíveis, escolha o botão Iniciar investigação. É exibida uma caixa de diálogo na qual você pode inserir sua meta de investigação e selecionar modelos sugeridos, como “Análise da causa raiz” ou “Problemas de desempenho”.
- No Agentic Chat, digite o comando de `/investigate` barra na entrada do bate-papo com seu objetivo de investigação.



Para obter mais informações sobre o Agentic Chat, consultethe section called “Chat Agêntico”.

Analisando os resultados da investigação

Quando a investigação é concluída, o Agente de Investigação apresenta uma hipótese primária com um nível de confiança e evidências de apoio. A página de resultados mostra as etapas de investigação realizadas, as descobertas relevantes classificadas por importância e as hipóteses alternativas.

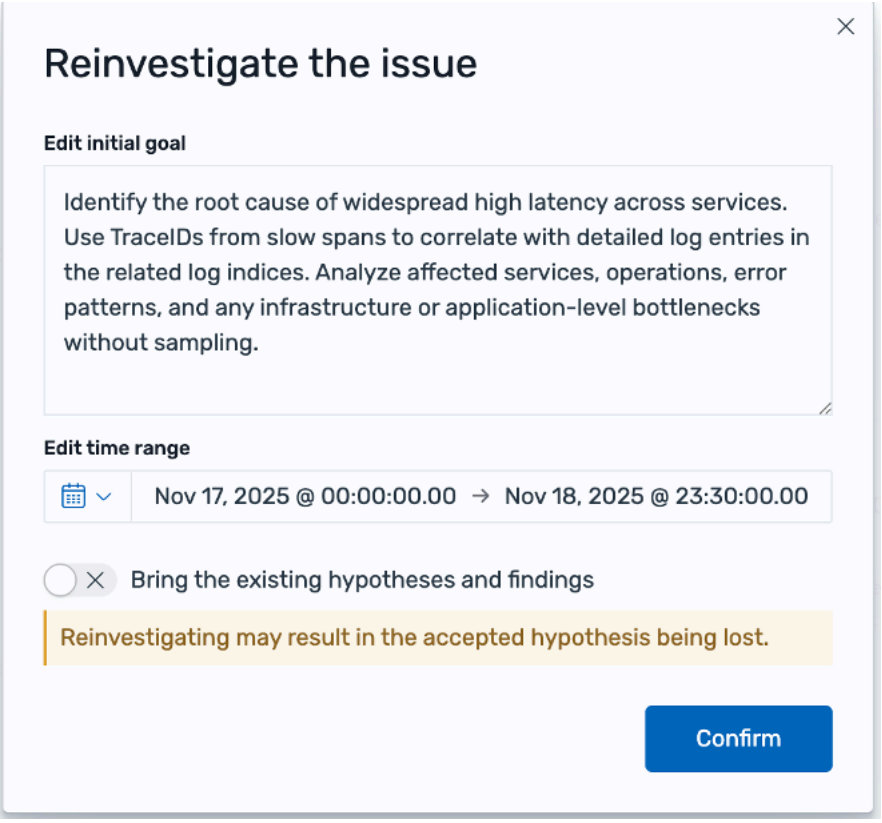


Você pode revisar as descobertas por trás de cada hipótese e escolher Aceitar para confirmar a hipótese ou Descartar para rejeitá-la. Hipóteses alternativas com menor probabilidade também estão disponíveis para revisão. Você pode selecionar qualquer hipótese alternativa como conclusão final, se ela corresponder melhor à sua avaliação.

Reinvestigando

Se os resultados da investigação exigirem mais esclarecimentos ou se o Agente de Investigação determinar que a pergunta da investigação não pode ser respondida pelos conjuntos de dados

disponíveis, você pode usar a opção Reinvestigar para ajustar e reexecutar a investigação. Escolha Reinvestigar para editar a meta inicial, ajustar o intervalo de tempo e, opcionalmente, trazer as hipóteses e descobertas existentes para a nova investigação.



Memória agente no Amazon Service OpenSearch

A Memória Agêntica é uma camada de memória gerenciada por serviços que alimenta o Agentic Chat e o Investigation Agent. Ele retém o contexto de sua conversa ou investigação, para que você tenha uma experiência consistente em diferentes páginas de recursos, guias do navegador e atualizações de página. A Memória Agentic funciona automaticamente e não requer configuração do usuário.

A memória agente é construída na estrutura de memória do OpenSearch agente. O armazenamento de memória é isolado pelo ID do usuário para fins de privacidade.

Proteção de dados

O Agentic Memory é de uso gratuito. Os dados do cliente armazenados na Memória Agentic são criptografados com uma chave gerenciada pelo serviço. Se você habilitou a criptografia de chave gerenciada pelo cliente (CMK) para seu aplicativo de OpenSearch interface de usuário, seus dados

de memória serão criptografados com sua CMK em vez disso. A memória é armazenada em uma coleção Amazon OpenSearch Serverless gerenciada por serviços.

Para obter mais informações sobre a criptografia CMK, consultethe section called “Criptografia e chave gerenciada pelo cliente”.

Limitações

A Memória Agêntica não pode reter o contexto em diferentes tópicos de conversa.

Criptografando metadados de aplicativos de OpenSearch interface de usuário com chaves gerenciadas pelo cliente

Os ativos visuais e as configurações são armazenados como metadados para seus aplicativos de OpenSearch interface do usuário. Isso inclui consultas, visualizações e painéis salvos. Os dados das fontes de dados associadas não são armazenados nos metadados. Para obter informações sobre criptografia de dados em suas fontes de dados, consulte Proteção de dados no Amazon OpenSearch Service para OpenSearch domínios e Criptografia no Amazon OpenSearch Serverless para coleções sem servidor.

Seus metadados de OpenSearch interface do usuário são protegidos com criptografia em repouso. Isso impede o acesso não autorizado. A criptografia usa AWS Key Management Service (AWS KMS) para armazenar e gerenciar as chaves de criptografia. Por padrão, os metadados da OpenSearch interface do usuário são criptografados com chaves AWS próprias.

Você também pode usar o recurso de chave gerenciada pelo cliente (CMK) para gerenciar suas próprias chaves de criptografia. Isso ajuda você a atender aos requisitos regulatórios e de conformidade. Para usar a CMK, você deve criar um novo aplicativo de OpenSearch interface de usuário e habilitar a CMK no processo de criação. No momento, não há suporte para atualizar um aplicativo de OpenSearch interface de usuário existente da chave AWS própria para a CMK.

Quando usar as chaves gerenciadas pelo cliente:

- Sua organização tem requisitos de conformidade regulatória para o gerenciamento de chaves
- Você precisa de trilhas de auditoria para o uso da chave de criptografia
- Você deseja controlar os principais horários de rotação
- Você precisa se integrar aos fluxos de trabalho de gerenciamento de chaves existentes

Ao usar uma chave gerenciada pelo cliente, você tem controle total sobre a chave. Isso inclui a capacidade de:

- Estabelecer e manter as políticas de chaves
- Estabelecer e manter políticas e concessões do IAM
- Ativar e desativar a chave
- Gire o material criptográfico da chave
- Adicione tags à chave
- Criar aliases de chaves
- Programe a chave para exclusão

> **Note**
>
> A chave gerenciada pelo cliente deve estar na Região da AWS mesma do aplicativo de OpenSearch interface do usuário. Você não pode usar uma chave de uma região diferente.

Tópicos

- Pré-requisitos para usar chaves gerenciadas pelo cliente
- Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o console
- Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o AWS CLI
- Monitorando o uso da chave gerenciada pelo cliente
- Atualizando as configurações de criptografia

Pré-requisitos para usar chaves gerenciadas pelo cliente

Antes de usar uma chave gerenciada pelo cliente para criptografar os metadados do seu aplicativo de OpenSearch interface de usuário, você deve criar uma chave de criptografia simétrica em. AWS KMS Para obter instruções sobre como criar chaves, consulte Criação de chaves no Guia do AWS KMS desenvolvedor.

A política de chaves da sua chave gerenciada pelo cliente deve conceder permissão à OpenSearch interface do usuário para usar a chave. Use a política principal a seguir, *`placeholder values`* substituindo-a por suas próprias informações:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowOpenSearchUIToUseKey",
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "application.opensearchservice.amazonaws.com"
                ]
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowKeyAdministration",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        }
    ]
}
```

A política inclui duas instruções:

- A primeira instrução permite que a OpenSearch UI use a chave para operações de criptografia.
- A segunda instrução permite que os usuários administrem a chave. Conta da AWS Isso inclui permissões para atualizar a política de chaves, ativar ou desativar a chave e programar a exclusão da chave. Você pode restringir ainda mais essas permissões substituindo o principal raiz por usuários ou funções específicas do IAM.

Para obter mais informações sobre políticas de chaves, consulte Como usar políticas de chaves AWS KMS no Guia do AWS KMS desenvolvedor.

Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o console

Ao criar um aplicativo de OpenSearch interface de usuário no console, você pode especificar uma chave gerenciada pelo cliente para criptografar os metadados do aplicativo.

Para criar um aplicativo de OpenSearch interface de usuário com criptografia de chave gerenciada pelo cliente usando o console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação esquerdo, escolha OpenSearch UI (painéis).
3. Selecione Criar aplicativo.
4. Em Nome da aplicação, insira um nome para a aplicação.
5. Defina as configurações de autenticação e administrador conforme necessário. Para obter mais informações, consulte the section called “Introdução”.
6. Na seção Criptografia, em Criptografia em repouso, escolha Usar chave gerenciada pelo cliente.
7. Selecione uma chave gerenciada pelo cliente existente na lista ou escolha Criar uma chave para criar uma nova chave AWS KMS.

   > **Note**
   >
   > A chave deve estar na Região da AWS mesma do aplicativo que você está criando.

8. (Opcional) Adicione tags ao aplicativo.
9. Escolha Criar.

Criação de um aplicativo com criptografia de chave gerenciada pelo cliente usando o AWS CLI

Para criar um aplicativo de OpenSearch interface de usuário com criptografia de chave gerenciada pelo cliente usando o AWS CLI, use o comando create-application com o parâmetro. `--kms-key-arn`

Substitua *`placeholder values`* por suas próprias informações.

```
aws opensearch create-application \
    --name my-application \
    --kms-key-arn arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012
```

Se você não especificar o `--kms-key-arn` parâmetro, OpenSearch usa uma chave AWS gerenciada para criptografar os metadados do aplicativo.

Monitorando o uso da chave gerenciada pelo cliente

Quando você usa uma chave gerenciada pelo cliente com um aplicativo de OpenSearch interface de usuário, AWS KMS registra cada uso da chave nos AWS CloudTrail registros. Você pode usar esses registros para monitorar como e quando sua chave é usada. Os registros mostram qual usuário ou serviço acessou a chave.

AWS AWS KMS gira automaticamente as chaves gerenciadas pelo cliente todos os anos. Você também pode girar manualmente as teclas conforme necessário. Para obter mais informações sobre rotação de chaves, consulte Como girar chaves KMS no Guia do AWS KMS desenvolvedor.

Para obter mais informações sobre como monitorar o uso de chaves, consulte Registrar chamadas de AWS KMS API AWS CloudTrail no Guia do AWS KMS desenvolvedor.

> **Note**
>
> O uso de chaves gerenciadas pelo cliente gera AWS KMS cobranças. As cobranças são baseadas no número de solicitações de API e chaves armazenadas. Para obter detalhes sobre preços, consulte Preços do serviço de gerenciamento de AWS chaves.

Atualizando as configurações de criptografia

Depois de criar um aplicativo de OpenSearch interface de usuário, você não pode alterar suas configurações de criptografia. Se você precisar usar uma chave gerenciada pelo cliente diferente, deverá criar um novo aplicativo. Se precisar alternar entre chaves AWS gerenciadas e gerenciadas pelo cliente, você também deverá criar um novo aplicativo com as configurações de criptografia desejadas.

> **Important**
>
> Antes de desativar ou excluir uma chave gerenciada pelo cliente, considere o seguinte:
>
> - Se você desativar a chave, o aplicativo perderá o acesso aos metadados criptografados. Você deve reativar a mesma chave para restaurar o acesso.
> - Se você excluir a chave, os objetos salvos do aplicativo ficarão permanentemente inacessíveis. Isso inclui consultas, visualizações e painéis. As chaves excluídas não podem ser recuperadas.
> - Recomendamos documentar o ARN da chave antes de fazer qualquer alteração no status da chave.

Próximas etapas

Depois de configurar a criptografia CMK para seu aplicativo, você pode:

- Associe fontes de dados ao seu aplicativo. Para obter mais informações, consulte the section called “Gerenciar associações de fontes de dados e permissões de acesso à VPC”.
- Crie espaços de trabalho para sua equipe. Para obter mais informações, consulte the section called “Usando espaços de trabalho OpenSearch do Amazon Service”.
- Configure o AWS CloudTrail monitoramento do uso da chave. Para obter mais informações, consulte the section called “Monitorando o uso da chave gerenciada pelo cliente”.

Habilitando a federação SAML com AWS Identity and Access Management

OpenSearch A interface do usuário oferece suporte à Security Assertion Markup Language 2.0 (SAML), um padrão aberto usado por muitos provedores de identidade. Isso habilita a federação de identidades com o AWS Identity and Access Management (IAM). Com esse suporte, os usuários da sua conta ou organização podem acessar diretamente a OpenSearch interface do usuário assumindo as funções do IAM. Você pode criar uma experiência de login único iniciada pelo provedor de identidade (IdP) para seus usuários finais, na qual eles podem se autenticar no provedor de identidade externo e serem roteados diretamente para sua página definida na interface do usuário. OpenSearch Você também pode implementar um controle de acesso refinado configurando

seus usuários finais ou grupos para assumirem diferentes funções do IAM com permissões diferentes para acessar a OpenSearch interface do usuário e as fontes de dados associadas.

Este tópico apresenta step-by-step instruções para configurar o uso do SAML com OpenSearch a interface do usuário. Nesses procedimentos, usamos as etapas para configurar a aplicação de gerenciamento de identidades e acesso Okta como exemplo. As etapas de configuração para outros provedores de identidades, como o Azure Active Directory e o Ping, são semelhantes.

Tópicos

- Etapa 1: configurar a aplicação de provedor de identidades (Okta)
- Etapa 2: AWS Configurar o Okta
- Etapa 3: criar a política OpenSearch de acesso ao Amazon Service no IAM
- Etapa 4: verificar a experiência de login único iniciada por provedor de identidades com o SAML
- Etapa 5: configurar o controle de acesso refinado baseado em atributos SAML

Etapa 1: configurar a aplicação de provedor de identidades (Okta)

Para usar o SAML com a OpenSearch interface do usuário, a primeira etapa é configurar seu provedor de identidade.

Tarefa 1: criar usuários do Okta

1. Entre na sua organização Okta em https://login.okta.com/como um usuário com privilégios administrativos.
2. No console do administrador, em Diretório no painel de navegação, escolha Pessoas.
3. Escolha Add person (Adicionar pessoa).
4. Em Nome, insira o nome do usuário.
5. Em Sobrenome, insira o sobrenome do usuário.
6. Em Nome do usuário, insira o nome de usuário do usuário no formato de e-mail.
7. Escolha Definirei a senha e insira uma senha
8. (Opcional) Desmarque a caixa O usuário deve alterar a senha no primeiro login se não quiser que o usuário altere a senha no primeiro login.
9. Escolha Salvar.

Tarefa 2: criar e atribuir grupos

1. Entre na sua organização Okta em https://login.okta.com/como um usuário com privilégios administrativos.
2. No console do administrador, em Diretório no painel de navegação, escolha Grupos.
3. Escolha Add Group (Adicionar grupo).
4. Insira um nome de grupo e selecione Salvar.
5. Escolha o grupo recém-criado e escolha Atribuir pessoas.
6. Escolha o sinal de adição (+) e depois Concluído.
7. (Opcional) Repita as etapas de 1 a 6 para adicionar mais grupos.

Tarefa 3: criar aplicações da Okta

1. Entre na sua organização Okta em https://login.okta.com/como um usuário com privilégios administrativos.
2. No console do administrador, em Applications no painel de navegação, escolha Applications.
3. Selecione Create App Integration (Criar integração de aplicações).
4. Escolha SAML 2.0 como método de login e, em seguida, escolha Next.
5. Insira um nome para a integração da aplicação (por exemplo, **`OpenSearch_UI`**) e escolha Avançar.
6. Insira os seguintes valores na aplicação; você não precisa alterar outros valores:
    a. 1. Para URL de login único, insira **`https://signin.aws.amazon.com/saml`** AWS as regiões comerciais ou a URL específica da sua região.
    b. 2. Em Audience URI (SP Entity ID), insira **`urn:amazon:webservices`**.
    c. 3. Em Name ID format, insira **`EmailAddress`**.
7. Escolha Próximo.
8. Escolha I'm an Okta customer adding an internal app e depois This is an internal app that we have created.
9. Escolha Terminar.
10. Escolha Assignments e depois escolha Assign.
11. Escolha Atribuir a grupos e depois selecione Assign ao lado dos grupos que você deseja adicionar.

12. Selecione Concluído.

Tarefa 4: definir a configuração avançada do Okta

Depois de criar a aplicação SAML personalizada, conclua as etapas a seguir:

1. Entre na sua organização Okta em https://login.okta.com/como um usuário com privilégios administrativos.

   No console do administrador, na área General, escolha Edit em SAML settings.
2. Escolha Próximo.
3. Defina o estado de retransmissão padrão para o endpoint da OpenSearch interface do usuário, usando o formato:

   `https://region.console.aws.amazon.com/aos/home?region=region#opensearch/applications/application-id/redirectToDashboardURL`.

   Este é um exemplo:

   ```
   https://us-east-2.console.aws.amazon.com/aos/home?region=us-east-2#opensearch/applications/abc123def4567EXAMPLE/redirectToDashboardURL
   ```
4. Em Group Attribute Statements (optional), adicione o seguinte atributo:

   a. Forneça o perfil do IAM e o provedor de identidades no formato separado por vírgula usando o atributo Role. Você usará essa mesma função e provedor de identidade do IAM em uma etapa posterior ao definir a AWS configuração.

   b. Defina user.login para. RoleSessionName Isso é usado como um identificador para as credenciais temporárias que são emitidas quando o perfil é assumido.

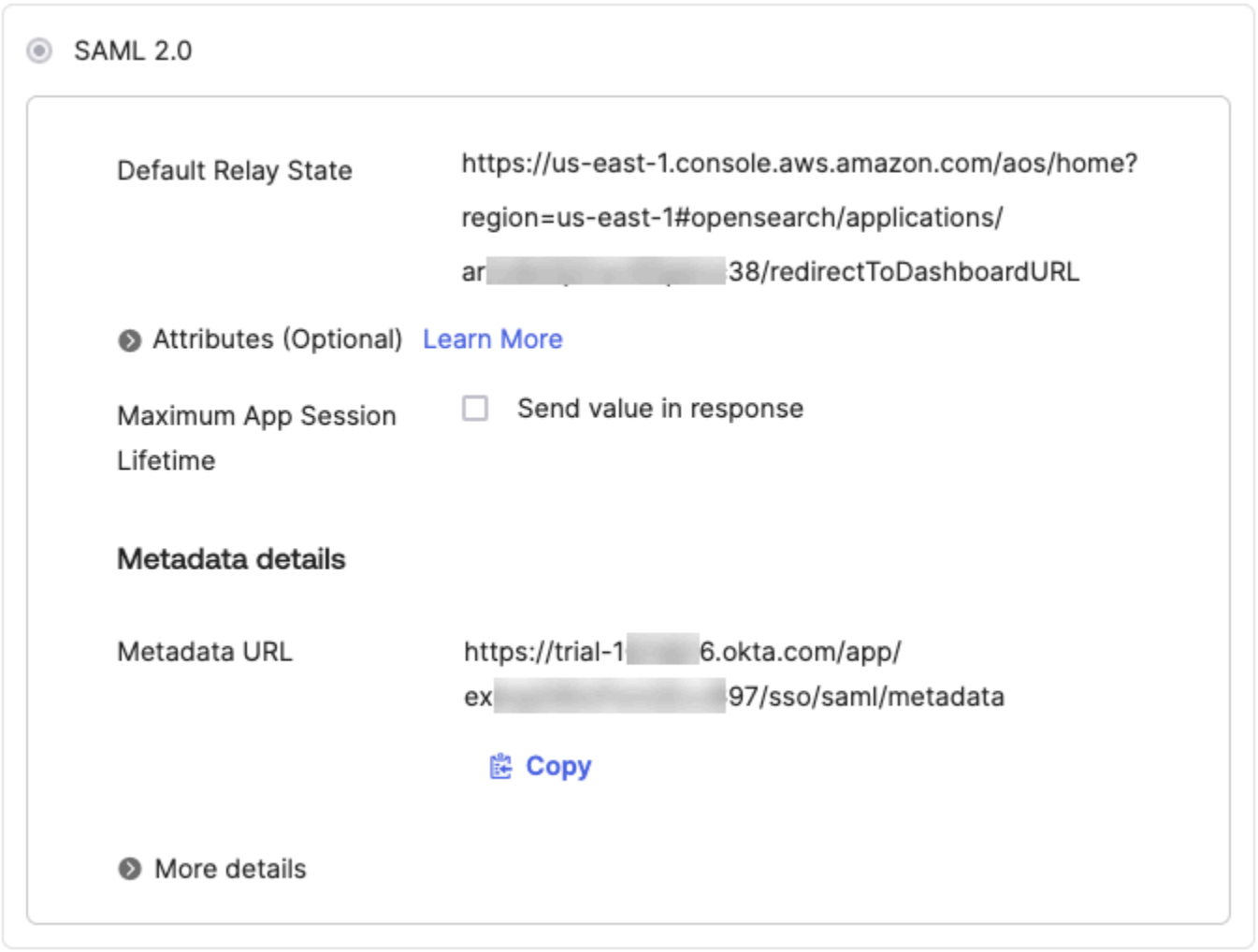
   Para referência:

Nome	Formato de nome	Formato	Exemplo
`https://aws.amazon`	Não especificado	`arn:aws:iam:: aws-accou`	`arn:aws:iam::11122`

Nome	Formato de nome	Formato	Exemplo
.com/SAML/Attributes/Role		*nt-id* :role/role-name,arn:aws:iam:: *aws-accou nt-id* :saml-provider/ *provider-name*	2333444:role/oktarole,arn:aws:iam::111222333444:saml-provider/oktaidp
https://aws.amazon.com/SAML/Attributes/RoleSessionName	Não especificado	user.login	user.login

5. Depois de adicionar as propriedades do atributo, escolha Avançar e, em seguida, escolha Finalizar.

Os atributos devem ter formato semelhante aos mostrados na imagem a seguir. O valor do Default Relay State é a URL para definir a página inicial para usuários finais em sua conta ou organização após concluírem a validação de login único da Okta. Você pode configurá-lo para qualquer página na OpenSearch interface do usuário e, em seguida, fornecer esse URL aos usuários finais pretendidos.



Etapa 2: AWS Configurar o Okta

Conclua as tarefas a seguir para definir sua AWS configuração para o Okta.

Tarefa 1: coletar informações do Okta

Para esta etapa, você precisará coletar suas informações do Okta para poder configurá-lo posteriormente na AWS.

1. Entre na sua organização Okta em https://login.okta.com/como um usuário com privilégios administrativos.
2. Na guia Sign On canto inferior direito da página, escolha View SAML setup instructions.
3. Anote o valor de Identity Provider Single Sign-on URL. Você pode usar essa URL ao se conectar a qualquer cliente SQL de terceiros, como o SQL Workbench/J.

4. Use os metadados do provedor de identidades no bloco 4 e salve o arquivo de metadados no formato.xml (por exemplo, `metadata.xml`).

Tarefa 2: criar o provedor do IAM

Para criar o provedor do IAM, conclua as seguintes etapas:

1. Faça login no Console de gerenciamento da AWS e abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação, em Gerenciamento de acesso, escolha Provedores de identidades.
3. Escolha Add provider (Adicionar provedor).
4. Para o tipo de provedor, selecione SAML.
5. Em Nome do provedor, insira um nome.
6. Em Documento de metadados, escolha Escolher arquivo e carregue o arquivo metadata file (.xml) que você baixou anteriormente.
7. Escolha Add provider (Adicionar provedor).

Tarefa 3: criar um perfil do IAM

Para criar sua AWS Identity and Access Management função, conclua as seguintes etapas:

1. Faça login no Console de gerenciamento da AWS e abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação, em Gerenciamento de acesso, escolha Perfis.
3. Selecione Criar perfil.
4. Em Tipo de entidade confiável, escolha Federação SAML 2.0.
5. Em Provedor baseado em SAML 2.0, escolha o provedor de identidades que você criou anteriormente.
6. Selecione Permitir programação e Console de gerenciamento da AWS acesso.
7. Escolha Próximo.
8. Na lista Políticas de permissões, marque as caixas de seleção das políticas que concedem permissões de OpenSearch serviço, por exemplo, política AWS gerenciada AmazonOpenSearchServiceFullAccess.
9. Escolha Próximo.

10. Na seção Revisar, em Nome do perfil, insira um nome para o perfil, por exemplo, **`oktarole`**.
11. (Opcional) Em Descrição da imagem, insira uma breve da finalidade do perfil.
12. Selecione Criar perfil.
13. Navegue até o perfil que você acabou de criar, escolha a guia Relações de Confiança e escolha Editar política de confiança.
14. No painel Editar instrução, em Adicionar ações para STS, selecione a caixa para TagSession.
15. Escolha Atualizar política.

Etapa 3: criar a política OpenSearch de acesso ao Amazon Service no IAM

Saiba como configurar suas funções do IAM para controle de OpenSearch acesso. Com as funções do IAM, você pode implementar um controle de acesso refinado para que seus grupos de usuários do Okta acessem os recursos. OpenSearch Este tópico demonstra a configuração baseada em perfil do IAM usando dois grupos de exemplo.

Sample group: Alice

Solicitação:

```
GET _plugins/_security/api/roles/alice-group
```

Resultado:

```
{
  "alice-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
        "index_patterns": [
          "alice*"
        ],
        "dls": "",
        "fls": [],
        "masked_fields": [],
        "allowed_actions": [
```

```
          "indices_all"
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

Sample group: Bob

Solicitação:

```
GET _plugins/_security/api/roles/bob-group
```

Resultado:

```
{
  "bob-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
        "index_patterns": [
          "bob*"
        ],
        "dls": "",
        "fls": [],
        "masked_fields": [],
        "allowed_actions": [
          "indices_all"
```

```
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

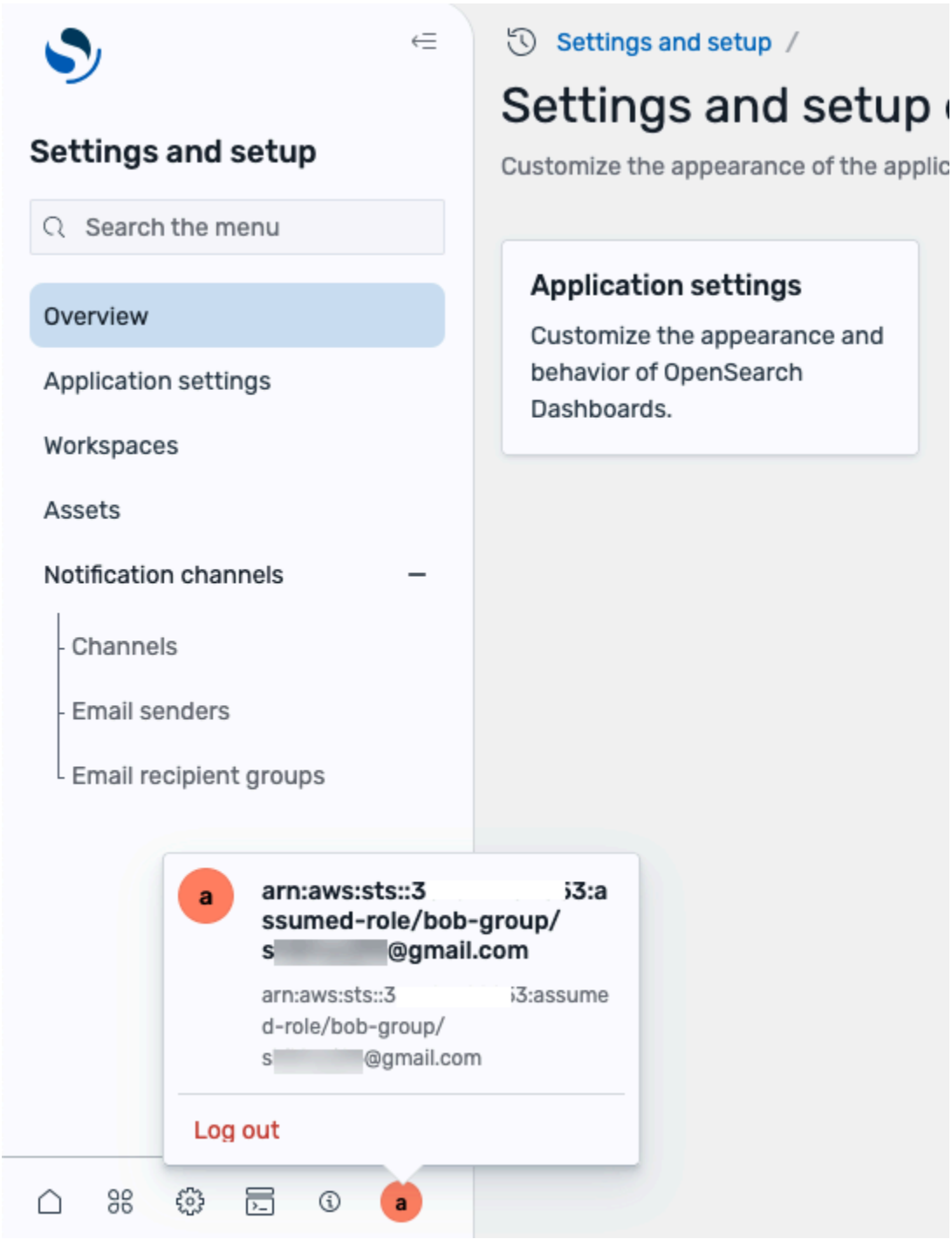
Você pode mapear as funções de domínio do Amazon OpenSearch Service para funções do IAM usando o mapeamento de funções de back-end, conforme demonstrado no exemplo a seguir:

```
{
  "bob-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/bob-group"
    ],
    "and_backend_roles": []
  },
  "alice-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/alice-group"
    ],
    "and_backend_roles": []
  }
}
```

Etapa 4: verificar a experiência de login único iniciada por provedor de identidades com o SAML

Abra a URL do Default Relay State para abrir a página de autenticação da Okta. Insira as credenciais de um usuário final. Você é redirecionado automaticamente para a OpenSearch interface do usuário.

Você pode verificar suas credenciais atuais escolhendo o ícone do usuário na parte inferior do painel de navegação, conforme ilustrado na seguinte imagem:



Você também pode verificar as permissões de controle de acesso refinadas para o usuário acessando as Ferramentas do Desenvolvedor na parte inferior do painel de navegação e executando consultas no console. Veja os exemplos de consultas.

Example 1: Displays information about the current user

Solicitação:

```
GET _plugins/_security/api/account
```

Resultado:

```
{
  "user_name": "arn:aws:iam::XXXXXXXXXXXX:role/bob-group",
  "is_reserved": false,
  "is_hidden": false,
  "is_internal_user": false,
  "user_requested_tenant": null,
  "backend_roles": [
    "arn:aws:iam::XXXXXXXXXXXX:role/bob-group"
  ],
  "custom_attribute_names": [],
  "tenants": {
    "global_tenant": true,
    "arn:aws:iam::XXXXXXXXXXXX:role/bob-group": true
  },
  "roles": [
    "bob-group"
  ]
}
```

Example 2: Displays actions permitted for a user

Solicitação:

```
GET bob-test/_search
```

Resultado:

```
{
  "took": 390,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
```

```
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 1,
    "hits": [
      {
        "_index": "bob-test",
        "_id": "ui01N5UBCIHpjO8Jlvfy",
        "_score": 1,
        "_source": {
          "title": "Your Name",
          "year": "2016"
        }
      }
    ]
  }
}
```

Example 3: Displays actions not permitted for a user

Solicitação:

```
GET alice-test
```

Resultado:

```
{
  "error": {
    "root_cause": [
      {
        "type": "security_exception",
        "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
      }
    ],
    "type": "security_exception",
    "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
  },
  "status": 403
```

```
}
```

Etapa 5: configurar o controle de acesso refinado baseado em atributos SAML

Com o Amazon OpenSearch Service, você pode usar um controle de acesso refinado com SAML para mapear usuários e grupos do seu provedor de identidade para usuários e funções de controle de acesso OpenSearch refinados de forma dinâmica. Você pode definir o escopo dessas funções para OpenSearch domínios específicos e coleções sem servidor, além de definir permissões em nível de índice e segurança em nível de documento.

> **Note**
>
> Para saber mais sobre controle de acesso da rede, consulte the section called “Controle de acesso refinado”.

Tópicos

- Atributos do SAML para controle de acesso de alta granularidade
- Tarefa 1: configurar o Okta para controle de acesso refinado
- Tarefa 2: Configurar SAML no domínio OpenSearch
- Tarefa 3: Configurar o SAML em coleções sem OpenSearch servidor

Atributos do SAML para controle de acesso de alta granularidade

subjectKey

Mapeia para um atributo de usuário exclusivo, como e-mail ou nome de usuário, que identifica o usuário para autenticação.

rolesKey

Mapeia para atributos de grupo ou perfil no IdP, que determinam os perfis ou permissões para autorização.

Tarefa 1: configurar o Okta para controle de acesso refinado

Para configurar o Okta para controle de acesso refinado

1. Adicione um novo atributo para o OpenSearch usuário principal na seção Declarações de atributos:

   - Nome: `UserName`
   - Valor: `${user-email}`

   Esse atributo é usado como a chave de assunto na configuração OpenSearch refinada de controle de acesso para autenticação.
2. Adiciona um atributo de grupo para perfis na seção Instrução de atributo de grupo:

   - Nome: `groups`
   - Filtro: `0penSearch_xxx`

   Esse atributo é usado como a chave de função para mapear grupos para funções de controle de acesso OpenSearch refinadas para autorização.

Tarefa 2: Configurar SAML no domínio OpenSearch

Para configurar o SAML no domínio OpenSearch

1. No AWS Management Console, identifique o domínio do OpenSearch serviço para o qual você deseja habilitar o controle de acesso detalhado para os usuários da interface do OpenSearch usuário.
2. Navegue até a página de detalhes do domínio específico.
3. Selecione a guia Configuração de segurança e clique em Editar.
4. Expanda o SAML via federação do IAM.
5. Insira a `subjectKey` e a `roleKey` que você definiu no Okta.
6. Selecione Salvar alterações.

Também é possível configurar um controle de acesso refinado usando o AWS CLI.

```
aws opensearch create-domain \
--domain-name testDomain \
--engine-version OpenSearch_1.3 \
--cluster-config
 InstanceType=r5.xlarge.search,InstanceCount=1,DedicatedMasterEnabled=false,ZoneAwarenessEnable
 \
--access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:12345678901:domain/neosaml10/*"}]}' \
--domain-endpoint-options '{"EnforceHTTPS":true,"TLSSecurityPolicy":"Policy-Min-
TLS-1-2-2019-07"}' \
--node-to-node-encryption-options '{"Enabled":true}' \
--encryption-at-rest-options '{"Enabled":true}' \
--advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"********","MasterUserPassword":"********"}, "IAMFederationOptions":
{"Enabled": true,"SubjectKey":"TestSubjectKey","RolesKey":"TestRolesKey"}}' \
--ebs-options "EBSEnabled=true,VolumeType=gp2,VolumeSize=300" \
--no-verify-ssl \
--endpoint-url https://es.us-east-1.amazonaws.com \
--region us-east-1
```

Para atualizar um domínio existente:

```
aws opensearch update-domain-config \
--domain-name testDomain \
--advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"********","MasterUserPassword":"********"}, "IAMFederationOptions":
{"Enabled": true,"SubjectKey":"TestSubjectKey","RolesKey":"TestRolesKey"}}' \
--ebs-options "EBSEnabled=true,VolumeType=gp2,VolumeSize=300" \
--no-verify-ssl \
--endpoint-url https://es.us-east-1.amazonaws.com \
--region us-east-1
```

Tarefa 3: Configurar o SAML em coleções sem OpenSearch servidor

Para configurar o controle de acesso refinado baseado em SAML no Serverless OpenSearch

1. Abra o Console de gerenciamento da AWS e navegue até o Amazon OpenSearch Service.
2. No painel de navegação, em Tecnologia sem servidor, escolha Segurança e depois Autenticação.

3. Na seção Federação do IAM, selecione Editar.

   É possível manusear com o controle de acesso refinado baseado em atributos do SAML usando essa configuração. A federação do IAM é desabilitada por padrão.

4. Selecione Habilitar federação do IAM.
5. Insira os valores `subjectKey` e `roleKey` que você definiu no Okta.

   Para saber mais, consulte the section called “Atributos do SAML para controle de acesso de alta granularidade”.

6. Selecione Salvar.
7. No painel de navegação, em Tecnologia sem servidor, escolha Política de acesso a dados.
8. Atualize uma política existente ou crie uma nova.
9. Expanda uma regra, escolha Adicionar entidades principais e selecione Usuários e grupos da Federação do IAM.
10. Adicione as entidades principais necessárias e escolha Salvar.
11. Selecione Conceder.
12. Nessa regra, faça o seguinte:

    - Selecione as permissões que você deseja definir para as entidades principais selecionadas.
    - Especifique as coleções nas quais você deseja aplicar as permissões.
    - Opcionalmente, defina as permissões em nível de índice.

    > **Note**
    >
    > Você pode criar várias regras para atribuir permissões diferentes a diferentes grupos de entidades principais.

13. Quando terminar, escolha Save (Salvar).
14. Escolha Criar.

Como alternativa, você pode usar a CLI para criar as configurações de segurança para coleções, conforme abaixo:

```
aws opensearchserverless create-security-config --region "region"  --type iamfederation
 --name "configuration_name" --description "description" --iam-federation-options
 '{"groupAttribute":"GroupKey","userAttribute":"UserKey"}'
```

Gerenciar associações de fontes de dados e permissões de acesso à nuvem privada virtual

Use os procedimentos desta seção para gerenciar associações de fontes de dados e configurar todas as permissões de acesso necessárias para uma nuvem privada virtual (VPC).

Tópicos

- Associando uma fonte de dados a um aplicativo de OpenSearch interface do usuário
- Gerenciar o acesso a domínios em uma VPC
- Configurando o acesso a coleções OpenSearch sem servidor em uma VPC

Associando uma fonte de dados a um aplicativo de OpenSearch interface do usuário

Depois de criar um aplicativo de OpenSearch interface de usuário, você pode usar o console ou associá-lo AWS CLI a uma ou mais fontes de dados. Depois disso, os usuários finais podem recuperar dados dessas fontes de dados para pesquisar, trabalhar com painéis e assim por diante.

Associar uma fonte de dados a um aplicativo de OpenSearch interface do usuário (console)

Para associar uma fonte de dados a um aplicativo de OpenSearch interface de usuário usando o console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Escolha OpenSearch UI (Painéis) e, em seguida, escolha o nome de um aplicativo de OpenSearch interface do usuário.
3. Na área Fontes de dados associadas, escolha Gerenciar fontes de dados.
4. Escolha entre os OpenSearch domínios e coleções que você deseja associar ao aplicativo.

> **Tip**
>
> Se não encontrar as fontes de dados que está procurando, entre em contato com os administradores para que concedam a você a permissão necessária. Para saber mais, consulte the section called “Permissões para criar uma aplicação que usa a autenticação do IAM Identity Center (opcional)”.

5. Escolha Avançar e depois Salvar.

Depois de associar uma fonte de dados à aplicação, o botão Iniciar aplicação é habilitado na página de detalhes da aplicação. Você pode escolher Iniciar aplicativo para abrir a OpenSearch página Bem-vindo, na qual você pode criar e gerenciar espaços de trabalho.

Para obter informações sobre como trabalhar com espaços de trabalho, consulte the section called “Usando espaços de trabalho OpenSearch do Amazon Service”.

Gerenciar o acesso a domínios em uma VPC

Se um OpenSearch domínio em uma VPC estiver associado ao aplicativo, um administrador da VPC deverá autorizar o acesso entre a interface do usuário e a VPC usando o OpenSearch console ou. AWS CLI

Gerenciar o acesso aos domínios de uma VPC (console)

Para configurar o acesso a um domínio da VPC usando o Console de gerenciamento da AWS:

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Domínios e escolha o nome do domínio da VPC.

   - ou -

   Escolha Criar domínio e configure os detalhes do domínio.
3. Escolha a guia Endpoints da VPC e depois Autorizar entidade principal.
4. Na caixa de diálogo Autorizar diretores, selecione Autorizar diretores de outros AWS serviços e, em seguida, escolha OpenSearch aplicativos (Painel) na lista.
5. Escolha Authorize.

Gerenciar o acesso aos domínios de uma VPC (AWS CLI)

Para autorizar um domínio VPC usando o AWS CLI

Para autorizar o domínio VPC usando AWS CLI o, execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch authorize-vpc-endpoint-access \
 --domain-name domain-name \
 --service application.opensearchservice.amazonaws.com \
 --region region-id
```

Para revogar associações de domínio de VPC usando o console

Quando uma associação não é mais necessária, o proprietário do domínio da VPC pode revogar o acesso usando o procedimento a seguir.

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Domínios e escolha o nome do domínio da VPC.
3. Escolha a guia VPC endpoints e, em seguida, selecione o botão para a linha de OpenSearch aplicativos (Painel).
4. Escolha Revogar acesso.

Para revogar uma associação de domínio VPC usando o AWS CLI

Para revogar uma associação de domínio VPC com OpenSearch o aplicativo de interface do usuário, execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch revoke-vpc-endpoint-access \
    --domain-name domain-name \
    --service application.opensearchservice.amazonaws.com \
    --region region-id
```

Configurando o acesso a coleções OpenSearch sem servidor em uma VPC

Se uma coleção Amazon OpenSearch Serverless em uma VPC estiver associada ao aplicativo, um administrador da VPC poderá autorizar o acesso criando uma nova política de rede e anexando-a à coleção.

Configurando o acesso a coleções OpenSearch sem servidor em uma VPC (console)

Para configurar o acesso às coleções OpenSearch sem servidor em uma VPC usando o console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Políticas de rede, escolha o nome da política de rede e escolha Editar.

   - ou -

   Escolha Criar política de rede e configure os detalhes da política.
3. Na área Tipo de acesso, escolha Privado (recomendado) e selecione Acesso privado ao serviço da AWS .
4. No campo de pesquisa, escolha Serviço e depois `application.opensearchservice.amazonaws.com`.
5. Na área Tipo de recurso, selecione a caixa Habilitar acesso ao OpenSearch endpoint.
6. Em Pesquisar coleções ou inserir termos de prefixo específicos, no campo de pesquisa, selecione Nome da coleção e insira ou selecione o nome das coleções a serem associadas à política de rede.
7. Escolha Criar para uma nova política de rede ou Atualizar para uma política de rede existente.

Configurando o acesso a coleções OpenSearch sem servidor em uma VPC ()AWS CLI

Para configurar o acesso às coleções OpenSearch sem servidor em uma VPC usando o AWS CLI

1. Crie um arquivo .json semelhante ao que se segue. Substitua *`placeholder values`* por suas próprias informações.

```
{
    "Description" : "policy-description",
    "Rules": [{
       "ResourceType" : "collection",
        "Resource" : ["collection/collection_name"]
     }],
    "SourceServices" : [
          "application.opensearchservice.amazonaws.com"
      ],
```

```
      "AllowFromPublic" : false
}
```

2. Crie ou atualize uma política de rede para uma coleção em uma VPC para trabalhar com aplicativos de OpenSearch interface do usuário.

   Create a network policy

   Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

   ```
   aws opensearchserverless create-security-policy \
       --type network  \
       --region region \
       --endpoint-url endpoint-url \
       --name network-policy-name \
       --policy file:/path_to_network_policy_json_file
   ```

   O comando retorna informações semelhantes às seguintes:

   ```
   {
       "securityPolicyDetail": {
           "createdDate": ******,
           "lastModifiedDate": ******,
           "name": "network-policy-name",
           "policy": [
               {
                   "SourceVPCEs": [],
                   "AllowFromPublic": false,
                   "Description": "",
                   "Rules": [
                       {
                           "Resource": [
                               "collection/network-policy-name"
                           ],
                           "ResourceType": "collection"
                       }
                   ],
                   "SourceServices": [
                       "application.opensearchservice.amazonaws.com"
                   ]
               }
   ```

```
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Update a network policy

Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearchserverless update-security-policy \
    --type network  \
    --region region \
    --endpoint-url endpoint-url \
    --name network-policy-name \
    --policy-version "policy_version_from_output_of_network_policy_creation" \
    --policy file:/path_to_network_policy_json_file
```

O comando retorna informações semelhantes às seguintes:

```
{
    "securityPolicyDetail": {
        "createdDate": ******,
        "lastModifiedDate": ******,
        "name": "network-policy-name",
        "policy": [
            {
                "SourceVPCEs": [],
                "AllowFromPublic": false,
                "Description": "",
                "Rules": [
                    {
                        "Resource": [
                            "collection/network-policy-name"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "SourceServices": [
                    "application.opensearchservice.amazonaws.com"
                ]
```

```
            }
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Usando espaços de trabalho OpenSearch do Amazon Service

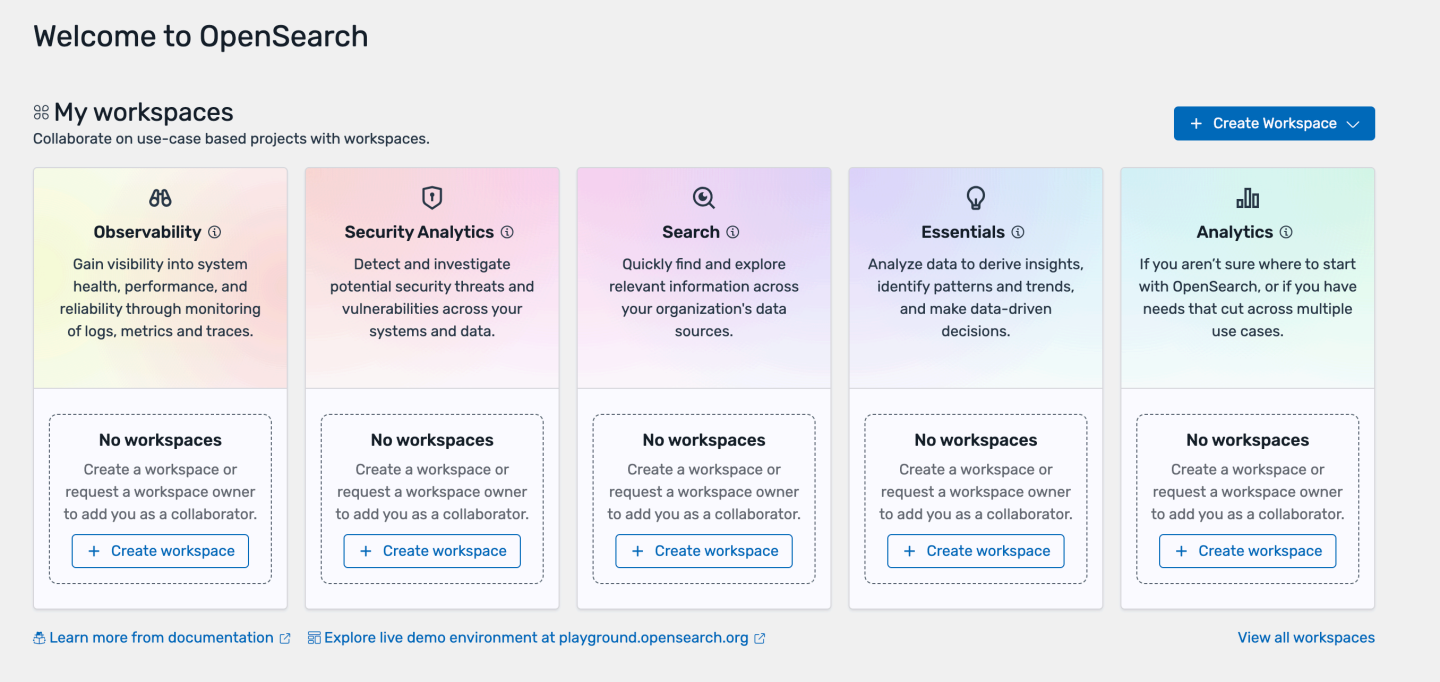
O Amazon OpenSearch Service oferece suporte à criação de vários espaços de trabalho específicos para casos de uso. Cada espaço de trabalho fornece uma experiência selecionada para casos de uso mais comuns, como Observabilidade, Analytics de segurança e pesquisa. O Workspace também é compatível com o gerenciamento de colaboradores, para que você possa compartilhar seu espaço de trabalho somente com os colaboradores pretendidos e gerenciar as permissões de cada um.

Criação de espaços de trabalho de aplicativos de OpenSearch UI

Depois que um aplicativo de OpenSearch interface de usuário for criado e associado às fontes de dados e as permissões de usuário tiverem sido configuradas para o aplicativo, você poderá iniciar o aplicativo de OpenSearch interface do usuário para criar espaços de trabalho.

Para começar a criar um espaço de trabalho, você pode selecionar o botão Iniciar aplicativo na página de detalhes do aplicativo ou usar o URL do aplicativo de OpenSearch interface de usuário para abrir a página inicial do aplicativo de OpenSearch interface de usuário em uma nova janela do navegador.

O aplicativo de OpenSearch interface do usuário fornece opções para criar espaços de trabalho e lista todos os espaços de trabalho existentes na página inicial, categorizados por caso de uso.



Para saber mais sobre os tipos de instâncias compatíveis, consulte the section called “Tipo de espeço ded trabalho”.

Privacidade do espaço de trabalho e colaboradores

Você pode definir uma configuração de privacidade para um espaço de trabalho como o nível de permissão padrão para todos os usuários. Você pode fazer isso ao criar um espaço de trabalho ou modificar um espaço de trabalho existente (na guia Colaboradores do espaço de trabalho). Existem três opções de privacidade:

- Privativo dos colaboradores: somente os colaboradores que você adiciona explicitamente ao espaço de trabalho podem acessar o espaço de trabalho. Você pode definir os níveis de permissão de cada colaborador.
- Qualquer pessoa pode ver — Qualquer pessoa que tenha acesso ao aplicativo de OpenSearch interface do usuário pode acessar o espaço de trabalho e visualizar seus ativos, mas não pode fazer nenhuma alteração no espaço de trabalho.
- Qualquer pessoa pode editar — Qualquer pessoa que tenha acesso ao aplicativo de OpenSearch interface do usuário pode acessar o espaço de trabalho, visualizar ativos nele e fazer alterações nos ativos no espaço de trabalho.

Na guia Colaboradores do espaço de trabalho, você pode adicionar usuários ou funções do IAM e Centro de Identidade do AWS IAM usuários como colaboradores em um espaço de trabalho. Há três níveis de permissões para os colaboradores escolherem:

- Somente leitura: o colaborador somente pode visualizar os ativos do espaço de trabalho. Essa configuração será substituída se o espaço de trabalho estiver configurado para usar a configuração de privacidade Qualquer pessoa pode editar.
- Ler e gravar: o colaborador pode visualizar e editar os ativos do espaço de trabalho. Se o espaço de trabalho estiver configurado para usar a configuração de privacidade Qualquer um pode visualizar, o colaborador ainda poderá editar.
- Administrador: o colaborador pode atualizar as configurações e excluir o espaço de trabalho. O colaborador também pode alterar as configurações de privacidade do espaço de trabalho e gerenciar os colaboradores. O usuário que cria o espaço de trabalho é automaticamente designado como administrador do espaço de trabalho.

Tipo de espeço ded trabalho

O Amazon OpenSearch Service fornece cinco tipos de espaço de trabalho, cada um com recursos diferentes para os diferentes casos de uso:

- O espaço de trabalho Observabilidade foi projetado para ter visibilidade da integridade, performance e confiabilidade do sistema por meio do monitoramento de logs, métricas e rastreamentos.
- O espaço de trabalho Analytics de segurança foi projetado para detectar e investigar possíveis ameaças e vulnerabilidades de segurança em seus sistemas e dados.
- O espaço de trabalho Pesquisa foi projetado para encontrar e explorar rapidamente informações relevantes nas fontes de dados da sua organização.
- O espaço de trabalho Essentials foi projetado para o OpenSearch Serverless como fonte de dados e permite analisar dados para obter insights, identificar padrões e tendências e tomar decisões baseadas em dados rapidamente. Você pode encontrar e explorar informações relevantes nas fontes de dados da sua organização em um espaço de trabalho Fundamentos.
- O espaço de trabalho do Analytics (todos os recursos) foi projetado para casos de uso multiuso e oferece suporte a todos os recursos disponíveis na interface OpenSearch de serviço (painéis).

Acesso a dados entre regiões e contas

OpenSearch A interface do usuário suporta o acesso a dados de OpenSearch domínios em diferentes Conta da AWS s e Região da AWS s. Você pode escolher entre duas abordagens, dependendo de suas necessidades. A tabela a seguir compara as duas abordagens.

> **Note**
>
> Tanto o acesso a dados entre contas quanto a pesquisa entre clusters funcionam somente com OpenSearch domínios. Nenhuma das abordagens oferece suporte a coleções OpenSearch sem servidor.

Aspecto	Acesso a dados entre contas	Pesquisa entre clusters
Recurso	Associe domínios de outras contas como fontes de dados diretas na OpenSearch interface do usuário	Consulte dados em domínios conectados usando conexões de pesquisa entre clusters
Mecanismo	Acesso direto — a OpenSearch interface do usuário se conecta diretamente ao domínio de destino em outra conta	Acesso indireto — requer um domínio local na mesma conta da OpenSearch interface do usuário para retransmitir solicitações para domínios remotos
Suporte entre contas.	Sim	Sim
Suporte entre regiões	Não — os domínios de origem e de destino devem estar no mesmo Região da AWS	Sim — os domínios de origem e destino podem estar em diferentes s Região da AWS
Unifique dados em todos os domínios	Não — cada domínio é consultado de forma independente como uma fonte de dados separada	Sim — uma única consulta pode agregar resultados de vários domínios conectados
Métodos de autenticação	IAM e Centro de Identidade do AWS IAM	IAM (com controle de acesso refinado)

Aspecto	Acesso a dados entre contas	Pesquisa entre clusters
Complexidade da configuração	Inferior — requer uma função do IAM entre contas para validação	Superior — requer conexões entre clusters, políticas de acesso em ambos os domínios e controle de acesso refinado
Visibilidade da fonte de dados na OpenSearch interface	Cada domínio entre contas aparece como uma fonte de dados separada	Os domínios remotos são acessados por meio dos aliases de conexão do domínio de origem local
Acesso de gravação ao domínio remoto	Sim — controlado pela política de acesso do domínio de destino	Não — a pesquisa entre clusters fornece acesso somente de leitura a domínios remotos

Tópicos

- Acesso a dados entre contas a domínios OpenSearch
- Pesquisa entre clusters

Acesso a dados entre contas a domínios OpenSearch

Você pode configurar seus aplicativos de OpenSearch interface de usuário em uma conta para acessar OpenSearch domínios em contas diferentes. Ao criar um aplicativo de OpenSearch interface de usuário com fontes de dados entre contas, você fornece um `iamRoleForDataSourceArn` que aponta para uma função do IAM na conta de destino. OpenSearch A interface do usuário valida a solicitação assumindo essa função e ligando `es:DescribeDomain` para verificar a acessibilidade do domínio. A função entre contas é usada somente para validação do plano de controle. O acesso ao plano de dados é controlado separadamente pela política de acesso do domínio de destino.

⚠ Código de exemplo

Os exemplos de código neste tópico são apenas para fins ilustrativos. Eles demonstram funcionalidade básica e podem não incluir tratamento de erros, melhores práticas de segurança ou recursos prontos para produção. Antes de usar o código de amostra na

produção, revise-o e modifique-o para atender aos seus requisitos específicos e teste minuciosamente em seu ambiente.

Principais conceitos

Conta de origem

O Conta da AWS que hospeda seu aplicativo de OpenSearch interface do usuário.

Conta de destino

O Conta da AWS local onde o OpenSearch domínio reside.

Função entre contas

Uma função do IAM na conta de destino que é usada somente para validação do plano de controle. Essa função requer somente a `es:DescribeDomain` permissão.

Função do aplicativo IAM Identity Center

Uma função do IAM na conta de origem que é usada para acessar o plano de dados do usuário do IAM Identity Center.

Pré-requisitos

Antes de configurar o acesso a dados entre contas, verifique se você tem o seguinte:

- AWS CLI instalado e configurado
- Acesso à origem e ao destino Conta da AWS
- Para fluxos do IAM Identity Center: uma instância Centro de Identidade do AWS IAM organizacional

Cenários

Escolha o cenário que corresponda ao método de autenticação e à configuração do domínio:

- the section called “Cenário 1: usuário do IAM acessando um domínio público”
- the section called “Cenário 2: usuário do IAM Identity Center acessando um domínio público”

- the section called “Cenário 3: usuário do IAM acessando um domínio VPC”
- the section called “Cenário 4: usuário do IAM Identity Center acessando um domínio VPC”

Cenário 1: usuário do IAM acessando um domínio público

Etapa 1: criar a função IAM entre contas (conta de destino)

Crie uma função do IAM na conta de destino que permita que a conta de origem a assuma para validação do domínio.

Para criar a função entre contas

1. Crie uma política de confiança que permita que a conta de origem assuma a função:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::source-account-id:root"
    },
    "Action": "sts:AssumeRole"
  }]
}
```

2. Crie a função:

```
aws iam create-role \
  --role-name OpenSearchUIAccessRole \
  --assume-role-policy-document file://trust-policy.json
```

3. Crie uma política de permissões com apenas a `es:DescribeDomain` ação:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "es:DescribeDomain",
    "Resource": "arn:aws:es:region:target-account-id:domain/*"
  }]
}
```

4. Anexe a política de permissões ao perfil:

```
aws iam put-role-policy \
  --role-name OpenSearchUIAccessRole \
  --policy-name ValidationOnly \
  --policy-document file://permissions-policy.json
```

Etapa 2: criar o OpenSearch domínio (conta de destino)

Crie um OpenSearch domínio na conta de destino com controle de acesso refinado e criptografia habilitados:

```
aws opensearch create-domain \
  --domain-name domain-name \
  --engine-version OpenSearch_2.19 \
  --cluster-config InstanceType=m5.large.search,InstanceCount=1 \
  --ebs-options "EBSEnabled=true,VolumeType=gp3,VolumeSize=100" \
  --advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"admin","MasterUserPassword":"master-password"}}' \
  --node-to-node-encryption-options '{"Enabled":true}' \
  --encryption-at-rest-options '{"Enabled":true}' \
  --domain-endpoint-options '{"EnforceHTTPS":true,"TLSSecurityPolicy":"Policy-Min-
TLS-1-2-2019-07"}' \
  --access-policies '{"Version":"2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":
{"AWS":"*"},"Action":"es:ESHttp*","Resource":"arn:aws:es:region:target-account-
id:domain/domain-name/*"}]}' \
  --region region
```

Aguarde até que o status do domínio se torne `Active` antes de continuar.

Etapa 3: criar o aplicativo de OpenSearch interface do usuário (conta de origem)

Crie o aplicativo na conta de origem com a fonte de dados entre contas:

```
aws opensearch create-application \
  --region region \
  --name "cross-account-iam-app" \
  --data-sources '[{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/domain-name",
```

```
    "dataSourceDescription":"Cross-account domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
  }]' \
  --app-configs '[{"key":"opensearchDashboards.dashboardAdmin.users","value":"[\"*
\"]"}]'
```

Etapa 4: verificar e acessar

Recupere os detalhes do aplicativo para obter o URL do endpoint:

```
aws opensearch get-application \
  --region region \
  --id application-id
```

- Navegue até o URL do endpoint do aplicativo a partir da resposta.
- Faça login com as credenciais do IAM.
- O usuário do IAM assina as solicitações do plano de dados com suas próprias credenciais.
- A política de acesso ao domínio de destino controla quais dados o usuário pode acessar.

Cenário 2: usuário do IAM Identity Center acessando um domínio público

Etapa 1: criar a função IAM entre contas (conta de destino)

Crie uma função do IAM na conta de destino que permita que a conta de origem a assuma para validação do domínio.

Para criar a função entre contas

1. Crie uma política de confiança que permita que a conta de origem assuma a função:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::source-account-id:root"
    },
    "Action": "sts:AssumeRole"
  }]
```

```
}
```

2. Crie a função:

```
aws iam create-role \
  --role-name OpenSearchUIAccessRole \
  --assume-role-policy-document file://trust-policy.json
```

3. Crie uma política de permissões com apenas a `es:DescribeDomain` ação:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "es:DescribeDomain",
    "Resource": "arn:aws:es:region:target-account-id:domain/*"
  }]
}
```

4. Anexe a política de permissões ao perfil:

```
aws iam put-role-policy \
  --role-name OpenSearchUIAccessRole \
  --policy-name ValidationOnly \
  --policy-document file://permissions-policy.json
```

Etapa 2: criar o OpenSearch domínio (conta de destino)

Crie um OpenSearch domínio na conta de destino. Use o mesmo comando dethe section called "Etapa 2: criar o OpenSearch domínio (conta de destino)", mas atualize a política de acesso para permitir o papel do aplicativo IAM Identity Center na conta de origem:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::source-account-id:role/NeoIdCAppRole"
    },
    "Action": "es:ESHttp*",
    "Resource": "arn:aws:es:region:target-account-id:domain/domain-name/*"
  }]
```

```
}
```

Aguarde até que o status do domínio se torne `Active` antes de continuar.

Etapa 3: criar a função do IAM para o aplicativo IAM Identity Center (conta de origem)

Crie uma função do IAM na conta de origem que a OpenSearch UI usa para acessar o plano de dados do usuário do IAM Identity Center.

Para criar a função do aplicativo IAM Identity Center

1. Crie uma política de confiança:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "application.opensearchservice.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "application.opensearchservice.amazonaws.com"
      },
      "Action": "sts:SetContext",
      "Condition": {
        "ForAllValues:ArnEquals": {
          "sts:RequestContextProviders": "arn:aws:iam::source-account-id:oidc-
provider/portal.sso.region.amazonaws.com/apl/application-id"
        }
      }
    }
  ]
}
```

2. Crie uma política de permissões:

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [{
    "Sid": "OpenSearchDomain",
    "Effect": "Allow",
    "Action": ["es:ESHttp*"],
    "Resource": "*"
  }]
}
```

3. Crie a função e anexe as políticas:

```
aws iam create-role \
  --role-name NeoIdCAppRole \
  --assume-role-policy-document file://neoidc-trust-policy.json

aws iam put-role-policy \
  --role-name NeoIdCAppRole \
  --policy-name NeoIdCAppPermissions \
  --policy-document file://neoidc-permissions-policy.json
```

Etapa 4: criar o aplicativo de OpenSearch interface do usuário com o IAM Identity Center (conta de origem)

```
aws opensearch create-application \
  --region region \
  --name "cross-account-idc-app" \
  --iam-identity-center-options '{
    "enabled":true,
    "iamIdentityCenterInstanceArn":"arn:aws:sso:::instance/ssoins-instance-id",
    "iamRoleForIdentityCenterApplicationArn":"arn:aws:iam::source-account-
id:role/NeoIdCAppRole"
  }' \
  --data-sources '[{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/domain-name",
    "dataSourceDescription":"Cross-account domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
  }]' \
  --app-configs '[{"key":"opensearchDashboards.dashboardAdmin.users","value":"[\"*
\"]"}]'
```

Etapa 5: criar e atribuir usuários e grupos do IAM Identity Center

Crie um usuário do IAM Identity Center

Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws identitystore create-user \
  --identity-store-id d-directory-id \
  --user-name user-email \
  --display-name "display-name" \
  --name Formatted=string,FamilyName=last-name,GivenName=first-name \
  --emails Value=user-email,Type=work,Primary=true
```

Crie um grupo do IAM Identity Center e adicione o usuário

Execute os seguintes comandos :

```
aws identitystore create-group \
  --identity-store-id d-directory-id \
  --display-name "OpenSearchUsers" \
  --description "Users with OpenSearch access"

aws identitystore create-group-membership \
  --identity-store-id d-directory-id \
  --group-id group-id \
  --member-id UserId=user-id
```

Atribua o usuário ou grupo ao aplicativo

Execute este comando: .

```
aws sso-admin create-application-assignment \
  --application-arn "arn:aws:sso:::source-account-id:application/ssoins-instance-id/
apl-application-id" \
  --principal-id user-id-or-group-id \
  --principal-type USER
```

Configurar o mapeamento de funções de back-end no domínio de destino

Mapeie o grupo do IAM Identity Center para uma função de OpenSearch segurança no domínio de destino:

```
curl -XPUT "https://domain-endpoint/_plugins/_security/api/rolesmapping/all_access" \
```

```
  -u admin:master-password \
  -H 'Content-Type: application/json' \
  -d '{
    "backend_roles": ["group-id"],
    "hosts": [],
    "users": []
  }'
```

Etapa 6: verificar e acessar

```
aws opensearch get-application \
  --region region \
  --id application-id
```

- Navegue até o URL do endpoint do aplicativo.
- Faça login com as credenciais de usuário do IAM Identity Center.
- As solicitações de dados dos usuários do IAM Identity Center são assinadas com a função de aplicativo do IAM Identity Center, não com a função entre contas.
- Mapeamentos de funções de back-end nas permissões de acesso aos dados de controle de domínio.

Cenário 3: usuário do IAM acessando um domínio VPC

Etapa 1: criar a função IAM entre contas (conta de destino)

Crie uma função do IAM na conta de destino que permita que a conta de origem a assuma para validação do domínio.

Para criar a função entre contas

1. Crie uma política de confiança que permita que a conta de origem assuma a função:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::source-account-id:root"
    },
    "Action": "sts:AssumeRole"
```

```
  }]
}
```

2. Crie a função:

```
aws iam create-role \
  --role-name OpenSearchUIAccessRole \
  --assume-role-policy-document file://trust-policy.json
```

3. Crie uma política de permissões com apenas a `es:DescribeDomain` ação:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "es:DescribeDomain",
    "Resource": "arn:aws:es:region:target-account-id:domain/*"
  }]
}
```

4. Anexe a política de permissões ao perfil:

```
aws iam put-role-policy \
  --role-name OpenSearchUIAccessRole \
  --policy-name ValidationOnly \
  --policy-document file://permissions-policy.json
```

Etapa 2: configurar a VPC (conta de destino)

Ignore essa etapa se já existir uma VPC na conta de destino.

```
# Create VPC
aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --region region

# Create subnet
aws ec2 create-subnet \
  --vpc-id vpc-id \
  --cidr-block 10.0.1.0/24 \
  --availability-zone regiona \
  --region region
```

```
# Create security group
aws ec2 create-security-group \
  --group-name opensearch-vpc-sg \
  --description "Security group for OpenSearch VPC domain" \
  --vpc-id vpc-id \
  --region region

# Allow inbound HTTPS
aws ec2 authorize-security-group-ingress \
  --group-id security-group-id \
  --protocol tcp \
  --port 443 \
  --cidr 10.0.0.0/16 \
  --region region
```

Saiba mais sobre a criação de domínios VPC.

Etapa 3: criar o domínio VPC (conta de destino)

```
aws opensearch create-domain \
  --domain-name vpc-domain-name \
  --engine-version OpenSearch_2.19 \
  --cluster-config InstanceType=m5.large.search,InstanceCount=1 \
  --ebs-options "EBSEnabled=true,VolumeType=gp3,VolumeSize=100" \
  --vpc-options "SubnetIds=subnet-id,SecurityGroupIds=security-group-id" \
  --advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"admin","MasterUserPassword":"master-password"}}' \
  --node-to-node-encryption-options '{"Enabled":true}' \
  --encryption-at-rest-options '{"Enabled":true}' \
  --domain-endpoint-options '{"EnforceHTTPS":true,"TLSSecurityPolicy":"Policy-Min-
TLS-1-2-2019-07"}' \
  --access-policies '{"Version":"2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":
{"AWS":"*"},"Action":"es:ESHttp*","Resource":"arn:aws:es:region:target-account-
id:domain/vpc-domain-name/*"}]}' \
  --region region
```

Aguarde até que o status do domínio se torne `Active` antes de continuar.

Etapa 4: autorizar o VPC endpoint para o principal do serviço de interface OpenSearch do usuário (conta de destino)

> **Important**
>
> Essa é uma etapa crítica exclusiva dos domínios VPC. O serviço de OpenSearch interface do usuário deve ser explicitamente autorizado a acessar o VPC endpoint.

```
# Authorize the service principal
aws opensearch authorize-vpc-endpoint-access \
  --domain-name vpc-domain-name \
  --service "application.opensearchservice.amazonaws.com" \
  --region region

# Verify authorization
aws opensearch list-vpc-endpoint-access \
  --domain-name vpc-domain-name \
  --region region
```

Resposta esperada:

```
{
  "AuthorizedPrincipalList": [
    {
      "PrincipalType": "AWS_SERVICE",
      "Principal": "application.opensearchservice.amazonaws.com"
    }
  ]
}
```

Etapa 5: criar o aplicativo de OpenSearch interface do usuário (conta de origem)

```
aws opensearch create-application \
  --region region \
  --name "cross-account-vpc-iam-app" \
  --data-sources '[{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/vpc-domain-name",
    "dataSourceDescription":"Cross-account VPC domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
```

```
  }]' \
  --app-configs '[{"key":"opensearchDashboards.dashboardAdmin.users","value":"[\"*
\"]"}]'
```

Etapa 6: verificar e acessar

Recupere os detalhes do aplicativo para obter o URL do endpoint:

```
aws opensearch get-application \
  --region region \
  --id application-id
```

- Navegue até o URL do endpoint do aplicativo a partir da resposta.
- Faça login com as credenciais do IAM.
- O usuário do IAM assina as solicitações do plano de dados com suas próprias credenciais.
- A política de acesso ao domínio de destino controla quais dados o usuário pode acessar.

Cenário 4: usuário do IAM Identity Center acessando um domínio VPC

Etapa 1: criar a função IAM entre contas (conta de destino)

Crie uma função do IAM na conta de destino que permita que a conta de origem a assuma para validação do domínio.

Para criar a função entre contas

1. Crie uma política de confiança que permita que a conta de origem assuma a função:

   ```
   {
     "Version": "2012-10-17",
     "Statement": [{
       "Effect": "Allow",
       "Principal": {
         "AWS": "arn:aws:iam::source-account-id:root"
       },
       "Action": "sts:AssumeRole"
     }]
   }
   ```

2. Crie a função:

```
aws iam create-role \
  --role-name OpenSearchUIAccessRole \
  --assume-role-policy-document file://trust-policy.json
```

3. Crie uma política de permissões com apenas a `es:DescribeDomain` ação:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "es:DescribeDomain",
    "Resource": "arn:aws:es:region:target-account-id:domain/*"
  }]
}
```

4. Anexe a política de permissões ao perfil:

```
aws iam put-role-policy \
  --role-name OpenSearchUIAccessRole \
  --policy-name ValidationOnly \
  --policy-document file://permissions-policy.json
```

Etapa 2: configurar a VPC (conta de destino)

Ignore essa etapa se já existir uma VPC na conta de destino.

```
# Create VPC
aws ec2 create-vpc \
  --cidr-block 10.0.0.0/16 \
  --region region

# Create subnet
aws ec2 create-subnet \
  --vpc-id vpc-id \
  --cidr-block 10.0.1.0/24 \
  --availability-zone regiona \
  --region region

# Create security group
aws ec2 create-security-group \
  --group-name opensearch-vpc-sg \
```

```
  --description "Security group for OpenSearch VPC domain" \
  --vpc-id vpc-id \
  --region region

# Allow inbound HTTPS
aws ec2 authorize-security-group-ingress \
  --group-id security-group-id \
  --protocol tcp \
  --port 443 \
  --cidr 10.0.0.0/16 \
  --region region
```

Saiba mais sobre a criação de domínios VPC.

Etapa 3: criar o domínio VPC (conta de destino)

Use o mesmo comando dethe section called “Etapa 3: criar o domínio VPC (conta de destino)”, mas atualize a política de acesso para permitir o papel do aplicativo IAM Identity Center na conta de origem:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::source-account-id:role/NeoIdCAppRole"
    },
    "Action": "es:ESHttp*",
    "Resource": "arn:aws:es:region:target-account-id:domain/vpc-domain-name/*"
  }]
}
```

Aguarde até que o status do domínio se torne `Active` antes de continuar.

Etapa 4: autorizar o VPC endpoint para o principal do serviço de interface OpenSearch do usuário (conta de destino)

> **Important**
>
> Essa é uma etapa crítica exclusiva dos domínios VPC. O serviço de OpenSearch interface do usuário deve ser explicitamente autorizado a acessar o VPC endpoint.

```
# Authorize the service principal
aws opensearch authorize-vpc-endpoint-access \
  --domain-name vpc-domain-name \
  --service "application.opensearchservice.amazonaws.com" \
  --region region

# Verify authorization
aws opensearch list-vpc-endpoint-access \
  --domain-name vpc-domain-name \
  --region region
```

Resposta esperada:

```
{
  "AuthorizedPrincipalList": [
    {
      "PrincipalType": "AWS_SERVICE",
      "Principal": "application.opensearchservice.amazonaws.com"
    }
  ]
}
```

Etapa 5: criar a função do IAM para o aplicativo IAM Identity Center (conta de origem)

Crie uma função do IAM na conta de origem que a OpenSearch UI usa para acessar o plano de dados do usuário do IAM Identity Center.

Para criar a função do aplicativo IAM Identity Center

1. Crie uma política de confiança:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "application.opensearchservice.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    },
    {
```

```
      "Effect": "Allow",
      "Principal": {
        "Service": "application.opensearchservice.amazonaws.com"
      },
      "Action": "sts:SetContext",
      "Condition": {
        "ForAllValues:ArnEquals": {
          "sts:RequestContextProviders": "arn:aws:iam::source-account-id:oidc-
provider/portal.sso.region.amazonaws.com/apl/application-id"
        }
      }
    }
  ]
}
```

2. Crie uma política de permissões:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "OpenSearchDomain",
    "Effect": "Allow",
    "Action": ["es:ESHttp*"],
    "Resource": "*"
  }]
}
```

3. Crie a função e anexe as políticas:

```
aws iam create-role \
  --role-name NeoIdCAppRole \
  --assume-role-policy-document file://neoidc-trust-policy.json

aws iam put-role-policy \
  --role-name NeoIdCAppRole \
  --policy-name NeoIdCAppPermissions \
  --policy-document file://neoidc-permissions-policy.json
```

Etapa 6: criar o aplicativo de OpenSearch interface do usuário com o IAM Identity Center (conta de origem)

```
aws opensearch create-application \
```

```
  --region region \
  --name "cross-account-vpc-idc-app" \
  --iam-identity-center-options '{
    "enabled":true,
    "iamIdentityCenterInstanceArn":"arn:aws:sso:::instance/ssoins-instance-id",
    "iamRoleForIdentityCenterApplicationArn":"arn:aws:iam::source-account-
id:role/NeoIdCAppRole"
  }' \
  --data-sources '[{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/vpc-domain-name",
    "dataSourceDescription":"Cross-account VPC domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
  }]' \
  --app-configs '[{"key":"opensearchDashboards.dashboardAdmin.users","value":"[\"*
\"]"}]'
```

Etapa 7: criar e atribuir usuários e grupos do IAM Identity Center

Crie um usuário do IAM Identity Center

Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws identitystore create-user \
  --identity-store-id d-directory-id \
  --user-name user-email \
  --display-name "display-name" \
  --name Formatted=string,FamilyName=last-name,GivenName=first-name \
  --emails Value=user-email,Type=work,Primary=true
```

Crie um grupo do IAM Identity Center e adicione o usuário

Execute os seguintes comandos :

```
aws identitystore create-group \
  --identity-store-id d-directory-id \
  --display-name "OpenSearchUsers" \
  --description "Users with OpenSearch access"

aws identitystore create-group-membership \
  --identity-store-id d-directory-id \
  --group-id group-id \
  --member-id UserId=user-id
```

Atribua o usuário ou grupo ao aplicativo

Execute este comando: .

```
aws sso-admin create-application-assignment \
  --application-arn "arn:aws:sso:::source-account-id:application/ssoins-instance-id/
apl-application-id" \
  --principal-id user-id-or-group-id \
  --principal-type USER
```

Configurar o mapeamento de funções de back-end no domínio de destino

Mapeie o grupo do IAM Identity Center para uma função de OpenSearch segurança no domínio de destino:

```
curl -XPUT "https://domain-endpoint/_plugins/_security/api/rolesmapping/all_access" \
  -u admin:master-password \
  -H 'Content-Type: application/json' \
  -d '{
    "backend_roles": ["group-id"],
    "hosts": [],
    "users": []
  }'
```

Etapa 8: verificar e acessar

```
aws opensearch get-application \
  --region region \
  --id application-id
```

- Navegue até o URL do endpoint do aplicativo.
- Faça login com as credenciais de usuário do IAM Identity Center.
- As solicitações de dados dos usuários do IAM Identity Center são assinadas com a função de aplicativo do IAM Identity Center, não com a função entre contas.
- Mapeamentos de funções de back-end nas permissões de acesso aos dados de controle de domínio.

Como gerenciar aplicações do

Atualizar um aplicativo com fontes de dados entre contas

Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearch update-application \
  --region region \
  --id application-id \
  --data-sources '[{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/domain-1",
    "dataSourceDescription":"First cross-account domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
  },{
    "dataSourceArn":"arn:aws:es:region:target-account-id:domain/domain-2",
    "dataSourceDescription":"Second cross-account domain",
    "iamRoleForDataSourceArn":"arn:aws:iam::target-account-
id:role/OpenSearchUIAccessRole"
  }]'
```

> **Important**
>
> A operação de atualização substitui toda a matriz de fontes de dados. Inclua todas as fontes de dados que você deseja manter.

Listar aplicativos

Execute este comando: .

```
aws opensearch list-applications \
  --region region
```

Deleta a aplicação

Execute este comando: .

```
aws opensearch delete-application \
  --region region \
  --id application-id
```

Revogar o acesso ao VPC endpoint

Execute este comando: .

```
aws opensearch revoke-vpc-endpoint-access \
  --domain-name vpc-domain-name \
  --service "application.opensearchservice.amazonaws.com" \
  --region region
```

Referência rápida

As tabelas a seguir resumem as principais diferenças entre os tipos de domínio e os métodos de autenticação.

Domínio público comparado ao domínio VPC

Aspecto	Domínio público	Domínio VPC
Autorização de VPC endpoint	Não obrigatório	Obrigatório — deve autorizar `application.opensearchservice.amazonaws.com`
Configuração da rede	Nenhum	VPC, sub-rede, grupo de segurança com entrada HTTPS (443)
Política de acesso do IAM	Obrigatório	Obrigatório
Função entre contas	Obrigatório para contas cruzadas	Obrigatório para contas cruzadas

Usuário do IAM comparado ao usuário do IAM Identity Center

Aspecto	IAM user (Usuário do IAM)	Usuários do IAM Identity Center
Credenciais do plano de dados	Credenciais do IAM do próprio usuário	Função do aplicativo IAM Identity Center
Controle de acesso	Política de acesso ao domínio	Política de acesso ao domínio e mapeamentos de funções de back-end

Aspecto	IAM user (Usuário do IAM)	Usuários do IAM Identity Center
Configuração adicional	Nenhum	Função do aplicativo IAM Identity Center, user/group criação, atribuição de aplicativos, mapeamento de funções de back-end
OpenSearch Configuração do aplicativo de interface do usuário	Nenhuma opção do IAM Identity Center	`--iam-identity-center-options` obrigatório

Observações importantes

- O `iamRoleForDataSourceArn` deve estar na mesma conta do`dataSourceArn`.
- Isso só `iamRoleForDataSourceArn` é necessário para fontes de dados entre contas. Omita-o para fontes de dados da mesma conta.
- A função entre contas só precisa da `es:DescribeDomain` permissão. Ele nunca é usado para acesso ao plano de dados.
- Para domínios VPC, tanto a política do IAM quanto a autorização do VPC endpoint devem ser configuradas.
- Versões de motor suportadas: OpenSearch 1.3 e superiores.

Solução de problemas

Problema	Resolução
A criação do aplicativo falha com “Não é possível acessar o domínio”	Verifique se a função entre contas tem a `es:DescribeDomain` permissão e se a política de confiança permite a conta de origem.
Falha na associação de domínio VPC	Certifique-se de que o VPC endpoint esteja autorizado para. `application.opensearchservice.amazonaws.com`

Problema	Resolução
Acesso ao plano de dados negado para o usuário do IAM	Verifique se a política de acesso ao domínio de destino permite o usuário ou o responsável pela função do IAM.
Acesso ao plano de dados negado para o usuário do IAM Identity Center	Verifique se o mapeamento da função de back-end inclui o ID do grupo do IAM Identity Center e se a política de domínio permite a função do aplicativo do IAM Identity Center.
Erro de incompatibilidade de conta	Certifique-se de que `iamRoleForDataSourceArn` esteja na mesma conta do domínio em`dataSourceArn` .

Pesquisa entre clusters

Usando a pesquisa entre clusters no Amazon OpenSearch Serverless, você pode realizar consultas e agregações em vários domínios conectados.

A pesquisa entre clusters no Amazon OpenSearch Serverless usa os conceitos de domínio de origem e domínio de destino. Uma solicitação de pesquisa entre clusters começa em um domínio de origem. O domínio de destino pode estar em um domínio diferente Conta da AWS ou Região da AWS (ou ambos) do domínio de origem para consulta. Usando a pesquisa entre clusters, você pode configurar um domínio de origem para associar à sua OpenSearch interface de usuário na mesma conta e, em seguida, criar conexões com os domínios de destino. Como resultado, você pode usar a OpenSearch interface do usuário com dados dos domínios de destino, mesmo se eles estiverem em uma conta ou região diferente.

Você paga taxas de transferência de AWS dados padrão para dados transferidos para dentro e para fora do Amazon OpenSearch Service. Você não é cobrado pelos dados transferidos entre os nós dentro do seu domínio OpenSearch de serviço. Para mais informações sobre as cobranças por dados “entrada” e “saída”, consulte Transferência de dados na página Preço sob demanda do Amazon EC2.

Você pode usar a pesquisa entre clusters como mecanismo para que sua OpenSearch interface de usuário seja associada a clusters em uma conta ou região diferente. Por padrão, as solicitações entre domínios são criptografadas em trânsito como parte da node-to-node criptografia.

> **Note**
>
> A OpenSearch ferramenta de código aberto também documenta a pesquisa entre clusters. Observe que a configuração da ferramenta de código aberto difere significativamente para clusters de código aberto em comparação com os domínios gerenciados do Amazon OpenSearch Serverless.
> Mais notavelmente, no Amazon OpenSearch Serverless, você configura conexões entre clusters usando o Console de gerenciamento da AWS em vez de usar solicitações. cURL Além disso, o serviço gerenciado usa o AWS Identity and Access Management (IAM) para autenticação entre clusters, além do controle de acesso refinado.
> Portanto, recomendamos usar o conteúdo deste tópico para configurar a pesquisa entre clusters para seus domínios em vez da documentação de código OpenSearch aberto.

Diferenças funcionais ao usar a pesquisa entre clusters

Em comparação com os domínios regulares, os domínios de destino criados usando a pesquisa entre clusters têm as seguintes diferenças e requisitos funcionais:

- Você não pode gravar nem executar comandos PUT no cluster remoto. Seu acesso ao cluster remoto é somente leitura.
- Tanto o domínio de origem quanto o de destino devem ser OpenSearch domínios. Você não pode conectar um domínio do Elasticsearch ou clusters autogerenciados do OpenSearch /Elasticsearch para interface do usuário. OpenSearch
- Um domínio pode ter um máximo de 20 conexões com outros domínios. Isso inclui conexões de saída e entrada.
- O domínio de origem deve estar na mesma versão ou em uma versão superior à OpenSearch do domínio de destino. Se você quiser configurar conexões bidirecionais entre dois domínios, os dois domínios devem ser da mesma versão. Recomendamos atualizar os dois domínios para a versão mais recente antes de fazer a conexão. Se você precisar atualizar domínios depois de configurar a conexão bidirecional, primeiro exclua e depois recrie a conexão.
- Não é possível usar dicionários personalizados ou SQL com clusters remotos.
- Você não pode usar CloudFormation para conectar domínios.
- Não é possível usar a pesquisa entre clusters em instâncias M3 ou expansíveis (T2 e T3).
- A pesquisa entre clusters não funciona para coleções Amazon OpenSearch Serverless.

Pré-requisitos de pesquisa entre clusters para UI OpenSearch

Antes de configurar a pesquisa entre clusters com dois OpenSearch domínios, certifique-se de que seus domínios atendam aos seguintes requisitos:

- O controle de acesso refinado está habilitado no domínio
- Node-to-node a criptografia está habilitada para ambos os domínios

Tópicos

- Configurar permissões de acesso para acesso a dados entre regiões e entre contas com pesquisa entre clusters
- Criar uma conexão entre domínios
- Testar a configuração de segurança para acesso a dados entre regiões e contas com pesquisa entre clusters
- Excluir uma conexão

Configurar permissões de acesso para acesso a dados entre regiões e entre contas com pesquisa entre clusters

Quando você envia uma solicitação de pesquisa entre clusters para o domínio de origem, o domínio avalia essa solicitação em relação à sua política de acesso ao domínio. A pesquisa entre clusters requer controle de acesso refinado. Este é um exemplo de uma política de acesso aberto no domínio de origem.

JSON

```
{
"Version":"2012-10-17",
"Statement": [
    {
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "*"
        ]
      },
      "Action": [
```

```
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111222333444:domain/src-domain/*"
    }
  ]
}
```

> **Note**
>
> Se você incluir índices remotos no caminho, deverá codificar em URL o URI no ARN do domínio.
> Por exemplo, use o seguinte formato do ARN: `:arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst%3Aremote_index`
> Não use o seguinte formato de ARN: `arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst:remote_index.`

Se você escolher por usar uma política de acesso restritiva além do controle de acesso refinado, sua política deverá permitir o acesso, pelo menos, a `es:ESHttpGet`. Este é um exemplo:

JSON

```
{
"Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111222333444:user/john-doe"
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/my-domain/*"
    }
  ]
}
```

O controle de acesso refinado no domínio de origem avalia a solicitação para determinar se ela está assinada com credenciais de autenticação básica HTTP ou do IAM. Se estiver, o controle de acesso refinado avalia então se o usuário tem permissão para realizar a pesquisa e acessar os dados.

Os requisitos de permissão para pesquisas são os seguintes:

- Se a solicitação pesquisar somente dados no domínio de destino (por exemplo, `dest-alias:dest-index/_search`)), serão necessárias permissões somente no domínio de destino.
- Se a solicitação pesquisar dados em ambos os domínios (por exemplo,`source-index,dest-alias:dest-index/_search`)) serão necessárias permissões em ambos os domínios.
- Para usar o controle de acesso detalhado, a permissão `indices:admin/shards/search_shards` é necessária além das permissões padrão de leitura ou pesquisa para os índices relevantes.

O domínio de origem passa a solicitação para o domínio de destino. O domínio de destino avalia essa solicitação em relação à política de acesso ao domínio. Para oferecer suporte a todos os recursos da OpenSearch interface do usuário, como indexação de documentos e realização de pesquisas padrão, é necessário definir permissões completas. Este é um exemplo de nossa política recomendada no domínio de destino:

JSON

```
{
"Version":"2012-10-17",
 "Statement": [
    {
       "Effect": "Allow",
       "Principal": {
         "AWS": [
           "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-2:111222333444:domain/my-destination-
domain/*"
    },
    {
```

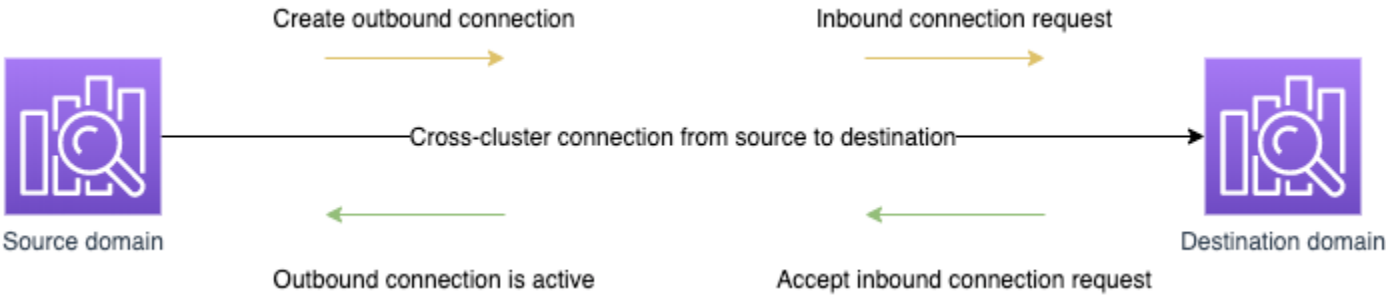
```
    "Effect": "Allow",
        "Principal": {
    "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-2:111222333444:domain/"
    }
  ]
}
```

Se você quiser realizar somente pesquisas básicas, o requisito mínimo de política é que a permissão `es:ESCrossClusterGet` seja aplicada ao domínio de destino, sem compatibilidade com curinga. Por exemplo, na política anterior, você especificaria o nome do domínio como */my-destination-domain* e não*/my-destination-domain/**.

Nesse caso, o domínio de destino realiza a pesquisa e retorna os resultados para o domínio de origem. O domínio de origem combina seus próprios resultados (se houver) com os resultados do domínio de destino e os retorna para você.

Criar uma conexão entre domínios

Uma conexão de pesquisa entre clusters é unidirecional, do domínio de origem para o domínio de destino. Isso significa que os domínios de destino (em uma conta ou região diferente) não podem consultar o domínio de origem, que é local para a OpenSearch interface do usuário. O domínio de origem cria uma conexão de saída com o domínio de destino. O domínio de destino recebe uma solicitação de conexão de entrada do domínio de origem.



Para criar uma conexão entre domínios

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, selecione Domínios.

3. Escolha o nome de um domínio para servir como domínio de origem e depois escolha a guia Conexões.
4. Na área Conexões de saída, escolha Solicitar.
5. Em Alias de conexão, insira um nome para a conexão. O alias de conexão é usado na OpenSearch interface do usuário para selecionar os domínios de destino.
6. Em Modo de conexão, escolha Direto para pesquisas ou replicação entre clusters.
7. Para especificar que a conexão deve ignorar clusters indisponíveis durante uma pesquisa, selecione a caixa Ignorar clusters indisponíveis. Essa configuração garante que as consultas entre clusters retornem resultados parciais, mesmo que ocorram falhas em um ou mais clusters remotos.
8. Em Cluster de destino, escolha entre Conectar a um cluster neste Conta da AWS e Conectar a um cluster em outro Conta da AWS.
9. Em ARN de domínio remoto, insira nome do recurso da Amazon (ARN)(ARN) para o cluster. O ARN do domínio pode estar localizado na área Informações gerais da página de detalhes do domínio.

   O domínio deve atender aos seguintes requisitos:

   - O ARN deve estar no formato `arn:`*`partition`*`:es:`*`regionaccount-id`*`:`*`type`*`/`*`domain-id`*. Por exemplo:

     ```
     arn:aws:es:us-east-2:111222333444:domain/my-domain
     ```
   - O domínio deve ser configurado para usar a OpenSearch versão 1.0 (ou posterior) ou a versão 6.7 do Elasticsearch (ou posterior).
   - O controle de acesso refinado deve estar habilitado no domínio.
   - O domínio deve estar em execução OpenSearch.
10. Escolha Solicitar.

A pesquisa entre clusters primeiro valida a solicitação de conexão para ter certeza de que os pré-requisitos são atendidos. Se os domínios forem incompatíveis, a solicitação de conexão entrará no estado `Validation failed`.

Se a solicitação de conexão for validada com êxito, ela é enviada para o domínio de destino, onde precisa ser aprovada. Até que essa aprovação seja fornecida, a conexão permanecerá em um estado `Pending acceptance`. Quando a solicitação de conexão é aceita no domínio de destino, o estado muda para `Active` e o domínio de destino torna-se disponível para consultas.

A página de domínio mostra os detalhes gerais da integridade do domínio e da instância do domínio de destino. Os proprietários de domínios têm a flexibilidade de criar, visualizar, remover e monitorar conexões de saída e de entrada de seus domínios.

Depois de estabelecida a conexão, todo o tráfego que flui entre os nós dos domínios conectados é criptografado. Se você conectar um domínio da VPC a um domínio que não seja de VPC e este domínio for um endpoint público que pode receber tráfego da Internet, o tráfego entre clusters entre os domínios ainda será criptografado e seguro.

Testar a configuração de segurança para acesso a dados entre regiões e contas com pesquisa entre clusters

Depois de configurar as permissões de acesso para acesso a dados entre regiões e contas com a pesquisa entre clusters, recomendamos testar a configuração usando o Postman, uma plataforma de terceiros para desenvolvimento colaborativo de APIs.

Para definir a configuração de segurança usando o Postman

1. No domínio de destino, indexe um documento. Este é um exemplo de solicitação:

```
POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1
{
"Dracula": "Bram Stoker"
}
```

2. Para consultar esse índice do domínio de origem, inclua o alias de conexão do domínio de destino dentro da consulta. Você pode encontrar o alias de conexão na guia Conexões no painel do domínio. A seguir está um exemplo de solicitação e resposta truncada:

```
GET https://src-domain.us-east-1.es.amazonaws.com/connection_alias:books/_search
{
...
  "hits": [
{
"_index": "source-destination:books",
  "_type": "_doc",
  "_id": "1",
  "_score": 1,
  "_source": {
"Dracula": "Bram Stoker"
  }
```

```
}
  ]
}
```

3. (Opcional) Você pode criar uma configuração que inclua vários domínios em uma única pesquisa. Por exemplo, digamos que você configurou o seguinte:

   Uma conexão entre `domain-a` e `domain-b`, com um alias de conexão denominado `cluster_b`

   Uma conexão entre `domain-a` e `domain-c`, com um alias de conexão denominado `cluster_c`

   Nesse caso, as pesquisas incluem o conteúdo `domain-a`, `domain-b` e `domain-c`. Este é um exemplo de solicitação e resposta truncada:

   Solicitação

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
"match": {
  "user": "domino"
}
  }
}
```

   Resposta:

```
{
"took": 150,
  "timed_out": false,
  "_shards": {
"total": 3,
"successful": 3,
"failed": 0,
"skipped": 0
  },
  "_clusters": {
"total": 3,
"successful": 3,
```

```
"skipped": 0
  },
  "hits": {
"total": 3,
"max_score": 1,
"hits": [
  {
"_index": "local_index",
    "_type": "_doc",
    "_id": "0",
    "_score": 1,
    "_source": {
"user": "domino",
      "message": "This is message 1",
      "likes": 0
    }
  },
  {
"_index": "cluster_b:b_index",
    "_type": "_doc",
    "_id": "0",
    "_score": 2,
    "_source": {
"user": "domino",
      "message": "This is message 2",
      "likes": 0
    }
  },
  {
"_index": "cluster_c:c_index",
    "_type": "_doc",
    "_id": "0",
    "_score": 3,
    "_source": {
"user": "domino",
      "message": "This is message 3",
      "likes": 0
    }
  }
]
  }
}
```

Se você não optou por ignorar clusters indisponíveis na sua configuração de conexão, todos os clusters de destino na sua pesquisa precisam estar disponíveis para que sua solicitação de pesquisa seja executada com êxito. Caso contrário, toda a solicitação falhará — mesmo que um dos domínios não esteja disponível, nenhum resultado da pesquisa será retornado.

Excluir uma conexão

A exclusão de uma conexão interrompe qualquer operação de pesquisa entre clusters no domínio de destino.

Você pode executar o seguinte procedimento no domínio de origem e no domínio de destino para remover a conexão. Depois que a conexão é removida, ela permanece visível com o status `Deleted` por 15 dias.

Não é possível excluir um domínio com conexões ativas entre clusters. Para excluir um domínio, primeiro remova todas as conexões de entrada e saída desse domínio. Isso garante que você leve em consideração os usuários de domínio entre clusters antes de excluir o domínio.

Para excluir uma conexão

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, selecione Domínios.
3. Escolha o nome de um domínio existente e depois a guia Conexões.
4. Selecione nome de uma conexão a ser excluída.
5. Escolha Excluir e depois confirme a exclusão.

Gerenciando o acesso à OpenSearch interface do usuário a partir de um VPC endpoint

Você pode criar uma conexão privada entre sua VPC e a OpenSearch interface do usuário usando o. AWS PrivateLink Usando essa conexão, você pode acessar aplicativos de OpenSearch interface do usuário como se estivessem na mesma VPC. Dessa forma, você não precisa configurar um gateway de internet, dispositivo NAT, conexão VPN ou AWS Direct Connect estabelecer a conexão. As instâncias na sua VPC não precisam de endereços IP públicos para acessar OpenSearch a interface do usuário.

Para estabelecer essa conexão privada, primeiro você cria um endpoint de interface alimentado por AWS PrivateLink. Uma interface de rede de endpoint é criada automaticamente em cada sub-rede que você especifica para o endpoint da interface. Essas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado a aplicativos de interface do usuário. OpenSearch

Criação de uma conexão privada entre uma VPC e uma interface do usuário OpenSearch

Você pode criar uma conexão privada para acessar a OpenSearch interface de usuário de uma VPC usando o Console de gerenciamento da AWS ou. AWS CLI

Criação de uma conexão privada entre uma VPC e uma OpenSearch interface de usuário (console)

Para criar uma conexão privada entre uma VPC e uma OpenSearch interface de usuário usando o console

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, em Tecnologia sem servidor, escolha Endpoints da VPC.
3. Escolha Criar endpoint da VPC.
4. Em Nome, insira um nome para o endpoint.
5. Para VPC, selecione a VPC a partir da qual você OpenSearch acessará os aplicativos de interface do usuário.
6. Em Sub-redes, selecione uma sub-rede a partir da qual você acessará os aplicativos de OpenSearch interface do usuário.

> **Note**
>
> O endereço IP e o tipo de DNS do endpoint são baseados no tipo de sub-rede:
>
> - Pilha dupla: se todas as sub-redes tiverem intervalos de endereços. IPv4 IPv6
> - IPv6: Se todas as sub-redes forem IPv6 somente sub-redes.
> - IPv4: se todas as sub-redes tiverem intervalos de IPv4 endereços.

7. Em Grupos de segurança, selecione um ou mais grupos de segurança para associar às interfaces de rede de endpoint.

> **Note**
>
> Nesta etapa, você limitará as portas, protocolos e fontes de tráfego de entrada que está autorizando para o endpoint. Certifique-se de que as regras do grupo de segurança permitam que os recursos que usarão o VPC endpoint para se comunicar com os aplicativos de OpenSearch interface do usuário também se comuniquem com a interface de rede do endpoint.

8. 8. Escolha Criar endpoint.

Criação de uma conexão privada entre uma VPC e uma OpenSearch UI ()AWS CLI

Para criar uma conexão privada entre uma VPC e uma OpenSearch interface de usuário usando o AWS CLI

Execute o comando a seguir. Substitua *placeholder values* por suas próprias informações.

```
aws opensearchserverless create-vpc-endpoint \
    --region region \
    --endpoint endpoint \
    --name vpc_endpoint_name \
    --vpc-id vpc_id \
    --subnet-ids subnet_ids
```

Atualização da política de VPC endpoint para permitir acesso ao aplicativo de interface do usuário OpenSearch

Depois de criar a conexão privada, atualize a política de VPC endpoint para permitir o acesso ao aplicativo de OpenSearch UI na política de VPC endpoint especificando o ID do aplicativo.

Para saber mais sobre a atualização de uma política de endpoint da VPC, consulte Atualizar uma política de endpoint da VPC no Guia do AWS PrivateLink .

Certifique-se de que a política de endpoint da VPC inclua a instrução a seguir. Substitua *placeholder value* por suas próprias informações.

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": ["opensearch-ui-application-id"]
            }
        }
    }]
}
```

Revogando o acesso à OpenSearch interface do usuário em uma política de VPC endpoint

OpenSearch A interface do usuário exige permissão explícita na política de endpoint da VPC para permitir que os usuários acessem o aplicativo a partir da VPC. Se você não quiser mais que os usuários acessem a OpenSearch interface do usuário da VPC, você pode remover a permissão na política de endpoint. Depois disso, os usuários encontram uma mensagem `403 forbidden` de erro ao tentar acessar a OpenSearch interface do usuário.

Para saber mais sobre a atualização de uma política de endpoint da VPC, consulte Atualizar uma política de endpoint da VPC no Guia do AWS PrivateLink .

Este é um exemplo de política de endpoint da VPC que nega acesso aos aplicações de interface do usuário na VPC:

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": [""]
            }
        }
    }]
```

```
}
```

Migração de objetos salvos dos OpenSearch painéis para a interface do usuário OpenSearch

Se você tiver painéis, visualizações, padrões de índice e outros objetos salvos nos OpenSearch Painéis, poderá migrá-los e reutilizá-los na interface do usuário. OpenSearch

Benefícios da migração para a OpenSearch interface do usuário:

- Alta disponibilidade — a OpenSearch interface do usuário é hospedada no Nuvem AWS e permanece disponível durante as atualizações e a manutenção do domínio, enquanto os OpenSearch painéis são hospedados no domínio e ficarão temporariamente indisponíveis.
- Várias fontes de dados — A OpenSearch interface do usuário pode fornecer um único painel consolidado em várias fontes de dados, incluindo OpenSearch domínios, coleções sem servidor e conexões de dados com o Amazon S3 e a CloudWatch Amazon; enquanto OpenSearch cada painel só pode funcionar com um domínio ou coleção.
- Recursos adicionais, como AI Assistant e Workspaces, estão disponíveis na OpenSearch interface do usuário. Saiba mais:OpenSearch UI.

Tópicos

- Visão geral da migração
- Pré-requisitos
- Etapa 1: exportar objetos salvos dos OpenSearch painéis
- Etapa 2: importar objetos salvos para a OpenSearch interface do usuário

Visão geral da migração

O processo de migração consiste nas seguintes etapas de alto nível:

1. Exporte objetos salvos dos OpenSearch painéis — Use a interface de gerenciamento de objetos salvos do OpenSearch painel ou a API de exportação para baixar seus painéis, visualizações, padrões de índice e outros objetos como um arquivo JSON delimitado por nova linha (NDJSON).

2. Crie um aplicativo de OpenSearch interface de usuário e um espaço de trabalho — Se você ainda não tiver feito isso, crie um aplicativo de OpenSearch interface de usuário e um espaço de trabalho para receber os objetos importados.
3. Registre a fonte de dados na OpenSearch interface do usuário — associe seu OpenSearch domínio ao aplicativo de OpenSearch interface do usuário e registre-o como uma fonte de dados no espaço de trabalho. Os padrões de índice em seus objetos importados fazem referência a essa fonte de dados.
4. Importar objetos salvos para a OpenSearch interface do usuário — Use a OpenSearch interface de gerenciamento de objetos salvos da interface do usuário ou a API de importação para carregar o arquivo NDJSON no espaço de trabalho de destino.
5. Valide os objetos importados — Abra seus painéis e visualizações na OpenSearch interface do usuário para verificar se eles são exibidos corretamente e se os dados estão fluindo do domínio ou da coleção associada.

Pré-requisitos

Antes da migração, verifique os seguintes pré-requisitos:

- Você tem as permissões do IAM necessárias para chamar o Amazon OpenSearch Service OpenSearch APIs e. Para obter mais informações, consulte the section called “Permissões necessárias para criar aplicativos do Amazon OpenSearch Service”.
- Você pode acessar o domínio ou a coleção e os OpenSearch painéis que deseja migrar.
- Você criou um aplicativo de OpenSearch interface do usuário. Para obter informações sobre como criar um aplicativo e um espaço de trabalho, consultethe section called “Introdução”.
- Você associou o mesmo domínio ou coleção ao aplicativo de OpenSearch interface do usuário. Para obter informações sobre a associação de fontes de dados, consultethe section called “Gerenciar associações de fontes de dados e permissões de acesso à VPC”.

> **Note**
>
> OpenSearch A interface do usuário só é compatível com a OpenSearch versão 1.3 e versões posteriores. Verifique se seu OpenSearch domínio está executando a versão 1.3 ou superior antes de tentar migrar objetos salvos.

Etapa 1: exportar objetos salvos dos OpenSearch painéis

Exporte seus objetos salvos dos OpenSearch painéis usando a interface de gerenciamento ou a API de exportação. A exportação produz um arquivo JSON delimitado por nova linha (NDJSON) que contém todos os tipos de objetos salvos selecionados e suas dependências.

Tópicos

- Exportar manualmente em OpenSearch painéis
- Exportação via API

Exportar manualmente em OpenSearch painéis

Para exportar objetos salvos usando a interface de gerenciamento de OpenSearch painéis

1. Abra sua instância do OpenSearch Dashboards.
2. No painel de navegação esquerdo, escolha Gerenciamento.
3. Em Gerenciamento de painéis, escolha Objetos salvos.
4. Selecione os objetos salvos que você deseja exportar. Para exportar todos os objetos de um tipo específico, filtre por tipo usando a barra de pesquisa. Para exportar todos os objetos, marque a caixa de seleção no cabeçalho da tabela.
5. Escolha Exportar.
6. Na caixa de diálogo Exportar objetos salvos, verifique se a opção Incluir objetos relacionados está selecionada. Essa opção inclui todos os objetos dos quais os objetos salvos selecionados dependem, como padrões de índice referenciados por visualizações. Desmarque essa opção somente se você pretende gerenciar dependências manualmente.
7. Escolha Exportar para baixar o `.ndjson` arquivo para sua máquina local.

> **Tip**
>
> Quando você escolhe Incluir objetos relacionados, o arquivo NDJSON exportado contém todos os objetos salvos necessários para renderizar os painéis e visualizações selecionados, incluindo seus padrões de índice dependentes, visualizações e objetos de pesquisa. Isso simplifica a etapa de importação e evita erros de referência ausentes.

Exportação via API

Você pode usar a API de exportação de objetos salvos do OpenSearch Dashboards para exportar objetos salvos de forma programática. Isso é útil para automatizar migrações ou integrar a etapa de exportação em um pipeline. CI/CD

> **Note**
>
> Se o seu OpenSearch domínio tiver um controle de acesso refinado ativado, você deverá fornecer credenciais de autenticação com sua solicitação de exportação. Use a autenticação básica HTTP adicionando a `-u` bandeira com seu nome de usuário e senha. Para obter mais informações sobre as opções de autenticação, consulte the section called “Controle de acesso refinado”.

O exemplo a seguir exporta todos os painéis com seus objetos relacionados. Substitua *`placeholder values`* por suas próprias informações.

```
curl -X POST \
    "https://dashboards-endpoint/_dashboards/api/saved_objects/_export" \
    -u 'master-username:master-password' \
    -H "Content-Type: application/json" \
    -H "osd-xsrf: true" \
    -d '{
        "type": ["dashboard", "visualization", "index-pattern", "search"],
        "includeReferencesDeep": true,
        "excludeExportDetails": false
    }' \
    -o saved-objects-export.ndjson
```

Se o seu domínio não tiver um controle de acesso refinado ativado, você poderá omitir a sinalização. `-u`

Para exportar objetos salvos específicos por ID, use o `objects` parâmetro em vez de`type`:

```
curl -X POST \
    "https://dashboards-endpoint/_dashboards/api/saved_objects/_export" \
    -u 'master-username:master-password' \
    -H "Content-Type: application/json" \
    -H "osd-xsrf: true" \
    -d '{
```

```
        "objects": [
            {"type": "dashboard", "id": "dashboard-id"},
            {"type": "visualization", "id": "visualization-id"}
        ],
        "includeReferencesDeep": true
    }' \
    -o saved-objects-export.ndjson
```

> **Note**
>
> Para encontrar um objeto salvo IDs, você pode usar a API Saved Objects para listar todos os objetos de um tipo específico. O exemplo a seguir lista todos os painéis:
>
> ```
> curl -X GET \
>     "https://dashboards-endpoint/_dashboards/api/saved_objects/_find?
> type=dashboard" \
>     -u 'master-username:master-password'
> ```
>
> A resposta inclui o ID de cada objeto salvo. Você também pode encontrar o ID na URL do navegador ao visualizar o objeto nos OpenSearch painéis.

Etapa 2: importar objetos salvos para a OpenSearch interface do usuário

Depois de exportar os objetos salvos, você pode importar o arquivo NDJSON para a OpenSearch interface do usuário manualmente ou via API.

Tópicos

- Importar manualmente na OpenSearch interface do usuário
- Importar via API

Importar manualmente na OpenSearch interface do usuário

Para importar objetos salvos usando a OpenSearch interface de usuário de gerenciamento de interface

1. Abra seu aplicativo de OpenSearch interface do usuário e navegue até o espaço de trabalho de destino.

2. No espaço de trabalho, escolha Ativos na navegação superior ou acesse a página de ativos do espaço de trabalho.
3. Escolha Importar para abrir a caixa de diálogo Importar ativos.
4. Escolha Selecionar arquivo e selecione o `.ndjson` arquivo que você exportou dos OpenSearch painéis.
5. Para Gerenciamento de conflitos, escolha uma das seguintes opções:
   - Crie novos ativos com recursos exclusivos IDs (padrão) — Gera novos IDs para todos os objetos importados, evitando conflitos com ativos existentes.
   - Verificar ativos existentes — Verifica se há conflitos com objetos existentes. Quando selecionado, escolha uma das seguintes subopções:
     - Substituir conflitos automaticamente — os ativos existentes com a mesma ID são substituídos automaticamente.
     - Solicitar ação em caso de conflito — Você será solicitado a resolver cada conflito individualmente.
6. Escolha Importar.
7. Revise o resumo da importação.

Importar via API

Para importar objetos salvos usando a API com a autenticação AWS Signature versão 4, você deve primeiro obter o ID da fonte de dados e depois usá-lo na solicitação de importação. Substitua *`placeholder values`* por suas próprias informações.

Etapa 1: Obtenha o ID da fonte de dados do seu espaço de trabalho:

```
curl -X GET \
    "https://opensearch-ui-endpoint/w/workspace-id/api/saved_objects/_find?type=data-
source" \
    --aws-sigv4 "aws:amz:region:opensearch" \
    --user "$AWS_ACCESS_KEY_ID:$AWS_SECRET_ACCESS_KEY" \
    -H "x-amz-security-token: $AWS_SESSION_TOKEN" \
    -H "osd-xsrf: true"
```

> **Note**
>
> A resposta inclui o ID da fonte de dados. Você também pode encontrar o ID da fonte de dados na URL do navegador ao visualizar a fonte de dados na OpenSearch interface do usuário.

Etapa 2: importar os objetos salvos usando o ID da fonte de dados da Etapa 1:

```
curl -X POST \
    "https://opensearch-ui-endpoint/w/workspace-id/api/saved_objects/_import?
overwrite=true&dataSourceId=data-source-id" \
    --aws-sigv4 "aws:amz:region:opensearch" \
    --user "$AWS_ACCESS_KEY_ID:$AWS_SECRET_ACCESS_KEY" \
    -H "x-amz-security-token: $AWS_SESSION_TOKEN" \
    -H "osd-xsrf: true" \
    -F "file=@saved-objects-export.ndjson"
```

> **Note**
>
> Esses exemplos usam a `--aws-sigv4` opção integrada do curl (disponível no curl 7.75 ou posterior) para assinar as solicitações. Defina suas AWS credenciais como variáveis de ambiente antes de executar os comandos:`AWS_ACCESS_KEY_ID`,`AWS_SECRET_ACCESS_KEY`, e `AWS_SESSION_TOKEN` (se estiver usando credenciais temporárias).

OpenSearch Endpoints e cotas de interface do usuário

Para se conectar programaticamente a um AWS serviço, você usa um endpoint. As service quotas, também chamadas de limites, correspondem ao número máximo de recursos ou operações de serviço para sua conta da AWS .

A Amazon OpenSearch UI é a interface de OpenSearch usuário da próxima geração para OpenSearch painéis. Ele fornece endpoints para acessar seus OpenSearch painéis. Use este tópico para encontrar os endpoints de serviço e as cotas de serviço para OpenSearch a interface do usuário.

Para obter mais informações sobre outros OpenSearch serviços, consulte Pontos finais e cotas de serviço.

OpenSearch Endpoints de interface do usuário

OpenSearch A interface do usuário está disponível nas seguintes regiões:

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Mumbai)	ap-south-1	opensearch.ap-south-1.amazonaws.com es.ap-south-1.amazonaws.com	HTTPS HTTPS
Europa (Paris)	eu-west-3	opensearch.eu-west-3.amazonaws.com es.eu-west-3.amazonaws.com	HTTPS HTTPS
Leste dos EUA (Ohio)	us-east-2	opensearch.us-east-2.amazonaws.com es.us-east-2.amazonaws.com	HTTPS HTTPS
Europa (Irlanda)	eu-west-1	opensearch.eu-west-1.amazonaws.com es.eu-west-1.amazonaws.com	HTTPS HTTPS
Europa (Frankfurt)	eu-central-1	opensearch.eu-central-1.amazonaws.com es.eu-central-1.amazonaws.com	HTTPS HTTPS
América do Sul (São Paulo)	sa-east-1	opensearch.sa-east-1.amazonaws.com es.sa-east-1.amazonaws.com	HTTPS HTTPS

Nome da região	Região	Endpoint	Protocolo
Leste dos EUA (Norte da Virgínia)	us-east-1	opensearch.us-east-1.amazonaws.com es.us-east-1.amazonaws.com	HTTPS HTTPS
Europa (Londres)	eu-west-2	opensearch.eu-west-2.amazonaws.com es.eu-west-2.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Tóquio)	ap-northeast-1	opensearch.ap-northeast-1.amazonaws.com es.ap-northeast-1.amazonaws.com	HTTPS HTTPS
Oeste dos EUA (Oregon)	us-west-2	opensearch.us-west-2.amazonaws.com es.us-west-2.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Singapura)	ap-southeast-1	opensearch.ap-southeast-1.amazonaws.com es.ap-southeast-1.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Sydney)	ap-southeast-2	opensearch.ap-southeast-2.amazonaws.com es.ap-southeast-2.amazonaws.com	HTTPS HTTPS
Canadá (Central)	ca-central-1	opensearch.ca-central-1.amazonaws.com es.ca-central-1.amazonaws.com	HTTPS HTTPS
Europa (Estocolmo)	eu-north-1	opensearch.eu-north-1.amazonaws.com es.eu-north-1.amazonaws.com	HTTPS HTTPS

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Hong Kong)	ap-east-1	opensearch.ap-east-1.amazonaws.com es.ap-east-1.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Seul)	ap-northeast-2	opensearch.ap-northeast-2.amazonaws.com es.ap-northeast-2.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Osaka)	ap-northeast-3	opensearch.ap-northeast-3.amazonaws.com es.ap-northeast-3.amazonaws.com	HTTPS HTTPS
Ásia-Pacífico (Hyderabad)	ap-south-2	opensearch.ap-south-2.amazonaws.com es.ap-south-2.amazonaws.com	HTTPS HTTPS
Europa (Espanha)	eu-south-2	opensearch.eu-south-2.amazonaws.com es.eu-south-2.amazonaws.com	HTTPS HTTPS
Oeste dos EUA (Norte da Califórnia)	us-west-1	opensearch.us-west-1.amazonaws.com es.us-west-1.amazonaws.com	HTTPS HTTPS
Europa (Zurique)	eu-central-2	opensearch.eu-central-2.amazonaws.com es.eu-central-2.amazonaws.com	HTTPS HTTPS
Europa (Milão)	eu-south-1	opensearch.eu-south-1.amazonaws.com es.eu-south-1.amazonaws.com	HTTPS HTTPS

OpenSearch Cotas de serviços de interface do usuário

Sua AWS conta tem as seguintes cotas relacionadas aos recursos de OpenSearch interface do usuário.

Nome	Padrão	Ajustável	Observações
OpenSearch Aplicativos de interface de usuário por conta por região	30	Sim	O número máximo de aplicativos de OpenSearch interface de usuário que você pode criar por conta por região. Você pode aumentar o limite para 50 usando a cota de serviço com aprovação automática. Para solicitar um limite maior, envie um ticket de suporte.

Gerenciamento de índices no Amazon Service OpenSearch

Depois de adicionar dados ao Amazon OpenSearch Service, você geralmente precisa reindexar esses dados, trabalhar com aliases de índice, mover um índice para um armazenamento mais econômico ou excluí-lo completamente. Este capítulo aborda soluções de armazenamento em camadas, gerenciamento do estado do índice e outras operações de indexação. Para obter informações sobre o OpenSearch índice APIs, consulte a OpenSearch documentação.

Tópicos

- OpenSearch Instâncias otimizadas para domínios OpenSearch do Amazon Service
- Armazenamento em vários níveis para Amazon Service OpenSearch
- UltraWarm armazenamento para Amazon OpenSearch Service
- Armazenamento a frio para Amazon OpenSearch Service
- Gerenciamento do estado do índice no Amazon OpenSearch Service
- Resumindo índices no Amazon OpenSearch Service com pacotes cumulativos de índices
- Transformação de índices no Amazon Service OpenSearch
- Replicação entre clusters para Amazon Service OpenSearch
- Migração de índices do Amazon OpenSearch Service usando reindexação remota
- Gerenciando dados de séries temporais no Amazon OpenSearch Service com fluxos de dados

OpenSearch Instâncias otimizadas para domínios OpenSearch do Amazon Service

A família de instâncias OpenSearch otimizada para o Amazon OpenSearch Service é uma solução econômica para armazenar grandes volumes de dados. Um domínio com instâncias OpenSearch otimizadas usa o armazenamento local como principal, com dados copiados de forma síncrona para o Amazon S3 à medida que chegam. Essa estrutura de armazenamento fornece maior produtividade de indexação com alta durabilidade. OR1, OR2, OM2 usa o Amazon Elastic Block Store (Amazon EBSgp3) `io1` ou volumes localmente, OI2 enquanto as instâncias usam discos locais. NVMe A família de instâncias OpenSearch otimizada também oferece suporte à recuperação automática de dados em caso de falha. Para obter informações sobre as opções OpenSearch otimizadas de tipo de instância, consultethe section called “Tipos de instâncias da geração atual”.

Se você estiver indexando cargas de trabalho de análise operacional pesadas, como análise de log, observabilidade ou análise de segurança, você pode se beneficiar do desempenho aprimorado e da eficiência computacional das instâncias otimizadas. OpenSearch Além disso, a recuperação automática de dados oferecida por instâncias OpenSearch otimizadas melhora a confiabilidade geral do seu domínio.

OpenSearch O serviço envia métricas de instância OpenSearch otimizadas relacionadas ao armazenamento para a Amazon. CloudWatch Para ver uma lista das métricas disponíveis, consulte ???.

OpenSearch as instâncias otimizadas estão disponíveis sob demanda ou com preços de instância reservada, com uma taxa horária para as instâncias e o armazenamento provisionados no Amazon EBS e no Amazon S3.

Tópicos

- Limitações
- Ajustar para obter melhor throughput de ingestão
- Como as instâncias OpenSearch otimizadas diferem de outras instâncias
- Como as instâncias OpenSearch otimizadas diferem das UltraWarm instâncias
- Provisionamento de um domínio com OpenSearch instâncias otimizadas

Limitações

Considere as seguintes limitações ao usar instâncias OpenSearch otimizadas para seu domínio.

- Os domínios recém-criados devem estar executando a OpenSearch versão 2.11 ou superior.
- Os domínios existentes devem estar executando a OpenSearch versão 2.15 ou superior.
- A criptografia em repouso deve estar habilitada no domínio. Para saber mais, consulte ???.
- Se o domínio usar nós principais dedicados, eles deverão usar instâncias do Graviton. Para saber mais sobre nós principais dedicados, consulte ???.
- O intervalo de atualização dos índices em instâncias OpenSearch otimizadas deve ser de 10 segundos ou mais. O intervalo de atualização padrão para instâncias OpenSearch otimizadas é de 10 segundos.

Ajustar para obter melhor throughput de ingestão

Para obter a melhor taxa de transferência de indexação de suas instâncias OpenSearch otimizadas, recomendamos que você faça o seguinte:

- Use grandes volumes para melhorar a utilização do buffer. O tamanho recomendado são 10 MB.
- Use vários clientes para melhorar a performance de processamento paralelo.
- Defina o número de fragmentos primários ativos de acordo com o número de nós de dados para maximizar a utilização dos recursos.

Como as instâncias OpenSearch otimizadas diferem de outras instâncias

OpenSearch as instâncias otimizadas diferem das instâncias não otimizadas das seguintes maneiras:

- Para instâncias OpenSearch otimizadas, a indexação é realizada somente em fragmentos primários.
- Se as instâncias OpenSearch otimizadas forem configuradas com réplicas, a taxa de indexação poderá parecer menor do que realmente é. Por exemplo, se houver um fragmento primário e um fragmento-réplica, a taxa de indexação poderá mostrar uma taxa de 1000 quando a taxa de indexação verdadeira é de 2000.
- OpenSearch instâncias otimizadas realizam operações de buffer antes de serem enviadas para uma fonte remota. Isso gera latências de ingestão mais altas.

  > **Note**
  >
  > A métrica `IndexingLatency` não é afetada, pois não inclui o tempo de sincronização do translog.

- Os fragmentos-réplica podem estar alguns segundos atrás dos fragmentos primários. Você pode monitorar o atraso usando a métrica da `ReplicationLagMaxTime` Amazon CloudWatch

Como as instâncias OpenSearch otimizadas diferem das UltraWarm instâncias

OpenSearch O serviço fornece UltraWarm instâncias que são uma forma econômica de armazenar grandes quantidades de dados somente para leitura. Tanto as OpenSearch otimizadas quanto as UltraWarm instâncias armazenam dados localmente no Amazon EBS e remotamente no Amazon S3. No entanto, as UltraWarm instâncias OpenSearch otimizadas e as instâncias diferem de várias maneiras importantes:

- OpenSearch as instâncias otimizadas mantêm uma cópia dos dados em sua loja local e remota. Em UltraWarm alguns casos, os dados são mantidos principalmente em uma loja remota para reduzir os custos de armazenamento. Dependendo de seus padrões de uso, os dados podem ser movidos para o armazenamento local.
- OpenSearch as instâncias otimizadas estão ativas e podem aceitar operações de leitura e gravação, enquanto os dados nas UltraWarm instâncias são somente para leitura até que você os mova manualmente de volta para o armazenamento dinâmico.
- UltraWarm depende de instantâneos de índice para durabilidade dos dados. OpenSearch instâncias otimizadas, em comparação, realizam replicação e recuperação nos bastidores. No caso de um índice vermelho, as instâncias OpenSearch otimizadas restaurarão automaticamente os fragmentos perdidos do seu armazenamento remoto no Amazon S3. O tempo de recuperação varia dependendo do volume de dados a ser recuperado.

Para obter mais informações sobre UltraWarm armazenamento, consulte???.

Provisionamento de um domínio com OpenSearch instâncias otimizadas

Você pode selecionar instâncias OpenSearch otimizadas para seus nós de dados ao criar um novo domínio com o Console de gerenciamento da AWS ou o AWS Command Line Interface (AWS CLI). É possível indexar e consultar os dados usando as ferramentas existentes.

Console

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, selecione Domínios.
3. Escolha Criar domínio.

4. Na seção Número de nós de dados, expanda o menu Família de instâncias e escolha OpenSearch otimizado.
5. Escolha o tipo de instância e outras configurações de armazenamento.
6. Na seção Criptografia, certifique-se de que a opção Habilitar criptografia de dados em repouso esteja selecionada.
7. Configure o resto do domínio e escolha Criar.

AWS CLI

Para provisionar um domínio que usa armazenamento OpenSearch otimizado usando o. AWS CLI, você deve fornecer o valor do tamanho específico do tipo de instância (como OR1 OR2 OM2,, ou OI2) no`InstanceType`.

O exemplo a seguir cria um domínio com OR1 instâncias de tamanho `2xlarge` e permite a criptografia em repouso.

```
aws opensearch create-domain \
  --domain-name test-domain \
  --engine-version OpenSearch_2.11 \
  --cluster-config
 "InstanceType=or1.2xlarge.search,InstanceCount=3,DedicatedMasterEnabled=true,DedicatedMasterTy
 \
  --ebs-options "EBSEnabled=true,VolumeType=gp3,VolumeSize=200" \
  --encryption-at-rest-options Enabled=true \
  --advanced-security-options
 "Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions={MasterUserName=test-
user,MasterUserPassword=test-password}" \
  --node-to-node-encryption-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:account-id:domain/test-domain/*"}]}'
```

O exemplo a seguir cria um domínio com OI2 instâncias de tamanho `large` e permite a criptografia em repouso. Observe que OI2 as instâncias não exigem a configuração do EBS, pois usam NVMe armazenamento local.

```
aws opensearch create-domain \
  --domain-name test-domain-oi2 \
```

```
  --engine-version OpenSearch_2.11 \
  --cluster-config
 "InstanceType=oi2.2xlarge.search,InstanceCount=3,DedicatedMasterEnabled=true,DedicatedMasterTy
 \
  --encryption-at-rest-options Enabled=true \
  --advanced-security-options
 "Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions={MasterUserName=test-
user,MasterUserPassword=test-password}" \
  --node-to-node-encryption-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:account-id:domain/test-domain-oi2/*"}]}'
```

Armazenamento em vários níveis para Amazon Service OpenSearch

O armazenamento multicamada para o Amazon OpenSearch Service é uma solução inteligente de gerenciamento de dados que otimiza o desempenho e os custos gerenciando dados em diferentes níveis de armazenamento. Essa arquitetura permite que as organizações equilibrem com eficiência a compensação entre desempenho e custo, mantendo os dados acessados com frequência em armazenamento quente de alto desempenho e transferindo os dados acessados com menos frequência para um armazenamento aquecido mais econômico.

O Amazon OpenSearch Service oferece duas opções de arquitetura para o nível hot/warm de armazenamento:

- OpenSearch Arquitetura de armazenamento de vários níveis
  - Combina o Amazon S3 com o armazenamento de instância local
  - Desenvolvido por instâncias OpenSearch otimizadas
  - Suporta operações de gravação em camadas quentes
  - Oferece suporte à migração de dados perfeita entre os níveis quente e quente
  - Disponível na versão OpenSearch 3.3 e superior
  - Não suporta Cold Tier
- UltraWarmArquitetura baseada
  - Combina o Amazon S3 com o armazenamento de instância local
  - Desenvolvido por UltraWarm instâncias

- Otimizado para cargas de trabalho de nível quente somente para leitura
- Disponível na versão 6.8 e superior do Elasticsearch e em todas as versões OpenSearch
- Suporta Cold Tier

> **Note**
>
> Essa documentação se concentra somente na arquitetura de várias camadas. Para a arquitetura de armazenamento Ultrawarm, consulte Ultrawarm and Cold Storage

Arquitetura de armazenamento de vários níveis

Tópicos

- Benefícios principais
- Pré-requisitos
- Limitações
- Pontos importantes
- Criação de um domínio de vários níveis
- Gerenciando migrações de níveis
- Configuração de segurança
- Práticas recomendadas
- Monitoramento de métricas

Benefícios principais

- Writable Warm: suporta operações de gravação em índices quentes
- Migração perfeita: movimentação contínua de dados entre níveis de armazenamento
- Otimização de custos: reduza os custos de armazenamento transferindo dados menos ativos para um armazenamento aquecido econômico
- Aprimoramento do desempenho: mantenha o alto desempenho dos dados acessados com frequência no nível atual
- Gerenciamento flexível de dados: escolha a arquitetura que melhor se adapta às suas necessidades de carga de trabalho

- Gerenciamento automatizado: gerenciamento simplificado do ciclo de vida dos dados em todos os níveis de armazenamento

Pré-requisitos

- Versão do motor: OpenSearch 3.3 ou posterior
- Famílias de instâncias:
  - Hot nodes: OR1, OR2, OM2, ou OI2
  - Nodos quentes: OI2
- Segurança: Node-to-node criptografia, criptografia em repouso, HTTPS aplicado

Limitações

- Funciona em todos os domínios com instâncias OpenSearch otimizadas que ainda não têm o Ultrawarm ativado
- Sem suporte para camada fria

Pontos importantes

- Migrações quentes para quentes não acionam uma fusão forçada na arquitetura de vários níveis. Se necessário, o usuário ainda pode orquestrar mesclagens forçadas usando a política de gerenciamento de índices.
- Além da indexação, os nós quentes agora também realizam operações de mesclagem em segundo plano (semelhantes às quentes).
- Todas as solicitações de pesquisa em índices quentes são roteadas para o fragmento primário e a réplica serve para leitura somente quando o fragmento primário está inativo.
- Instantâneos automatizados para índices quentes também são compatíveis com essa arquitetura.
- A replicação entre clusters só é compatível com índices ativos
- Índices APIs como Shrink, Split e Clone não funcionam com índices quentes.

Criação de um domínio de vários níveis

Etapa 1: criar o domínio

```
aws opensearch create-domain \
      --domain-name my-domain \
      --engine-version OpenSearch_3.3 \
      --cluster-config
 InstanceCount=3,InstanceType=or2.large.search,DedicatedMasterEnabled=true,DedicatedMasterType=
 \
      --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
      --node-to-node-encryption-options Enabled=true \
      --encryption-at-rest-options Enabled=true \
      --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
      --advanced-security-options
 '{"Enabled":true,"InternalUserDatabaseEnabled":true,"MasterUserOptions":
{"MasterUserName":"user_name","MasterUserPassword":"your_pass"}}' \
      --access-policies '{"Version": "2012-10-17",        "Statement":
[{"Effect":"Allow","Principal":"*","Action":"es:*","Resource":"*"}]}' \
      --region us-east-1
```

Etapa 2: verificar os nós quentes

```
aws opensearch describe-domain-nodes --domain-name my-domain --region us-east-1
```

Exemplo de resposta (trecho):

```
{
      "NodeType": "Warm",
      "InstanceType": "oi2.large.search",
      "NodeStatus": "Active"
    }
```

Gerenciando migrações de níveis

Suporte a domínios de vários níveis:

- Nova hierarquização APIs para uma experiência simplificada
- Legacy UltraWarm APIs para compatibilidade

Nova hierarquização APIs

Migre um índice para aquecer:

```
curl -XPOST 'https://localhost:9200/index-name/_tier/warm'
```

Resposta:

```
{"acknowledged": true}
```

Migre um índice para hot:

```
curl -XPOST 'https://localhost:9200/index-name/_tier/hot'
```

Resposta:

```
{"acknowledged": true}
```

Verifique o status da classificação por níveis:

```
curl -XGET 'https://localhost:9200/index-name/_tier'
```

Exemplo de resposta:

```
{
      "tiering_status": {
         "index": "index-name",
         "state": "RUNNING_SHARD_RELOCATION",
         "source": "HOT",
         "target": "WARM",
         "start_time": 1745836500563,
         "shard_level_status": {
           "running": 0,
           "total": 100,
           "pending": 100,
           "succeeded": 0
         }
      }
    }
```

Visualização detalhada do fragmento:

```
curl 'https://localhost:9200/index1/_tier?detailed=true'
```

Liste todas as migrações em andamento (texto):

```
curl 'https://localhost:9200/_tier/all'
```

Liste todas as migrações em andamento (JSON):

```
curl 'https://localhost:9200/_tier/all?format=json'
```

Filtrar por nível de destino:

```
curl 'https://localhost:9200/_tier/all?target=_warm'
```

Legacy UltraWarm APIs para compatibilidade

Migre para aquecer:

```
curl -XPOST localhost:9200/_ultrawarm/migration/index2/_warm
```

Migre para hot:

```
curl -XPOST localhost:9200/_ultrawarm/migration/index2/_hot
```

Verifique o status:

```
curl -XGET localhost:9200/_ultrawarm/migration/index2/_status
```

Configuração de segurança

Se você habilitar o armazenamento em vários níveis em um domínio preexistente do Amazon OpenSearch Service, a `storage_tiering_manager` função pode não estar definida no domínio. Os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices warm usando o controle de aceso detalhado. Para criar manualmente a função `storage_tiering_manager`, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Nome do grupo	Permissões
cluster_hierarquizado de armazenamento	índices: admin/_tier/all
armazenamento_hierarquizado_index_read	índices: admin/_ tier/get, indices:admin/get
armazenamento_hierarquizado_index_write	índices: admin/_ _to_hot tier/hot_to_warm, indices:admin/_tier/warm

3. Escolha Funções e, em seguida, Criar função.
4. Nomeie a função `storage_tiering_manager`.
5. Para Permissões de cluster, selecione `storage_tiering_cluster` e `cluster_monitor`.
6. Para Índice, digite `*`.
7. Para permissões de índice`storage_tiering_index_read`, selecione `storage_tiering_index_write` `indices_monitor` e.
8. Escolha Criar.
9. Depois de criar a função, mapeie-a para qualquer função de usuário ou back-end que gerenciará índices de várias camadas.

Práticas recomendadas

Ao implementar o armazenamento em vários níveis em seus domínios do Amazon OpenSearch Service, considere as seguintes melhores práticas:

- Revise regularmente seus padrões de acesso aos dados para otimizar a alocação de níveis
- Monitore as métricas de desempenho para garantir a utilização eficiente dos recursos
- Use a nova hierarquização APIs para controle granular sobre a migração de dados

Monitoramento de métricas

Os domínios de armazenamento de vários níveis fornecem métricas adicionais para monitorar o desempenho do nível quente. Essas métricas incluem métricas existentes UltraWarm e novas métricas específicas da arquitetura de instâncias OpenSearch otimizadas:

Novas métricas

Nome da métrica	Estatísticas do Node Level	Estatísticas em nível de cluster	Granularity
WarmIndexingLatency	Média	Média	1 min
WarmIndexingRate	Média	Média, Máxima, Soma	1 min
WarmThreadpoolIndexingQueue	Máximo	Soma, Máximo, Média	1 min
WarmThreadpoolIndexingRejected	Máximo	Soma	1 min
WarmThreadpoolIndexingThreads	Máximo	Soma, média	1 min

UltraWarm armazenamento para Amazon OpenSearch Service

UltraWarm fornece uma maneira econômica de armazenar grandes quantidades de dados somente para leitura no Amazon Service. OpenSearch Os nós de dados padrão usam o armazenamento de atividade muito alta, o qual assume a forma de armazenamentos de instâncias ou volumes do Amazon EBS anexados a cada nó. O armazenamento de atividade muito alta fornece a performance mais rápida possível para indexar e pesquisar novos dados.

Em vez de armazenamento conectado, UltraWarm os nós usam o Amazon S3 e uma solução sofisticada de armazenamento em cache para melhorar o desempenho. Para índices nos quais você não está gravando ativamente, consultar com menos frequência e dos quais não precisa do mesmo desempenho UltraWarm oferece custos significativamente mais baixos por GiB de dados. Como os índices quentes são somente para leitura, a menos que você os devolva ao armazenamento ativo, UltraWarm é mais adequado para dados imutáveis, como registros.

Em OpenSearch, os índices quentes se comportam como qualquer outro índice. Você pode consultá-los usando o mesmo APIs ou usá-los para criar visualizações em OpenSearch painéis.

Tópicos

- Pré-requisitos
- UltraWarm requisitos de armazenamento e considerações de desempenho
- UltraWarm preços
- Habilitando UltraWarm
- Migração de índices para armazenamento UltraWarm
- Automatização de migrações
- Ajuste de migrações
- Cancelamento de migrações
- Listagem de índices quentes e mornos
- Retorno de índices warm para o armazenamento quente
- Restauração de índices quentes de snapshots
- Snapshots manuais de índices mornos
- Migração de índices mornos para o armazenamento frio
- Práticas recomendadas para índices KNN
- Desativando UltraWarm

Pré-requisitos

UltraWarm tem alguns pré-requisitos importantes:

- UltraWarm requer o Elasticsearch 6.8 OpenSearch ou superior.
- Para usar o armazenamento de alta atividade (warm), os domínios devem ter nós principais dedicados.
- Ao usar um domínio multi-AZ com modo de espera, o número de nós mornos deve ser um múltiplo do número de zonas de disponibilidade que estão usadas.
- Se o domínio usar um tipo de instância T2 ou T3 para os nós de dados, não será possível usar o armazenamento de alta atividade.
- Se o índice usar aproximação de k-NN (`"index.knn":true`), você poderá movê-lo da versão 2.17 e posteriores para o armazenamento morno. Domínios em versões anteriores à 2.17 podem

ser atualizados para a 2.17 para usar essa funcionalidade, mas os índices KNN criados em versões anteriores à 2.x não podem migrar para. UltraWarm

- Se o domínio usa controle de acesso refinado, os usuários devem ser mapeados para a `ultrawarm_manager` função nos OpenSearch painéis para fazer chamadas de API. UltraWarm

> **Note**
>
> A `ultrawarm_manager` função pode não estar definida em alguns domínios de OpenSearch serviço preexistentes. Se você não vir a função no Dashboards, será necessário criá-la manualmente.

Configurar permissões

Se você habilitar UltraWarm em um domínio OpenSearch de serviço preexistente, a `ultrawarm_manager` função pode não estar definida no domínio. Os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices warm usando o controle de aceso detalhado. Para criar manualmente a função `ultrawarm_manager`, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Group name	Permissões
`ultrawarm_cluster`	• `cluster:admin/ultrawarm/migration/list` • `cluster:monitor/nodes/stats`
`ultrawarm_index_read`	• `indices:admin/ultrawarm/migration/get` • `indices:admin/get`
`ultrawarm_index_write`	• `indices:admin/ultrawarm/migration/warm` • `indices:admin/ultrawarm/migration/hot` • `indices:monitor/stats` • `indices:admin/ultrawarm/migration/cancel`

3. Escolha Funções e, em seguida, Criar função.

4. Nomeie a função como ultrawarm_manager.
5. Para Permissões de cluster, selecione `ultrawarm_cluster` e `cluster_monitor`.
6. Para Índice, digite `*`.
7. Para Permissões de índice, selecione `ultrawarm_index_read`, `ultrawarm_index_write`, e `indices_monitor`.
8. Escolha Criar.
9. Depois de criar a função, mapeie-a para qualquer função de usuário ou back-end que gerenciará UltraWarm índices.

UltraWarm requisitos de armazenamento e considerações de desempenho

Conforme abordado emthe section called “Cálculo de requisitos de armazenamento”, os dados no armazenamento dinâmico geram uma sobrecarga significativa: réplicas, espaço reservado do Linux e espaço reservado do OpenSearch serviço. Por exemplo, um fragmento primário de 20 GiB com um fragmento de réplica requer aproximadamente 58 GiB de armazenamento de atividade muito alta.

Como ele usa o Amazon S3, não UltraWarm incorre em nenhuma dessas despesas gerais. Ao calcular os requisitos UltraWarm de armazenamento, você considera somente o tamanho dos fragmentos primários. A durabilidade dos dados no S3 elimina a necessidade de réplicas e o S3 abstrai qualquer consideração de sistema operacional ou de serviço. Esse mesmo fragmento de 20 GiB exige 20 GiB de armazenamento de alta atividade. Se você provisionar uma instância `ultrawarm1.large.search`, poderá usar todos os 20 TiB de seu armazenamento máximo para fragmentos primários. Consulte the section called “UltraWarm cotas de armazenamento” para obter um resumo dos tipos de instância e a quantidade máxima de armazenamento que cada um pode atender.

Com UltraWarm, ainda recomendamos um tamanho máximo de fragmento de 50 GiB. O número de núcleos de CPU e a quantidade de RAM alocada para cada tipo de UltraWarm instância dão uma ideia do número de fragmentos que eles podem pesquisar simultaneamente. Observe que, embora apenas os fragmentos primários contem para o UltraWarm armazenamento no S3, os OpenSearch painéis `_cat/indices` ainda relatam o tamanho do UltraWarm índice como o total de todos os fragmentos primários e de réplica.

Por exemplo, cada instância de `ultrawarm1.medium.search` tem dois núcleos de CPU e pode endereçar até 1,5 TiB de armazenamento no S3. Duas dessas instâncias têm uma combinação de 3 TiB de armazenamento, o que funcionará para aproximadamente 62 fragmentos se o tamanho

de cada fragmento for 50 GiB. Se uma solicitação para o cluster pesquisar apenas quatro desses fragmentos, a performance poderá ser excelente. Se a solicitação for ampla e pesquisar todos os 62, os quatro núcleos da CPU poderão ter dificuldade para executar a operação. Monitore as `WarmJVMMemoryPressure` UltraWarm métricas `WarmCPUUtilization` e para entender como as instâncias lidam com suas cargas de trabalho.

Se as suas pesquisas forem amplas ou frequentes, considere deixar os índices no armazenamento quente. Assim como qualquer outra OpenSearch carga de trabalho, a etapa mais importante para determinar se UltraWarm atende às suas necessidades é realizar testes representativos do cliente usando um conjunto de dados realista.

UltraWarm preços

Com o armazenamento de alta atividade, você paga pelo que provisiona. Algumas instâncias exigem um volume do Amazon EBS vinculado, enquanto outras incluem um armazenamento de instâncias. Se esse armazenamento estiver vazio ou cheio, você pagará o mesmo preço.

Com o UltraWarm armazenamento, você paga pelo que usa. Uma instância `ultrawarm1.large.search` pode processar até 20 TiB de armazenamento no S3, mas se você armazenar apenas 1 TiB de dados, será cobrado somente por 1 TiB de dados. Como todos os outros tipos de nós, você também paga uma taxa horária por cada UltraWarm nó. Para obter mais informações, consulte the section called “Preços”.

Habilitando UltraWarm

O console é a maneira mais simples de criar um domínio que usa o armazenamento de alta atividade. Ao criar o domínio, escolha Habilitar nós de dados mornos e o número de nós mornos desejados. O mesmo processo básico funciona em domínios existentes, desde que eles atendam aos pré-requisitos. Mesmo depois que o estado do domínio mudar de Processamento para Ativo, UltraWarm pode não estar disponível para uso por várias horas.

Ao usar um domínio multi-AZ com modo de espera, o número de nós mornos deve ser um múltiplo do número de zonas de disponibilidade. Para obter mais informações, consulte the section called “Multi-AZ com modo de espera”.

Você também pode usar a API de configuração AWS CLIou para habilitar UltraWarm, especificamente`WarmEnabled`, as `WarmType` opções`WarmCount`, e em`ClusterConfig`.

> **Note**
>
> Os domínios oferecem suporte a um número máximo de nós de alta atividade. Para obter detalhes, consulte the section called “Cotas”.

Exemplo de comando da CLI

O AWS CLI comando a seguir cria um domínio com três nós de dados, três nós principais dedicados, seis nós quentes e um controle de acesso refinado ativado:

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedMasterEnabled=true,DedicatedMasterType=
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":["123456789012"]},"Action":
["es:*"],"Resource":"arn:aws:es:us-west-1:123456789012:domain/my-domain/*"}]}' \
  --region us-east-1
```

Para saber mais, consulte a Referência de comandos da AWS CLI.

Exemplo de solicitação da API de configuração

A solicitação a seguir à API de configuração cria um domínio com três nós de dados, três nós principais dedicados e seis nós de alta atividade com o controle de acesso refinado habilitado e uma política de acesso restritiva:

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
```

```
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
    },
    "WarmEnabled": true,
    "WarmCount": 6,
    "WarmType": "ultrawarm1.medium.search"
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain",
  "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/my-domain/*\"}]}"
}
```

Para obter informações detalhadas, consulte a Amazon OpenSearch Service API Reference.

Migração de índices para armazenamento UltraWarm

Se você terminou de escrever em um índice e não precisa mais do desempenho de pesquisa mais rápido possível, migre-o de hot para UltraWarm:

```
POST _ultrawarm/migration/my-index/_warm
```

Depois, verifique o status da migração:

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_SHARD_RELOCATION",
    "migration_type": "HOT_TO_WARM",
    "shard_level_status": {
      "running": 0,
      "total": 5,
      "pending": 3,
      "failed": 0,
      "succeeded": 2
    }
  }
}
```

A integridade do índice deve ser verde para que seja possível executar uma migração. Se você migrar vários índices em sucessão rápida, poderá obter um resumo de todas as migrações em texto não criptografado, semelhante à API `_cat`:

```
GET _ultrawarm/migration/_status?v

index    migration_type state
my-index HOT_TO_WARM    RUNNING_SHARD_RELOCATION
```

OpenSearch O serviço migra um índice por vez para o. UltraWarm É possível ter até 200 migrações na fila. Qualquer solicitação que exceda o limite será rejeitada. Para verificar o número de migrações atual, monitore a métrica `HotToWarmMigrationQueueSize`. Os índices permanecem disponíveis durante todo o processo de migração, sem tempo de inatividade.

O processo de migração tem os seguintes estados:

```
PENDING_INCREMENTAL_SNAPSHOT
RUNNING_INCREMENTAL_SNAPSHOT
FAILED_INCREMENTAL_SNAPSHOT
PENDING_FORCE_MERGE
RUNNING_FORCE_MERGE
FAILED_FORCE_MERGE
PENDING_FULL_SNAPSHOT
RUNNING_FULL_SNAPSHOT
FAILED_FULL_SNAPSHOT
PENDING_SHARD_RELOCATION
RUNNING_SHARD_RELOCATION
FINISHED_SHARD_RELOCATION
```

Como esses estados indicam, as migrações podem falhar durante os snapshots, as realocações de fragmentos ou as uniões de força. As falhas durante os snapshots ou as realocações de fragmentos geralmente ocorrem devido a falhas de nós ou a problemas de conectividade do S3. A falta de espaço em disco geralmente é a causa subjacente das falhas de união de força.

Após o término da migração, a mesma solicitação `_status` retornará um erro. Se você verificar o índice nesse momento, verá algumas configurações exclusivas dos índices mornos:

```
GET my-index/_settings

{
  "my-index": {
    "settings": {
      "index": {
        "refresh_interval": "-1",
        "auto_expand_replicas": "false",
        "provided_name": "my-index",
        "creation_date": "1599241458998",
        "unassigned": {
          "node_left": {
            "delayed_timeout": "5m"
          }
        },
        "number_of_replicas": "1",
        "uuid": "GswyCdR0RSq0SJYmzsIpiw",
        "version": {
          "created": "7070099"
        },
        "routing": {
```

```
          "allocation": {
            "require": {
              "box_type": "warm"
            }
          }
        },
        "number_of_shards": "5",
        "merge": {
          "policy": {
            "max_merge_at_once_explicit": "50"
          }
        }
      }
    }
  }
}
```

- `number_of_replicas`, nesse caso, é o número de réplicas passivas, que não consomem espaço em disco.
- `routing.allocation.require.box_type` especifica que o índice deve usar nós de alta atividade em vez de nós de dados padrão.
- `merge.policy.max_merge_at_once_explicit` especifica o número de segmentos a serem mesclados simultaneamente durante a migração.

Os índices no armazenamento quente são somente para leitura, a menos que você os retorne ao armazenamento ativo, o que os torna UltraWarm mais adequados para dados imutáveis, como registros. Você pode consultar os índices e excluí-los, mas não pode adicionar, atualizar ou excluir documentos individuais. Se tentar, você poderá encontrar a seguinte mensagem de erro:

```
{
  "error" : {
    "root_cause" : [
      {
        "type" : "cluster_block_exception",
        "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage
 exceeded flood-stage watermark, index has read-only-allow-delete block];"
      }
    ],
    "type" : "cluster_block_exception",
```

```
    "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage exceeded
 flood-stage watermark, index has read-only-allow-delete block];"
  },
  "status" : 429
}
```

Automatização de migrações

Recomendamos usar the section called “Gerenciamento de estados de índice” para automatizar o processo de migração depois que um índice atinge uma determinada idade ou atende a outras condições. Consulte a política de exemplo que demonstra este fluxo de trabalho.

Ajuste de migrações

As migrações de índice para o UltraWarm armazenamento exigem uma mesclagem forçada. Cada OpenSearch índice é composto por um certo número de fragmentos, e cada fragmento é composto por algum número de segmentos do Lucene. A operação de forças mesclagem expurga documentos que foram marcados para exclusão e conserva espaço em disco. Por padrão, UltraWarm mescla índices em um segmento, exceto os índices kNN, nos quais um valor padrão de 20 é usado.

Você pode alterar esse valor para até 1.000 segmentos usando a configuração `index.ultrawarm.migration.force_merge.max_num_segments`. Valores mais altos aceleram o processo de migração, mas aumentam a latência de consulta para o índice de alta atividade após a conclusão da migração. Para alterar a configuração, faça a seguinte solicitação:

```
PUT my-index/_settings
{
  "index": {
    "ultrawarm": {
      "migration": {
        "force_merge": {
          "max_num_segments": 1
        }
      }
    }
  }
}
```

Para verificar a duração desse estágio do processo de migração, monitore a métrica `HotToWarmMigrationForceMergeLatency`.

Cancelamento de migrações

UltraWarm lida com migrações sequencialmente, em uma fila. Se uma migração estiver na fila, mas ainda não tiver sido iniciada, você poderá removê-la da fila usando a seguinte solicitação:

```
POST _ultrawarm/migration/_cancel/my-index
```

Se o domínio usa controle de acesso refinado, você precisará da permissão `indices:admin/ultrawarm/migration/cancel` para fazer essa solicitação.

Listagem de índices quentes e mornos

UltraWarm adiciona duas opções adicionais, semelhantes a`_all`, para ajudar a gerenciar índices quentes e quentes. Para obter uma lista de todos os índices mornos ou quentes, faça as seguintes solicitações:

```
GET _warm
GET _hot
```

Você pode usar essas opções em outras solicitações que especificam índices, como:

```
_cat/indices/_warm
_cluster/state/_all/_hot
```

Retorno de índices warm para o armazenamento quente

Se você precisar gravar em um índice novamente, migre-o de volta para o armazenamento de atividade muito alta:

```
POST _ultrawarm/migration/my-index/_hot
```

Você pode ter até 10 migrações em fila do armazenamento quente para o armazenamento quente por vez. OpenSearch O serviço processa as solicitações de migração uma de cada vez, na ordem em que foram colocadas na fila. Para verificar o número atual, monitore a `WarmToHotMigrationQueueSize` métrica.

Após a conclusão da migração, verifique as configurações de índice para garantir que atendam às suas necessidades. Os índices retornam ao armazenamento quente com uma réplica.

Restauração de índices quentes de snapshots

Além do repositório padrão para instantâneos automatizados, UltraWarm adiciona um segundo repositório para índices quentes,. `cs-ultrawarm` Cada snapshot neste repositório contém apenas um índice. Se você excluir um índice de alta atividade, seu instantâneo permanecerá no repositório `cs-ultrawarm` por 14 dias, assim como qualquer outro snapshot automatizado.

Quando você restaura um snapshot de `cs-ultrawarm`, ele é restaurado no armazenamento de alta atividade (warm), não no armazenamento de atividade muito alta (hot). Os snapshots nos repositórios `cs-automated` e `cs-automated-enc` são restaurados no armazenamento de atividade muito alta.

Para restaurar um UltraWarm instantâneo para um armazenamento aquecido

1. Identifique o snapshot mais recente que contém o índice que você deseja restaurar:

```
GET _snapshot/cs-ultrawarm/_all?verbose=false

{
  "snapshots": [{
    "snapshot": "snapshot-name",
    "version": "1.0",
    "indices": [
      "my-index"
    ]
  }]
}
```

> **Note**
>
> Por padrão, a operação `GET _snapshot/<repo>` exibe informações detalhadas de dados, como hora de início, hora de término e duração de cada instantâneo em um repositório. A operação `GET _snapshot/<repo>` recupera informações dos arquivos de cada instantâneo contido em um repositório. Se você não precisar do horário de início, horário de término e duração e precisar apenas do nome e das informações de índice de um snapshot, recomendamos usar o parâmetro `verbose=false` ao listar snapshots para minimizar o tempo de processamento e evitar o tempo limite.

2. Se o índice já existir, exclua-o:

```
DELETE my-index
```

Se não quiser excluir o índice, devolva-o ao armazenamento de atividade muito alta e reindexe-o.

3. Restaure o snapshot:

```
POST _snapshot/cs-ultrawarm/snapshot-name/_restore
```

UltraWarm ignora todas as configurações de índice que você especificar nessa solicitação de restauração, mas você pode especificar opções como `rename_pattern` e. `rename_replacement` Para obter um resumo das opções de restauração de OpenSearch instantâneos, consulte a OpenSearch documentação.

Snapshots manuais de índices mornos

Você pode obter snapshots manuais de índices mornos, mas não recomendamos fazer isso. O repositório `cs-ultrawarm` automatizado já contém um snapshot para cada índice de alta atividade, obtido durante a migração, sem custo adicional.

Por padrão, o OpenSearch Serviço não inclui índices quentes em instantâneos manuais. Por exemplo, a chamada a seguir inclui apenas índices quentes:

```
PUT _snapshot/my-repository/my-snapshot
```

Se você optar por obter snapshots manuais de índices mornos, diversas considerações importantes serão aplicáveis.

- Você não pode misturar índices quentes e mornos. Por exemplo, a solicitação a seguir falha:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,hot-index-1",
  "include_global_state": false
}
```

Se eles incluírem uma mistura de índices quentes e mornos, as instruções universais (*) também falharão.

- Você só pode incluir um índice de alta atividade por snapshot. Por exemplo, a solicitação a seguir falha:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,warm-index-2,other-warm-indices-*",
  "include_global_state": false
}
```

  Esta solicitação é bem-sucedida:

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1",
  "include_global_state": false
}
```

- Os snapshots manuais são sempre restaurados para o armazenamento de atividade muito alta, mesmo que tenham originalmente incluído um índice de alta atividade.

Migração de índices mornos para o armazenamento frio

Se você tiver dados UltraWarm que não consulta com frequência, considere migrá-los para o armazenamento a frio. O armazenamento de baixa atividade é destinado a dados que você acessa apenas ocasionalmente ou que não estão mais em uso ativo. Você não pode ler nem gravar em índices frios, mas pode migrá-los de volta para o armazenamento morno sem nenhum custo sempre que precisar consultá-los. Para obter instruções, consulte Migração de índices para o armazenamento frio.

Práticas recomendadas para índices KNN

- O nível Ultrawarm/Frio está disponível para todos os tipos de mecanismos de índice KNN. Nós o recomendamos para índices KNN que usam o mecanismo Lucene e pesquisa vetorial otimizada para disco, o que não exige o carregamento total dos dados de gráfico na memória off-heap. Ao usá-lo com mecanismos nativos de memória, como FAISS e NMSLIB, você deve considerar o tamanho do gráfico de fragmentos que será pesquisado ativamente e provisionar as UltraWarm instâncias, preferencialmente do tipo de instância, adequadamente. `uw.large` Por exemplo, se os clientes tiverem duas instâncias `uw.large` configuradas, cada uma terá aproximadamente `knn.memory.circuit_breaker.limit * 61` GiB de memória off-heap disponível. Você terá

a performance ideal se todas as suas consultas mornas visarem fragmentos cujo tamanho gráfico cumulativo não exceda a memória off-heap disponível. A latência será afetada se a memória disponível for menor que a necessária para carregar o gráfico devido a remoções e à espera até a memória off-heap ficar disponível. É por isso que não recomendamos o uso de instâncias `uw.medium` para casos de uso em que mecanismos na memória são usados ou casos de maior throughput de pesquisa, independentemente dos mecanismos.

- Os índices KNN migrados para não UltraWarm serão mesclados à força em um único segmento. Isso evita qualquer impacto nos nós quentes e mornos que enfrentam problemas de OOM devido ao tamanho do gráfico se tornar grande demais para os mecanismos em memória. Devido ao aumento no número de segmentos por fragmento, isso pode resultar no consumo de mais espaço de cache local e permitir que menos índices migrem para a camada morna. Você pode escolher forçar a mesclagem de índices em um único segmento usando a configuração existente e substituindo-a antes de migrar os índices para a camada morna. Para obter mais informações, consulte the section called “Ajuste de migrações”.
- Se você tiver um caso de uso em que os índices são pesquisados com pouca frequência e não atendem a uma carga de trabalho sensível à latência, você pode optar por migrar esses índices para o nível. UltraWarm Isso ajudará você a reduzir as instâncias de computação de nível ativo e permitir que a computação de UltraWarm nível gerencie a consulta nesses índices de baixa prioridade. Isso também pode ajudar a isolar os recursos consumidos por consultas de índices de baixa e alta prioridade de modo que não afetem um ao outro.

Desativando UltraWarm

O console é a maneira mais simples de desativar UltraWarm. Escolha o domínio, Ações, e Editar configuração do cluster. Desmarque Habilitar nós de dados mornos e escolha Salvar alterações. Você também pode usar a opção `WarmEnabled` na AWS CLI e na API de configuração.

Antes de desativar UltraWarm, você deve excluir todos os índices quentes ou migrá-los de volta para o armazenamento ativo. Depois que o armazenamento quente estiver vazio, aguarde cinco minutos antes de tentar UltraWarm desativá-lo.

Armazenamento a frio para Amazon OpenSearch Service

O armazenamento a frio permite que você armazene qualquer quantidade de dados históricos ou acessados com pouca frequência em seu domínio do Amazon OpenSearch Service e os analise sob demanda, a um custo menor do que outros níveis de armazenamento. O armazenamento de

baixa atividade é apropriado se você precisa fazer pesquisas periódicas ou análises forenses em seus dados mais antigos. Exemplos práticos de dados adequados para armazenamento de baixa atividade incluem logs acessados com pouca frequência, dados que devem ser preservados para atender a requisitos de compatibilidade ou registros com valor histórico.

Semelhante ao UltraWarmarmazenamento, o armazenamento a frio é apoiado pelo Amazon S3. Quando precisar consultar dados frios, você pode anexá-los seletivamente aos UltraWarm nós existentes. Você pode gerenciar a migração e o ciclo de vida de seus dados de baixa atividade manualmente ou com políticas de gerenciamento de estado de índice.

Tópicos

- Pré-requisitos
- Requisitos de armazenamento e considerações de performance do armazenamento de baixa atividade
- Preços do armazenamento de baixa atividade
- Habilitação do armazenamento de baixa atividade
- Gerenciando índices frios em painéis OpenSearch
- Migração de índices para o armazenamento frio
- Automatização de migrações para o armazenamento frio
- Cancelando migrações para armazenamento frio
- Listagem de índices de baixa atividade
- Migração de índices frios para o armazenamento warm
- Restauração de índices frios de snapshots
- Cancelamento de migrações do armazenamento de baixa atividade para o armazenamento de alta atividade
- Atualizando metadados de índice de baixa atividade
- Exclusão de índices de baixa atividade
- Desabilitação do armazenamento de baixa atividade

Pré-requisitos

O armazenamento de baixa atividade apresenta os seguintes pré-requisitos:

- O armazenamento a frio requer o Elasticsearch versão 7.9 OpenSearch ou posterior.

- Para habilitar o armazenamento frio em um domínio de OpenSearch serviço, você também deve habilitar o armazenamento a quente no mesmo domínio.
- Para que seja possível usar o armazenamento de baixa atividade, os domínios deverão ter nós principais dedicados.
- Se o domínio usar um tipo de instância T2 ou T3 para os nós de dados, não será possível usar o armazenamento de baixa atividade .
- Se o índice usar aproximação de k-NN (`"index.knn":true`), você não poderá movê-lo da versão 2.17 e posteriores para o armazenamento frio. Os domínios em versões anteriores à 2.17 podem ser atualizados para a 2.17 para usar essa funcionalidade, mas os índices KNN criados em versões anteriores à 2.x não podem migrar para armazenamento frio.
- Se o domínio usar um controle de acesso refinado, os usuários não administradores devem ser mapeados para a `cold_manager` função nos OpenSearch painéis para gerenciar índices frios.

> **Note**
>
> A `cold_manager` função pode não existir em alguns domínios de OpenSearch serviço preexistentes. Se você não vir a função no Dashboards, será necessário criá-la manualmente.

Configurar permissões

Se você habilitar o armazenamento a frio em um domínio OpenSearch de serviço preexistente, a `cold_manager` função pode não estar definida no domínio. Se o domínio usar controle de acesso refinado, os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices de baixa atividade. Para criar manualmente a função `cold_manager`, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Group name	Permissões
`cold_cluster`	• `cluster:monitor/nodes/stats` • `cluster:admin/ultrawarm*`

Group name	Permissões
	• `cluster:admin/cold/*`
`cold_index`	• `indices:monitor/stats` • `indices:data/read/minmax` • `indices:admin/ultrawarm/migration/get` • `indices:admin/ultrawarm/migration/cancel`

3. Escolha Funções e, em seguida, Criar função.
4. Nomeie a função como cold_manager.
5. Em Permissões de cluster, escolha o grupo `cold_cluster` que você criou.
6. Em Índice, insira `*`.
7. Em Permissões de índice, escolha o grupo `cold_index` que você criou.
8. Escolha Criar.
9. Depois de criar a função, mapeie-a em qualquer função de usuário ou backend que gerencie índices de baixa atividade.

Requisitos de armazenamento e considerações de performance do armazenamento de baixa atividade

Como o armazenamento a frio usa o Amazon S3, ele não incorre na sobrecarga do armazenamento dinâmico, como réplicas, espaço reservado do Linux e espaço reservado do serviço. OpenSearch O armazenamento de baixa atividade não tem tipos de instância específicos porque não há nenhuma capacidade computacional anexada a ele. Você pode armazenar qualquer quantidade de dados em armazenamento de baixa atividade. Monitore a `ColdStorageSpaceUtilization` métrica na Amazon CloudWatch para ver quanto espaço de armazenamento refrigerado você está usando.

Preços do armazenamento de baixa atividade

Semelhante ao UltraWarm armazenamento, com o armazenamento a frio, você paga apenas pelo armazenamento de dados. Não há custo de computação para dados de baixa atividade e você não será cobrado se não houver dados no armazenamento de baixa atividade.

Você não incorre em cobranças de transferência ao mover dados entre os armazenamentos de baixa e de alta atividade. Enquanto os índices estão sendo migrados entre o armazenamento warm

e o frio, você continua pagando por apenas uma cópia do índice. Após a conclusão da migração, o índice é cobrado de acordo com o nível de armazenamento para o qual foi migrado. Para obter mais informações sobre preços de armazenamento a frio, consulte os preços do Amazon OpenSearch Service.

Habilitação do armazenamento de baixa atividade

O console é a maneira mais simples de criar um domínio que usa o armazenamento de baixa atividade. Ao criar o domínio, escolha primeiro Habilitar nós de dados mornos, porque é preciso habilitar o armazenamento morno no mesmo domínio. Depois, escolha Habilitar armazenamento frio.

O mesmo processo funciona em domínios existentes, desde que você atenda aos pré-requisitos. Mesmo depois que o estado do domínio mudar de Em processamento para Ativo, o UltraWarm poderá permanecer indisponível por várias horas.

Você também pode usar a AWS CLI ou a API de configuração para habilitar o armazenamento de baixa atividade.

Exemplo de comando da CLI

O AWS CLI comando a seguir cria um domínio com três nós de dados, três nós principais dedicados, armazenamento a frio ativado e controle de acesso refinado habilitado:

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-
config ColdStorageOptions={Enabled=true},WarmEnabled=true,WarmCount=4,WarmType=ultrawarm1.mediu
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --region us-east-2
```

Para saber mais, consulte a Referência de comandos da AWS CLI.

Exemplo de solicitação da API de configuração

A seguinte solicitação à API de configuração cria um domínio com três nós de dados, três nós principais dedicados, armazenamento de baixa atividade habilitado e controle de acesso refinado habilitado:

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
     },
    "WarmEnabled": true,
    "WarmCount": 4,
    "WarmType": "ultrawarm1.medium.search",
    "ColdStorageOptions": {
       "Enabled": true
     }
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
```

```
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain"
}
```

Para obter informações detalhadas, consulte a Amazon OpenSearch Service API Reference.

Gerenciando índices frios em painéis OpenSearch

Você pode gerenciar índices quentes, quentes e frios com a interface de painéis existente em seu domínio de OpenSearch serviço. O Dashboards permite que você migre índices entre armazenamentos warm e frio e monitore o status da migração do índice sem usar a CLI ou a API de configuração. Para obter mais informações, consulte Gerenciamento de índices em OpenSearch painéis.

Migração de índices para o armazenamento frio

Ao migrar índices para o armazenamento frio, você deve fornecer um intervalo de tempo para os dados para facilitar a descoberta. Você pode selecionar um campo de timestamp com base nos dados em seu índice, fornecer manualmente um carimbo de data/hora inicial e final ou optar por não especificar um.

Parâmetro	Valor compatível	Description
`timestamp_field`	O date/time campo do mapeamento do índice.	Os valores mínimo e máximo do campo fornecido são calculados e armazenad os como os metadados `start_time` e `end_time` para o índice de baixa atividade.
`start_time` e `end_time`	Use um dos seguintes formatos:	Os valores são fornecido s como os metadados `start_time` e `end_time`

Parâmetro	Valor compatível	Description
	• strict_date_optional_time. Por exemplo: `yyyy-MM-dd'T'HH:mm:ss.SSSZ` ou `yyyy-MM-dd` • Tempo de época em milissegundos	para o índice de baixa atividade.

Se não quiser especificar um carimbo de data/hora, adicione `?ignore=timestamp` à solicitação em vez disso.

A seguinte solicitação migra um índice de alta atividade para o armazenamento de baixa atividade e fornece horários de início e término para os dados nesse índice:

```
POST _ultrawarm/migration/my-index/_cold
  {
    "start_time": "2020-03-09",
    "end_time": "2020-03-09T23:00:00Z"
  }
```

Depois, verifique o status da migração:

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_METADATA_RELOCATION",
    "migration_type": "WARM_TO_COLD"
  }
}
```

OpenSearch O serviço migra um índice por vez para o armazenamento refrigerado. É possível ter até 100 migrações na fila. Qualquer solicitação que exceda o limite será rejeitada. Para verificar o número de migrações atual, monitore a métrica `WarmToColdMigrationQueueSize`. O processo de migração tem os seguintes estados:

```
ACCEPTED_COLD_MIGRATION - Migration request is accepted and queued.
```

```
RUNNING_METADATA_MIGRATION - The migration request was selected for execution and
 metadata is migrating to cold storage.
FAILED_METADATA_MIGRATION - The attempt to add index metadata has failed and all
 retries are exhausted.
PENDING_INDEX_DETACH - Index metadata migration to cold storage is completed. Preparing
 to detach the warm index state from the local cluster.
RUNNING_INDEX_DETACH - Local warm index state from the cluster is being removed. Upon
 success, the migration request will be completed.
FAILED_INDEX_DETACH - The index detach process failed and all retries are exhausted.
```

Automatização de migrações para o armazenamento frio

Você pode usar o Gerenciamento de estados de índices para automatizar o processo de migração depois que um índice atinge uma determinada idade ou atende a outras condições. Veja o exemplo de política, que demonstra como migrar automaticamente os índices do armazenamento a quente para o armazenamento UltraWarm a frio.

> **Note**
>
> Um `timestamp_field` explícito é necessário para mover índices para o armazenamento frio usando uma política de gerenciamento de estados de índices.

Cancelando migrações para armazenamento frio

Se uma migração para armazenamento frio estiver enfileirada ou em um estado de falha, você poderá cancelar a migração usando a seguinte solicitação:

```
POST _ultrawarm/migration/_cancel/my-index

{
  "acknowledged" : true
}
```

Se o domínio usa controle de acesso refinado, você precisará da permissão `indices:admin/ultrawarm/migration/cancel` para fazer essa solicitação.

Listagem de índices de baixa atividade

Antes de consultar, você pode listar os índices no armazenamento refrigerado para decidir para quais migrar UltraWarm para análise posterior. A seguinte solicitação lista todos os índices de baixa atividade classificados por nome de índice:

```
GET _cold/indices/_search
```

Exemplo de resposta

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 3,
  "indices" : [
    {
      "index" : "my-index-1",
      "index_cold_uuid" : "hjEoh26mRRCFxRIMdgvLmg",
      "size" : 10339,
      "creation_date" : "2021-06-28T20:23:31.206Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-2",
      "index_cold_uuid" : "0vIS2n-oROmOWDFmwFIgdw",
      "size" : 6068,
      "creation_date" : "2021-07-15T19:41:18.046Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-3",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

Filtrar

Você pode filtrar índices frios com base em um padrão de índice baseado em prefixos e em deslocamentos de intervalo de tempo.

A seguinte solicitação lista índices que correspondem ao padrão de prefixo de `event-*`:

```
GET _cold/indices/_search
 {
   "filters":{
      "index_pattern": "event-*"
   }
 }
```

Exemplo de resposta

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "events-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

A seguinte solicitação retorna índices com campos de metadados `start_time` e `end_time` entre `2019-03-01` e `2020-03-01`:

```
GET _cold/indices/_search
{
  "filters": {
    "time_range": {
      "start_time": "2019-03-01",
      "end_time": "2020-03-01"
    }
  }
```

```
}
```

Exemplo de resposta

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "my-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2019-05-09T00:00Z",
      "end_time" : "2019-09-09T23:00Z"
    }
  ]
}
```

Classificação

Você pode classificar índices frios por campos de metadados, como nome ou tamanho do índice. A seguinte solicitação lista todos os índices classificados por tamanho em ordem decrescente:

```
GET _cold/indices/_search
 {
 "sort_key": "size:desc"
 }
```

Exemplo de resposta

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 5,
  "indices" : [
    {
      "index" : "my-index-6",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
```

```
    },
    {
      "index" : "my-index-9",
      "index_cold_uuid" : "mbD3ZRVDRI6ONqgEOsJyUA",
      "size" : 57922,
      "creation_date" : "2021-07-07T23:41:35.640Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-5",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

Outras chaves de classificação válidas são `start_time:asc/desc`, `end_time:asc/desc` e `index_name:asc/desc`.

Paginação

Você pode paginar uma lista de índices de baixa atividade. Configure o número de índices a serem retornados por página com o parâmetro `page_size` (o padrão é 10). Cada solicitação `_search` em seus índices frios retorna um `pagination_id` que você pode usar para chamadas subsequentes.

A seguinte solicitação pagina os resultados de uma solicitação `_search` de seus índices frios e exibe os próximos 100 resultados:

```
GET _cold/indices/_search?page_size=100
{
"pagination_id": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
}
```

Migração de índices frios para o armazenamento warm

Depois de restringir sua lista de índices frios com os critérios de filtragem da seção anterior, migre-os de volta para UltraWarm onde você possa consultar os dados e usá-los para criar visualizações.

A solicitação a seguir migra dois índices frios de volta para o armazenamento warm:

```
POST _cold/migration/_warm
 {
 "indices": "my-index1,my-index2"
 }


{
  "acknowledged" : true
}
```

Para verificar o status da migração e recuperar o ID de migração, envie a seguinte solicitação:

```
GET _cold/migration/_status
```

Exemplo de resposta

```
{
  "cold_to_warm_migration_status" : [
    {
      "migration_id" : "tyLjXCA-S76zPQbPVHkOKA",
      "indices" : [
        "my-index1,my-index2"
      ],
      "state" : "RUNNING_INDEX_CREATION"
    }
  ]
}
```

Para obter informações de migração específicas do índice, inclua o nome do índice:

```
GET _cold/migration/my-index/_status
```

Em vez de especificar um índice, você pode listar os índices por seu status de migração atual. Os valores válidos são `_failed`, `_accepted` e `_all`.

O comando a seguir obtém o status de todos os índices em uma única solicitação de migração:

```
GET _cold/migration/_status?migration_id=my-migration-id
```

Recupere o ID de migração usando a solicitação de status. Para obter informações detalhadas sobre migração, adicione `&verbose=true`.

Você pode migrar índices do armazenamento frio para o armazenamento morno em lotes de 10, com, no máximo, 100 índices sendo migados simultaneamente. Qualquer solicitação que exceda o limite será rejeitada. Para verificar o número de migrações que estão ocorrendo no momento, monitore a métrica `ColdToWarmMigrationQueueSize`. O processo de migração tem os seguintes estados:

```
ACCEPTED_MIGRATION_REQUEST - Migration request is accepted and queued.
RUNNING_INDEX_CREATION - Migration request is picked up for processing and will create
 warm indexes in the cluster.
PENDING_COLD_METADATA_CLEANUP - Warm index is created and the migration service will
 attempt to clean up cold metadata.
RUNNING_COLD_METADATA_CLEANUP - Cleaning up cold metadata from the indexes migrated to
 warm storage.
FAILED_COLD_METADATA_CLEANUP - Failed to clean up metadata in the cold tier.
FAILED_INDEX_CREATION - Failed to create an index in the warm tier.
```

Restauração de índices frios de snapshots

Se precisar restaurar um índice de baixa atividade excluído, você pode restaurá-lo de volta ao nível de maior atividade seguindo as instruções the section called “Restauração de índices quentes de snapshots” e, em seguida, migrando o índice de volta para o nível de baixa atividade novamente. Você não pode restaurar um índice frio excluído diretamente para o nível frio. OpenSearch O serviço retém índices frios por 14 dias após a exclusão.

Cancelamento de migrações do armazenamento de baixa atividade para o armazenamento de alta atividade

Se uma migração de índice do armazenamento de baixa atividade para o armazenamento de alta atividade estiver enfileirada ou em um estado de falha, você poderá cancelá-la com a seguinte solicitação:

```
POST _cold/migration/my-index/_cancel

{
  "acknowledged" : true
}
```

Para cancelar a migração de um lote de índices (máximo de 10 por vez), especifique o ID de migração:

```
POST _cold/migration/_cancel?migration_id=my-migration-id

{
  "acknowledged" : true
}
```

Recupere o ID de migração usando a solicitação de status.

Atualizando metadados de índice de baixa atividade

Você pode atualizar os campos `start_time` e `end_time` para um índice de baixa atividade:

```
PATCH _cold/my-index
 {
 "start_time": "2020-01-01",
 "end_time": "2020-02-01"
 }
```

Não é possível atualizar o `timestamp_field` de um índice no armazenamento de baixa atividade.

> **Note**
>
> OpenSearch Os painéis não são compatíveis com o método PATCH. Use curl, Postman ou algum outro método para atualizar metadados de baixa atividade.

Exclusão de índices de baixa atividade

Se você não estiver usando uma política do ISM, poderá excluir índices frios manualmente. A seguinte solicitação exclui um índice de baixa atividade:

```
DELETE _cold/my-index

{
  "acknowledged" : true
}
```

Desabilitação do armazenamento de baixa atividade

O console OpenSearch de serviço é a maneira mais simples de desativar o armazenamento a frio. Selecione o domínio e escolha Ações, Editar configuração do cluster, depois desmarque a opção Habilitar armazenamento estático.

Para usar a AWS CLI ou a API de configuração, em`ColdStorageOptions`, defina. `"Enabled"="false"`

Antes de desabilitar o armazenamento frio, você deve excluir todos os índices frios ou migrá-los de volta para o armazenamento warm, caso contrário, a ação de desabilitar falhará.

Gerenciamento do estado do índice no Amazon OpenSearch Service

O Index State Management (ISM) no Amazon OpenSearch Service permite definir políticas de gerenciamento personalizadas que automatizam tarefas rotineiras e as aplicam a índices e padrões de indexação. Não é mais necessário configurar e gerenciar processos externos para executar operações de índice.

Uma política contém um estado padrão e uma lista de estados entre os quais o índice transita. Dentro de cada estado, é possível definir uma lista de ações a serem realizadas e das condições que acionam essas transições. Um caso de uso típico é excluir periodicamente índices antigos após um determinado período. Por exemplo, é possível definir uma política que mova seu índice para o estado `read_only` após 30 dias e, por fim, excluí-lo após 90 dias.

Depois de anexar uma política a um índice, o ISM cria um trabalho que é executado em intervalos de 5 a 8 minutos (ou 30 a 48 minutos para clusters pré-1.3) para executar ações de política, verificar condições e fazer a transição do índice para estados diferentes. O tempo base para que esse trabalho seja executado é a cada 5 minutos. Além disso, uma variação aleatória de 0 a 60% é adicionada a ele para garantir que não ocorra um surto de atividade de todos os seus índices ao mesmo tempo. O ISM não executa tarefas se o estado do cluster for vermelho.

O ISM exige o Elasticsearch 6.8 OpenSearch ou posterior.

> **Note**
>
> Esta documentação fornece uma breve visão geral do ISM e de vários exemplos de políticas. Também explica como o ISM para domínios do Amazon OpenSearch Service difere do ISM

em clusters autogerenciados OpenSearch. Para obter a documentação completa do ISM, incluindo uma referência abrangente de parâmetros, descrições de cada configuração e uma referência de API, consulte Index State Management na OpenSearch documentação.

Important

Você não pode mais utilizar modelos de índice para aplicar políticas de ISM a índices recém-criados. Você pode continuar gerenciando automaticamente índices recém-criados com o campo do modelo de ISM. Esta atualização introduz uma alteração significativa que afeta os CloudFormation modelos existentes usando essa configuração.

Criar uma política do IAM

Para começar a usar o gerenciamento de estados de índices

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Selecione o domínio para o qual você deseja criar uma política do ISM.
3. No painel do domínio, navegue até a URL dos OpenSearch painéis e faça login com seu nome de usuário e senha principais. O URL segue este formato:

```
domain-endpoint/_dashboards/
```

4. Abra o painel de navegação esquerdo em OpenSearch Painéis e escolha Gerenciamento de índices e, em seguida, Criar política.
5. Use o editor visual ou o editor JSON para criar políticas. Recomendamos que você use o editor visual, pois ele oferece uma maneira mais estruturada de definir políticas. Para obter ajuda com a criação de políticas, consulte as políticas de exemplo abaixo.
6. Depois de criar uma política, anexe-a a um ou mais índices:

```
POST _plugins/_ism/add/my-index
{
  "policy_id": "my-policy-id"
}
```

> **Note**
>
> Se o seu domínio estiver executando uma versão herdada do Elasticsearch, use `_opendistro` em vez de `_plugins`.

Como alternativa, selecione o índice em OpenSearch Painéis e escolha Aplicar política.

Políticas de exemplo

As políticas de exemplo a seguir demonstram como automatizar casos de uso comuns do ISM.

Armazenamento de atividade muito alta para alta atividade para baixa atividade

Esse exemplo de política move um índice do armazenamento dinâmico para e UltraWarm, eventualmente, para armazenamento refrigerado. Em seguida, ele exclui o índice.

O índice está inicialmente no estado `hot`. Após dez dias, o ISM o transfere para o estado `warm`. 80 dias depois, quando o índice tiver 90 dias, o ISM move o índice para o estado `cold`. Após um ano, o serviço envia uma notificação para uma sala do Amazon Chime informando que o índice está sendo excluído e, depois, o exclui permanentemente.

Observe que os índices frios exigem a operação `cold_delete` em vez da operação normal `delete`. Observe também que um `timestamp_field` explícito é necessário em seus dados para gerenciar índices frios com ISM.

```
{
  "policy": {
    "description": "Demonstrate a hot-warm-cold-delete workflow.",
    "default_state": "hot",
    "schema_version": 1,
    "states": [{
        "name": "hot",
        "actions": [],
        "transitions": [{
          "state_name": "warm",
          "conditions": {
            "min_index_age": "10d"
          }
```

```
        }]
      },
      {
        "name": "warm",
        "actions": [{
          "warm_migration": {},
          "retry": {
            "count": 5,
            "delay": "1h"
          }
        }],
        "transitions": [{
          "state_name": "cold",
          "conditions": {
            "min_index_age": "90d"
          }
        }]
      },
      {
        "name": "cold",
        "actions": [{
            "cold_migration": {
              "timestamp_field": "<your timestamp field>"
            }
          }
        ],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
             "min_index_age": "365d"
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "notification": {
            "destination": {
              "chime": {
                "url": "<URL>"
              }
            },
            "message_template": {
              "source": "The index {{ctx.index}} is being deleted."
```

```
            }
          }
        },
        {
          "cold_delete": {}
        }]
      }
    ]
  }
}
```

Reduzir a contagem de réplicas

Esta política de exemplo mais simples reduz a contagem de réplicas para zero após sete dias para conservar espaço em disco e exclui o índice após 21 dias. Essa política pressupõe que seu índice não seja crítico e não receba mais solicitações de gravação. Ter réplicas zero traz algum risco de perda de dados.

```
{
  "policy": {
    "description": "Changes replica count and deletes.",
    "schema_version": 1,
    "default_state": "current",
    "states": [{
        "name": "current",
        "actions": [],
        "transitions": [{
          "state_name": "old",
          "conditions": {
            "min_index_age": "7d"
          }
        }]
      },
      {
        "name": "old",
        "actions": [{
          "replica_count": {
            "number_of_replicas": 0
          }
        }],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
```

```
            "min_index_age": "21d"
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "delete": {}
        }],
        "transitions": []
      }
    ]
  }
}
```

Obter o snapshot de um índice

Esta política de exemplo usa a operação `snapshot` para obter um instantâneo de um índice assim que ele passa a conter pelo menos um documento. `repository` é o nome do repositório manual de snapshots que você registrou no Amazon S3. `snapshot` é o nome do snapshot. Para obter pré-requisitos para a obtenção de snapshot e etapas para registrar um repositório, consulte the section called “Criação de snapshots de índices”.

```
{
  "policy": {
    "description": "Takes an index snapshot.",
    "schema_version": 1,
    "default_state": "empty",
    "states": [{
        "name": "empty",
        "actions": [],
        "transitions": [{
          "state_name": "occupied",
          "conditions": {
            "min_doc_count": 1
          }
        }]
      },
      {
        "name": "occupied",
        "actions": [{
          "snapshot": {
            "repository": "<my-repository>",
```

```
            "snapshot": "<my-snapshot>"
            }
          }],
          "transitions": []
      }
    ]
  }
}
```

Modelos do ISM

Você pode configurar um campo `ism_template` em uma política para que, quando criar um índice que corresponda ao padrão do modelo, a política seja anexada automaticamente a esse índice. Neste exemplo, qualquer índice que você criar com um nome começando com “log” é automaticamente correspondido à política do ISM `my-policy-id`:

```
PUT _plugins/_ism/policies/my-policy-id
{
  "policy": {
    "description": "Example policy.",
    "default_state": "...",
    "states": [...],
    "ism_template": {
      "index_patterns": ["log*"],
      "priority": 100
    }
  }
}
```

Para obter um exemplo mais detalhado, consulte Exemplo de política com modelo de ISM para rolagem automática.

Diferenças

Em comparação com o OpenSearch Elasticsearch, o ISM for Amazon OpenSearch Service tem várias diferenças.

Operações do ISM

- OpenSearch O serviço oferece suporte a três operações ISM exclusivas`warm_migration`,`cold_migration`,, `cold_delete`:

- Se seu domínio estiver UltraWarmativado, a `warm_migration` ação fará a transição do índice para armazenamento aquecido.
- Se o seu domínio tiver armazenamento frio habilitado, a ação `cold_migration` passará o índice para o armazenamento frio, e a ação `cold_delete` excluirá um índice do armazenamento frio.

  Mesmo que uma dessas ações não seja concluída dentro do período de tempo limite definido, a migração ou exclusão dos índices ainda continuará. Definir uma error_notification para uma das ações acima vai notificar você de que a ação falhou se não tiver sido concluída em um período de tempo limite, mas a notificação é apenas para sua própria referência. A operação real não tem tempo limite inerente e continua a ser executada até que eventualmente seja bem-sucedida ou falhe.

- Se seu domínio OpenSearch executa o Elasticsearch 7.4 ou posterior, o OpenSearch Service oferece suporte ao ISM `open` e às operações. `close`

- Se o seu domínio executar OpenSearch o Elasticsearch 7.7 ou posterior, o OpenSearch Service suportará a operação do ISM. `snapshot`

Operações ISM de armazenamento de baixa atividade

Para índices frios, você deve especificar um `?type=_cold` parâmetro ao usar o seguinte ISM APIs:

- Adicionar política
- Remover política
- Atualizar política
- Repetir índice com falha
- Explicar índice

Estes APIs para índices frios têm as seguintes diferenças adicionais:

- Operadores curingas não são aceitos, exceto quando usados no final. Por exemplo, `_plugins/_ism/<add, remove, change_policy, retry, explain>/logstash-*` é aceito, mas `_plugins/_ism/<add, remove, change_policy, retry, explain>/iad-*-prod` não.

- Não há suporte a índices e padrões de vários índices. Por exemplo, `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs` é aceito, mas `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs,sample-data` não.

Configurações do ISM

OpenSearch e o Elasticsearch permitem que você altere todas as configurações de ISM disponíveis usando a `_cluster/settings` API. No Amazon OpenSearch Service, você só pode alterar as seguintes configurações do ISM:

- Configurações no nível do cluster:
  - `plugins.index_state_management.enabled`
  - `plugins.index_state_management.history.enabled`
- Configurações no nível do índice:
  - `plugins.index_state_management.rollover_alias`

Tutorial: como automatizar processos do Gerenciamento de estados de índice

Este tutorial demonstra como implementar uma política do ISM que automatiza tarefas de rotina de gerenciamento de índices e as aplica a índices e padrões de índices.

O Index State Management (ISM) no Amazon OpenSearch Service permite automatizar atividades recorrentes de gerenciamento de índices, evitando o uso de ferramentas adicionais para gerenciar os ciclos de vida dos índices. Você pode criar uma política que automatize essas operações com base na idade, tamanho e outras condições do índice, tudo de dentro do seu domínio do Amazon OpenSearch Service.

OpenSearch O serviço oferece suporte a três níveis de armazenamento: o estado “ativo” padrão para gravação ativa e análise de baixa latência, UltraWarm para dados somente para leitura de até três petabytes e armazenamento frio para arquivamento ilimitado a longo prazo.

Este tutorial apresenta um exemplo de caso de uso do tratamento de dados de séries temporais em índices diários. No tutorial, você configurará uma política que captura um instantâneo automatizado de cada índice anexado após 24 horas. Em seguida, ele migra o índice do estado quente padrão

para o UltraWarm armazenamento após dois dias, o armazenamento frio após 30 dias e, finalmente, exclui o índice após 60 dias.

Pré-requisitos

- Seu domínio OpenSearch de serviço deve estar executando a versão 6.8 ou posterior do Elasticsearch.
- Seu domínio deve ter um UltraWarmarmazenamento a frio ativado.
- É necessário registrar um repositório de snapshots manuais para seu domínio.
- Sua função de usuário precisa de permissões suficientes para acessar o console OpenSearch de serviço. Se necessário, valide e configure o acesso ao seu domínio.

Etapa 1: configurar a política do ISM

Primeiro, configure uma política do ISM nos OpenSearch painéis.

1. No painel do seu domínio no console OpenSearch de serviços, navegue até a URL dos OpenSearch painéis e faça login com seu nome de usuário e senha principais. O URL segue este formato: *`domain-endpoint`*`/_dashboards/`.
2. Em OpenSearch Painéis, escolha Adicionar dados de amostra e adicione um ou mais índices de amostra ao seu domínio.
3. Abra o painel de navegação esquerdo e escolha IGerenciamento de índices e Criar política.
4. Atribua o nome `ism-policy-example` à política.
5. Substitua a política padrão pela seguinte política:

```
{
  "policy": {
    "description": "Move indexes between storage tiers",
    "default_state": "hot",
    "states": [
      {
        "name": "hot",
        "actions": [],
        "transitions": [
          {
            "state_name": "snapshot",
            "conditions": {
              "min_index_age": "24h"
```

```
            }
          }
        ]
      },
      {
        "name": "snapshot",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
              "delay": "30m"
            },
            "snapshot": {
              "repository": "snapshot-repo",
              "snapshot": "ism-snapshot"
            }
          }
        ],
        "transitions": [
          {
            "state_name": "warm",
            "conditions": {
              "min_index_age": "2d"
            }
          }
        ]
      },
      {
        "name": "warm",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
              "delay": "1h"
            },
            "warm_migration": {}
          }
        ],
        "transitions": [
          {
            "state_name": "cold",
            "conditions": {
```

```
            "min_index_age": "30d"
          }
        }
      ]
    },
    {
      "name": "cold",
      "actions": [
        {
          "retry": {
            "count": 5,
            "backoff": "exponential",
            "delay": "1h"
          },
          "cold_migration": {
            "start_time": null,
            "end_time": null,
            "timestamp_field": "@timestamp",
            "ignore": "none"
          }
        }
      ],
      "transitions": [
        {
          "state_name": "delete",
          "conditions": {
            "min_index_age": "60d"
          }
        }
      ]
    },
    {
      "name": "delete",
      "actions": [
        {
          "cold_delete": {}
        }
      ],
      "transitions": []
    }
  ],
  "ism_template": [
    {
      "index_patterns": [
```

```
          "index-*"
        ],
        "priority": 100
      }
    ]
  }
}
```

> **Note**
>
> O campo `ism_template` anexa automaticamente a política a qualquer índice recém-criado que corresponda a um dos `index_patterns` especificados. Nesse caso, todos os índices que começam com `index-`. É possível modificar esse campo para corresponder a um formato de índice em seu ambiente. Para saber mais, consulte Modelos do ISM.

6. Na seção `snapshot` da política, substitua *`snapshot-repo`* pelo nome do repositório de snapshots que você registrou para o seu domínio. Se quiser, você também pode substituir *`ism-snapshot`*, que será o nome do snapshot quando ele for criado.
7. Escolha Criar. A política agora está visível na página Políticas de gerenciamento de estado.

Etapa 2: anexar a política a um ou mais índices.

Agora que você criou a política, anexe-a a um ou mais índices no cluster.

1. Vá para a guia Índices quentes e procure `opensearch_dashboards_sample`, que lista todos os índices de exemplo adicionados na etapa 1.
2. Selecione todos os índices e escolha Aplicar política e, em seguida, escolha a ism-policy-examplepolítica que você acabou de criar.
3. Escolha Aplicar.

É possível monitorar os índices à medida que eles avançam pelos vários estados na página Índices gerenciados por políticas.

Resumindo índices no Amazon OpenSearch Service com pacotes cumulativos de índices

Os pacotes cumulativos de índices no Amazon OpenSearch Service permitem que você reduza os custos de armazenamento acumulando periodicamente dados antigos em índices resumidos.

Você escolhe os campos que interessam e usa uma totalização de índices para criar um novo índice com apenas esses campos agregados em buckets de tempo menos detalhados. Você pode armazenar meses ou anos de dados históricos por uma fração do custo com a mesma performance de consulta.

Os pacotes cumulativos de índices exigem o Elasticsearch 7.9 OpenSearch ou posterior.

> **Note**
>
> Essa documentação ajuda você a começar a criar um trabalho de rollup de índices no Amazon OpenSearch Service. Para obter uma documentação abrangente, incluindo uma lista de todas as configurações disponíveis e uma referência completa da API, consulte Index Rollups na OpenSearch documentação.

Criação de um trabalho de totalização de índices

Para começar, escolha Gerenciamento de índices em OpenSearch painéis. Selecione Rollup Jobs (Trabalhos de totalização) e escolha Create rollup job (Criar trabalho de totalização).

Etapa 1: configurar índices

Configure os índices de origem e de destino. O índice de origem é aquele que você deseja totalizar. O índice de destino é onde os resultados do conjunto de índices são salvos.

Depois de criar um trabalho de totalização de índices, você não poderá alterar suas seleções de índice.

Etapa 2: Definir agregações e métricas

Selecione os atributos com as agregações (termos e histogramas) e métricas (média, soma, máximo, mínimo e contagem de valores) que deseja totalizar. Certifique-se de não adicionar muitos atributos altamente granulares, porque você não economizará muito espaço.

Etapa 3: Especificar agendamentos

Especifique um agendamento para agrupar seus índices à medida que são ingeridos. O trabalho de totalização é habilitado por padrão.

Etapa 4: revisar e criar

Revise sua configuração e selecione Create (Criar).

Etapa 5: Pesquisar o índice de destino

Você pode usar a API `_search` padrão para pesquisar o índice de destino. Você não pode acessar a estrutura interna dos dados no índice de destino porque o plug-in reescreve automaticamente a consulta em segundo plano para se adequar ao índice de destino. Isso é para garantir que você possa usar a mesma consulta para o índice de origem e de destino.

Para consultar o índice de destino, defina `size` como 0:

```
GET target_index/_search
{
  "size": 0,
  "query": {
    "match_all": {}
  },
  "aggs": {
    "avg_cpu": {
      "avg": {
        "field": "cpu_usage"
      }
    }
  }
}
```

> **Note**
>
> OpenSearch as versões 2.2 e posteriores oferecem suporte à pesquisa de vários índices cumulativos em uma solicitação. OpenSearch as versões anteriores à 2.2 e as versões legadas do Elasticsearch OSS oferecem suporte apenas a um índice cumulativo por pesquisa.

Transformação de índices no Amazon Service OpenSearch

Considerando que os trabalhos de recolhimento de índices permitem que você reduza a granularidade dos dados totalizando dados antigos em índices condensados, os trabalhos de transformação permitem criar uma visualização resumida e diferente dos dados centrada em determinados campos para que você possa visualizar ou analisar os dados de diferentes maneiras.

As transformações de índice têm uma interface de usuário de OpenSearch painéis e uma API REST. O recurso requer OpenSearch 1.0 ou posterior.

> **Note**
>
> Esta documentação fornece uma breve visão geral das transformações de índice para ajudar você a começar a usá-las em um domínio do Amazon OpenSearch Service. Para obter uma documentação abrangente e uma referência da API REST, consulte Index transforms na OpenSearch documentação de código aberto.

Criação de um trabalho de transformação de índice

Se você não tiver dados em seu cluster, use os dados de voo de amostra nos OpenSearch painéis para experimentar trabalhos de transformação. Depois de adicionar os dados, inicie o OpenSearch Dashboards. Em seguida, escolha Index management (Gerenciamento de índices),Transform Jobs (Trabalhos de transformação) e Create Transform Job (Criar trabalho de transformação).

Etapa 1: escolher índices

Na seção Indexes (Índices), selecione o índice de origem e o índice de destino. Você pode selecionar um índice de destino existente, ou criar um novo inserindo um nome para ele.

Se você quiser transformar apenas um subconjunto do seu índice de origem, escolha Adicionar filtro de dados e use a OpenSearch consulta DSL para especificar um subconjunto do seu índice de origem.

Etapa 2: Escolher campos

Depois de escolher seus índices, escolha os campos que deseja usar no trabalho de transformação, bem como se deseja usar agrupamentos ou agregações.

- Você pode usar agrupamentos para colocar seus dados em buckets separados em seu índice transformado. Por exemplo, se você quiser agrupar todos os destinos aeroportuários nos dados de voo de amostra, agrupe o `DestAirportID` campo em um `DestAirportID_terms` campo de destino e você poderá encontrar o aeroporto agrupado IDs em seu índice transformado após a conclusão do trabalho de transformação.
- Por outro lado, as agregações permitem realizar cálculos simples. Por exemplo, você pode incluir uma agregação no trabalho de transformação para definir um novo campo de `sum_of_total_ticket_price` que calcula a soma de todas as passagens aéreas. Em seguida, você pode analisar os novos dados em seu índice transformado.

Etapa 3: Especificar um agendamento

Os trabalhos de transformação são habilitados por padrão e executados de acordo com agendamentos. Em Transform execution interval (Transformar intervalo de execução), especifique um intervalo em minutos, horas ou dias.

Etapa 4: Revisar e monitorar

Revise sua configuração e selecione Create (Criar). Em seguida, monitore a coluna Transform job status (Status do trabalho de transformação).

Etapa 5: Pesquisar o índice de destino

Após a conclusão do trabalho, você pode usar a API `_search` padrão para pesquisar o índice de destino.

Por exemplo, após executar um trabalho de transformação que transforma os dados de voo com base no campo `DestAirportID`, você poderá executar a seguinte solicitação para retornar todos os campos que têm um valor `SFO`:

```
GET target_index/_search
{
  "query": {
    "match": {
      "DestAirportID_terms" : "SFO"
    }
  }
}
```

Replicação entre clusters para Amazon Service OpenSearch

Com a replicação entre clusters no Amazon OpenSearch Service, você pode replicar índices, mapeamentos e metadados de usuários de um domínio do Serviço para outro. OpenSearch Usar a replicação entre clusters ajuda a garantir a recuperação de desastres em caso de interrupção, e permite replicar dados em datacenters geograficamente distantes para reduzir a latência. Você paga taxas AWS de transferência de dados padrão pelos dados transferidos entre domínios.

A replicação entre clusters segue um modelo de replicação ativo-passivo no qual o índice local ou seguidor extrai dados do índice remoto ou líder. O índice líder se refere à fonte dos dados ou ao índice do qual você deseja replicar os dados. O índice seguidor se refere ao destino dos dados ou ao índice para o qual você deseja replicar os dados.

A replicação entre clusters está disponível em domínios que executam o Elasticsearch 7.10 ou 1.1 ou posterior. OpenSearch

> **Note**
>
> Esta documentação descreve como configurar a replicação entre clusters a partir da perspectiva do Amazon OpenSearch Service. Isso inclui usar o Console de gerenciamento da AWS para configurar conexões entre clusters, o que não é possível em um cluster autogerenciado OpenSearch . Para obter a documentação completa, incluindo uma referência de configurações e uma referência abrangente de API, consulte Replicação entre clusters na OpenSearch documentação.

Tópicos

- Limitações
- Pré-requisitos
- Requisitos de permissão
- Configurar uma conexão entre clusters
- Como iniciar a replicação
- Confirmar replicação
- Interromper e retomar a replicação
- Encerrar a replicação
- Seguir automaticamente

- Atualizar domínios conectados

Limitações

A replicação entre clusters tem as seguintes limitações:

- Você não pode replicar dados entre domínios do Amazon OpenSearch Service e clusters autogerenciados OpenSearch ou do Elasticsearch.
- Você não pode replicar um índice de um domínio seguidor para outro domínio seguidor. Se você quiser replicar um índice para vários domínios seguidores, só poderá replicá-lo a partir do único domínio líder.
- Um domínio pode ser conectado, por meio de uma combinação de conexões de entrada e saída, a um máximo de 20 outros domínios.
- Quando você configura inicialmente uma conexão entre clusters, o domínio líder deve estar na mesma versão ou em uma versão superior à do domínio seguidor.
- Você não pode usar CloudFormation para conectar domínios.
- Não é possível usar a replicação entre clusters em instâncias M3 ou expansíveis (T2 e T3).
- Você não pode replicar dados entre índices UltraWarm ou índices frios. Ambos os índices devem estar em um armazenamento quente.
- Quando você exclui um índice do domínio líder, o índice correspondente no domínio seguidor não é excluído automaticamente.
- A replicação entre clusters não é suportada entre o padrão e o opcional. Ambos os domínios devem estar em regiões padrão ou em regiões opcionais.

Pré-requisitos

Antes de configurar a replicação entre clusters, verifique se os domínios atendem aos seguintes requisitos:

- Elasticsearch 7.10 ou 1.1 ou posterior OpenSearch
- Controle de acesso refinado habilitado
- Node-to-node criptografia ativada
- Os índices de líderes devem ter sido `index.soft_deletes.enabled` definidos como. `true` Os índices criados no Elasticsearch 7.0 ou OpenSearch 1.0 e versões posteriores têm essa

configuração ativada por padrão. No entanto, os índices criados no Elasticsearch 6.x e depois atualizados permanecem. `soft_deletes=false` Para replicar esses índices, você deve primeiro reindexá-los.

Para verificar se um índice tem as exclusões reversíveis ativadas:

```
GET <index-name>/_settings?
include_defaults=true&flat_settings=true&filter_path=*.settings.index.soft_deletes.enabled
```

Se `soft_deletes` estiver`false`, reindexe os dados em um novo índice antes de iniciar a replicação.

Requisitos de permissão

Para iniciar a replicação, você deve incluir a permissão `es:ESCrossClusterGet` no domínio remoto (líder). Recomendamos a seguinte política do IAM no domínio remoto. Essa política também permite executar outras operações, como indexar documentos e executar pesquisas padrão:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/leader-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
```

```
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/leader-domain"
    }
  ]
}
```

Verifique se a permissão `es:ESCrossClusterGet` é aplicada a `/leader-domain` e não a `/leader-domain/*`.

Para que usuários não administradores realizem atividades de replicação, também é preciso que eles sejam mapeados às permissões apropriadas. A maioria das permissões corresponde a operações de API REST específicas. Por exemplo, a permissão `indices:admin/plugins/replication/index/_resume` permite que você retome a replicação de um índice. Para obter uma lista completa de permissões, consulte Permissões de replicação na OpenSearch documentação.

> **Note**
>
> Os comandos para iniciar a replicação e criar uma regra de replicação são casos especiais. Como eles invocam processos em segundo plano nos domínios do líder e do seguidor, você deve passar um `leader_cluster_role` e `follower_cluster_role` na solicitação. OpenSearch O serviço usa essas funções em todas as tarefas de replicação de back-end. Para obter informações sobre mapeamento e uso dessas funções, consulte Mapear as funções do cluster de líder e seguidor na OpenSearch documentação.

Configurar uma conexão entre clusters

Para replicar índices de um domínio para outro, você precisa configurar uma conexão entre clusters entre os domínios. A maneira mais fácil de conectar domínios é através da guia Conexões do painel de domínio. Você também pode usar a API de configuração ou a CLI da AWS. Como a replicação entre clusters segue um modelo “pull”, você inicia as conexões a partir do domínio seguidor.

> **Note**
>
> Se você conectou anteriormente dois domínios para executar pesquisas entre clusters, essa mesma conexão não pode ser usada para replicação. A conexão é marcada como `SEARCH_ONLY` no console. Para executar a replicação entre dois domínios conectados

anteriormente, você deve excluir a conexão e recriá-la. Assim que você tiver feito isso, a conexão estará disponível para a pesquisa entre clusters e a replicação entre clusters.

Como configurar uma conexão

1. No console do Amazon OpenSearch Service, selecione o domínio do seguidor, vá até a guia Conexões e escolha Solicitar.
2. Em Alias de conexão, insira um nome para a conexão.
3. Escolha entre se conectar a um domínio em sua Conta da AWS região ou em outra conta ou região.

   - Para se conectar a um domínio em sua Conta da AWS região, selecione o domínio e escolha Solicitar.
   - Para se conectar a um domínio em outra região Conta da AWS ou em outra, especifique o ARN do domínio remoto e escolha Solicitar.

OpenSearch O serviço valida a solicitação de conexão. Se os domínios forem incompatíveis, a conexão falhará. Se a validação for bem-sucedida, ela será enviada ao domínio de destino para aprovação. Quando o domínio de destino aprova a solicitação, você pode iniciar a replicação.

A replicação entre clusters oferece suporte à replicação bidirecional. Isso significa que você pode criar uma conexão de saída do domínio A para o domínio B e outra conexão de saída do domínio B para o domínio A. Você pode então configurar a replicação para que o domínio A siga um índice no domínio B e o domínio B siga um índice no domínio A.

Como iniciar a replicação

Depois de estabelecer uma conexão entre clusters, você pode começar a replicar dados. Primeiro, crie um índice no domínio líder a ser replicado:

```
PUT leader-01
```

Para replicar esse índice, envie esse comando ao domínio seguidor:

```
PUT _plugins/_replication/follower-01/_start
{
   "leader_alias": "connection-alias",
```

```
   "leader_index": "leader-01",
   "use_roles":{
      "leader_cluster_role": "all_access",
      "follower_cluster_role": "all_access"
   }
}
```

Você pode encontrar o alias de conexão na guia Conexões no painel do domínio.

Este exemplo pressupõe que um administrador esteja emitindo a solicitação e usa `all_access` para `leader_cluster_role` e `follower_cluster_role` para simplificar. Em ambientes de produção, no entanto, recomendamos que você crie usuários de replicação nos índices líder e seguidor e os mapeie de acordo. Os nomes de usuário devem ser idênticos. Para obter informações sobre essas funções e como mapeá-las, consulte Mapear as funções do cluster de líderes e seguidores na OpenSearch documentação.

Confirmar replicação

Para confirmar se a replicação está acontecendo, obtenha o status da replicação:

```
GET _plugins/_replication/follower-01/_status

{
  "status" : "SYNCING",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01",
  "syncing_details" : {
    "leader_checkpoint" : -5,
    "follower_checkpoint" : -5,
    "seq_no" : 0
  }
}
```

Os valores de ponto de verificação de líder e seguidor começam como números inteiros negativos e refletem o número de fragmentos que você tem (-1 para um fragmento, -5 para cinco fragmentos e assim por diante). Os valores aumentam para números inteiros positivos a cada alteração que você fizer. Se os valores forem os mesmos, significa que os índices estão totalmente sincronizados. Você pode usar esses valores de ponto de verificação para medir a latência de replicação em seus domínios.

Para validar ainda mais a replicação, adicione um documento ao índice líder:

```
PUT leader-01/_doc/1
{
   "Doctor Sleep":"Stephen King"
}
```

E confirme que ele aparece no índice seguidor:

```
GET follower-01/_search

{
    ...
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "follower-01",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 1.0,
        "_source" : {
          "Doctor Sleep" : "Stephen King"
        }
      }
    ]
  }
}
```

Interromper e retomar a replicação

Você pode interromper temporariamente a replicação se precisar corrigir problemas ou reduzir a carga no domínio líder. Envie essa solicitação ao domínio seguidor. Certifique-se de incluir um corpo da solicitação vazio:

```
POST _plugins/_replication/follower-01/_pause
{}
```

Em seguida, obtenha o status para garantir que a replicação seja interrompida:

```
GET _plugins/_replication/follower-01/_status
```

```
{
  "status" : "PAUSED",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01"
}
```

Quando terminar de fazer as alterações, retome a replicação. Envie essa solicitação ao domínio seguidor. Certifique-se de incluir um corpo da solicitação vazio:

```
POST _plugins/_replication/follower-01/_resume
{}
```

Não será possível retomar a replicação depois que ela for pausada por mais de 12 horas. Você deve interromper a replicação, excluir o índice seguidor e reiniciar a replicação do líder.

Encerrar a replicação

Quando você encerra completamente a replicação, o índice seguidor deixa de seguir o líder e torna-se um índice padrão. Você não pode reiniciar uma replicação depois de encerrá-la.

Encerre a replicação do domínio seguidor. Certifique-se de incluir um corpo da solicitação vazio:

```
POST _plugins/_replication/follower-01/_stop
{}
```

Seguir automaticamente

Você pode definir um conjunto de regras de replicação em um único domínio líder que replica automaticamente índices correspondentes a um padrão especificado. Quando um índice no domínio líder corresponde a um dos padrões (por exemplo,`books*`), um índice de seguidores correspondente é criado no domínio seguidor. OpenSearch O serviço replica todos os índices existentes que correspondam ao padrão, bem como os novos índices criados por você. Não replica índices que já existem no domínio seguidor.

Para replicar todos os índices (com exceção dos índices criados pelo sistema e aqueles que já existem no domínio seguidor), use um padrão curinga (`*`).

Criar uma regra de replicação

Crie uma regra de replicação no domínio do seguidor e especifique o nome da conexão entre clusters:

```
POST _plugins/_replication/_autofollow
{
   "leader_alias" : "connection-alias",
   "name": "rule-name",
   "pattern": "books*",
   "use_roles":{
      "leader_cluster_role": "all_access",
      "follower_cluster_role": "all_access"
   }
}
```

Você pode encontrar o alias de conexão na guia Conexões no painel do domínio.

Este exemplo pressupõe que um administrador esteja emitindo a solicitação e usa `all_access` como as funções de domínio líder e seguidor para simplificar. Em ambientes de produção, no entanto, recomendamos que você crie usuários de replicação nos índices líder e seguidor e os mapeie de acordo. Os nomes de usuário devem ser idênticos. Para obter informações sobre essas funções e como mapeá-las, consulte Mapear as funções do cluster de líderes e seguidores na OpenSearch documentação.

Para recuperar uma lista de regras de replicação existentes em um domínio, use a operação da API de estatísticas de auto-follow.

Para testar a regra, crie um índice que corresponda ao padrão no domínio líder:

```
PUT books-are-fun
```

E confira se sua réplica aparece no domínio seguidor:

```
GET _cat/indices

health status index          uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   books-are-fun  ldfHO78xYYdxRMULuiTvSQ     1   1          0            0
     208b           208b
```

Excluir uma regra de replicação

Quando você exclui uma regra de replicação, o OpenSearch Serviço interrompe a replicação de novos índices que correspondam ao padrão, mas continua a atividade de replicação existente até que você interrompa a replicação desses índices.

Exclua regras de replicação do domínio seguidor:

```
DELETE _plugins/_replication/_autofollow
{
   "leader_alias" : "connection-alias",
   "name": "rule-name"
}
```

Atualizar domínios conectados

Para atualizar a versão do mecanismo de dois domínios que têm uma conexão entre clusters, atualize primeiro o domínio seguidor e depois o líder. Não exclua a conexão entre eles, caso contrário, a replicação será interrompida e não será possível retomá-la.

Migração de índices do Amazon OpenSearch Service usando reindexação remota

A reindexação remota permite copiar índices de um domínio do Amazon OpenSearch Service para outro. Você pode migrar índices de qualquer domínio de OpenSearch serviço ou clusters autogerenciados OpenSearch e do Elasticsearch.

Com domínio e índice remotos, se referem à fonte dos dados ou ao domínio e índice dos quais você deseja copiar os dados. Um domínio e índice local referem-se ao destino dos dados ou ao domínio e índice para os quais você deseja copiar os dados.

A reindexação remota exige OpenSearch 1.0 ou posterior, ou Elasticsearch 6.7 ou posterior, no domínio local. O domínio remoto deve ser inferior ou da mesma versão principal que o domínio local. As versões do Elasticsearch são consideradas inferiores às OpenSearch versões, o que significa que você pode reindexar dados de domínios do Elasticsearch para domínios. OpenSearch Dentro da mesma versão principal, o domínio remoto pode ser qualquer versão secundária. Por exemplo, a reindexação remota do Elasticsearch 7.10.x para 7.9 é suportada, mas OpenSearch 1.0 para o Elasticsearch 7.10.x não é suportada.

> **Note**
>
> Esta documentação descreve como reindexar dados entre os domínios do Amazon OpenSearch Service. Para obter a documentação completa da `reindex` operação, incluindo etapas detalhadas e opções suportadas, consulte o documento Reindex na OpenSearch documentação.

Tópicos

- Pré-requisitos
- Reindexar dados entre os domínios da Internet OpenSearch do Serviço
- Reindexe dados entre domínios OpenSearch de serviço quando o controle remoto está em uma VPC
- Reindexe dados entre domínios que não são OpenSearch de serviço
- Reindexar conjuntos de dados grandes
- Configurações da reindexação remota

Pré-requisitos

A reindexação remota tem os seguintes requisitos:

- O domínio remoto deve ser acessível pelo domínio local. Para um domínio remoto que reside em uma VPC, o domínio local deve ter acesso à VPC. Este processo varia de acordo com a configuração de rede, mas geralmente envolve a conexão a uma VPN ou rede gerenciada ou o uso a conexão de endpoint da VPC nativa. Para saber mais, consulte the section called “Suporte à VPC”.
- A solicitação deve ser autorizada pelo domínio remoto como qualquer outra solicitação REST. Se o domínio remoto tiver o controle de acesso detalhado habilitado, você deve ter permissão para executar a reindexação no domínio remoto e ler o índice no domínio local. Para obter mais considerações de segurança, consulte the section called “Controle de acesso refinado”.
- Recomendamos criar um índice com a configuração desejada no domínio local antes de iniciar o processo de reindexação.
- Se o domínio usar um tipo de instância T2 ou T3 para os nós de dados, não será possível usar a reindexação remota.

Reindexar dados entre os domínios da Internet OpenSearch do Serviço

O cenário mais básico é que o índice remoto esteja no mesmo Região da AWS que seu domínio local com um endpoint acessível ao público e você tenha assinado as credenciais do IAM.

A partir do domínio remoto, especifique o índice remoto do qual a reindexação será feita e o índice local para o qual reindexar:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Você deve adicionar 443 no final do endpoint do domínio remoto para verificar a validade.

Para verificar se o índice foi copiado para o domínio local, envie essa solicitação para o domínio local:

```
GET local_index/_search
```

Se a reindexação remota estiver em uma região diferente do domínio local, passe seu nome de região, como nesta solicitação de exemplo:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "region": "eu-west-1"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
```

```
  }
}
```

No caso de regiões isoladas, como AWS GovCloud (US) regiões da China, o endpoint pode não estar acessível porque seu usuário do IAM não é reconhecido nessas regiões.

Se o domínio remoto estiver protegido por autenticação básica, especifique o nome de usuário e senha:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Reindexe dados entre domínios OpenSearch de serviço quando o controle remoto está em uma VPC

Cada domínio OpenSearch de serviço é composto por sua própria infraestrutura interna de nuvem privada virtual (VPC). Quando você cria um novo domínio em uma OpenSearch Service VPC existente, uma interface de rede elástica é criada para cada nó de dados na VPC.

Como a operação de reindexação remota é executada a partir do domínio de OpenSearch serviço remoto e, portanto, dentro de sua própria VPC privada, você precisa de uma forma de acessar a VPC do domínio local. Você pode fazer isso usando o recurso de conexão de endpoint VPC integrado para estabelecer uma conexão ou configurando um proxy. AWS PrivateLink

Se o seu domínio local usa a OpenSearch versão 1.0 ou posterior, você pode usar o console ou o AWS CLI para criar uma AWS PrivateLink conexão. Uma AWS PrivateLink conexão permite que os recursos na VPC local se conectem de forma privada aos recursos na VPC remota dentro da mesma. Região da AWS

Para criar uma conexão de endpoint da VPC, o domínio de origem a ser reindexado deve estar em uma VPC local, e os domínios de origem e de destino devem estar na mesma Região da AWS.

Reindexe os dados com o Console de gerenciamento da AWS

Você pode usar a reindexação remota com o console para copiar índices entre dois domínios que compartilham uma conexão de endpoint da VPC.

1. Navegue até o console do Amazon OpenSearch Service emhttps://console.aws.amazon.com/aos/.
2. No painel de navegação à esquerda, escolha Domínios.
3. Selecione o domínio local ou o domínio para o qual você deseja copiar dados. Isso abre a página de detalhes do domínio. Selecione a guia Conexões abaixo das informações gerais e escolha Solicitar.
4. Na página Solicitar conexão, selecione Conexão de endpoint da VPC para seu modo de conexão e insira outros detalhes relevantes. Esses detalhes incluem o domínio remoto, que é o domínio do qual você deseja copiar dados. Em seguida, escolha Solicitar.
5. Navegue até a página de detalhes do domínio remoto, selecione a guia Conexões e encontre a tabela Conexões de entrada. Marque a caixa de seleção ao lado do nome do domínio do qual você acabou de criar a conexão (o domínio local). Escolha Aprovar.
6. Retorne ao domínio local, escolha a guia Conexões e encontre a tabela Conexões de saída. Depois que a conexão entre os dois domínios estiver ativa, um endpoint ficará disponível na coluna Endpoint na tabela. Copie o endpoint.
7. Abra o painel do domínio local e escolha Ferramentas de desevolvedor na barra de navegação à esquerda. Para confirmar que o índice do domínio remoto ainda não existe no seu domínio local, execute a seguinte solicitação GET. *remote-domain-index-name*Substitua pelo seu próprio nome de índice.

   ```
   GET remote-domain-index-name/_search
   {
      "query":{
         "match_all":{}
      }
   }
   ```

   Na saída, você verá um erro que indica que o índice não foi encontrado.

8. Abaixo da sua solicitação GET, crie uma solicitação POST e use seu endpoint como host remoto, da seguinte maneira.

```
POST _reindex
{
   "source":{
      "remote":{
         "host":"connection-endpoint",
         "username":"username",
         "password":"password"
      },
      "index":"remote-domain-index-name"
   },
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

   Execute essa solicitação.

9. Execute a solicitação GET novamente. A saída agora deve indicar que o índice local existe. Você pode consultar esse índice para verificar se OpenSearch copiou todos os dados do índice remoto.

Reindexe dados com operações da API OpenSearch de serviço

Você pode usar a reindexação remota com a API para copiar índices entre dois domínios que compartilham uma conexão de endpoint da VPC.

1. Use a operação da CreateOutboundConnectionAPI para solicitar uma nova conexão do seu domínio local com seu domínio remoto.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/cc/outboundConnection

{
   "ConnectionAlias": "remote-reindex-example",
   "ConnectionMode": "VPC_ENDPOINT",
   "LocalDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "local-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
```

```
      }
   },
   "RemoteDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "remote-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
      }
   }
}
```

Você recebe um `ConnectionId` na resposta. Salve essa ID para a próxima etapa.

2. Use a operação da AcceptInboundConnectionAPI com seu ID de conexão para aprovar a solicitação do domínio local.

```
PUT https://es.region.amazonaws.com/2021-01-01/opensearch/cc/
inboundConnection/ConnectionId/accept
```

3. Use a operação DescribeOutboundConnectionsda API para recuperar o endpoint do seu domínio remoto.

```
{
    "Connections": [
        {
            "ConnectionAlias": "remote-reindex-example",
            "ConnectionId": "connection-id",
            "ConnectionMode": "VPC_ENDPOINT",
            "ConnectionProperties": {
                "Endpoint": "connection-endpoint"
            },
            ...
        }
    ]
}
```

Salve o *connection-endpoint* para usar na Etapa 5.

4. Para confirmar que o índice do domínio remoto ainda não existe no seu domínio local, execute a seguinte solicitação GET. *remote-domain-index-name*Substitua pelo seu próprio nome de índice.

```
GET local-domain-endpoint/remote-domain-index-name/_search
{
   "query":{
      "match_all":{}
   }
}
```

Na saída, você verá um erro que indica que o índice não foi encontrado.

5. Crie uma solicitação POST e use seu endpoint como host remoto, da seguinte maneira.

```
POST local-domain-endpoint/_reindex
{
   "source":{
      "remote":{
         "host":"connection-endpoint",
         "username":"username",
         "password":"password"
      },
      "index":"remote-domain-index-name"
   },
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

Execute essa solicitação.

6. Execute a solicitação GET novamente. A saída agora deve indicar que o índice local existe. Você pode consultar esse índice para verificar se OpenSearch copiou todos os dados do índice remoto.

Se o domínio remoto estiver hospedado em uma VPC e você não quiser usar o atributo de conexão endpoint da VPC, você deverá configurar um proxy com um endpoint acessível publicamente. Nesse caso, o OpenSearch Service exige um endpoint público porque não tem a capacidade de enviar tráfego para sua VPC.

Quando você executa um domínio no modo de VPC, um ou mais endpoints são colocados na sua VPC. No entanto, esses endpoints são apenas para tráfego que entra no domínio dentro da VPC e não permitem tráfego na própria VPC.

O comando remote reindex é executado a partir do domínio local, portanto, o tráfego de origem não consegue usar esses endpoints para acessar o domínio remoto. É por isso que um proxy é necessário nesse caso de uso. O domínio proxy deve ter um certificado assinado por uma autoridade de certificação (CA) pública. Não há suporte a certificados CA autoassinados ou privados.

Reindexe dados entre domínios que não são OpenSearch de serviço

Se o índice remoto estiver hospedado fora do OpenSearch Service, como em uma instância EC2 autogerenciada, defina o `external` parâmetro como: `true`

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password",
      "external": true
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Nesse caso, somente a autenticação básica com um nome de usuário e senha é suportada. O domínio remoto deve ter um endpoint acessível ao público (mesmo que esteja na mesma VPC do domínio de serviço OpenSearch local) e um certificado assinado por uma CA pública. Não há suporte para certificados CA autoassinados ou privados.

Reindexar conjuntos de dados grandes

A reindexação remota envia uma solicitação de rolagem para o domínio remoto com os seguintes valores padrão:

- Contexto de pesquisa de 5 minutos
- Tempo limite de soquete de 30 segundos
- Tamanho do lote 1.000

Recomendamos ajustar esses parâmetros para acomodar seus dados. Para documentos grandes, considere um tamanho de lote menor, and/or maior tempo limite. Para saber mais, consulte Resultados paginados.

```
POST _reindex?pretty=true&scroll=10h&wait_for_completion=false
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "socket_timeout": "60m"
    },
    "size": 100,
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Também recomendamos adicionar as seguintes configurações ao índice local para melhorar a performance:

```
PUT local_index
{
  "settings": {
    "refresh_interval": -1,
    "number_of_replicas": 0
  }
}
```

Após a conclusão do processo de reindexação, você poderá definir a contagem de réplicas desejada e remover a configuração de intervalo de atualização.

Para reindexar somente um subconjunto de documentos selecionados por meio de uma consulta, envie esta solicitação para o domínio local:

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443"
```

```
    },
    "index": "remote_index",
    "query": {
      "match": {
        "field_name": "text"
      }
    }
  },
  "dest": {
    "index": "local_index"
  }
}
```

A reindexação remota não oferece suporte a fatiamento. Por isso, você não pode executar várias operações de rolagem para a mesma solicitação em paralelo.

Configurações da reindexação remota

Além das opções de reindexação padrão, o OpenSearch Service oferece suporte às seguintes opções:

Opções	Valores válidos	Description	Obrigatório
externo	Booleano	Se o domínio remoto não for um domínio OpenSearch de serviço ou se você estiver reindexando entre dois domínios VPC, especifique como. `true`	Não
region	String	Se o domínio remoto estiver em uma região diferente, especifique o nome da região.	Não

Gerenciando dados de séries temporais no Amazon OpenSearch Service com fluxos de dados

Um fluxo de trabalho típico para gerenciar dados de séries temporais envolve várias etapas, como criar um alias de índice de sobreposição, definir um índice de gravação e definir mapeamentos e configurações comuns para os índices de apoio.

Os fluxos de dados no Amazon OpenSearch Service ajudam a simplificar esse processo de configuração inicial. Os fluxos de dados funcionam “fora da caixa” para dados baseados em tempo, como logs de aplicações que, normalmente, são de natureza somente anexação.

Os fluxos de dados exigem a OpenSearch versão 1.0 ou posterior.

> **Note**
>
> Esta documentação fornece etapas básicas para ajudar você a começar a usar fluxos de dados em um domínio do Amazon OpenSearch Service. Para obter uma documentação abrangente, consulte Fluxos de dados na OpenSearch documentação.

Conceitos básicos de fluxos de dados

Um fluxo de dados é composto internamente por vários índices de apoio. As solicitações de pesquisa são roteadas para todos os índices de apoio, enquanto as solicitações de indexação são roteadas para o índice de gravação mais recente.

Etapa 1: Criar um modelo de índice

Para criar um fluxo de dados, primeiro você precisa criar um modelo de índice que configura um conjunto de índices como um fluxo de dados. O objeto `data_stream` indica que ele é um fluxo de dados, e não um modelo de índice regular. O padrão de índice corresponde ao nome do fluxo de dados:

```
PUT _index_template/logs-template
{
  "index_patterns": [
    "my-data-stream",
    "logs-*"
  ],
  "data_stream": {},
```

```
  "priority": 100
}
```

Nesse caso, cada documento ingerido deve ter um campo `@timestamp`. Você também pode definir seu campo de datação personalizado como uma propriedade no objeto `data_stream`:

```
PUT _index_template/logs-template
{
  "index_patterns": "my-data-stream",
  "data_stream": {
    "timestamp_field": {
      "name": "request_time"
    }
  }
}
```

Etapa 2: Criar um stream de dados

Depois de criar um modelo de índice, você poderá começar a ingerir dados diretamente sem criar um fluxo de dados.

Como temos um modelo de índice correspondente com um `data_stream` objeto, cria OpenSearch automaticamente o fluxo de dados:

```
POST logs-staging/_doc
{
  "message": "login attempt failed",
  "@timestamp": "2013-03-01T00:00:00"
}
```

Etapa 3: Ingerir dados no fluxo de dados

Para ingerir dados em um stream de dados, você pode usar a indexação APIs regular. Certifique-se de que todos os documentos indexados tenham um campo de carimbo de data/hora. Se tentar ingerir um documento que não tenha um campo de carimbo de data/hora, você receberá uma mensagem de erro.

```
POST logs-redis/_doc
{
  "message": "login attempt",
  "@timestamp": "2013-03-01T00:00:00"
```

```
}
```

Etapa 4: Pesquisar um fluxo de dados

Você pode pesquisar um fluxo de dados da mesma forma que pesquisa um índice regular ou um alias de índice. A operação de pesquisa aplica-se a todos os índices de apoio (todos os dados presentes no fluxo).

```
GET logs-redis/_search
{
  "query": {
    "match": {
      "message": "login"
    }
  }
}
```

Etapa 5: Rolar um fluxo de dados

Você pode configurar um Gerenciamento de estados de índices (ISM) para automatizar o processo de rolagem para o fluxo de dados. A política do ISM é aplicada aos índices de apoio no momento da sua criação. Quando você associa uma política a um fluxo de dados, ela afeta apenas os índices de apoio futuros desse fluxo de dados. Você também não precisa fornecer a configuração `rollover_alias`, porque a política ISM infere essas informações do índice de suporte.

> **Note**
>
> Se você migrar um índice de apoio para o armazenamento a frio, OpenSearch removerá esse índice do fluxo de dados. Mesmo se você mover o índice de volta para UltraWarm, o índice permanecerá independente e não fará parte do fluxo de dados original. Depois que um índice for removido do fluxo de dados, a pesquisa no fluxo não retornará nenhum dado do índice.

> **Warning**
>
> O índice de gravação de um fluxo de dados não pode ser migrado para o armazenamento de baixa atividade. Se deseja migrar dados do seu fluxo de dados para o armazenamento de baixa atividade, você deve reverter o fluxo de dados antes da migração.

Etapa 6: gerenciar fluxos de dados em painéis OpenSearch

Para gerenciar fluxos de dados a partir de OpenSearch painéis, abra OpenSearchpainéis, escolha Gerenciamento de índices, selecione Índices ou índices gerenciados por políticas.

Etapa 7: Excluir um fluxo de dados

A operação de exclusão primeiro exclui os índices de apoio de um fluxo de dados e, em seguida, exclui o próprio fluxo de dados.

Para excluir um fluxo de dados e todos os seus índices de apoio ocultos:

```
DELETE _data_stream/name_of_data_stream
```

Monitoramento de dados no Amazon OpenSearch Service

Monitore proativamente seus dados no Amazon OpenSearch Service com alertas e detecção de anomalias. Configure alertas para receber notificações quando seus dados excederem determinados limites. A detecção de anomalias usa o machine learning para detectar automaticamente todas as discrepâncias em seus dados de streaming. Você pode emparelhar a detecção de anomalias com alertas para garantir que seja notificado assim que uma anomalia for detectada.

Tópicos

- Configurando alertas no Amazon Service OpenSearch
- Detecção de anomalias no Amazon Service OpenSearch

Configurando alertas no Amazon Service OpenSearch

Configure alertas no Amazon OpenSearch Service para ser notificado quando os dados de um ou mais índices atenderem a determinadas condições. Por exemplo, talvez você queira receber um e-mail se a aplicação registrar mais de cinco erros HTTP 503 em uma hora, ou talvez queira notificar um desenvolvedor caso nenhum documento novo tenha sido indexado nos últimos 20 minutos.

O alerta requer o Elasticsearch 6.2 OpenSearch ou posterior.

> **Note**
>
> Esta documentação fornece uma breve visão geral dos alertas e destaca como os alertas em um domínio do Amazon OpenSearch Service são diferentes dos alertas em um cluster de código aberto. OpenSearch Para obter a documentação completa de alertas, incluindo uma referência abrangente da API, uma lista dos campos de solicitação disponíveis para monitores compostos e descrições das variáveis de gatilho e ações disponíveis, consulte Alertas na documentação. OpenSearch

Tópicos

- Permissões de alertas
- Conceitos básicos dos alertas
- Notificações
- Diferenças

Permissões de alertas

O recurso de alertas oferece suporte ao controle de acesso refinado. Para obter detalhes sobre como combinar e combinar permissões de acordo com seu caso de uso, consulte Segurança de alertas na OpenSearch documentação.

Para acessar a página de alertas nos OpenSearch painéis, você deve pelo menos estar mapeado para a função `alerting_read_access` predefinida ou receber permissões equivalentes. Essa função concede permissões para visualizar alertas, destinos e monitores, mas não para reconhecer alertas ou modificar destinos ou monitores.

Conceitos básicos dos alertas

Para criar um alerta, você configura um monitor, que é um trabalho executado em uma programação definida e consulta OpenSearch índices. Você também configura um ou mais acionadores, que definem as condições que geram os eventos. Finalmente, você configura ações, que é o que acontece depois que um alerta é acionado.

Para começar a usar alertas

1. Escolha Alertas no menu principal OpenSearch Painéis e escolha Criar monitor.
2. Crie um monitor por consulta, por bucket, por métrica de cluster ou por documento. Para obter instruções, consulte Criar um monitor.
3. Em Triggers (Acionadores), crie um ou mais acionadores. Para obter instruções, consulte Criação de acionadores.
4. Em Actions (Ações), configure um canal de notificação para o alerta. Escolha entre Slack, Amazon Chime, um webhook personalizado ou Amazon SNS. Como você pode imaginar, as notificações exigem conectividade com o canal. Por exemplo, para notificar um canal do Slack ou enviar um webhook personalizado para um servidor de terceiros, seu domínio de OpenSearch serviço precisa de conectividade de rede adequada. O webhook personalizado deve ter um endereço IP público para que um domínio OpenSearch de serviço envie alertas para ele.

> **Tip**
>
> Após uma ação enviar uma mensagem com êxito, proteger o acesso a essa mensagem (por exemplo, acesso a um canal do Slack) é sua responsabilidade. Se o seu domínio

contiver dados confidenciais, considere usar acionadores sem ações e verificar periodicamente o Dashboards em busca de alertas.

Notificações

O alerta se integra ao Notifications, que é um sistema unificado para OpenSearch notificações. O Notifications permite que você configure qual serviço de comunicação você deseja usar e veja estatísticas relevantes e informações de solução de problemas. Para obter uma documentação abrangente, consulte Notificações na OpenSearch documentação.

Seu domínio deve estar executando a OpenSearch versão 2.3 ou posterior para usar as notificações.

> **Note**
>
> OpenSearch as notificações são separadas das notificações de OpenSearch serviço, que fornecem detalhes sobre atualizações do software de serviço, aprimoramentos do Auto-Tune e outras informações importantes em nível de domínio. OpenSearch as notificações são específicas do plug-in.

Os canais de notificação substituíram os destinos de alerta a partir da OpenSearch versão 2.0. Os destinos foram oficialmente descontinuados e todas as notificações de alertas serão gerenciadas por meio de canais daqui para frente.

Quando você atualiza seus domínios para a versão 2.3 ou posterior (já que o suporte do OpenSearch Service para 2.x começa com 2.3), seus destinos existentes são migrados automaticamente para os canais de notificação. Se houver falha na migração deum destino, o monitor continuará a usá-lo até que o monitor seja migrado para um canal de notificação. Para obter mais informações, consulte Perguntas sobre destinos na OpenSearch documentação.

Para começar a usar as notificações, faça login nos OpenSearch painéis e escolha Notificações, Canais e Criar canal.

Amazon Simple Notification Service (Amazon SNS) é um tipo de canal compatível com notificações. Para autenticar usuários, você precisa fornecer ao usuário acesso total ao Amazon SNS, ou permitir que ele assuma um perfil do IAM que tenha permissões para acessar o Amazon SNS. Para obter instruções, consulte Amazon SNS como um tipo de canal.

Diferenças

Em comparação com a versão de código aberto do OpenSearch, os alertas no Amazon OpenSearch Service têm algumas diferenças notáveis.

Configurações de alertas

OpenSearch O serviço permite que você modifique as seguintes configurações de alerta:

- `plugins.scheduled_jobs.enabled`
- `plugins.alerting.alert_history_enabled`
- `plugins.alerting.alert_history_max_age`
- `plugins.alerting.alert_history_max_docs`
- `plugins.alerting.alert_history_retention_period`
- `plugins.alerting.alert_history_rollover_period`
- `plugins.alerting.filter_by_backend_roles`

Todas as outras configurações usam os valores padrão que não podem ser alterados.

Para desabilitar o alerta, envie a seguinte solicitação:

```
PUT _cluster/settings
{
  "persistent" : {
    "plugins.scheduled_jobs.enabled" : false
  }
}
```

A solicitação a seguir configura o alerta para excluir automaticamente os índices de histórico após sete dias, em vez dos 30 dias padrão:

```
PUT _cluster/settings
{
  "persistent": {
    "plugins.alerting.alert_history_retention_period": "7d"
  }
}
```

Se você criou monitores anteriormente e deseja interromper a criação de índices de alertas diários, exclua todos os índices de histórico de alertas:

```
DELETE .plugins-alerting-alert-history-*
```

Para reduzir a contagem de fragmentos para índices históricos, crie um modelo de índice. A solicitação a seguir define índices de histórico para alertas em um fragmento e uma réplica:

```
PUT _index_template/template-name
{
  "index_patterns": [".opendistro-alerting-alert-history-*"],
  "template": {
    "settings": {
      "number_of_shards": 1,
      "number_of_replicas": 1
    }
  }
}
```

Dependendo da sua tolerância à perda de dados, você pode até considerar o uso de réplicas zero. Para obter mais informações sobre como criar e gerenciar modelos de índice, consulte Modelos de índice na OpenSearch documentação.

Detecção de anomalias no Amazon Service OpenSearch

A detecção de anomalias no Amazon OpenSearch Service detecta automaticamente anomalias em seus OpenSearch dados quase em tempo real usando o algoritmo Random Cut Forest (RCF). O RCF é um algoritmo de machine learning não supervisionado que modela um esboço do fluxo de dados de entrada. O algoritmo calcula um valor de `anomaly grade` e `confidence score` para cada ponto de dados de entrada. A detecção de anomalias usa esses valores para diferenciar uma anomalia de variações normais nos dados.

Você pode emparelhar o plug-in de detecção de anomalias com o Plug-in de geração de alertas para receber uma notificação assim que for detectada uma anomalia.

A detecção de anomalias está disponível em domínios que executam qualquer OpenSearch versão do Elasticsearch 7.4 ou posterior. Todos os tipos de instâncias oferecem suporte à detecção de anomalias, exceto `t2.micro` e `t2.small`.

> **Note**
>
> Esta documentação fornece uma breve visão geral da detecção de anomalias no contexto do Amazon OpenSearch Service. Para obter uma documentação abrangente, incluindo etapas detalhadas, uma referência de API, uma referência de todas as configurações disponíveis e etapas para criar visualizações e painéis, consulte Detecção de anomalias na documentação de código aberto. OpenSearch

Pré-requisitos

A detecção de anomalias apresenta os seguintes pré-requisitos:

- A detecção de anomalias requer o Elasticsearch 7.4 OpenSearch ou posterior.
- A detecção de anomalias só oferece suporte ao controle de acesso refinado nas versões 7.9 e posteriores do Elasticsearch e em todas as versões do. OpenSearch Antes do Elasticsearch 7.9, somente usuários administradores podiam criar, visualizar e gerenciar detectores.
- Se o seu domínio usa controle de acesso refinado, os usuários não administradores devem ser mapeados para a `anomaly_read_access` função nos OpenSearch painéis para visualizar detectores ou `anomaly_full_access` para criar e gerenciar detectores.

Conceitos básicos da detecção de anomalias

Para começar, escolha Detecção de anomalias em OpenSearch painéis.

Etapa 1: Criar um detector

Um detector é uma tarefa individual de detecção de anomalias. Você pode criar vários detectores, e todos os detectores podem ser executados simultaneamente, com cada um efetuando análises de dados de diferentes fontes.

Etapa 2: Adicionar recursos ao detector

Um recurso é o campo no índice que você verifica em busca de anomalias. Um detector pode descobrir anomalias em um ou mais recursos. Você deve escolher uma das agregações a seguir para cada recurso: `average()`, `sum()`, `count()`, `min()` ou `max()`.

> **Note**
>
> O método `count()` de agregação está disponível somente no OpenSearch Elasticsearch 7.7 ou posterior. Para o Elasticsearch 7.4, use uma expressão personalizada como a seguinte:

```
{
  "aggregation_name": {
     "value_count": {
        "field": "field_name"
     }
  }
}
```

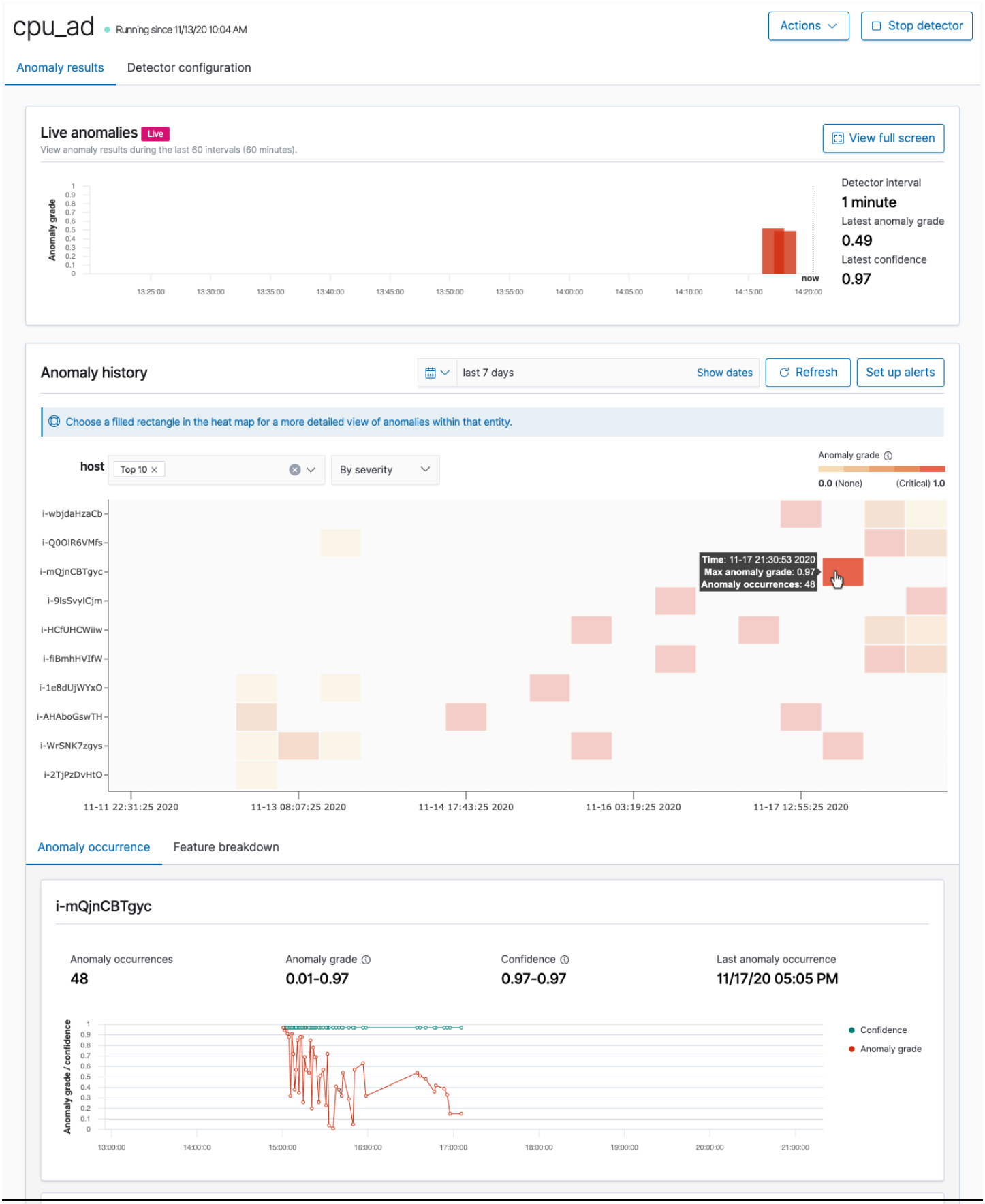
O método de agregação determina o que constitui uma anomalia. Por exemplo, se você escolher `min()`, o detector se concentrará em encontrar anomalias com base nos valores mínimos de seu recurso. Se você escolher `average()`, o detector encontrará anomalias com base nos valores médios de seu recurso. Você pode adicionar um máximo de cinco recursos por detector.

Você pode definir as seguintes configurações opcionais (disponíveis no Elasticsearch 7.7 e posterior):

- Category (Categoria): categorize ou corte seus dados com uma dimensão como endereço IP, ID do produto, código do país e assim por diante.
- Window size (Tamanho da janela): defina o número de intervalos de agregação do fluxo de dados a considerar em uma janela de detecção.

Depois de configurar seus recursos, visualize anomalias de amostra e ajuste as configurações do recurso, se necessário.

Etapa 3: Observar os resultados



- Live anomalies (Anomalias em tempo real): exibe os resultados das anomalias em tempo real dos últimos 60 intervalos. Por exemplo, se o intervalo for definido como 10, ele mostra os resultados dos últimos 600 minutos. Esse gráfico é atualizado a cada 30 segundos.
- Anomaly history (Histórico da anomalia): plota o grau da anomalia com a medida de confiança correspondente.
- Feature breakdown (Detalhamento de recurso): plota os recursos com base no método de agregação. É possível variar o intervalo de data e hora do detector.
- Anomaly occurrence (Ocorrência das anomalias) mostra os valores de `Start time`, `End time`, `Data confidence` e `Anomaly grade` para cada anomalia detectada.

  Se você definir o campo de categoria, verá um gráfico de Mapa de calor adicional que correlaciona resultados para entidades anômalas. Escolha um retângulo preenchido para obter uma visualização mais detalhada da anomalia.

Etapa 4: Configurar alertas

Para criar um monitor para enviar notificações quando qualquer anomalia for detectada, escolha Configurar alertas. O plug-in redireciona você para a página Adicionar monitor, na qual você pode configurar um alerta.

Tutorial: Detectar uso elevado da CPU com detecção de anomalias

Este tutorial demonstra como criar um detector de anomalias no Amazon OpenSearch Service para detectar o alto uso da CPU. Você usará OpenSearch painéis para configurar um detector para monitorar o uso da CPU e gerar um alerta quando o uso da CPU ultrapassar um limite especificado.

> **Note**
>
> Essas etapas se aplicam à versão mais recente do OpenSearch e podem ser um pouco diferentes das versões anteriores.

Pré-requisitos

- Você deve ter um domínio OpenSearch de serviço executando o Elasticsearch 7.4 ou posterior, ou qualquer versão. OpenSearch

- Também é necessário estar ingerindo arquivos de log de aplicação em seu cluster que contêm dados de uso da CPU.

Etapa 1: Criar um detector

Primeiro, crie um detector que identifique anomalias nos dados de uso da CPU.

1. Abra o menu do painel esquerdo em OpenSearch Painéis e escolha Detecção de anomalias e, em seguida, escolha Criar detector.
2. Nomeie o detector como **`high-cpu-usage`**.
3. Para sua fonte de dados, escolha o índice que contém os arquivos de log de uso da CPU em que deseja identificar anomalias.
4. Selecione o Timestamp field (Campo de identificação de data/hora) dos dados. Opcionalmente, é possível adicionar um filtro de dados. Esse filtro de dados analisa apenas um subconjunto da fonte de dados e reduz o ruído dos dados que não são relevantes.
5. Defina o Detector interval (Intervalo do detector) como 2 minutos. Esse intervalo define o tempo (por intervalo de minutos) para o detector coletar os dados.
6. Em Window delay (Atraso da janela), adicione um atraso de 1 minuto. Esse atraso adiciona tempo de processamento extra para garantir que todos os dados dentro da janela estejam presentes.
7. Escolha Próximo. No painel de detecção de anomalias, embaixo do nome do detector, escolha Configure model (Configurar modelo).
8. Em Feature name (Nome do recurso), insira **`max_cpu_usage`**. Em Feature state (Estado do recurso), selecione Enable feature (Habilitar recurso).
9. Em Find anomalies based on (Encontrar anomalias com base em), escolha Field value (Valor do campo).
10. Em Aggregation method (Método de agregação), escolha **`max()`**.
11. Em Field (Campo), selecione o campo nos dados que será verificado em busca de anomalias. Por exemplo, ele pode ser chamado de `cpu_usage_percentage`.
12. Mantenha todas as outras configurações em seus valores padrão e escolha Next (Próximo).
13. Ignore a configuração de trabalhos do detector e escolha Next (Próximo).
14. Na janela pop-up, escolha quando iniciar o detector (automática ou manualmente) e escolha Confirm (Confirmar).

Agora que o detector está configurado, depois que ele inicializar, você poderá ver os resultados em tempo real de uso da CPU na seção Real-time results (Resultados em tempo real) do painel do detector. A seção Live anomalies (Anomalias ao vivo) exibe todas as anomalias que ocorrem à medida que os dados são ingeridos em tempo real.

Etapa 2: Configurar um alerta

Agora que você criou um detector, crie um monitor que invoque um alerta para enviar uma mensagem ao Slack quando ele detectar uso da CPU que atenda às condições especificadas nas configurações do detector. Você receberá notificações do Slack quando os dados de um ou mais índices atenderem às condições que invocam o alerta.

1. Abra o menu do painel esquerdo em OpenSearch Painéis e escolha Alertas e, em seguida, escolha Criar monitor.
2. Informe um nome para o monitor.
3. Em Monitor type (Tipo de monitor), escolha Per-query monitor (Monitor por consulta). Um monitor por consulta executa uma consulta especificada e define os acionadores.
4. Em Monitor defining method (Método de definição do monitor), escolha Anomaly detector (Detector de anomalias) e, em seguida, selecione no menu suspenso Detector o detector criado na seção anterior.
5. Para Schedule (Programação), escolha a frequência com que o monitor coleta dados e a frequência com que você recebe alertas. Para este tutorial, defina a programação para executar a cada 7minutos.
6. Na seção Triggers (Acionadores), escolha Add trigger (Adicionar acionador). Em Trigger name (Nome do acionador), insira **`High CPU usage`**. Para fins deste tutorial, em Severity level (Nível de severidade), escolha1, o nível mais elevado de severidade.
7. Em Anomaly grade threshold (Limite de grau da anomalia), escolha IS ABOVE (ESTÁ ACIMA). No menu embaixo dessa opção, escolha o limite de grau a ser aplicado. Para este tutorial, defina Anomaly grade (Grau da anomalia) como 0,7.
8. Em Anomaly confidence threshold (Limite de confiança da anomalia), escolha IS ABOVE (ESTÁ ACIMA). No menu embaixo dessa opção, escolha o mesmo número que o grau da anomalia. Para este tutorial, defina Anomaly confidence threshold (Limite de confiança da anomalia) como 0,7.
9. Na seção Actions (Ações), escolha Destination (Destino). No campo Name (Nome), escolha o nome do destino. No menu Type (Tipo), escolha Slack. No campo Webhook URL (URL do webhook), insira um URL de webhook para receber alertas. Para saber mais, consulte Sending messages using incoming webhooks (Enviar mensagens usando webhooks recebidos).

10Escolha Criar.

Recursos relacionados

- the section called “Geração de alertas”

- the section called “Detecção de anomalias”
- Anomaly detection API (API de detecção de anomalias)

Assistente de IA para Amazon OpenSearch Service

O AI Assistant é um assistente generativo baseado em IA que ajuda desenvolvedores e profissionais de TI a realizar uma variedade de tarefas em todo o ciclo de vida do desenvolvimento de software. O AI Assistant ajuda AWS os clientes a codificar, testar, implantar, solucionar problemas e otimizar seus aplicativos em AWS execução.

A integração do AI Assistant com o Amazon OpenSearch Service oferece os seguintes recursos generativos:

- the section called “Gerar visualizações usando linguagem natural”
- the section called “Visualizar resumos e insights de alertas”
- the section called “Veja os resumos dos resultados da consulta gerados pelo AI Assistant na página Discover”
- the section called “Visualizar os detectores de anomalias recomendados”
- the section called “Acesse o chat do AI Assistant para perguntas sobre OpenSearch serviços”

Para acessar os recursos do AI Assistant no OpenSearch Service que são relevantes para sua tarefa, procure o seguinte ícone contextual nas páginas de visualização Alertas, Descobrir e Criar. Clique no ícone para solicitar assistência e usar os atributos descritos nesta seção.

Você também pode conversar diretamente com o AI Assistant clicando no ícone no canto superior direito do console de OpenSearch serviço. O chat do AI Assistant oferece suporte a respostas a perguntas sobre recursos e funcionalidades do OpenSearch serviço.

Suportado Regiões da AWS

O suporte do AI Assistant para OpenSearch serviços está disponível no seguinte Regiões da AWS:

- Leste dos EUA (N. da Virgínia)
- Oeste dos EUA (Oregon)
- Ásia-Pacífico (Mumbai)
- Ásia-Pacífico (Sydney)
- Ásia-Pacífico (Tóquio)
- Canadá (Central)

- Europa (Frankfurt)
- Europa (Londres)
- Europa (Paris)
- América do Sul (São Paulo)

Configurando o AI Assistant for OpenSearch Service

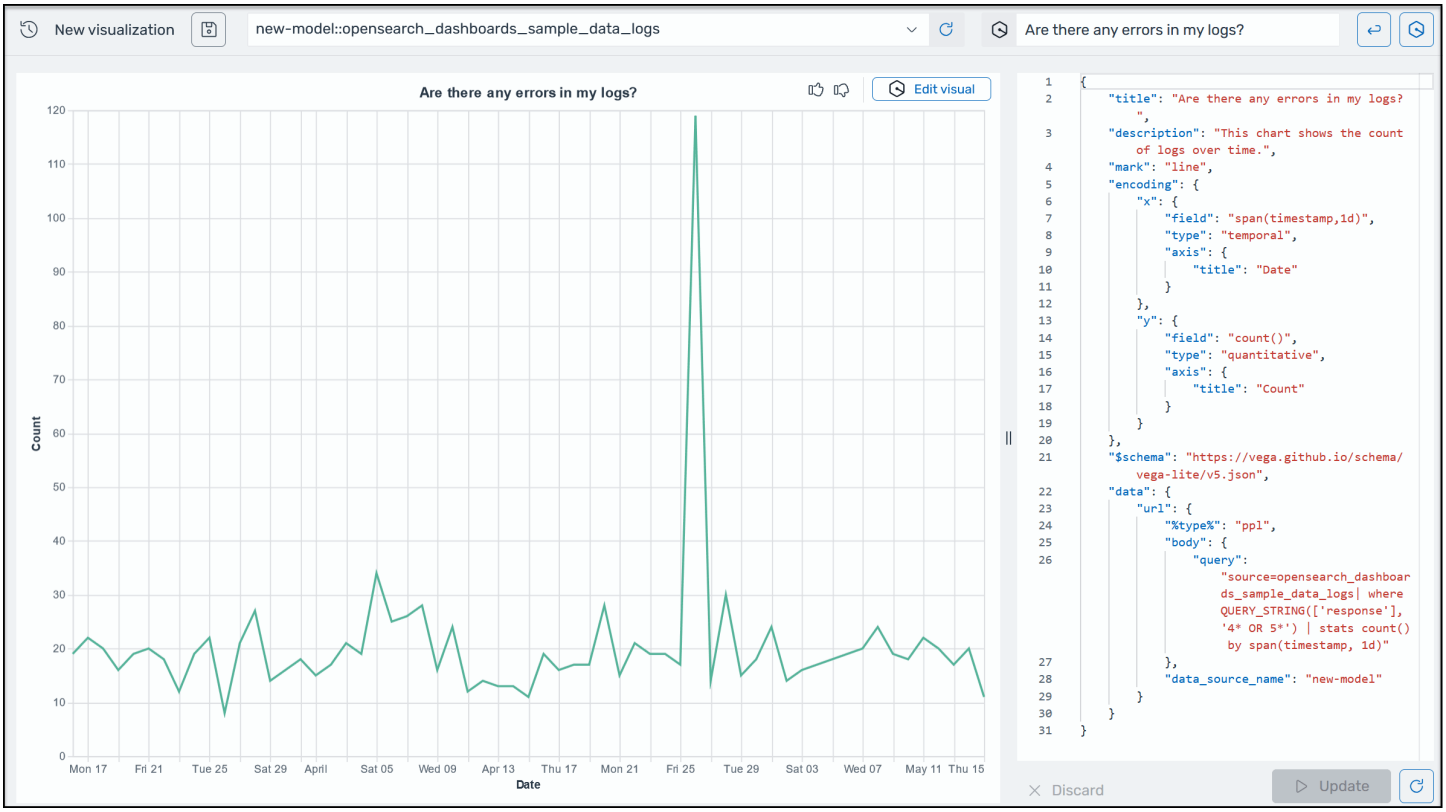
Conclua as etapas a seguir para configurar o AI Assistant for OpenSearch Service.

1. Verifique se você configurou o controle OpenSearch de acesso refinado do Serviço. Para obter mais informações, consulte Controle de acesso refinado no Amazon Service OpenSearch.
2. Verifique se sua fonte de dados está na OpenSearch versão 2.17 ou posterior.
3. Verifique se você marcou a caixa de seleção Habilitar geração de consultas em linguagem natural na seção Inteligência artificial (AI) e machine learning (ML) durante a criação do domínio ou ao editar a configuração de cluster.

> **Note**
>
> Os OpenSearch recursos do Amazon Q for Developer estão disponíveis com o nível gratuito Q. Para saber mais, consulte Como entender as camadas de serviço no Guia do usuário do Amazon Q User.

Gerar visualizações usando linguagem natural

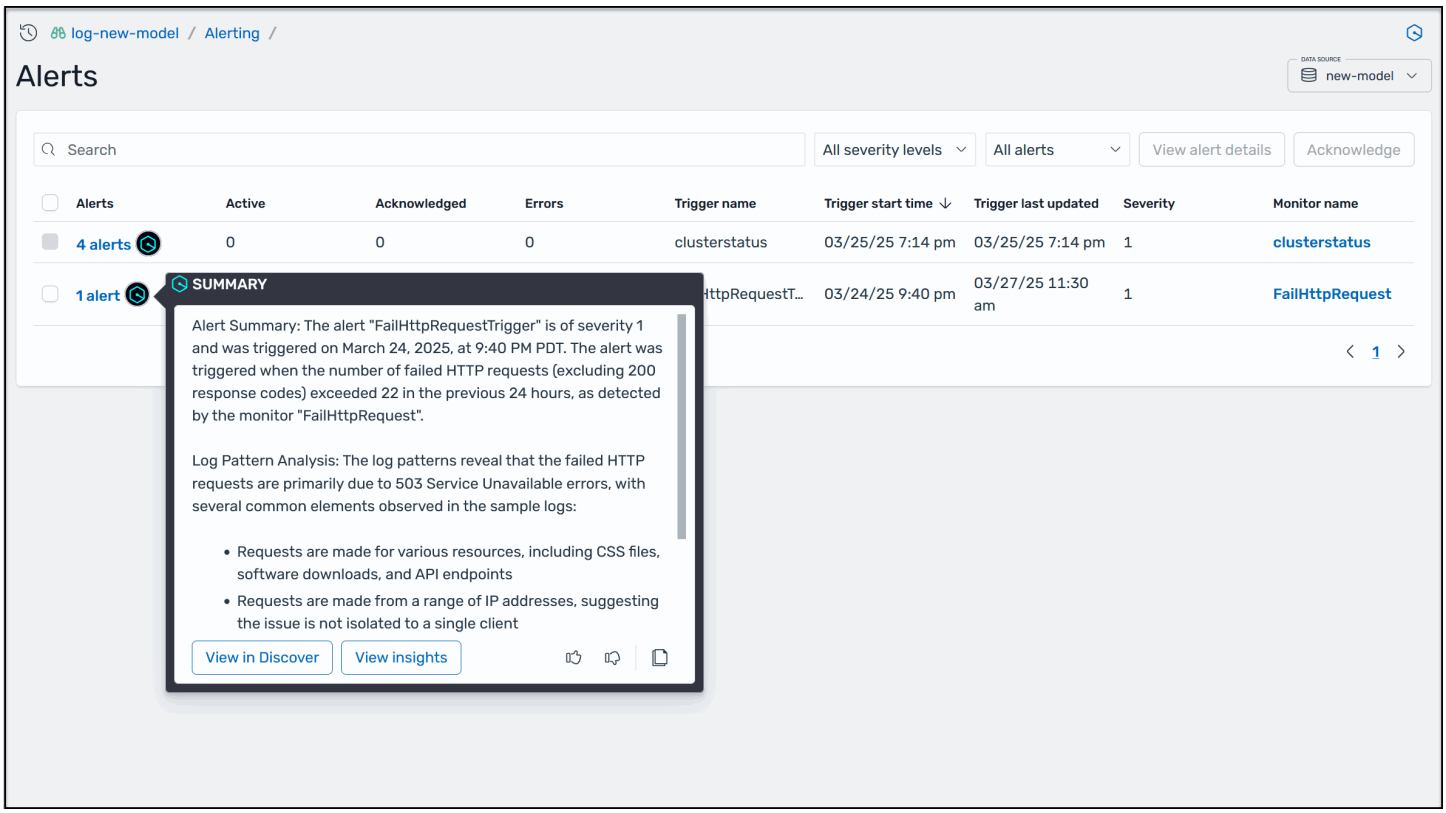
Para ajudá-lo a obter mais informações sobre seus dados operacionais, o AI Assistant for OpenSearch Service oferece suporte ao uso de instruções em linguagem natural para criar visualizações. Antes de começar, verifique se você configurou o AI Assistant for OpenSearch Service. Depois de confirmado, inicie seu aplicativo OpenSearch Service UI, navegue até a página de visualizações, selecione o botão + Criar visualização e escolha a linguagem natural. Agora você pode especificar uma fonte de dados e inserir um requisito em linguagem natural para gerar as visualizações. Aqui está um exemplo.



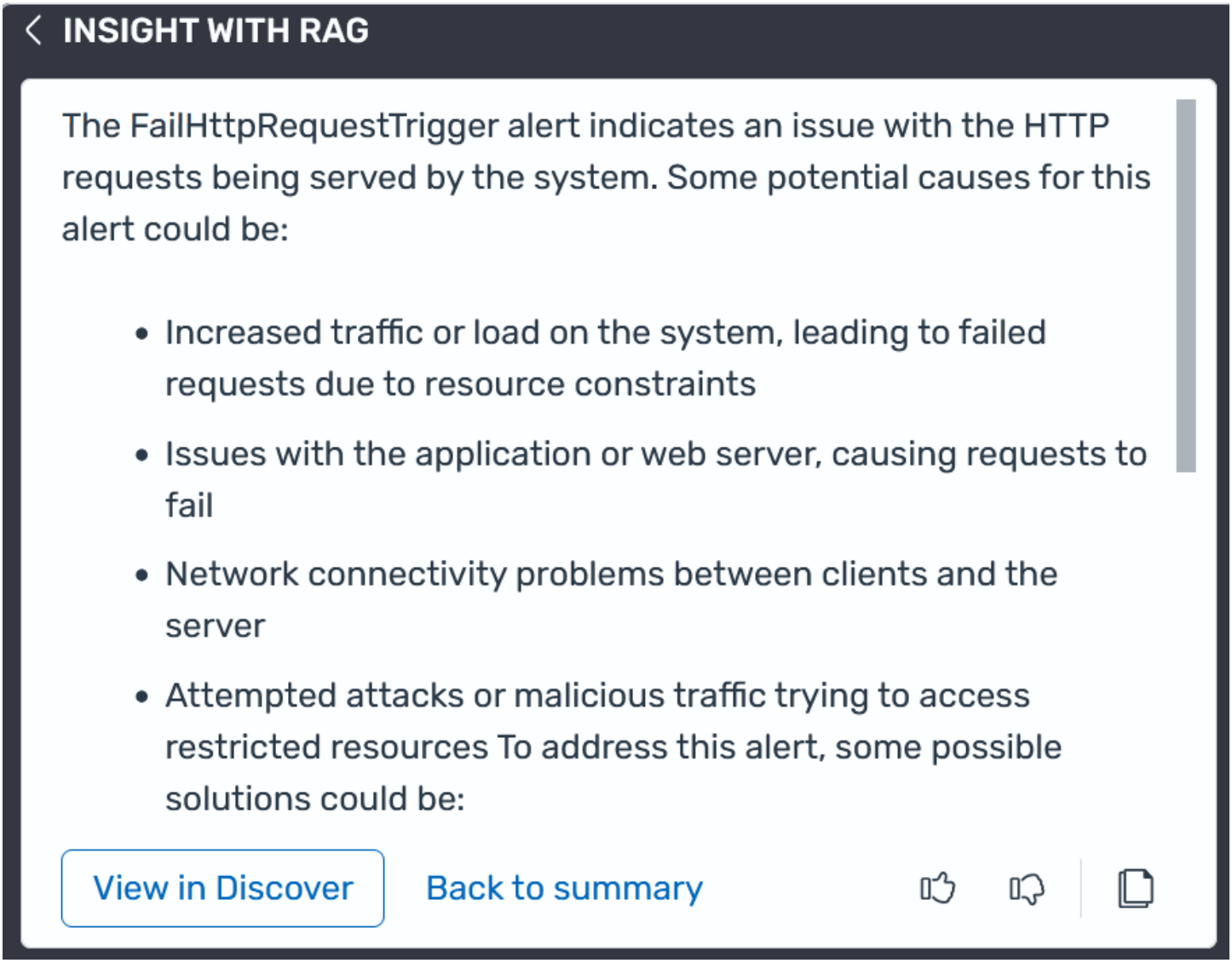
As visualizações podem agilizar a solução de problemas dividindo a análise de erros de acordo com diferentes dimensões. As visualizações também podem ajudar a identificar padrões e tendências, acelerar a tomada de decisões, mostrar relacionamentos e simplificar dados complexos. Como alternativa, você também pode criar visualizações na página Descobrir como parte dos resultados da consulta.

Visualizar resumos e insights de alertas

Você pode configurar o OpenSearch Serviço para criar um monitor de alerta quando os dados de um ou mais índices atenderem a determinadas condições. Para ajudá-lo a entender e solucionar rapidamente um alerta, você pode ver um resumo do alerta clicando no ícone do AI Assistant ao lado de um alerta. Um resumo fornece detalhes sobre o problema subjacente que acionou o alerta e, quando disponíveis, análises adicionais para ajudar a localizar a causa raiz do problema. A captura de tela a seguir mostra um exemplo de um resumo de alertas criado pelo AI Assistant.



Se você conectar uma base de conhecimento para fornecer contexto adicional sobre seu ambiente, conforme descrito posteriormente neste tópico, o AI Assistant cria insights sobre um alerta. Os insights fornecem detalhes e opções de solução de problemas para ajudar a corrigir a causa raiz de um alerta. Para o alerta mostrado anteriormente, o AI Assistant também produziu os seguintes insights.



> **Note**
>
> Dependendo da natureza do alerta e das informações disponíveis, o AI Assistant pode oferecer a opção de visualizar os dados do alerta na página Discover nos OpenSearch painéis. Se você ver o botão Exibir no Discover na parte inferior de um resumo de alertas do AI Assistant, clique no botão para abrir o conjunto de dados correspondente no Discover com um filtro ativo para os dados do alerta.

Tópicos

- Antes de começar
- Visualizar resumos e insights de alertas

Antes de começar

Conclua as etapas a seguir para configurar uma base de conhecimento do Amazon Bedrock para que o AI Assistant possa criar insights para alertas OpenSearch de serviço.

Etapa 1: criar a LambdaInvokeOpenSearch MLCommons função Role IAM

Crie uma nova função chamada `LambdaInvokeOpenSearchMLCommonsRole` in AWS Identity and Access Management (IAM). OpenSearch O serviço usa essa função para criar um conector de IA OpenSearch que ajuda a produzir insights com base em artigos configurados da base de conhecimento. Você deve mapear essa função para a `ml_full_access` função OpenSearch Serviço, conforme descrito na etapa 2.

Quando você criar o novo perfil, em Tipo de entidade confiável, escolha Conta da AWS . Não é necessário especificar uma política de permissão. Na página Adicionar permissões, escolha Próximo. Para obter mais informações sobre a criação de uma nova função, consulte Criação de uma função para um AWS serviço (console).

Etapa 2: mapear a LambdaInvokeOpenSearch MLCommons função da função para a função OpenSearch Service ml_full_access

Use o procedimento a seguir para mapear a `LambdaInvokeOpenSearchMLCommonsRole` função para a `ml_full_access` função OpenSearch Serviço. Esse mapeamento também ajuda o OpenSearch Service a criar o conector AI.

Para mapear a função do IAM necessária para a função OpenSearch Service ml_full_access

1. Abra a página de administração de dados do OpenSearch Service Dashboard.
2. Em Acesso a dados e usuário, escolha Perfis.
3. Use a caixa de pesquisa para encontrar o perfil `ml_full_access`.
4. Na página ml_full_access, escolha a guia Usuários mapeados.
5. Escolha Mapear usuários.
6. No campo Perfis de backend, cole o nome do recurso da Amazon (ARN) do perfil `LambdaInvokeOpenSearchMLCommonsRole` e escolha Mapear.

Etapa 3: configurar uma base OpenSearch de conhecimento do serviço usando CloudFormation

Use o procedimento a seguir para configurar uma base OpenSearch de conhecimento de serviço usando AWS CloudFormation para que o AI Assistant possa gerar insights.

Para configurar uma base de conhecimentos para insights

1. Faça login na https://console.aws.amazon.com/aos/página inicial do console do Amazon OpenSearch Service em um local compatível Região da AWS. Para obter mais informações, consulte Suportado Regiões da AWS.
2. No painel de navegação, selecione Integrações.
3. Na seção Modelos de integração, escolha o modelo Integrar com a base de conhecimentos por meio do Amazon Bedrock. Se não vir esse modelo, certifique-se de que você está em uma região compatível.
4. No quadro Integrar com a base de conhecimento por meio do Amazon Bedrock, escolha Configurar domínio e, em seguida, escolha uma das opções disponíveis. OpenSearch O serviço abre o modelo de CloudFormation pilha com os campos obrigatórios pré-preenchidos. A CloudFormation pilha oferece suporte à integração de domínios públicos e VPC.
5. Selecione Criar pilha. Depois de CloudFormation criar os recursos, o serviço exibe o agente Amazon Bedrock AgentIdConnectorId, e. ModelId

Quando aplicável, o AI Assistant agora cria insights para alertas OpenSearch de serviço.

Visualizar resumos e insights de alertas

Use o procedimento a seguir para visualizar resumos de alertas e insights no OpenSearch Service.

Visualizar resumos e insights de alertas

1. Verifique se você configurou o AI Assistant for OpenSearch Service.
2. Verifique se você configurou alertas para o OpenSearch Serviço.
3. No menu principal OpenSearch Painéis, escolha Alertas e, em seguida, escolha Alertas.
4. Escolha o ícone do AI Assistant ao lado de um alerta. O AI Assistant pode levar até 10 segundos para gerar o resumo.
5. Se estiver presente no resumo do alerta, escolha Visualizar insights para ver mais detalhes sobre o alerta segundo a base de conhecimentos configurada.

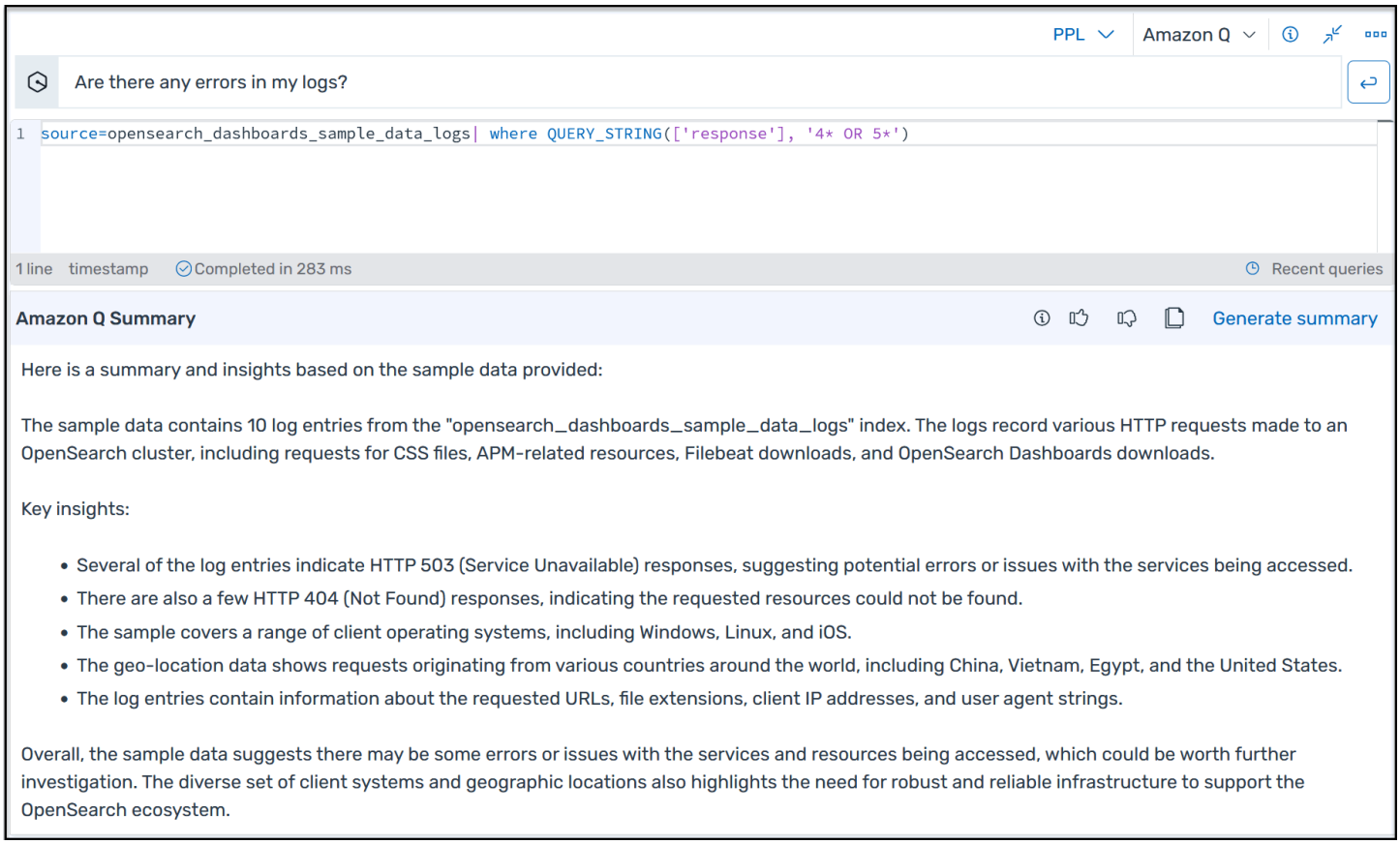
6. Se estiver presente no resumo do alerta, escolha Exibir no Discover para visualizar os dados do alerta na página Discover nos OpenSearch painéis.

Veja os resumos dos resultados da consulta gerados pelo AI Assistant na página Discover

OpenSearch O serviço permite que você consulte seus dados com solicitações em linguagem natural usando a linguagem de consulta Piped Processing Language (PPL) na página Discover. Por exemplo, você pode escrever consultas como as seguintes:

- Há algum erro nos meus logs de erros?
- Qual é o tamanho médio de solicitação por semana?
- Quantas solicitações foram agrupadas por código de resposta a semana passada?

Em resposta, o AI Assistant gera resumos em linguagem natural dos resultados da sua consulta com base nos primeiros dez registros, como os seguintes:



A combinação da geração de consultas em linguagem natural com os resumos de consultas pode adicionar um nível de consulta ao solucionar um alerta ou um meio fácil de entender os dados sem precisar escrever consultas complexas.

Para ver os resumos dos resultados da consulta gerados pelo AI Assistant na página Discover

1. Verifique se você configurou o AI Assistant for OpenSearch Service.
2. No menu principal dos OpenSearch Painéis, escolha Descobrir.
3. Na lista suspensa de linguagens de consulta, escolha PPL.
4. Na caixa de texto do AI Assistant, insira uma solicitação e clique no botão ao lado da caixa de texto para executar a consulta. O AI Assistant pode levar até 10 segundos para retornar um resumo. Após a consulta inicial, você deve escolher Gerar resumo para os resumos subsequentes.

> **Note**
>
> Você pode desativar a geração de resumos na lista suspensa do AI Assistant no Discover.

Visualizar os detectores de anomalias recomendados

A detecção de anomalias no Amazon OpenSearch Service detecta automaticamente anomalias em seus OpenSearch dados quase em tempo real usando o algoritmo Random Cut Forest (RCF). O RCF é um algoritmo de machine learning não supervisionado que modela um esboço do fluxo de dados de entrada. O algoritmo calcula um valor de `anomaly grade` e `confidence score` para cada ponto de dados de entrada. A detecção de anomalias usa esses valores para diferenciar uma anomalia de variações normais nos dados.

Para simplificar o processo de criação de detectores de anomalias, o AI Assistant pode gerar detectores sugeridos com base na fonte de dados selecionada na página Descobrir. O AI Assistant oferece suporte a detectores de anomalias sugeridos para qualquer idioma.

Para ver os detectores de anomalias recomendados pelo AI Assistant

1. Verifique se você configurou o AI Assistant for OpenSearch Service.
2. No menu principal da OpenSearch interface do usuário, escolha a página Descobrir e, em seguida, escolha uma fonte de dados.

3. No menu AI Assistant, escolha Sugerir detector de anomalias, conforme mostrado na captura de tela a seguir.

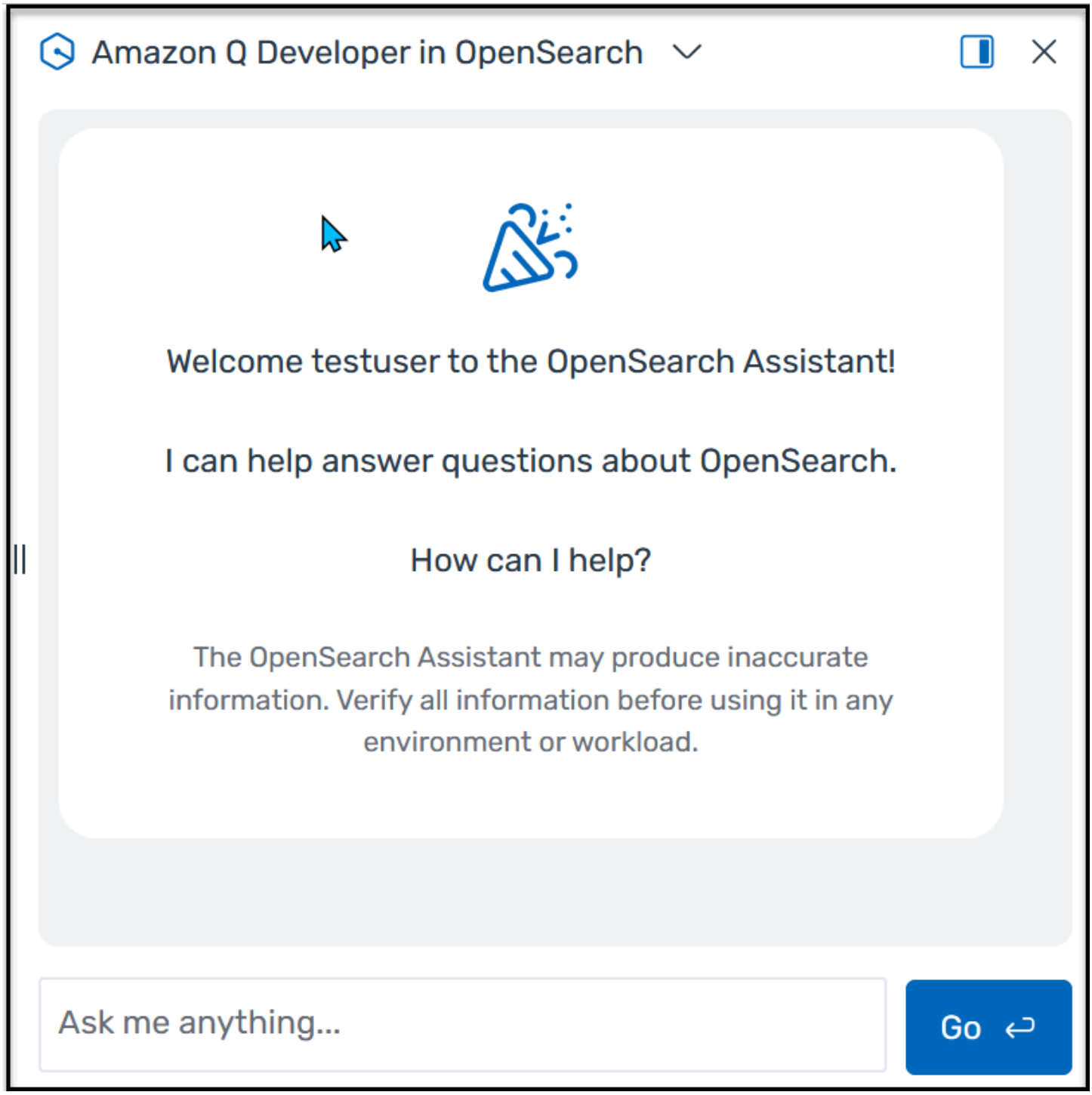


   O AI Assistant pode levar alguns segundos para gerar os recursos do detector.

4. Escolha Criar detector.

Acesse o bate-papo do AI Assistant para perguntas OpenSearch relacionadas

Se você tiver dúvidas sobre o Amazon OpenSearch Service, incluindo questões conceituais ou processuais relacionadas aos recursos ou funcionalidades do OpenSearch serviço, peça informações ao AI Assistant. O chatbot é compatível com um chat conversacional que preserva o contexto da conversa. Ele também salva um histórico das conversas para referência posterior. Para acessar o chatbot, clique no ícone do AI Assistant no canto superior direito de qualquer página de OpenSearch serviço. Digite uma pergunta na caixa de texto e clique em Ir. O AI Assistant pode levar até 30 segundos para gerar uma resposta.



Aqui está um exemplo de resposta do AI Assistant.

Can you suggest some node management best practices?

Based on your context of being in the OpenSearch Dashboard in the us-west-2 region, here are some node management best practices for Amazon OpenSearch Service: [1]

1. Implement Dedicated Manager Nodes:
    - Use dedicated manager nodes to improve cluster stability. [2]
    - Configure three dedicated manager nodes for optimal domain stability across three Availability Zones.
    - This offloads cluster management tasks from data nodes, enhancing overall performance.
2. Multi-AZ Deployment:
    - Deploy your domain across multiple Availability Zones for increased availability and resilience.
    - Consider using the Multi-AZ with Standby option for 99.99% availability and consistent performance.
3. Proper Sizing:

Aprendizado de máquina para Amazon OpenSearch Service

O ML Commons é um OpenSearch plug-in que fornece um conjunto de algoritmos comuns de aprendizado de máquina (ML) por meio de chamadas de transporte e da API REST. Essas chamadas escolhem os nós e os recursos certos para cada solicitação de ML e monitoram as tarefas de ML para garantir o tempo de atividade. Isso permite que você aproveite os algoritmos de ML de código aberto existentes e reduza o esforço necessário para desenvolver novos atributos de ML. Para saber mais sobre o plug-in, consulte Aprendizado de máquina na OpenSearch documentação. Este capítulo aborda como usar o plug-in com o Amazon OpenSearch Service.

Tópicos

- Conectores Amazon OpenSearch Service ML para Serviços da AWS
- Conectores Amazon OpenSearch Service ML para plataformas de terceiros
- Usando CloudFormation para configurar a inferência remota para pesquisa semântica
- Configurações do ML Commons não compatíveis
- OpenSearch Modelos de estrutura de fluxo de serviço

Conectores Amazon OpenSearch Service ML para Serviços da AWS

Ao usar conectores de aprendizado de máquina (ML) do Amazon OpenSearch Service com outro AWS service (Serviço da AWS), você precisa configurar uma função do IAM para conectar o serviço com segurança a esse OpenSearch serviço. Serviços da AWS que você pode configurar um conector para incluir o Amazon SageMaker AI e o Amazon Bedrock. Neste tutorial, abordamos como criar um conector do OpenSearch Service ao SageMaker Runtime. Para saber mais sobre conectores, consulte Conectores compatíveis.

Tópicos

- Pré-requisitos
- Crie um conector OpenSearch de serviço

Pré-requisitos

Para criar um conector, você deve ter um endpoint do Amazon SageMaker AI Domain e uma função do IAM que conceda acesso ao OpenSearch serviço.

Configurar um domínio Amazon SageMaker AI

Consulte Implantar um modelo na Amazon SageMaker AI no Amazon SageMaker AI Developer Guide para implantar seu modelo de aprendizado de máquina. Observe o URL do endpoint do seu modelo, que você precisa para criar um conector de IA.

Criar um perfil do IAM

Configure uma função do IAM para delegar permissões SageMaker de tempo de execução ao OpenSearch serviço. Para criar um novo perfil, consulte Como criar um perfil do IAM (console) no Guia do usuário do IAM. Opcionalmente, você pode usar um perfil existente, desde que tenha o mesmo conjunto de privilégios. Se você criar uma nova função em vez de usar uma função AWS gerenciada, substitua `opensearch-sagemaker-role` neste tutorial pelo nome da sua própria função.

1. Anexe a seguinte política gerenciada de IAM à sua nova função para permitir que o OpenSearch Serviço acesse seu endpoint de SageMaker IA. Para anexar uma política ao perfil, consulte Como adicionar permissões de identidade do IAM.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "sagemaker:InvokeEndpointAsync",
                "sagemaker:InvokeEndpoint"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

2. Siga as instruções em Modificação da política de confiança de um perfil para editar a relação de confiança do perfil. Na política a seguir, *service-principal* substitua por um dos seguintes princípios de serviço para OpenSearch Service ou OpenSearch Serverless:

   Para OpenSearch serviço

   ```
   opensearchservice.amazonaws.com
   ```

   Para OpenSearch servidores sem servidor

   ```
   ml.opensearchservice.amazonaws.com
   ```

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Action": [
                   "sts:AssumeRole"
               ],
               "Effect": "Allow",
               "Principal": {
                   "Service": [
                       "ml.opensearchservice.amazonaws.com"
                   ]
               }
           }
       ]
   }
   ```

   Recomendamos que você use as chaves de condição `aws:SourceAccount` e `aws:SourceArn` para limitar o acesso a um domínio específico. `SourceAccount`É o Conta da AWS ID que pertence ao proprietário do domínio e `SourceArn` o ARN do domínio. Por exemplo, você pode adicionar o bloco de condições a seguir na política de confiança:

   ```
   "Condition": {
       "StringEquals": {
           "aws:SourceAccount": "account-id"
       },
       "ArnLike": {
           "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
       }
   ```

```
}
```

Configurar permissões do

Para criar o conector, você precisa de permissão para passar a função do IAM para o OpenSearch Serviço. Você também precisa de acesso à ação `es:ESHttpPost`. Para conceder ambas as permissões, anexe a seguinte política ao perfil do IAM cujas credenciais estão sendo usadas para assinar a solicitação:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/opensearch-sagemaker-
role"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttpPost",
            "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
        }
    ]
}
```

Se seu usuário ou função não tiver permissões `iam:PassRole` para passar sua função, talvez você encontre o seguinte erro de autorização ao tentar registrar um repositório na próxima etapa.

Mapeie a função de ML em OpenSearch painéis (se estiver usando controle de acesso refinado)

O controle minucioso de acesso introduz uma etapa adicional ao configurar um conector. Mesmo que você use a autenticação básica HTTP para todos os outros fins, será necessário mapear o perfil `ml_full_access` para o seu perfil do IAM que tem permissões `iam:PassRole` para passar `opensearch-sagemaker-role`.

1. Navegue até o plug-in OpenSearch Dashboards do seu domínio OpenSearch de serviço. Você pode encontrar o endpoint do Dashboards no painel do seu domínio no console de OpenSearch serviços.
2. No menu principal, escolha Segurança, Funções e selecione a função ml_full_access.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Em Funções de backend, adicione o ARN da função que tem permissão para aprovar `opensearch-sagemaker-role`.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Crie um conector OpenSearch de serviço

Para criar um conector, envie uma `POST` solicitação para o endpoint do domínio OpenSearch Service. Você pode usar curl, o cliente Python de amostra, o Postman ou outro método para enviar uma solicitação assinada. Você não pode usar uma solicitação `POST` no console do Kibana. A solicitação assume o seguinte formato:

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
```

```
        "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
        "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
}
```

Se o domínio residir em uma nuvem privada virtual (VPC), o computador deverá estar conectado à VPC para que a solicitação crie o conector de IA com êxito. O acesso a uma VPC varia de acordo com a configuração de rede, mas geralmente requer uma conexão com VPN ou rede corporativa. Para verificar se você pode acessar seu domínio OpenSearch de serviço, navegue até `https://your-vpc-domain.region.es.amazonaws.com` em um navegador da Web e verifique se você recebeu a resposta JSON padrão.

Exemplo de cliente do Python

O cliente Python é mais simples de automatizar que uma solicitação HTTP, além de ser mais fácil reutilizá-lo. Para criar o conector AI com o cliente Python, salve o código de exemplo a seguir em um arquivo Python. O cliente requer os pacotes AWS SDK para Python (Boto3), `requests` e `requests-aws4auth`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository
path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
```

```
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
         "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
         "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
}
headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Conectores Amazon OpenSearch Service ML para plataformas de terceiros

Neste tutorial, abordamos como criar um conector do OpenSearch Service ao Cohere. Para saber mais sobre conectores, consulte Conectores compatíveis.

Ao usar um conector de aprendizado de máquina (ML) do Amazon OpenSearch Service com um modelo remoto externo, você precisa armazenar suas credenciais de autorização específicas em AWS Secrets Manager. Isso pode ser uma chave de API ou uma combinação de nome de usuário e senha. Isso significa que você também precisa criar uma função do IAM que permita que o OpenSearch serviço acesse a leitura do Secrets Manager.

Tópicos

- Pré-requisitos

- Crie um conector OpenSearch de serviço

Pré-requisitos

Para criar um conector para o Cohere ou qualquer provedor externo com o OpenSearch Service, você deve ter uma função do IAM que conceda acesso ao OpenSearch Service AWS Secrets Manager, onde você armazena suas credenciais. Você também deve armazenar suas credenciais no Secrets Manager.

Criar um perfil do IAM

Configure uma função do IAM para delegar permissões do Secrets Manager ao OpenSearch Service. Você também pode usar a função `SecretManagerReadWrite` existente. Para criar um novo perfil, consulte Como criar um perfil do IAM (console) no Guia do usuário do IAM. Se você criar uma nova função em vez de usar uma função AWS gerenciada, substitua `opensearch-secretmanager-role` neste tutorial pelo nome da sua própria função.

1. Anexe a seguinte política gerenciada do IAM à sua nova função para permitir que o OpenSearch Service acesse seus valores do Secrets Manager. Para anexar uma política ao perfil, consulte Como adicionar permissões de identidade do IAM.

   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Action": [
                   "secretsmanager:GetSecretValue"
               ],
               "Effect": "Allow",
               "Resource": "*"
           }
       ]
   }
   ```

2. Siga as instruções em Modificação da política de confiança de um perfil para editar a relação de confiança do perfil. Na política a seguir, *`service-principal`* substitua por um dos seguintes princípios de serviço para OpenSearch Service ou OpenSearch Serverless:

Para OpenSearch serviço

```
opensearchservice.amazonaws.com
```

Para OpenSearch servidores sem servidor

```
ml.opensearchservice.amazonaws.com
```

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "service-principle"
                ]
            }
        }
    ]
}
```

Recomendamos que você use as chaves de condição `aws:SourceAccount` e `aws:SourceArn` para limitar o acesso a um domínio específico. `SourceAccount`É o Conta da AWS ID que pertence ao proprietário do domínio e `SourceArn` o ARN do domínio. Por exemplo, você pode adicionar o bloco de condições a seguir na política de confiança:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

Configurar permissões do

Para criar o conector, você precisa de permissão para passar a função do IAM para o OpenSearch Serviço. Você também precisa de acesso à ação `es:ESHttpPost`. Para conceder ambas as permissões, anexe a seguinte política ao perfil do IAM cujas credenciais estão sendo usadas para assinar a solicitação:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::111122223333:role/opensearch-secretmanager-role"
    },
    {
      "Effect": "Allow",
      "Action": "es:ESHttpPost",
      "Resource": "arn:aws:es:us-east-1:111122223333:domain/domain-name/*"
    }
  ]
}
```

Se seu usuário ou função não tiver permissões `iam:PassRole` para passar sua função, talvez você encontre o seguinte erro de autorização ao tentar registrar um repositório na próxima etapa.

Configurar AWS Secrets Manager

Para armazenar suas credenciais de autorização no Secrets Manager, consulte Como criar um segredo da AWS Secrets Manager no Guia do usuário da AWS Secrets Manager .

Depois que o Secrets Manager aceitar seu par de valores-chave como segredo, você recebe um ARN com o formato: `arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret-a1b2c3` Mantenha um registro desse ARN, conforme você o usa, e da sua chave ao criar um conector na próxima etapa.

Mapeie a função de ML em OpenSearch painéis (se estiver usando controle de acesso refinado)

O controle minucioso de acesso introduz uma etapa adicional ao configurar um conector. Mesmo que você use a autenticação básica HTTP para todos os outros fins, será necessário mapear o perfil `ml_full_access` para o seu perfil do IAM que tem permissões `iam:PassRole` para passar `opensearch-sagemaker-role`.

1. Navegue até o plug-in OpenSearch Dashboards do seu domínio OpenSearch de serviço. Você pode encontrar o endpoint do Dashboards no painel do seu domínio no console de OpenSearch serviços.
2. No menu principal, escolha Segurança, Funções e selecione a função ml_full_access.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Em Funções de backend, adicione o ARN da função que tem permissão para aprovar `opensearch-sagemaker-role`.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Crie um conector OpenSearch de serviço

Para criar um conector, envie uma `POST` solicitação para o endpoint do domínio OpenSearch Service. Você pode usar curl, o cliente Python de amostra, o Postman ou outro método para enviar uma solicitação assinada. Você não pode usar uma solicitação `POST` no console do Kibana. A solicitação assume o seguinte formato:

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
```

```
            "action_type": "predict",
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
    ]
}
```

De duas maneiras, o corpo dessa solicitação é diferente do de uma solicitação de conector de código aberto. Dentro do `credential` campo, você passa o ARN para a função do IAM que permite que o OpenSearch Service leia do Secrets Manager, junto com o ARN para o segredo de quê. No campo `headers`, você se refere ao segredo usando a chave secreta e o fato de ser proveniente de um ARN.

Se o domínio residir em uma nuvem privada virtual (VPC), seu computador deverá estar conectado à VPC para que a solicitação crie o conector de IA com êxito. O acesso a uma VPC varia de acordo com a configuração de rede, mas geralmente requer uma conexão com VPN ou rede corporativa. Para verificar se você pode acessar seu domínio OpenSearch de serviço, navegue até `https://your-vpc-domain.region.es.amazonaws.com` em um navegador da Web e verifique se você recebeu a resposta JSON padrão.

Exemplo de cliente do Python

O cliente Python é mais simples de automatizar que uma solicitação HTTP, além de ser mais fácil reutilizá-lo. Para criar o conector AI com o cliente Python, salve o código de exemplo a seguir em um arquivo Python. O cliente requer os pacotes AWS SDK para Python (Boto3), `requests` e `requests-aws4auth`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
```

```
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
            "action_type": "predict",
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
    ]
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Usando CloudFormation para configurar a inferência remota para pesquisa semântica

A partir da OpenSearch versão 2.9, você pode usar a inferência remota com pesquisa semântica para hospedar seus próprios modelos de aprendizado de máquina (ML). A inferência remota usa o plug-in ML Commons.

Com a inferência remota, você pode hospedar suas inferências de modelo remotamente em serviços de ML, como Amazon AI SageMaker e Amazon Bedrock, e conectá-las ao Amazon OpenSearch Service com conectores de ML.

Para facilitar a configuração da inferência remota, o Amazon OpenSearch Service fornece um AWS CloudFormationmodelo no console. CloudFormation é um AWS service (Serviço da AWS) local onde você pode provisionar AWS e gerenciar recursos de terceiros tratando a infraestrutura como código.

O OpenSearch CloudFormation modelo automatiza o processo de provisionamento do modelo para que você possa criar facilmente um modelo em seu domínio de OpenSearch serviço e, em seguida, usar o ID do modelo para ingerir dados e executar consultas de pesquisa neural.

Ao usar codificadores neurais esparsos com a versão 2.12 e posteriores do OpenSearch Service, recomendamos que você use o modelo de tokenizador localmente em vez de implantá-lo remotamente. Para obter mais informações, consulte Modelos de codificação esparsa na OpenSearch documentação.

Tópicos

- CloudFormation Modelos disponíveis
- Pré-requisitos
- Modelos do Amazon Bedrock
- Configurando a pesquisa de agentes com Bedrock Claude
- Modelos de integração do servidor MCP
- SageMaker Modelos da Amazon
- Inferência remota para modelos de destaque semântico

CloudFormation Modelos disponíveis

Os seguintes modelos AWS CloudFormation de aprendizado de máquina (ML) estão disponíveis para uso:

Modelos do Amazon Bedrock

Integração do Incorporador de Texto do Amazon Titan

> Conecta-se aos modelos de ML hospedados do Amazon Bedrock, elimina a necessidade de implantação separada de modelos e usa endpoints predeterminados do Amazon Bedrock. Para

saber mais, consulte Incorporador de texto do Amazon Titan no Guia do usuário do Amazon Bedrock.

Integração do Cohere Embed

Fornece acesso aos modelos do Cohere Embed e é otimizado para fluxos de trabalho específicos de processamento de texto. Para saber mais, consulte Embed no site Cohere docs.

Incorporador Multimodal do Amazon Titan

É compatível com incorporações de texto e imagem, e habilita recursos de pesquisa multimodal. Para saber mais, consulte Incorporador Multimodal do Amazon Titan no Guia do usuário do Amazon Bedrock.

Modelos de integração do servidor MCP

Integração de servidor MCP

Implanta um Amazon Bedrock AgentCore Runtime, fornece um endpoint de agente, gerencia a autenticação de entrada e saída e oferece suporte à autenticação corporativa. OAuth

SageMaker Modelos da Amazon

Integração com modelos de incorporação de texto por meio da Amazon SageMaker

Implanta modelos de incorporação de texto no Amazon SageMaker Runtime, cria funções do IAM para acesso ao artefato do modelo e estabelece conectores de ML para pesquisa semântica.

Integração com codificadores esparsos por meio de SageMaker

Configura modelos de codificação esparsos para pesquisa neural, cria AWS Lambda funções para gerenciamento de conectores e retorna o modelo IDs para uso imediato.

Pré-requisitos

Para usar um CloudFormation modelo com o OpenSearch Serviço, preencha os pré-requisitos a seguir.

Configurar um domínio OpenSearch de serviço

Antes de usar um CloudFormation modelo, você deve configurar um domínio do Amazon OpenSearch Service com a versão 2.9 ou posterior e um controle de acesso refinado ativado. Crie

uma função OpenSearch de back-end de serviço para dar permissão ao plug-in ML Commons para criar seu conector para você.

O CloudFormation modelo cria uma função do Lambda IAM para você com o nome padrão`LambdaInvokeOpenSearchMLCommonsRole`, que você pode substituir se quiser escolher um nome diferente. Depois que o modelo criar essa função do IAM, você precisa dar permissão à função Lambda para chamar seu domínio de OpenSearch serviço. Para fazer isso, mapeie a função nomeada `ml_full_access` para sua função OpenSearch de back-end de serviço com as seguintes etapas:

1. Navegue até o plug-in OpenSearch Dashboards do seu domínio OpenSearch de serviço. Você pode encontrar o endpoint do Dashboards no painel do seu domínio no console de OpenSearch serviços.
2. No menu principal, escolha Segurança, Funções e selecione a função ml_full_access.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Em Funções de backend, adicione o ARN da função do Lambda que precisa de permissão para chamar seu domínio.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Depois de mapear a função, navegue até a configuração de segurança do seu domínio e adicione a função Lambda IAM à OpenSearch sua política de acesso ao serviço.

Ative as permissões em seu Conta da AWS

Você Conta da AWS deve ter permissão para acessar CloudFormation o Lambda, junto com o que AWS service (Serviço da AWS) você escolher para seu modelo — Runtime SageMaker ou Amazon Bedrock.

Se estiver usando o Amazon Bedrock, você também deve registrar seu modelo. Consulte Acesso aos modelos no Guia do usuário do Amazon Bedrock para registrar seu modelo.

Se você estiver usando seu próprio bucket do Amazon S3 para fornecer artefatos de modelo, deverá adicionar a função do CloudFormation IAM à sua política de acesso do S3. Para saber mais, consulte Adicionar e remover permissões de identidade do IAM no Guia do usuário do IAM.

Modelos do Amazon Bedrock

Os CloudFormation modelos do Amazon Bedrock provisionam os AWS recursos necessários para criar conectores entre o OpenSearch Service e o Amazon Bedrock.

Primeiro, o modelo cria uma função do IAM que permite que a futura função Lambda acesse seu domínio de OpenSearch serviço. Em seguida, o modelo cria a função do Lambda, que faz com que o domínio crie um conector usando o plug-in ML Commons. Depois que o OpenSearch Service cria o conector, a configuração da inferência remota é concluída e você pode executar pesquisas semânticas usando as operações da API Amazon Bedrock.

> **Note**
>
> Como o Amazon Bedrock hospeda seus próprios modelos de ML, você não precisa implantar um modelo no SageMaker Runtime. Em vez disso, o modelo usa um endpoint predeterminado para o Amazon Bedrock e ignora as etapas de provisionamento do endpoint.

Para usar o modelo Amazon Bedrock CloudFormation

1. Abra o console do Amazon OpenSearch Service.
2. No painel de navegação à esquerda, selecione Integrações.
3. Em Integrar com o modelo Incorporador de Texto do Amazon Titan por meio do Amazon Bedrock, escolha Configurar domínio, Configurar domínio público.
4. Siga o prompt para configurar seu modelo.

> **Note**
>
> OpenSearch O serviço também fornece um modelo separado para configurar um domínio da Amazon VPC. Se você usar esse modelo, precisará fornecer o ID da Amazon VPC para a função do Lambda.

Além disso, o OpenSearch Service fornece os seguintes modelos do Amazon Bedrock para se conectar ao modelo Cohere e ao modelo Amazon Titan Multimodal Embeddings:

- `Integration with Cohere Embed through Amazon Bedrock`

- `Integrate with Amazon Bedrock Titan Multi-modal`

Configurando a pesquisa de agentes com Bedrock Claude

O Agentic Search utiliza agentes autônomos para executar pesquisas complexas em seu nome, entendendo a intenção do usuário, orquestrando as ferramentas certas, gerando consultas otimizadas e fornecendo resumos transparentes de suas decisões por meio de uma interface de linguagem natural. Esses agentes são movidos por modelos de raciocínio, como Bedrock Claude.

Siga as etapas abaixo para abrir e executar um CloudFormation modelo que configura automaticamente os modelos da Bedrock Claude para pesquisa de agentes e saiba como configurar e criar seus agentes no plug-in AI Search Flows em painéis. OpenSearch

Habilitando o acesso ao Bedrock Claude

1. Pré-requisito: se seu domínio usa controle de acesso refinado, mapeie `arn:aws:iam::`*`your-account-id`*`:role/`*`LambdaInvokeOpenSearchMLCommonsRole`* como uma função de back-end para a função antes de executar o `ml_full_access` modelo. Essa função do IAM será criada automaticamente, CloudFormation caso ainda não exista. Para obter mais informações sobre como configurar o mapeamento, consulteMapeie a função de ML em OpenSearch painéis (se estiver usando controle de acesso refinado).
2. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
3. No painel de navegação à esquerda, escolha Integrações.
4. Em Integração com Bedrock Claude for Agentic Search, escolha Configurar domínio. Certifique-se de que seu domínio esteja na versão 3.3 ou superior.
5. No CloudFormation modelo, insira seu endpoint do domínio de OpenSearch serviço e selecione um modelo. Os campos restantes são opcionais ou pré-preenchidos. Clique em Create Stack e aguarde a conclusão do provisionamento.
6. No console do Amazon OpenSearch Service, selecione Domínios e selecione seu domínio. Clique na URL dos OpenSearch painéis para acessar os OpenSearch painéis.

Criando agentes e executando o Agentic Search

1. Em OpenSearch Painéis, abra o menu no lado esquerdo. Selecione OpenSearch Plugins > AI Search Flows para acessar o plug-in.

2. Na página Fluxos de trabalho, selecione a guia Novo fluxo de trabalho e, no cartão Agentic Search, clique em Criar.
3. Forneça um nome exclusivo para sua configuração de pesquisa e clique em Criar.
4. Em Configurar agente, clique em Criar novo agente. Selecione seu modelo Bedrock Claude recém-criado e clique em Criar agente. Se o botão estiver desativado, marque Configurações avançadas > Interface LLM e verifique se há uma interface válida selecionada. Todos os modelos CloudFormation serão modelos Bedrock Claude, então você pode selecionar Bedrock Claude, se ainda não estiver, e clicar em Criar agente.
5. Em Fluxo de teste, tente executar pesquisas agentes. Forneça uma consulta de pesquisa em linguagem natural e clique em Pesquisar.

Para obter a documentação completa do plug-in AI Search Flows, consulte Como configurar o Agentic Search na OpenSearch documentação.

Para obter mais informações sobre como o Agentic Search funciona, consulte Agentic Search na OpenSearch documentação.

Modelos de integração do servidor MCP

Com os modelos de servidor Model Context Protocol (MCP), você pode implantar um servidor MCP OpenSearch hospedado no Amazon Bedrock AgentCore, reduzindo a complexidade da integração entre agentes e ferramentas de IA. OpenSearch Para obter mais informações, consulte O que é o Amazon Bedrock AgentCore? .

Atributos do modelo

Esse modelo inclui os seguintes atributos principais para implantação e gerenciamento do servidor MCP.

Implantação gerenciada do servidor MCP

> Implanta opensearch-mcp-server-pyusando o Amazon Bedrock AgentCore Runtime e fornece um endpoint de agente que envia solicitações por proxy para o servidor MCP subjacente. Para obter mais informações, consulte opensearch-mcp-server-pyem GitHub.

Autenticação e segurança

Lida com a autenticação de entrada (dos usuários para o servidor MCP) e a autenticação de saída (do servidor MCP para OpenSearch), além de oferecer suporte OAuth à autenticação corporativa.

> **Note**
>
> O modelo do servidor MCP só está disponível nas seguintes Regiões da AWS:
>
> - Leste dos EUA (N. da Virgínia)
> - Oeste dos EUA (Oregon)
> - Europa (Frankfurt)
> - Ásia-Pacífico (Sydney)

Para usar o modelo do servidor MCP

Siga estas etapas para implantar o modelo de servidor MCP e conectá-lo ao seu OpenSearch domínio.

1. Abra o console do Amazon OpenSearch Service.
2. No painel de navegação à esquerda, selecione Integrações.
3. Localize o modelo de integração do servidor MCP.
4. Escolha Configurar domínio. Em seguida, insira o endpoint OpenSearch do seu domínio.

O modelo cria um AgentCore Runtime e os seguintes componentes, se os parâmetros opcionais correspondentes não forem especificados:

- Um repositório do Amazon ECR
- Um grupo de usuários do Amazon Cognito como autorizador OAuth
- Uma função de execução usada pelo AgentCore Runtime

Depois de concluir esse procedimento, você deve seguir estas etapas de pós-criação:

1. Para Amazon OpenSearch Service: mapeie o ARN da sua função de execução para uma função de OpenSearch back-end para controlar o acesso ao seu domínio.

   Para Amazon OpenSearch Serverless: crie uma política de acesso a dados que permita que sua função de execução acesse sua coleção.
2. Obtenha um token de OAuth acesso do seu autorizador. Em seguida, use esse token para acessar o servidor MCP na URL listada na saída da sua CloudFormation pilha.

Para obter mais informações, consulte Ações políticas para OpenSearch Serverless.

Integração com agentes de IA

Após a implantação, você pode integrar o servidor MCP com qualquer agente compatível com MCP. Para saber mais, consulte Invocar o servidor MCP implantado no Guia do desenvolvedor do Amazon Bedrock.

Integração de desenvolvedores

Você pode adicionar o endpoint do servidor MCP à configuração do agente. Você também pode usá-lo com a CLI do Amazon Q Developer, com agentes personalizados ou outros agentes compatíveis com MCP.

Implantação corporativa

Agentes hospedados centralmente podem se conectar a vários serviços com OpenSearch um único componente. Esse agente oferece suporte a sistemas de autenticação corporativos OAuth e é escalável para oferecer suporte a vários usuários e casos de uso.

Exemplo de uso da estrutura do Strands Agents

```
import os
import requests
from strands import Agent
from strands.tools.mcp import MCPClient
from mcp.client.streamable_http import streamablehttp_client

def get_bearer_token(discovery_url: str, client_id: str, client_secret: str):
    response = requests.get(discovery_url)
    discovery_data = response.json()
    token_endpoint = discovery_data['token_endpoint']
```

```
    data = {
        'grant_type': 'client_credentials',
        'client_id': client_id,
        'client_secret': client_secret
    }
    headers = {
        'Content-Type': 'application/x-www-form-urlencoded'
    }

    response = requests.post(token_endpoint, data=data, headers=headers)
    token_data = response.json()
    return token_data['access_token']

if __name__ == "__main__":
    discovery_url = os.environ["DISCOVERY_URL"]
    client_id = os.environ["CLIENT_ID"]
    client_secret = os.environ["CLIENT_SECRET"]
    mcp_url = os.environ["MCP_URL"]

    bearer_token = get_bearer_token(discovery_url, client_id, client_secret)

    opensearch_mcp_client = MCPClient(lambda: streamablehttp_client(mcp_url, {
        "authorization": f"Bearer {bearer_token}",
        "Content-Type": "application/json"
    }))

    with opensearch_mcp_client:
        tools = opensearch_mcp_client.list_tools_sync()
        agent = Agent(tools=tools)
        agent("list indices")
```

Para obter mais informações, consulte Hospedagem do OpenSearch MCP Server com o Amazon Bedrock AgentCore no OpenSearch site.

SageMaker Modelos da Amazon

Os SageMaker CloudFormation modelos da Amazon definem vários AWS recursos para configurar o plug-in neural e a pesquisa semântica para você.

Comece usando a integração com modelos de incorporação de texto por meio do modelo da Amazon para implantar um SageMaker modelo de incorporação de texto no SageMaker Runtime como um servidor. Se você não fornecer um endpoint de modelo, CloudFormation cria uma função

do IAM que permite ao SageMaker Runtime baixar artefatos de modelo do Amazon S3 e implantá-los no servidor. Se você fornecer um endpoint, CloudFormation cria uma função do IAM que permite que a função Lambda acesse OpenSearch o domínio do Serviço ou, se a função já existir, atualize e reutilize a função. O endpoint serve o modelo remoto usado para o conector ML com o plug-in ML Commons.

Em seguida, use o SageMaker modelo Integration with Sparse Encoders through Amazon para criar uma função Lambda que faça com que seu domínio configure conectores de inferência remotos. Depois que o conector é criado no OpenSearch Service, a inferência remota pode executar a pesquisa semântica usando o modelo remoto no SageMaker Runtime. O modelo retorna o ID do modelo em seu domínio para que você possa começar a pesquisar.

Para usar os CloudFormation modelos de SageMaker IA da Amazon

1. Abra o console do Amazon OpenSearch Service.
2. No painel de navegação à esquerda, selecione Integrações.
3. Em cada um dos modelos do Amazon SageMaker AI, escolha Configurar domínio, Configurar domínio público.
4. Siga as instruções no CloudFormation console para provisionar sua pilha e configurar um modelo.

> **Note**
>
> OpenSearch O serviço também fornece um modelo separado para configurar o domínio VPC. Se você usar esse modelo, precisará fornecer o ID da VPC para a função do Lambda.

Inferência remota para modelos de destaque semântico

O destaque semântico é um atributo de pesquisa avançada que aumenta a relevância dos resultados analisando o significado e o contexto das consultas, em vez de confiar apenas em correspondências exatas de palavras-chave. Esse recurso usa modelos de machine learning para avaliar a semelhança semântica entre as consultas da pesquisa e o conteúdo do documento ao identificar e destacar as frases ou passagens mais contextualmente relevantes dos documentos. Ao contrário dos métodos tradicionais de destaque que se concentram nas correspondências exatas de termos, o destaque semântico utiliza modelos de IA para avaliar cada frase usando informações contextuais da consulta e o texto ao redor, fazendo com que ela exiba informações pertinentes

mesmo quando os termos de pesquisa exatos não estão presentes nas passagens destacadas. Essa abordagem é especialmente relevante valiosa para implementações de pesquisa baseada em IA, nas quais os usuários priorizam o significado semântico em vez da correspondência literal de palavras, permitindo que os administradores de pesquisa ofereçam experiências de pesquisa mais inteligentes que levam em conta o contexto e destacam extensões de conteúdo significativas em vez de apenas ocorrências de palavras-chave. Para saber mais, consulte Usar destaque semântico.

Use o procedimento a seguir para abrir e executar um CloudFormation modelo que configura automaticamente os SageMaker modelos da Amazon para realce semântico.

Para usar o modelo de realce CloudFormation semântico

1. Abra o console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. No painel de navegação à esquerda, escolha Integrações.
3. Em Ativar destaque semântico por meio da SageMaker integração com a Amazon, escolha Configurar domínio, Configurar domínio público.
4. Siga o prompt para configurar seu modelo.

> **Note**
>
> OpenSearch O serviço também fornece um modelo separado para configurar o domínio VPC. Se você usar esse modelo, precisará fornecer o ID da VPC para a função do Lambda.

Configurações do ML Commons não compatíveis

O Amazon OpenSearch Service não suporta o uso das seguintes configurações do ML Commons:

- `plugins.ml_commons.allow_registering_model_via_url`
- `plugins.ml_commons.allow_registering_model_via_local_file`

> **Important**
>
> Em clusters de produção, não desabilite a configuração do cluster `plugins.ml_commons.only_run_on_ml_node` (não a defina como `false`). A opção de desabilitar essa proteção é para facilitar o desenvolvimento, mas os clusters de produção

devem usar os conectores. Para saber mais, consulte the section called “Conectores para Serviços da AWS”.

Para saber mais sobre as configurações do ML Commons, consulte Configurações de cluster do ML Commons.

OpenSearch Modelos de estrutura de fluxo de serviço

Os modelos de estrutura de fluxo do Amazon OpenSearch Service permitem que você automatize tarefas complexas de configuração e pré-processamento de OpenSearch serviços fornecendo modelos para casos de uso comuns. Por exemplo, você pode usar modelos de estrutura de fluxo para automatizar as tarefas de configuração do machine learning. Os modelos de estrutura de fluxo do Amazon OpenSearch Service fornecem uma descrição compacta do processo de configuração em um documento JSON ou YAML. Esses modelos descrevem configurações automatizadas de fluxo de trabalho para bate-papo conversacional ou geração de consultas, conectores de IA, ferramentas, agentes e outros componentes que preparam o OpenSearch Serviço para uso de back-end em modelos generativos.

Os modelos de estrutura de fluxo do Amazon OpenSearch Service podem ser personalizados para atender às suas necessidades específicas. Para ver um exemplo de um modelo de estrutura de fluxo personalizado, consulte flow-framework. Para modelos fornecidos pelo OpenSearch serviço, consulte modelos de fluxo de trabalho. Para obter uma documentação abrangente, incluindo etapas detalhadas, uma referência de API e uma referência de todas as configurações disponíveis, consulte Como automatizar a configuração na documentação de código OpenSearch aberto.

> **Note**
>
> O Flow-framework não oferece suporte à filtragem de funções de back-end para o Service 2.17. OpenSearch

Criação de conectores de ML no Service OpenSearch

Os modelos de estrutura de fluxo do Amazon OpenSearch Service permitem que você configure e instale conectores de ML utilizando a API de criação de conectores oferecida no ml-commons. Você pode usar conectores de ML para conectar o OpenSearch Serviço a outros AWS serviços ou plataformas de terceiros. Para saber mais sobre isso, consulte Como criar conectores para

plataformas de ML de terceiros. A API da estrutura OpenSearch de fluxo do Amazon Service permite automatizar as tarefas de configuração e pré-processamento do OpenSearch serviço e pode ser usada para criar conectores de ML.

Antes de criar um conector no OpenSearch Service, você deve fazer o seguinte:

- Crie um domínio Amazon SageMaker AI.
- Criar um perfil do IAM.
- Configure a permissão de função de transmissão.
- Mapeie a estrutura de fluxo e as funções do ml-commons nos painéis. OpenSearch

Para obter mais informações sobre como configurar conectores de ML para AWS serviços, consulte Conectores de ML do Amazon OpenSearch Service para AWS serviços. Para saber mais sobre o uso OpenSearch de conectores Service ML com plataformas de terceiros, consulte Conectores Amazon OpenSearch Service ML para plataformas de terceiros.

Como criar um conector por meio de um serviço de estrutura de fluxo

Para criar um modelo de estrutura de fluxo com conector, você precisará enviar uma `POST` solicitação para o endpoint do domínio OpenSearch Service. Você pode usar cURL, um cliente Python de amostra, o Postman ou outro método para enviar uma solicitação assinada. A solicitação `POST` assume o seguinte formato:

```
POST /_plugins/_flow_framework/workflow
{
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
  "workflows": {
    "provision": {
      "nodes": [
        {
          "id": "create_claude_connector",
```

```
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to Bedrock service for Claude model",
            "actions": [
              {
                "headers": {
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
                },
                "method": "POST",
                "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
                "action_type": "predict",
                "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
              }
            ],
            "credential": {
                "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
             },
            "parameters": {
              "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
              "content_type": "application/json",
              "auth": "Sig_V4",
              "max_tokens_to_sample": "8000",
              "service_name": "bedrock",
              "temperature": "0.0001",
              "response_filter": "$.completion",
              "region": "us-west-2",
              "anthropic_version": "bedrock-2023-05-31"
            }
          }
        }
      ]
    }
  }
}
```

Se o domínio residir em uma nuvem privada virtual (Amazon VPC), você deverá estar conectado à Amazon VPC para que a solicitação crie o conector de IA com êxito. O acesso a uma Amazon VPC varia de acordo com a configuração de rede, mas geralmente requer uma conexão com VPN ou rede corporativa. Para verificar se você pode acessar seu domínio OpenSearch de serviço, navegue até `https://your-vpc-domain.region.es.amazonaws.com` em um navegador da Web e verifique se você recebeu a resposta JSON padrão. (Substitua *placeholder text* o por seus próprios valores.

Exemplo de cliente do Python

O cliente Python é mais simples de automatizar do que uma solicitação HTTP, além de ser mais fácil reutilizá-lo. Para criar o conector AI com o cliente Python, salve o código de exemplo a seguir em um arquivo Python. O cliente requer os pacotes AWS SDK for Python (Boto3), Requests:HTTP for Humans e requests-aws4auth 1.2.3.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

path = '_plugins/_flow_framework/workflow'
url = host + path

payload = {
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
  "workflows": {
    "provision": {
```

```
      "nodes": [
        {
          "id": "create_claude_connector",
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to Bedrock service for Claude model",
            "actions": [
              {
                "headers": {
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
                },
                "method": "POST",
                "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
                "action_type": "predict",
                "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
              }
            ],
            "credential": {
                "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
             },
            "parameters": {
              "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
              "content_type": "application/json",
              "auth": "Sig_V4",
              "max_tokens_to_sample": "8000",
              "service_name": "bedrock",
              "temperature": "0.0001",
              "response_filter": "$.completion",
              "region": "us-west-2",
              "anthropic_version": "bedrock-2023-05-31"
            }
          }
        }
      ]
    }
```

```
  }
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Modelos de fluxo de trabalho predefinidos

O Amazon OpenSearch Service fornece vários modelos de fluxo de trabalho para alguns casos de uso comuns de aprendizado de máquina (ML). O uso de um modelo simplifica configurações complexas e fornece muitos valores padrão para casos de uso, como pesquisa semântica ou conversacional. Você pode especificar um modelo de fluxo de trabalho ao chamar a API Create Workflow.

- Para usar um modelo de fluxo de trabalho fornecido pelo OpenSearch serviço, especifique o caso de uso do modelo como parâmetro de `use_case` consulta.
- Para usar um modelo de fluxo de trabalho personalizado, forneça o modelo completo no corpo da solicitação. Para ver um exemplo de modelo personalizado, consulte um exemplo de modelo JSON ou um exemplo de modelo YAML.

Casos de uso de modelos

Esta tabela fornece uma visão geral dos diferentes modelos disponíveis, uma descrição dos modelos e os parâmetros necessários.

Caso de uso do modelo	Description	Parâmetros necessários
`bedrock_titan_embedding_model_deploy`	Cria e implanta um modelo de incorporação do Amazon Bedrock (por padrão, `titan-embed-text-v1`).	`create_connector.credential.roleArn`
`bedrock_titan_embe`	Cria e implanta um modelo de incorporação multimodal Amazon	`create_connector.credential.roleArn`

Caso de uso do modelo	Description	Parâmetros necessários
`dding_model_deploy`	Bedrock (por padrão, `titan-embed-text-v1`).	
`cohere_embedding_model_deploy`	Cria e implanta um modelo de incorporação Cohere (por padrão, embed-english-v 3.0).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`cohere_chat_model_deploy`	Cria e implanta um modelo de chat Cohere (por padrão, Cohere Command).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`open_ai_embedding_model_deploy`	Cria e implanta um modelo de incorporação OpenAI (por padrão text-embedding-ada, -002).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`openai_chat_model_deploy`	Cria e implanta um modelo de chat OpenAI (por padrão, gpt-3.5-turbo).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`semantic_search_with_cohere_embedding`	Configura a pesquisa semântica e implanta um modelo de incorporação Cohere. Você deve fornecer a chave de API para o modelo Cohere.	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`

Caso de uso do modelo	Description	Parâmetros necessários
`semantic_search_with_cohere_embedding_query_enricher`	Configura a pesquisa semântica e implanta um modelo de incorporação Cohere. Adiciona um processador de pesquisa query_enricher que define um ID de modelo padrão para consultas neurais. Você deve fornecer a chave de API para o modelo Cohere.	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`multimodal_search_with_bedrock_titan`	Implanta um modelo multimodal do Amazon Bedrock e configura um pipeline de ingestão com um processador text_image_embedding e um índice k-NN para pesquisa multimodal. Você deve fornecer suas credenciais da AWS .	`create_connector.credential.roleArn`

> **Note**
>
> Para todos os modelos que exigem um ARN secreto, o padrão é armazenar o segredo com o nome de chave “chave” no AWS Secrets Manager.

Modelos padrão com modelos pré-treinados

O Amazon OpenSearch Service oferece dois modelos adicionais de fluxo de trabalho padrão não disponíveis no serviço de código aberto OpenSearch .

Caso de uso do modelo	Description
`semantic_search_with_local_model`	Configura a pesquisa semântica e implanta um modelo pré-treinado (`msmarco-distilbert-base-tas-b`). Adiciona um processador de `neural_query_enricher` pesquisa que

Caso de uso do modelo	Description
	define um ID de modelo padrão para consultas neurais e cria um índice k-NN vinculado chamado ". my-nlp-index
`hybrid_search_with_local_model`	Configura a pesquisa híbrida e implanta um modelo pré-treinado (`msmarco-distilbert-base-tas-b`). Adiciona um processador de `neural_query_enricher` pesquisa que define um ID de modelo padrão para consultas neurais e cria um índice k-NN vinculado chamado ". my-nlp-index

Configurar permissões do

Se você criar um novo domínio com a versão 2.13 ou posterior, as permissões já estarão em vigor. Se você habilitar a estrutura de fluxo em um domínio de OpenSearch serviço preexistente com a versão 2.11 ou anterior e, em seguida, atualizar para a versão 2.13 ou posterior, deverá definir a função. `flow_framework_manager` Os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices warm usando o controle de aceso detalhado. Para criar manualmente a função `flow_framework_manager`, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Group name	Permissões
`flow_framework_full_access`	• `cluster:admin/opensearch/flow_framework/*` • `cluster_monitor`
`flow_framework_read_accesss`	• `cluster:admin/opensearch/flow_framework/workflow/get` • `cluster:admin/opensearch/flow_framework/workflow/search`

Group name	Permissões
	• `cluster:admin/opensearch/flow_framework/workflow_state/get` • `cluster:admin/opensearch/flow_framework/workflow_state/search`

3. Escolha Funções e, em seguida, Criar função.
4. Nomeie a função flow_framework_manager.
5. Para Permissões de cluster, selecione `flow_framework_full_access` e `flow_framework_read_access`.
6. Para Índice, digite `*`.
7. Para Permissões de índice, selecione `indices:admin/aliases/get`, `indices:admin/mappings/get`, e `indices_monitor`.
8. Escolha Criar.
9. Depois de criar a função, mapeie-a em qualquer função de usuário ou de backend que gerencie índices de estrutura de fluxo.

Análise de segurança para Amazon OpenSearch Service

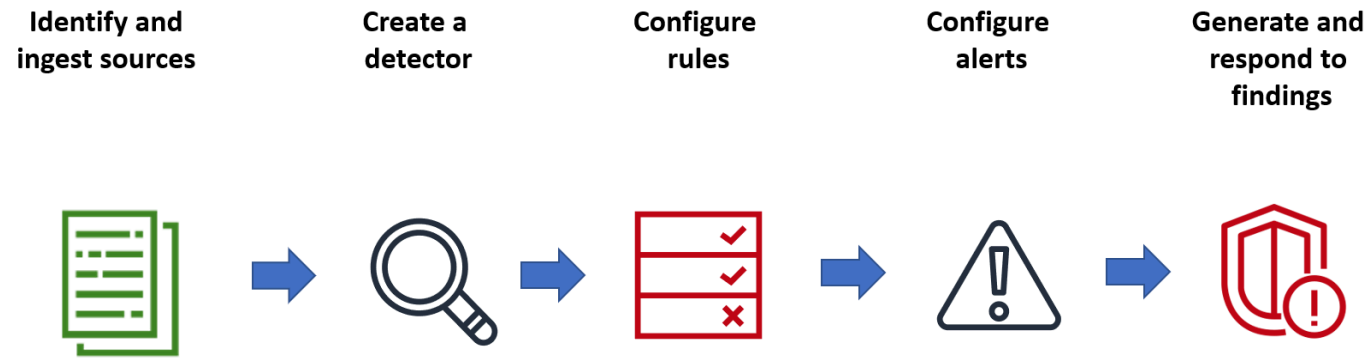
O Security Analytics é uma OpenSearch solução que fornece visibilidade da infraestrutura da sua organização, monitora atividades anômalas, detecta possíveis ameaças à segurança em tempo real e aciona alertas para destinos pré-configurados. Você pode monitorar atividades maliciosas nos seus logs de eventos de segurança avaliando continuamente as regras e revisando as descobertas de segurança geradas automaticamente. Além disso, o Security Analytics pode gerar alertas automatizados e enviá-los para um canal de notificação específico, como Slack ou e-mail.

Você pode usar o plug-in Security Analytics para detectar ameaças comuns out-of-the-box e gerar informações críticas de segurança a partir de seus registros de eventos de segurança existentes, como registros de firewall, registros do Windows e registros de auditoria de autenticação. Para usar o Security Analytics, seu domínio deve estar executando a OpenSearch versão 2.5 ou posterior.

> **Note**
>
> Esta documentação fornece uma breve visão geral do Security Analytics for Amazon OpenSearch Service. Ele define os principais conceitos e fornece etapas para configurar as permissões. Para obter uma documentação abrangente, incluindo um guia de configuração, uma referência de API e uma referência de todas as configurações disponíveis, consulte Security Analytics na OpenSearch documentação.

Componentes e conceitos de Security Analytics

Várias ferramentas e atributos fornecem a base para a operação do Security Analytics. Os principais componentes que compõem o plug-in incluem detectores, tipos de log, regras, descobertas e alertas.



Tipos de log

OpenSearch oferece suporte a vários tipos de registros e fornece out-of-the-box mapeamentos para cada tipo. Você especifica o tipo de log e configura um intervalo de tempo ao criar um detector e, a partir daí, o Security Analytics ativa automaticamente um conjunto relevante de regras que são executadas nesse intervalo.

Detectores

Os detectores identificam uma variedade de ameaças à segurança cibernética para um tipo de log em seus índices de dados. Você configura seu detector para usar regras personalizadas e regras Sigma prontas para uso que avaliam eventos que ocorrem no sistema. O detector então gera descobertas de segurança a partir desses eventos. Para obter mais informações sobre detectores, consulte Criação de detectores na OpenSearch documentação.

Regras

As regras de detecção de ameaças definem as condições que os detectores aplicam aos dados de log ingeridos para identificar um evento de segurança. O Security Analytics oferece suporte à importação, criação e personalização de regras para atender às suas necessidades e também fornece regras Sigma predefinidas e de código aberto para detectar ameaças comuns em seus logs. O Security Analytics mapeia muitas regras para uma base de conhecimento cada vez maior de táticas e técnicas adversárias mantida pela organização MITRE ATT&CK. Você pode usar os OpenSearch painéis ou o APIs para criar e usar regras. Para obter mais informações sobre regras, consulte Como trabalhar com regras na OpenSearch documentação.

Conclusões

Quando um detector combina uma regra com um evento de log, ele gera uma descoberta. Cada descoberta inclui uma combinação exclusiva de regras selecionadas, um tipo de log e uma severidade da regra. As descobertas não apontam necessariamente para ameaças iminentes no sistema, mas sempre isolam um evento de interesse. Para obter mais informações sobre descobertas, consulte Trabalhando com descobertas na OpenSearch documentação.

Alertas

Ao criar um detector, você pode especificar uma ou mais condições que acionam um alerta. Um alerta é uma notificação enviada para um canal preferencial, como Slack ou e-mail. Você configura o

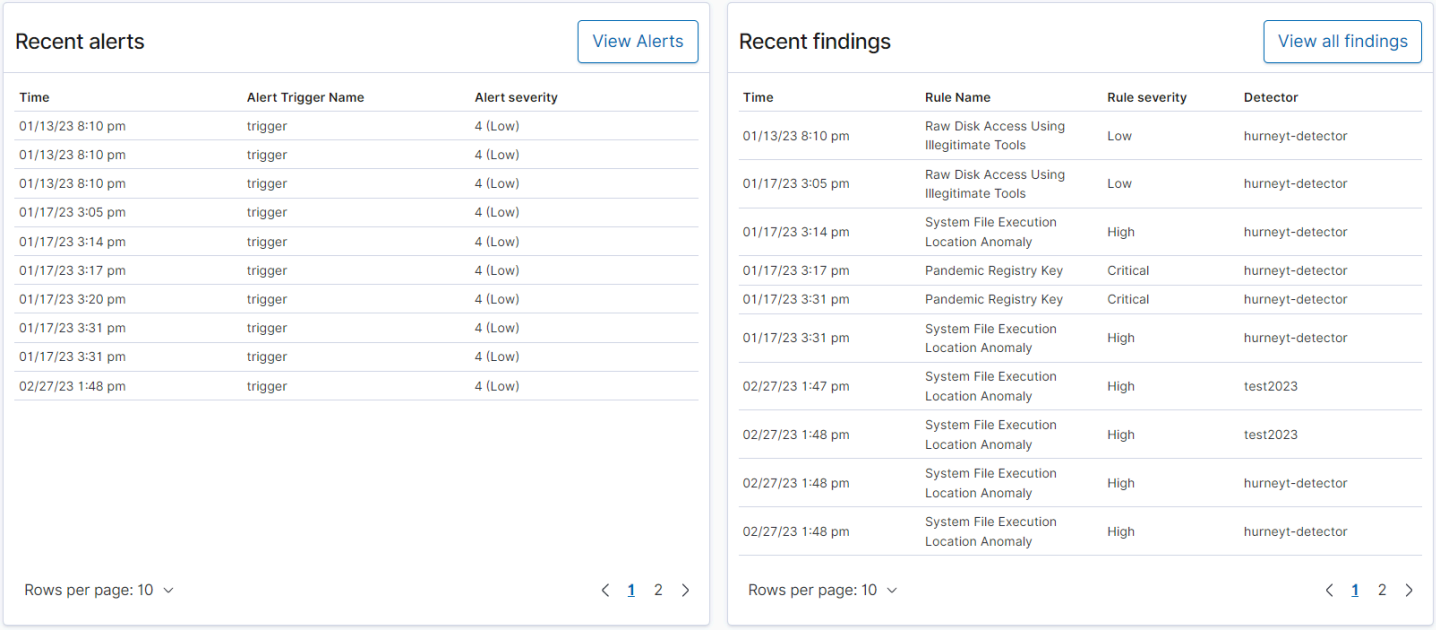
alerta para ser acionado quando o detector corresponde a uma ou várias regras e pode personalizar a mensagem de notificação. Para obter mais informações sobre alertas, consulte Como trabalhar com alertas na OpenSearch documentação.

Explorando o Security Analytics

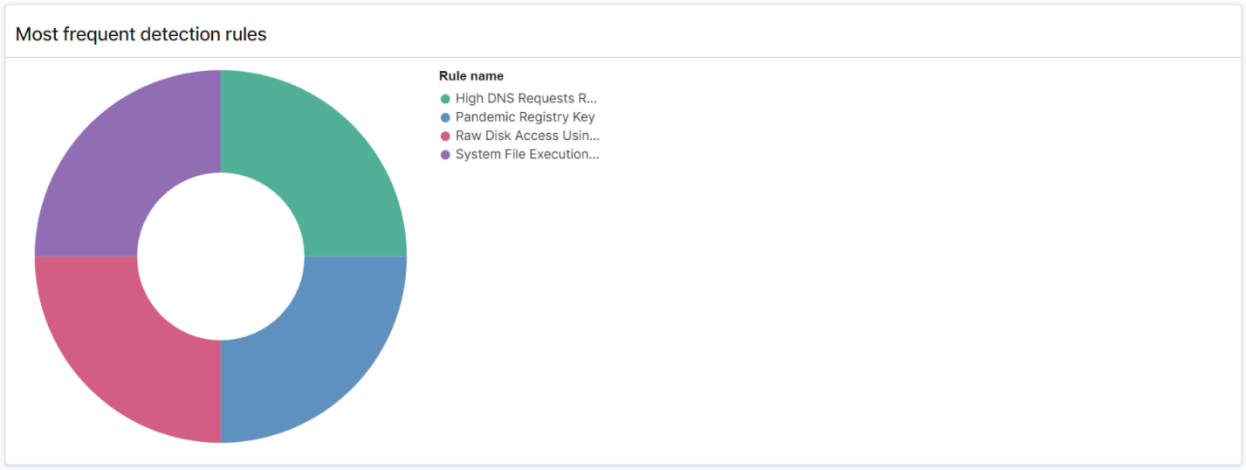
Você pode usar OpenSearch painéis para visualizar e obter informações sobre seu plug-in de análise de segurança. O Visão geral fornece informações como descobertas e contagens de alertas, descobertas e alertas recentes, regras de detecção frequentes e uma lista de seus detectores. Você pode ver uma exibição resumida composta por várias visualizações. O gráfico a seguir, por exemplo, mostra a tendência de descobertas e alertas para vários tipos de logs em um determinado período de tempo.



Mais abaixo na página, você pode revisar suas descobertas e alertas mais recentes.



Além disso, você pode ver uma distribuição das regras acionadas com mais frequência em todos os detectores ativos. Isso pode ajudar você a detectar e investigar diferentes tipos de atividades maliciosas em todos os tipos de log.



Finalmente, é possível visualizar o status dos detectores configurados. Nesse painel, você também pode navegar até o fluxo de trabalho de criação de detectores.



Para configurar sua configuração do Security Analytics, crie regras com a página Regras e use essas regras para escrever detectores na página Detectors. Para uma visão mais focada dos resultados do Security Analytics, você pode usar as páginas Descobertas e Alertas.

Configurar permissões do

Se você habilitar o Security Analytics em um domínio OpenSearch de serviço preexistente, a `security_analytics_manager` função pode não estar definida no domínio. Os usuários não administradores deverão ser mapeados nessa função para poderem gerenciar índices warm usando o controle de aceso detalhado. Para criar manualmente a função `security_analytics_manager`, faça o seguinte:

1. Em OpenSearch Painéis, acesse Segurança e escolha Permissões.
2. Escolha Criar grupo de ações e configure os seguintes grupos:

Group name	Permissões
`security_analytics_full_access`	• `cluster:admin/opensearch/securityanalytics/alerts/*` • `cluster:admin/opensearch/securityanalytics/detector/*` • `cluster:admin/opensearch/securityanalytics/findings/*` • `cluster:admin/opensearch/securityanalytics/mapping/*` • `cluster:admin/opensearch/securityanalytics/rule/*`

Group name	Permissões
`security_analytics_read_access`	• `cluster:admin/opensearch/securityanalytics/alerts/get` • `cluster:admin/opensearch/securityanalytics/detector/get` • `cluster:admin/opensearch/securityanalytics/detector/search` • `cluster:admin/opensearch/securityanalytics/findings/get` • `cluster:admin/opensearch/securityanalytics/mapping/get` • `cluster:admin/opensearch/securityanalytics/mapping/view/get` • `cluster:admin/opensearch/securityanalytics/rule/get` • `cluster:admin/opensearch/securityanalytics/rule/search`

3. Escolha Funções e, em seguida, Criar função.
4. Nomeie o perfil security_analytics_manager.
5. Para Permissões de cluster, selecione `security_analytics_full_access` e `security_analytics_read_access`.
6. Para Índice, digite `*`.
7. Para Permissões de índice, selecione `indices:admin/mapping/put` e `indices:admin/mappings/get`,
8. Escolha Criar.
9. Depois de criar a função, mapeie-a em qualquer função de usuário ou de backend que gerencie índices de Security Analytics.

Solução de problemas

Esse erro de índice não existe

Se você não tiver detectores e abrir o painel do Security Analytics, verá uma notificação no canto inferior direito que diz `[index_not_found_exception] no such index [.opensearch-sap-detectors-config]`. Você pode ignorar essa notificação, que desaparece em alguns segundos e não aparecerá novamente depois que um detector for criado.

Observabilidade no Amazon Service OpenSearch

Observabilidade é a prática de obter informações sobre o estado interno e o desempenho de seus sistemas complexos examinando suas saídas. O monitoramento tradicional pode dizer que seu sistema está inativo. A observabilidade ajuda você a entender por que ele está inativo, permitindo que você faça novas perguntas sobre seus dados.

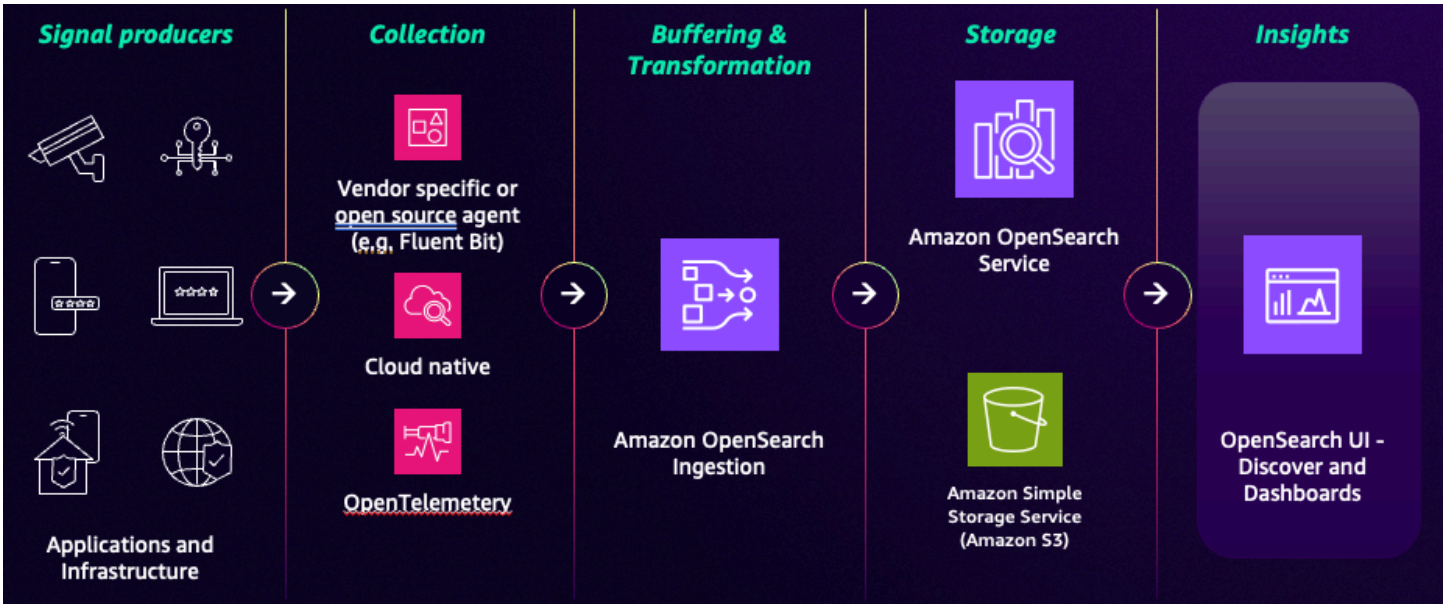
O Amazon OpenSearch Service fornece uma solução unificada coletando e correlacionando os principais tipos de dados de telemetria.

- Os registros fornecem registros com data e hora de eventos, como erros de aplicativos, solicitações de usuários ou mensagens de status do sistema.
- Os rastreamentos representam a end-to-end jornada de uma solicitação à medida que ela percorre os diferentes serviços em um sistema distribuído.

Ao reunir esses tipos de dados, o Amazon OpenSearch Service ajuda equipes operacionais, engenheiros de confiabilidade do site e desenvolvedores a detectar, diagnosticar e resolver problemas operacionais com mais rapidez.

O fluxo de trabalho de observabilidade no Service OpenSearch

Colocar dados de seus aplicativos no OpenSearch Service para análise usa um pipeline de vários estágios:



1. Coleção

   O processo começa na fonte com agentes ou coletores leves que coletam dados de telemetria de seus produtores de sinal, como aplicativos e infraestrutura. Dois agentes de código aberto comuns são:

   - OpenTelemetry(OTel) coletor — O padrão da indústria e o método preferido para coletar registros e rastros.
   - Fluent Bit — Um processador e encaminhador de registros leve e de alto desempenho que suporta o OTel esquema para registros e rastreamentos.

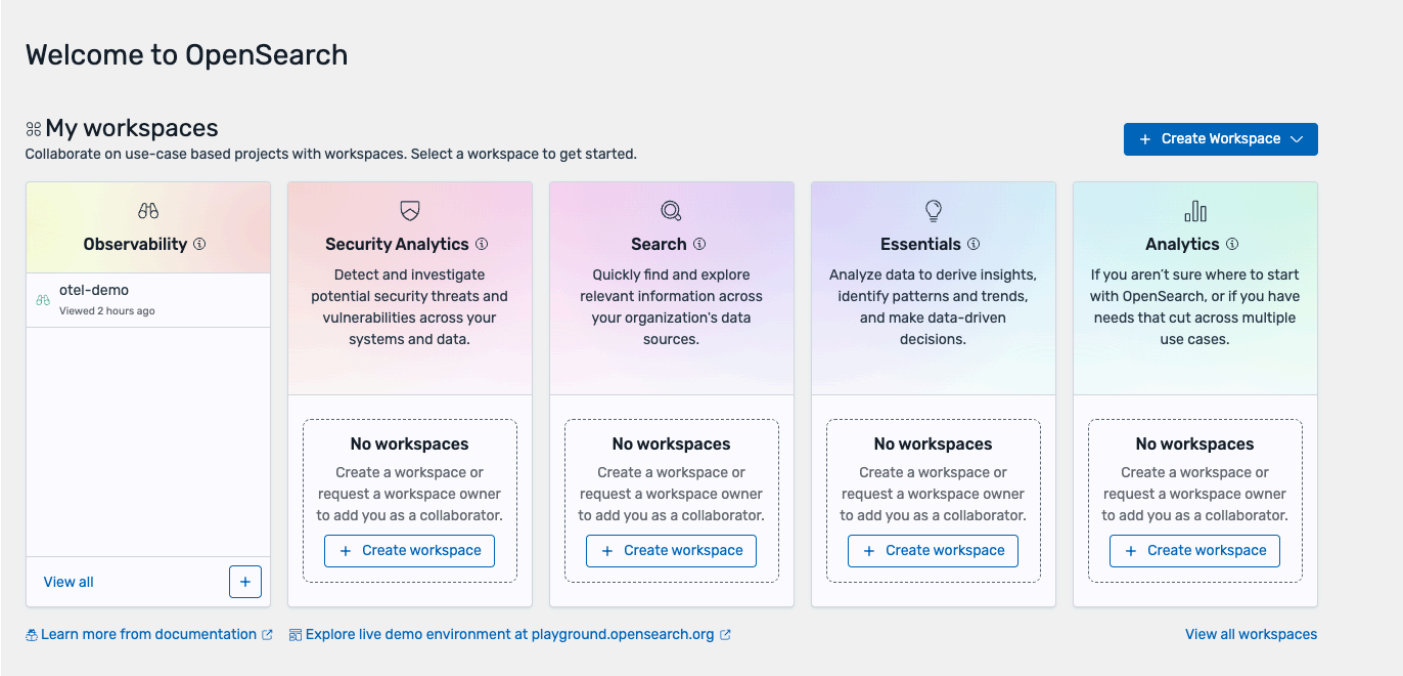
2. Ingestão (OpenSearch ingestão da Amazon)

   Depois que a telemetria é coletada, os dados são enviados para o OpenSearch Ingestion, um pipeline de dados totalmente gerenciado e sem servidor. Você pode criar pipelines personalizados para:

   - Filtrar — remova dados desnecessários para reduzir os custos de armazenamento.
   - Enriqueça — adicione metadados valiosos, como informações geográficas de um endereço IP.
   - Transforme e normalize — estruture registros não estruturados em um formato consistente.
   - Rota — Envie diferentes tipos de dados para diferentes índices de OpenSearch serviços ou para o Amazon S3.

3. Análise e visualização

   Após o processamento, os dados são carregados em um domínio ou coleção do OpenSearch Serviço. Você pode armazenar, indexar e analisar grandes quantidades de dados quase em tempo real. Você interage com esses dados por meio de uma interface de visualização, como o espaço de trabalho de observabilidade da interface do OpenSearch usuário, para executar consultas, criar painéis e configurar alertas.



OpenSearch UI e OpenSearch painéis

OpenSearch O serviço fornece duas interfaces de usuário distintas para observabilidade. Recomendamos que você use a OpenSearch interface do usuário e configure um espaço de trabalho de observabilidade para novas instalações e migre dos painéis existentes. OpenSearch Abaixo está uma tabela descrevendo os benefícios da OpenSearch interface do usuário em relação aos OpenSearch painéis tradicionais.

Recurso	OpenSearch UI	OpenSearch Painéis
Fontes de dados	Várias fontes — pode se conectar a vários domínios OpenSearch de serviço, coleções OpenSearch sem servidor e outras fontes de dados.	Fonte única — co-localizada com um único domínio de OpenSearch serviço.
Atualizações	Os novos recursos chegam aqui primeiro porque não estão vinculados a uma	Os novos recursos estão vinculados à OpenSearc h versão. As atualizações

Recurso	OpenSearch UI	OpenSearch Painéis
	OpenSearch versão específica.	podem ser descontinuadas no futuro.
Disponibilidade	Hospedado na Nuvem AWS garantia de zero tempo de inatividade durante as atualizações do cluster.	Pode ficar temporariamente indisponível durante a manutenção e as atualizações do domínio
Colaboração	Apresenta espaços de trabalho para colaboração em equipe organizada em fluxos de trabalho específicos.	A colaboração é baseada no compartilhamento de objetos salvos em um único domínio.

Observação — Para facilitar a introdução, criamos um novo fluxo de trabalho Get Started para registros no console do Amazon OpenSearch Service, que configurará um novo pipeline de ingestão OTel personalizado, permitirá que você selecione um OpenSearch cluster existente e crie um novo aplicativo de OpenSearch interface de usuário com um espaço de trabalho de observabilidade criado. Tudo o que você precisa fazer é direcionar seus OTel agentes para o novo endpoint de ingestão e você estará pronto para desbloquear insights sobre seus dados OTel formatados.

Como a ingestão e a análise são tratadas de forma diferente em registros e rastreamentos, criamos seções separadas para aprofundar.

Logs

OpenSearch A ingestão pode transformar dados de registro não estruturados em um formato estruturado durante a ingestão. OpenSearch A ingestão fornece processadores que normalizam e enriquecem seus dados antes de serem indexados. Exemplos de processadores úteis são:

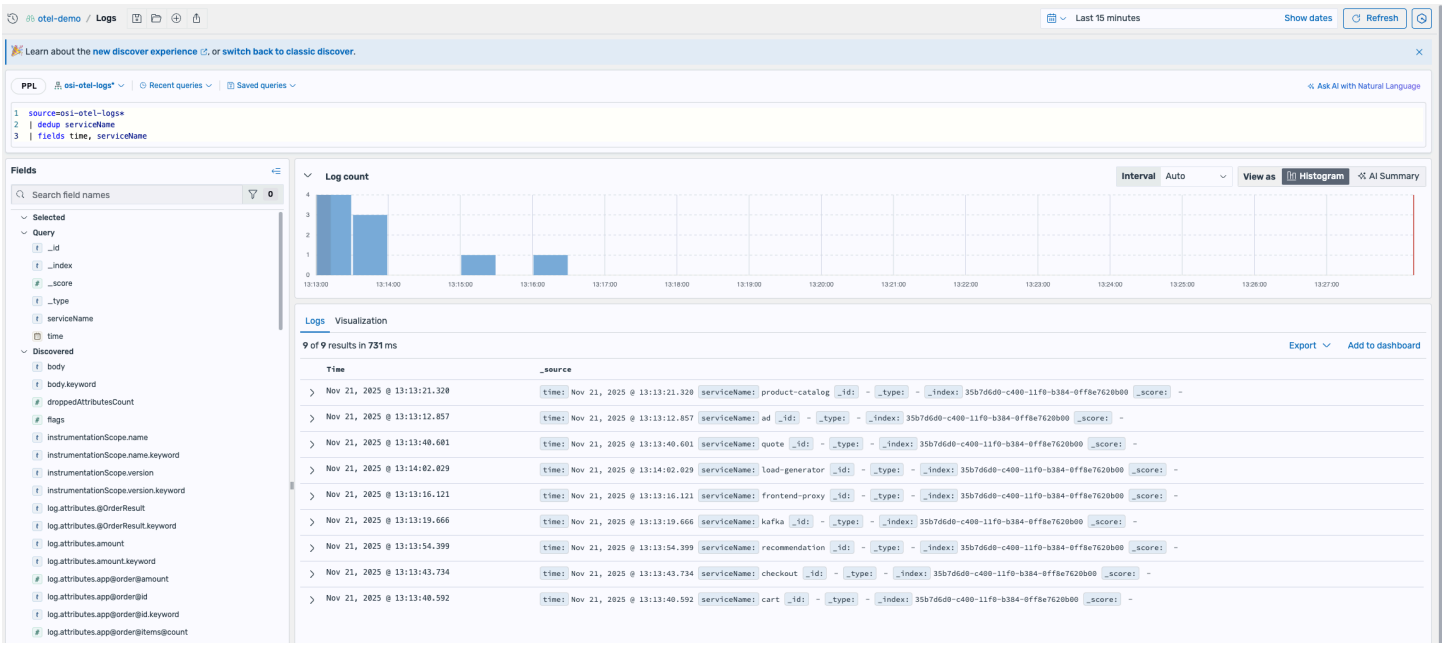
- `grok`— analisa e estrutura dados de texto não estruturados, como registros de acesso ao servidor web, em campos distintos.
- `date`— Analisa uma data de um campo de registro e a define como a data e hora do evento.
- `parse_json`— Analisa um campo de string que contém um objeto JSON.

Observação — Para facilitar a introdução, criamos um novo fluxo de trabalho de introdução para registros no console do Amazon OpenSearch Service, que configurará um novo pipeline de ingestão personalizado da Otel, o direcionará para um OpenSearch cluster existente e criará um novo aplicativo de OpenSearch interface de usuário com um espaço de trabalho de observabilidade criado. Tudo o que você precisa fazer é direcionar seus agentes do Otel para o novo terminal de ingestão.

OpenSearch UI e espaço de trabalho de observabilidade

Depois que seus dados de registros são ingeridos no Amazon OpenSearch Service, você usa as ferramentas fornecidas pelo espaço de trabalho de observabilidade do Amazon OpenSearch Service na OpenSearch interface do usuário para analisá-los. O espaço de trabalho de observabilidade fornece ferramentas especializadas projetadas para extrair insights significativos no Discover e nos painéis.

O espaço de trabalho de observabilidade vem com uma nova experiência Discover que usa linguagem de processamento canalizada (PPL) complementada com um assistente de linguagem natural desenvolvido pelo Amazon Q Developer for Business. A assistência linguística facilita que qualquer pessoa comece a usar linguagens canalizadas. Depois de refinar sua consulta, crie visualizações e painéis diretamente do novo Discover sem ir para outras partes da ferramenta. Para consultar seus dados usando DQL ou SQL, mude para a antiga experiência Discover.



Consultando seus registros usando PPL

Você tem várias opções para consultar seus registros para coletar informações sobre a operação do seu aplicativo ou serviço.

A linguagem de processamento canalizado (PPL) é uma linguagem de consulta com sintaxe baseada em tubos (|) para encadear comandos. Você pode usá-lo para criar expressões poderosas para analisar seus registros.

Observação: para desbloquear o PPL mais novo commands/functions na OpenSearch versão 2.19, você precisará alterar um sinalizador de recurso nas Ferramentas do OpenSearch Desenvolvedor usando a seguinte consulta (não é necessária para OpenSearch a versão 3.3):

```
PUT /_plugins/_query/settings { "transient" : { "plugins.calcite.enabled" : true } }
```

Encontre os hosts com mais erros

Este exemplo analisa seus registros para determinar os hosts de serviço com o maior número de erros totais.

```
source = my-index |
    where level = "ERROR" |
    stats count() as error_count by host |
    sort -error_count |
    head 5
```

Calcular o tempo médio de solicitação

Este exemplo analisa seus registros para calcular o tempo médio de solicitação para cada código de status no registro.

```
source = my-index |
    stats avg(request_time) by status_code
```

Para obter mais informações sobre o PPL, consulte o manual de referência do PPL em opensearch.org.

Consultando seus registros usando a IA

Este exemplo analisa seus registros para mostrar os erros registrados nos últimos 5 minutos.

```
Show me all of the error logs from the last 5 minutes
```



Consultando seus registros usando SQL

O SQL fornece uma maneira familiar de consultar dados de log.

Este exemplo analisa seus registros para mostrar erros por data e hora.

```
SELECT timestamp, severity_text, body, service_name
FROM opentelemetry_logs
WHERE severity_text = 'ERROR' AND service_name = 'my-service'
ORDER BY timestamp DESC;
```

Para obter mais informações sobre SQL, consulte o manual de referência de SQL em GitHub.

Consultando seus registros usando DQL

O DQL é bom para pesquisa e filtragem rápidas.

Este exemplo analisa seus registros e retorna erros e exceções.

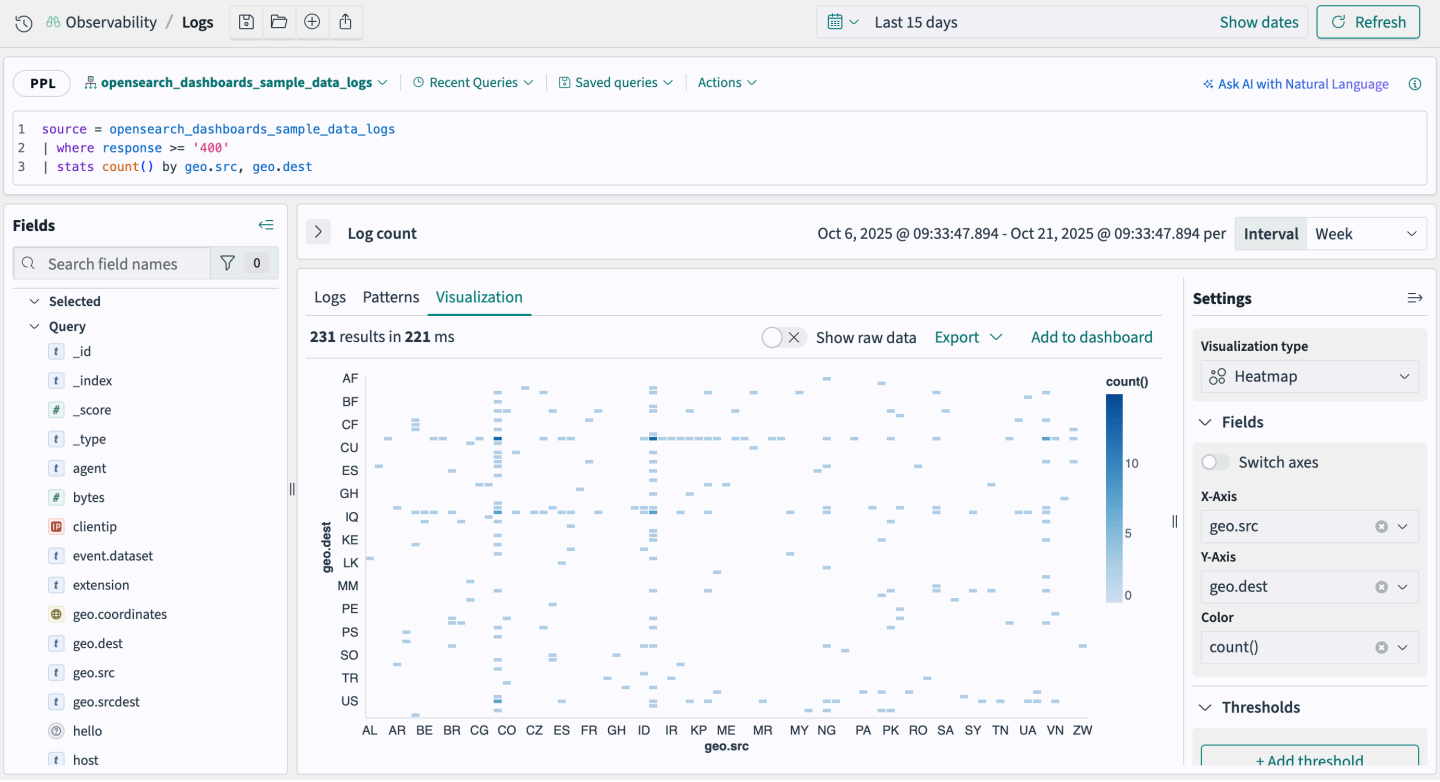
```
error OR exception
```

Para obter mais informações sobre o DQL, consulte o manual de referência do DQL em opensearch.org.

Painéis e alertas para registros

Na nova experiência Discover com PPL, você pode criar visualizações na guia de visualizações no Discover. Escolha entre 12 tipos de visualização e edite rapidamente antes de adicioná-los a um

painel. Na antiga experiência do Discover, você navegará até Visualize na navegação à esquerda para criar uma nova visualização e até Painéis para adicionar as visualizações aos seus painéis.



Você pode definir monitores de alerta usando PPL ou a DSL OpenSearch de consulta de serviço para executar consultas agendadas. Uma condição de gatilho, como um número específico de registros de erros, dispara um alerta. Você pode enviar notificações por meio de canais como o Amazon Simple Notification Service ou webhooks.

Para obter mais informações sobre alertas, consulte a documentação de alertas em opensearch.org.

Rastreamentos

O Amazon OpenSearch Service fornece recursos abrangentes de rastreamento distribuído que ajudam você a entender o desempenho do aplicativo e diagnosticar problemas em toda a sua arquitetura de microsserviços. Ao ingerir OpenTelemetry (OTel) dados de rastreamento com o OpenSearch Ingestion, o OpenSearch Service processa e estrutura automaticamente suas informações de telemetria, oferecendo end-to-end visibilidade dos fluxos de solicitações por meio de seus sistemas distribuídos.

Processamento e ingestão de dados de rastreamento

OpenSearch A ingestão fornece processadores especializados que normalizam e enriquecem seus dados de rastreamento durante a ingestão, garantindo que sua telemetria siga padrões consistentes e esteja pronta para análise. Os principais processadores para dados de rastreamento incluem:

- `service_map`— cria automaticamente gráficos de dependência de serviços a partir de relacionamentos abrangentes, mostrando como as solicitações fluem entre os serviços.
- `trace_group`— agrega extensões relacionadas em grupos de rastreamento lógico com base nos atributos da extensão de entrada, como método e caminho HTTP.
- `otel_trace_raw`— processa dados de OpenTelemetry rastreamento brutos e extrai atributos de amplitude, atributos de recursos e informações do escopo da instrumentação em campos pesquisáveis.

OpenSearch UI e espaço de trabalho de observabilidade

Depois que seus dados de rastreamento são ingeridos no Amazon OpenSearch Service, você usa as ferramentas fornecidas pelo espaço de trabalho de observabilidade do Amazon OpenSearch Service na OpenSearch interface do usuário para analisá-los. O espaço de trabalho de observabilidade fornece visualizações especializadas e ferramentas de análise projetadas para ajudá-lo a entender o desempenho do serviço, identificar gargalos e solucionar problemas em sua arquitetura distribuída.

O espaço de trabalho de observabilidade inclui uma visualização de Serviços que exibe métricas RED (taxa, taxa de erro, duração) para todos os serviços instrumentados, junto com um mapa de serviço interativo mostrando dependências e padrões de comunicação. A visualização Traces permite que você pesquise traços específicos usando trace IDs ou span e, em seguida IDs, se aprofunde em gráficos detalhados em cascata e análises de extensão para entender a jornada completa da solicitação em seu sistema.

Recursos principais

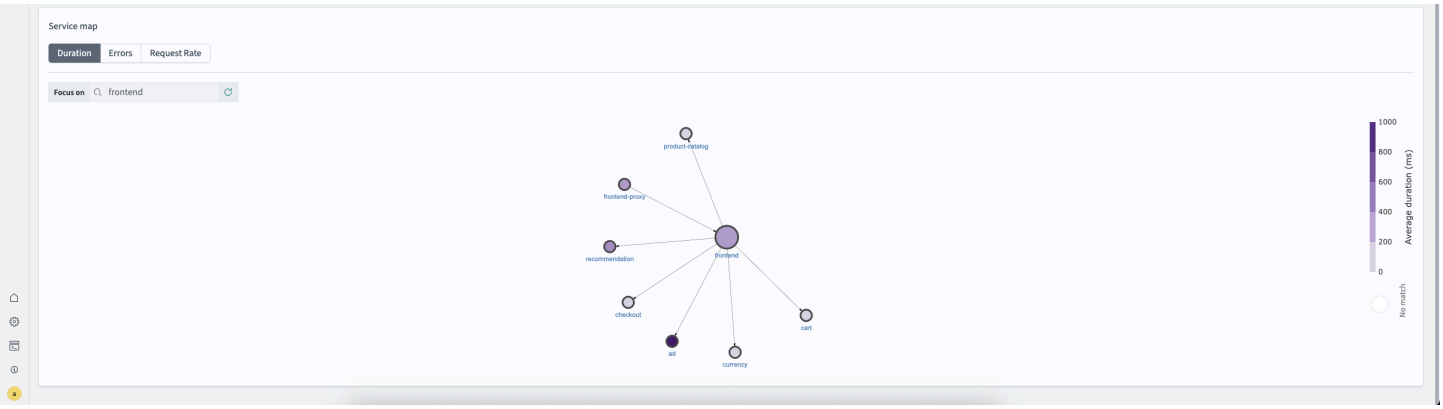
Visualização de serviços

A visualização Serviços fornece uma visão geral abrangente da integridade e do desempenho do seu aplicativo por meio de:

- Painel de métricas RED — Monitore a taxa (solicitações por segundo), a taxa de erro (porcentagem de solicitações com falha) e a duração (percentis de latência) de cada serviço em

seu sistema distribuído. Essas métricas fornecem uma visão imediata da integridade do serviço e ajudam a identificar rapidamente a degradação do desempenho.

- Mapa de serviços interativo — Visualize como seus serviços se comunicam entre si por meio de um gráfico de dependências gerado automaticamente. O mapa de serviços mostra os fluxos de solicitações entre os serviços, ajudando você a entender a arquitetura do sistema e identificar gargalos ou falhas em cascata.
- Indicadores de integridade do serviço — identifique rapidamente serviços problemáticos com base nas taxas de erro e nos limites de latência. Os serviços são codificados por cores para destacar aqueles que precisam de atenção imediata, facilitando a priorização dos esforços de solução de problemas.
- Diálogo de correlação de serviços — faça uma análise detalhada de qualquer serviço para analisar registros e rastreamentos relacionados. Essa visão integrada conecta métricas de nível de serviço com dados de rastreamento detalhados e entradas de registro associadas, permitindo uma análise mais rápida da causa raiz sem alternar entre ferramentas diferentes.



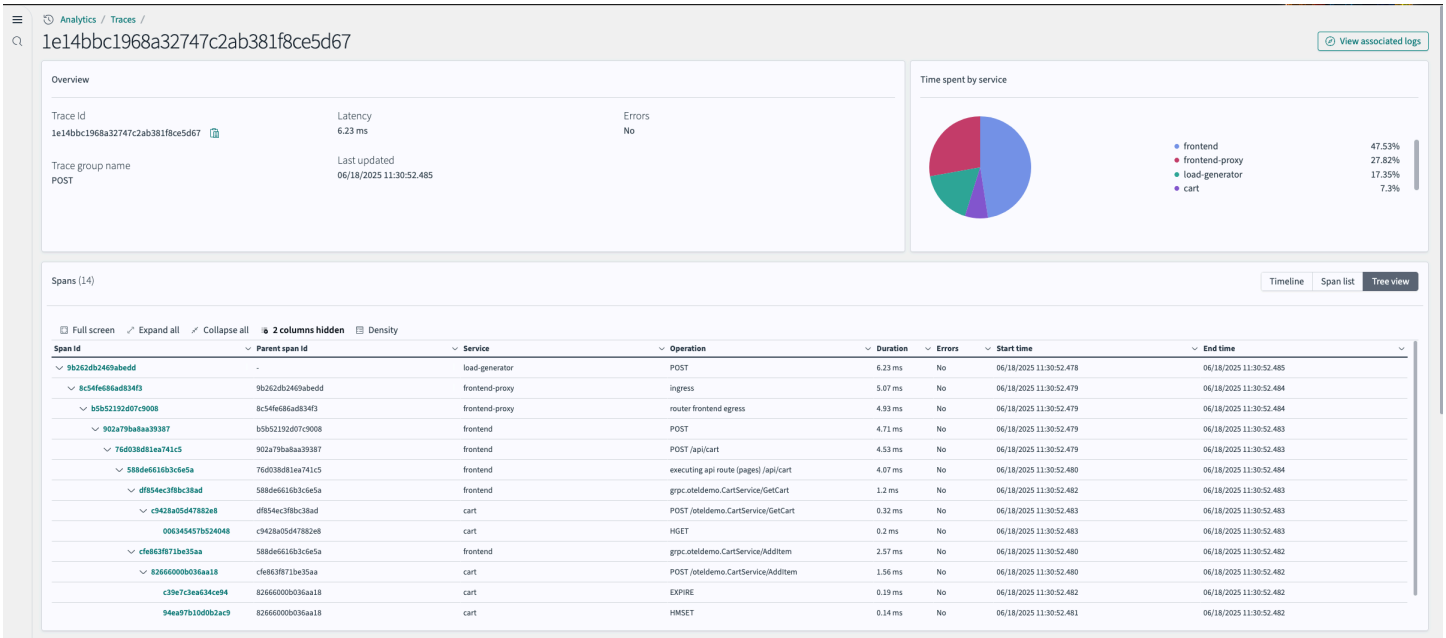
Visualização de traços

A visualização do Traces permite uma investigação profunda de solicitações individuais por meio de seu sistema distribuído:

- Agrupamento de rastreamento por método e caminho HTTP — organiza automaticamente os rastreamentos em grupos lógicos com base nos endpoints da API, mostrando métricas agregadas como latência média, taxa de erro e tendências de desempenho ao longo do tempo. Isso ajuda você a identificar quais endpoints estão enfrentando problemas e acompanhar as melhorias de desempenho.
- Pesquisa de ID de rastreamento e ID de extensão — localize rapidamente traços específicos usando identificadores de rastreamento ou identificadores de extensão. Isso é particularmente

útil ao investigar problemas relatados por usuários ou correlacioná-los com registros de erros que contêm contexto de rastreamento.

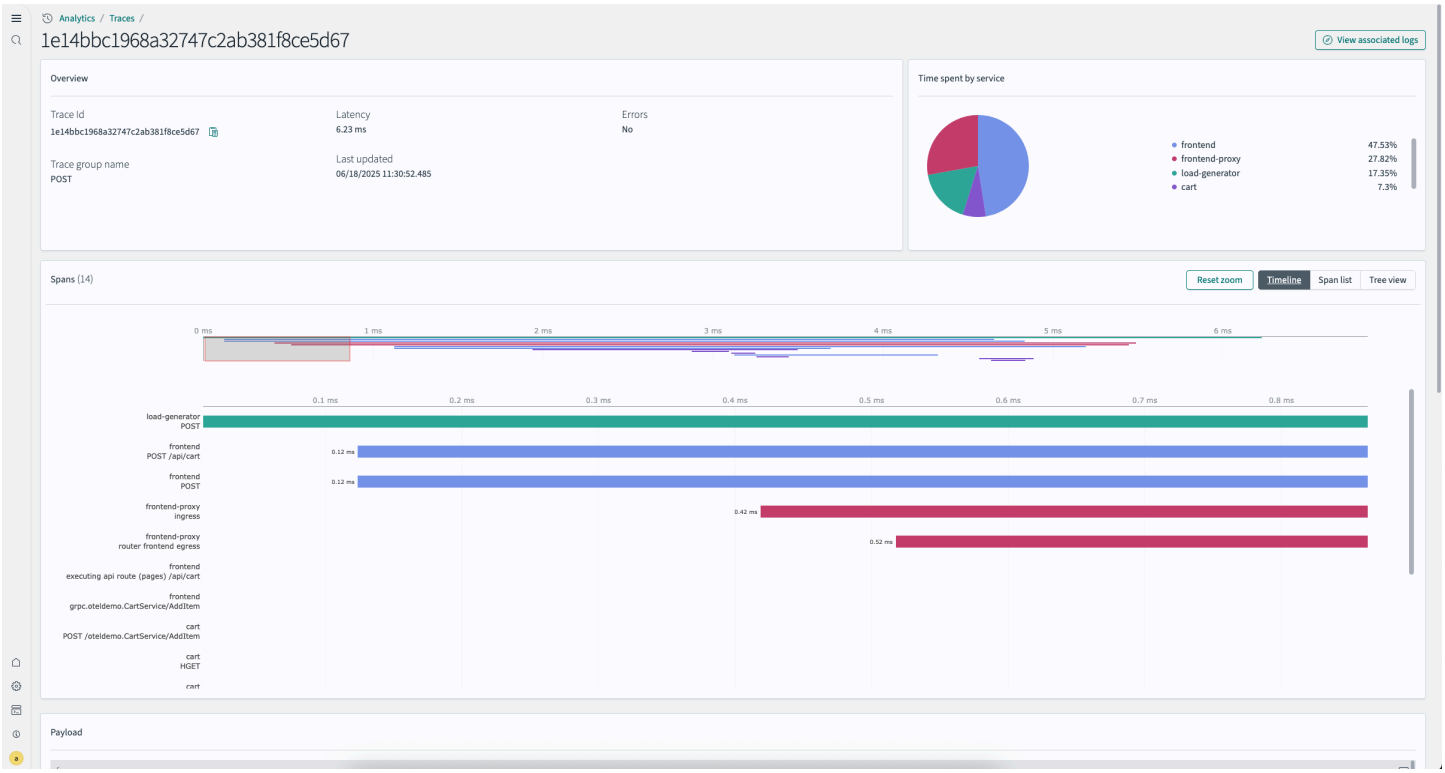
- Gráficos em cascata — Visualize o cronograma completo de uma solicitação à medida que ela flui pelos seus serviços. A visualização em cascata mostra o tempo e a duração do intervalo, facilitando a identificação de operações lentas, processamento sequencial versus processamento paralelo e onde o tempo está sendo gasto em seu sistema distribuído.
- Visualização em árvore com divisão hierárquica da extensão — Navegue pelas relações pai-filho entre as extensões para entender a hierarquia de chamadas em um rastreamento. Essa visualização ajuda você a ver como uma solicitação se ramifica entre os serviços e a identificar quais chamadas de serviço estão contribuindo para a latência geral.
- Painel de registros associados — Visualize os registros que ocorreram durante o mesmo período do rastreamento, filtrados por serviço relevante e contexto de rastreamento. Essa correlação entre rastreamentos e registros melhora significativamente a solução de problemas, fornecendo o fluxo de solicitações e os registros detalhados do aplicativo em uma única interface.



Capacidades avançadas

- Análise de correlação — vincule facilmente rastreamentos, extensões e serviços aos registros correspondentes. O espaço de trabalho de observabilidade correlaciona automaticamente os dados de telemetria usando o contexto de rastreamento, permitindo que você alterne entre diferentes visualizações da mesma solicitação sem perder o contexto.

- Nomes de índice personalizados e suporte entre clusters — Configure o OpenSearch serviço para ler dados de rastreamento de padrões de índice personalizados ou em vários OpenSearch clusters. Essa flexibilidade suporta cenários complexos de implantação e permite que você organize seus dados de telemetria de acordo com suas necessidades operacionais.
- Limites configuráveis do mapa de serviços — ajuste o número de serviços e conexões exibidos no mapa de serviços para lidar com topologias de grande escala. Para sistemas com centenas de serviços, você pode filtrar o mapa para se concentrar em subconjuntos de serviços específicos ou ajustar os limites de renderização para manter o desempenho.
- Navegação em minimapa para gráficos de Gantt — Navegue por gráficos em cascata grandes de forma eficiente usando a visão geral do minimapa. Esse recurso é especialmente útil ao analisar traços com muitos intervalos, permitindo que você pule rapidamente para diferentes seções da linha do tempo.



O Traces fornece at-a-glance visibilidade do desempenho do aplicativo com base nos dados do protocolo OpenTelemetry (OTel). Ele ajuda você a entender como as solicitações fluem pelo seu sistema distribuído, rastreando sua end-to-end jornada entre os serviços.

Melhores práticas operacionais para o Amazon OpenSearch Service

Este capítulo fornece as melhores práticas para operar domínios do Amazon OpenSearch Service e inclui diretrizes gerais que se aplicam a muitos casos de uso. Cada workload é única e tem características particulares, portanto, nenhuma recomendação genérica é exatamente certa para cada caso de uso. A prática recomendada mais importante é implantar, testar e ajustar seus domínios em um ciclo contínuo para encontrar a configuração, a estabilidade e o custo ideais para a workload.

Tópicos

- Monitoramento e alertas
- Estratégia de fragmentação
- Estabilidade
- desempenho
- Segurança
- Otimização de custos
- CloudWatch Alarmes recomendados para o Amazon Service OpenSearch
- Dimensionamento de domínios do Amazon OpenSearch Service
- Escala de petabytes no Amazon Service OpenSearch
- Nódulos de coordenação dedicados no Amazon OpenSearch Service
- Nodes mestres dedicados no Amazon OpenSearch Service

Monitoramento e alertas

As práticas recomendadas a seguir se aplicam ao monitoramento de seus domínios OpenSearch de serviço.

Configurar CloudWatch alarmes

OpenSearch O serviço emite métricas de desempenho para a Amazon CloudWatch. Analise regularmente as métricas do cluster e da instância e configure CloudWatch os alarmes recomendados com base no desempenho da sua carga de trabalho.

Habilitar a publicação de logs

OpenSearch O serviço expõe registros OpenSearch de erros, pesquisa registros lentos, indexação de registros lentos e registros de auditoria no Amazon CloudWatch Logs. Os logs lentos de pesquisa, logs lentos de indexação e logs de erros são úteis para solucionar problemas de performance e estabilidade. Os logs de auditoria, que estarão disponíveis apenas se você habilitar o controle de acesso detalhado para rastrear a atividade do usuário. Para obter mais informações, consulte Registros na OpenSearch documentação.

Logs lentos de pesquisa e logs lentos de indexação são ferramentas importantes para que você entenda e solucione problemas relacionados à performance de suas operações de pesquisa e indexação. Habilite a entrega de logs de lentidão de pesquisa e indexação para todos os domínios de produção. Você também deve configurar limites de registro — caso contrário, CloudWatch não capturará os registros.

Estratégia de fragmentação

Os fragmentos distribuem sua carga de trabalho pelos nós de dados em seu domínio OpenSearch de serviço. Índices configurados corretamente podem auxiliar no aumento da performance geral do domínio.

Quando você envia dados para o OpenSearch Serviço, você envia esses dados para um índice. Um índice é semelhante a uma tabela de banco de dados, com documentos como linhas e campos como colunas. Ao criar o índice, você OpenSearch informa quantos fragmentos primários deseja criar. Os fragmentos primários são partições independentes do conjunto de dados completo. OpenSearch O serviço distribui automaticamente seus dados pelos fragmentos primários em um índice. Você também pode configurar réplicas do índice. Cada fragmento de réplica compreende um conjunto completo de cópias dos fragmentos primários desse índice.

OpenSearch O serviço mapeia os fragmentos de cada índice nos nós de dados do seu cluster. Ele garante que os fragmentos primários e de réplica do índice sejam inerentes a nós de dados diferentes. A primeira réplica garante que você tenha duas cópias dos dados no índice. Você sempre deve usar pelo menos uma réplica. Réplicas adicionais fornecem redundância e capacidade de leitura adicionais.

OpenSearch envia solicitações de indexação para todos os nós de dados que contêm fragmentos que pertencem ao índice. Ele envia solicitações de indexação primeiro para nós de dados que contenham fragmentos primários e depois para nós de dados que contenham fragmentos de réplica.

As solicitações de pesquisa são encaminhadas pelo nó coordenador para um fragmento primário ou de réplica para todos os fragmentos pertencentes ao índice.

Por exemplo, para um índice com cinco fragmentos primários e uma réplica, cada solicitação de indexação toca em dez fragmentos. Por outro lado, as solicitações de pesquisa são enviadas para n fragmentos, onde n é o número de fragmentos primários. Para um índice com cinco fragmentos primários e uma réplica, cada consulta de pesquisa toca em cinco fragmentos (primários ou de réplica) desse índice.

Determinar as contagens de fragmentos e de nós de dados

Use as seguintes práticas recomendadas para determinar contagens de fragmentos e nós de dados para seu domínio.

Tamanho do fragmento: o tamanho dos dados no disco é um resultado direto do tamanho dos dados de origem e se altera à medida que você indexa mais dados. A source-to-index proporção pode variar muito, de 1:10 a 10:1 ou mais, mas geralmente é em torno de 1:1,10. Você pode usar essa proporção para prever o tamanho do índice no disco. Você pode indexar alguns dados e recuperar os tamanhos reais do índice para determinar a proporção de sua workload. Após prever um tamanho de índice, defina uma contagem de fragmentos de modo que cada fragmento tenha entre 10 e 30 GiB (para workloads de pesquisa) ou entre 30 e 50 GiB (para workloads de logs). O máximo deve ser 50 GiB; não se esqueça de planejar o crescimento.

Contagem de fragmentos: a distribuição de fragmentos para nós de dados tem um grande impacto na performance de um domínio. Quando tiver índices com vários fragmentos, tente fazer com que o número de fragmentos seja um múltiplo do número de nós de dados. Isso ajuda a garantir que os fragmentos sejam distribuídos uniformemente entre os nós de dados e evita os nós quentes. Por exemplo, se você tiver 12 fragmentos primários, sua contagem de nós de dados deverá ser 2, 3, 4, 6 ou 12. No entanto, a contagem de fragmentos é secundária ao tamanho do fragmento. Se você tiver 5 GiB de dados, ainda deverá usar um único fragmento.

Fragmentos por nó de dados: o número total de fragmentos que um nó pode conter é proporcional à memória heap da máquina virtual Java (JVM) do nó. Busque 25 fragmentos ou menos para cada GiB de memória heap. Por exemplo, um nó com 32 GiB de memória heap não deve conter mais de 800 fragmentos. Embora a distribuição de fragmentos possa variar de acordo com seus padrões de carga de trabalho, há um limite de 1.000 fragmentos por nó para o Elasticsearch e de OpenSearch 1,1 a 2,15 e 4.000 para 2,17 e superior. OpenSearch A API cat/allocation fornece uma visualização rápida do número de fragmentos e do armazenamento total de fragmentos nos nós de dados.

Proporção de fragmentos para CPU: quando um fragmento está envolvido em uma solicitação de indexação ou pesquisa, ele utiliza uma vCPU para processar a solicitação. Como prática recomendada, use um ponto de escala inicial de 1,5 vCPU por fragmento. Se o tipo de instância tiver 8 vCPUs, defina a contagem de nós de dados para que cada nó tenha no máximo seis fragmentos. Observe que isso é uma aproximação. Certifique-se de testar sua workload e escalar seu cluster adequadamente.

Para obter recomendações sobre volume de armazenamento, tamanho do fragmento e tipo de instância, consulte os seguintes recursos:

- the section called “Dimensionamento de domínios”
- the section called “Escala de petabytes”

Evitar distorções de armazenamento

A distorção de armazenamento ocorre quando um ou mais nós de um cluster mantêm uma proporção maior de armazenamento para um ou mais índices do que para outros. Podem indicar distorções de armazenamento: utilização de CPU desigual, latência intermitente e desigual e enfileiramento desigual entre nós de dados. Para determinar se você tem problemas de distorção, consulte as seguintes seções de resolução de problemas:

- the section called “Distorção de armazenamento e de fragmentos do nó”
- the section called “Distorção de armazenamento e de fragmentos de índices”

Estabilidade

As melhores práticas a seguir se aplicam à manutenção de um domínio de OpenSearch serviço estável e íntegro.

Mantenha-se atualizado com OpenSearch

Atualizações de software de serviço

OpenSearch O serviço lança regularmente atualizações de software que adicionam recursos ou melhoram seus domínios. As atualizações não alteram a versão do mecanismo OpenSearch ou do Elasticsearch. Recomendamos que você agende um horário recorrente para executar a operação da DescribeDomainAPI e inicie uma atualização do software de serviço, se for o `UpdateStatus`

caso. `ELIGIBLE` Se você não atualizar seu domínio dentro de um determinado período de tempo (normalmente duas semanas), o OpenSearch Serviço executará a atualização automaticamente.

OpenSearch atualizações de versão

OpenSearch O serviço adiciona regularmente suporte para versões mantidas pela comunidade do. OpenSearch Sempre atualize para as OpenSearch versões mais recentes quando elas estiverem disponíveis.

OpenSearch O serviço atualiza simultaneamente os OpenSearch OpenSearch painéis (ou o Elasticsearch e o Kibana, se seu domínio estiver executando um mecanismo legado). Se o cluster tiver nós principais dedicados, as atualizações serão concluídas sem tempo de inatividade. Caso contrário, o cluster pode não responder por vários segundos após a atualização enquanto elege um nó principal. OpenSearch Os painéis podem estar indisponíveis durante parte ou toda a atualização.

Há duas maneiras de atualizar um domínio:

- Atualização no local: esta opção é mais fácil porque você mantém o mesmo cluster.
- Atualização de snapshot/restauração: esta opção serve para testar novas versões em um novo cluster ou realizar migrações entre clusters.

Independentemente do processo de atualização usado, recomendamos manter um domínio exclusivamente para desenvolvimento e teste e atualizá-lo para a nova versão antes de atualizar o domínio de produção. Escolha Desenvolvimento e teste como tipo de implantação ao criar o domínio de teste. Certifique-se de atualizar todos os clientes para versões compatíveis imediatamente após a atualização do domínio.

Melhore a performance do snapshot

Para evitar que seu snapshot fique preso no processamento, o tipo de instância do nó principal dedicado deve corresponder à contagem de fragmentos. Para saber mais, consulte the section called “Escolher tipos de instâncias para nós principais dedicados”. Além disso, cada nó não deve ter mais de 25 fragmentos recomendados por GiB de memória heap de Java. Para saber mais, consulte the section called “Como escolher o número de fragmentos”.

Habilite nós principais dedicados

Os nós principais dedicados melhoram a estabilidade do cluster. Um nó principal dedicado executa tarefas de gerenciamento de cluster, mas não retém dados de índice nem responde a solicitações de

clientes. Essa transferência de tarefas de gerenciamento de cluster aumenta a estabilidade de seu domínio e possibilita que algumas alterações de configuração ocorram sem tempo de inatividade.

Habilite e use três nós principais dedicados para obter estabilidade de domínio ideal em três zonas de disponibilidade. A implantação com multi-AZ com modo de espera configura três nós principais dedicados para você. Para obter recomendações sobre tipos de instâncias, consulte the section called “Escolher tipos de instâncias para nós principais dedicados”.

Implantar em diversas zonas de disponibilidade

Para evitar a perda de dados e minimizar o tempo de inatividade do cluster em caso de interrupção do serviço, você pode distribuir nós em duas ou três zonas de disponibilidade na mesma Região da AWS. A melhor prática é implantar o multi-AZ com modo de espera, que configura três zonas de disponibilidade, com duas zonas ativas e uma atuando em espera, e com dois fragmentos de réplica por índice. Essa configuração permite que o OpenSearch Service distribua fragmentos de réplica para fragmentos AZs diferentes dos principais correspondentes. Não há cobranças de transferência de dados entre AZs para comunicações de cluster entre zonas de disponibilidade.

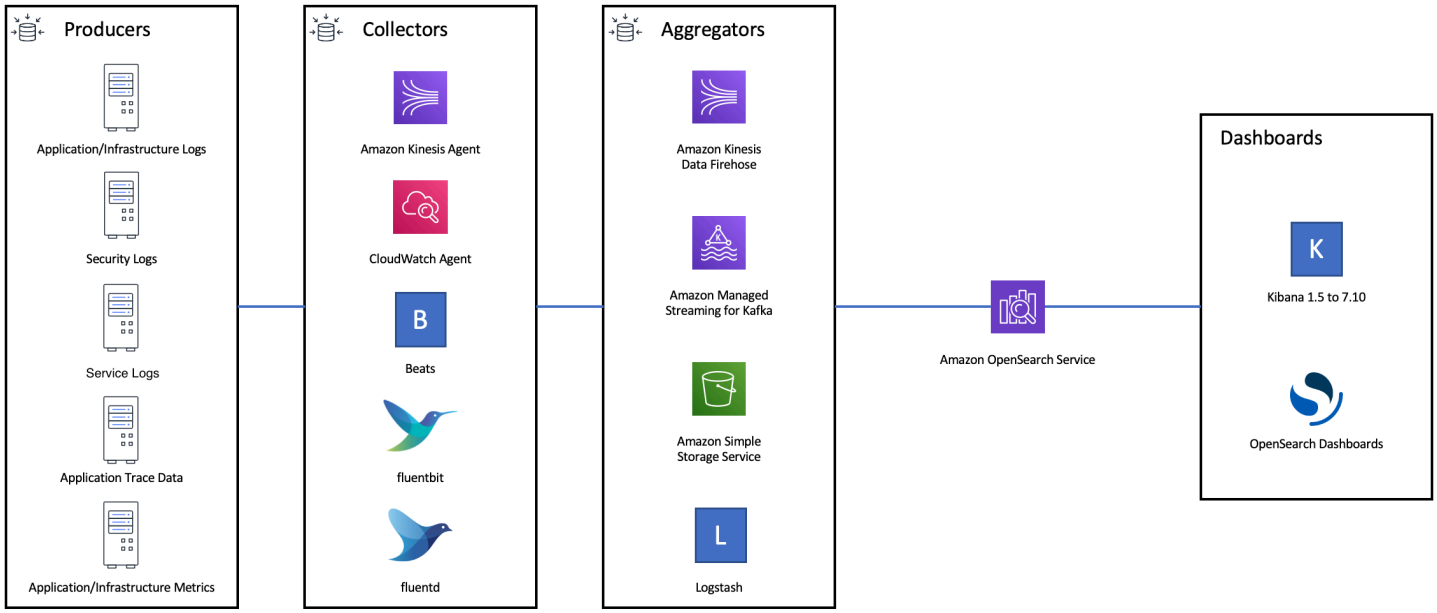
As zonas de disponibilidade são vários locais isolados dentro de cada região da . Com uma configuração de duas AZ (zonas de disponibilidade), perder uma zona de disponibilidade significa que você perde metade de toda a capacidade do domínio. A mudança para três zonas de disponibilidade reduz ainda mais o impacto da perda de uma única zona de disponibilidade.

Controlar o fluxo de ingestão e o armazenamento em buffer

Recomendamos limitar a contagem geral de solicitações usando a operação de API _bulk. É mais eficiente enviar uma solicitação `_bulk` contendo 5 mil documentos do que enviar 5 mil solicitações contendo um único documento.

Para uma estabilidade operacional ideal, às vezes é necessário limitar ou até mesmo pausar o fluxo de envio de informação de solicitações de indexação. Limitar a taxa de solicitações de indexação é um mecanismo importante para lidar com picos inesperados ou ocasionais nas solicitações que poderiam sobrecarregar o cluster. Considere a criação de um mecanismo de controle de fluxo em sua arquitetura de envio de informação.

O diagrama a seguir mostra diversas opções de componentes para uma arquitetura de ingestão de log. Configure a camada de agregação para permitir espaço suficiente para armazenar em buffer os dados de entrada para picos de tráfego repentinos e breve manutenção de domínio.



Criar mapeamentos para workloads de pesquisa

Para cargas de trabalho de pesquisa, crie mapeamentos que definam como OpenSearch armazena e indexa documentos e seus campos. Defina `dynamic` como `strict` para evitar adicionar novos campos acidentalmente.

```
PUT my-index
{
  "mappings": {
    "dynamic": "strict",
    "properties": {
      "title": { "type" : "text" },
      "author": { "type" : "integer" },
      "year": { "type" : "text" }
    }
  }
}
```

Usar modelos de índice

Você pode usar um modelo de índice como forma de saber OpenSearch como configurar um índice quando ele é criado. Configure modelos de índice antes de criar índices. Então, quando você cria um índice, ele herda as configurações e mapeamentos do modelo. É possível aplicar mais de um modelo a um único índice, assim você pode especificar configurações em um modelo e mapeamentos em

outro. Essa estratégia permite o uso de um modelo para configurações comuns em diversos índices e modelos separados para configurações e mapeamentos mais específicos.

As configurações a seguir são úteis para configurar em modelos:

- Número de fragmentos primários e de réplica
- Intervalo de atualização (com que frequência atualizar e realizar alterações recentes no índice disponível para pesquisa)
- Controle de mapeamento dinâmico
- Mapeamentos de campos explícitos

O modelo de exemplo a seguir contém cada uma destas configurações:

```
{
   "index_patterns":[
      "index-*"
   ],
   "order": 0,
   "settings": {
      "index": {
         "number_of_shards": 3,
         "number_of_replicas": 1,
         "refresh_interval": "60s"
      }
   },
   "mappings": {
      "dynamic": false,
      "properties": {
         "field_name1": {
            "type": "keyword"
         }
      }
   }
}
```

Mesmo que raramente mudem, ter configurações e mapeamentos definidos centralmente OpenSearch é mais simples de gerenciar do que atualizar vários clientes upstream.

Gerenciar índices com o Index State Management

Se você estiver gerenciando logs ou dados de séries temporais, recomendamos usar o ISM – Gerenciamento de estados de índice. O ISM permite automatizar tarefas regulares de gerenciamento do ciclo de vida do índice. Com o ISM, você pode criar políticas que invocam sobreposições de alias de índice, obter snapshots de índices, mover índices entre camadas de armazenamento e excluir índices antigos. Você pode até mesmo usar a operação de sobreposição do ISM como uma estratégia alternativa de gerenciamento do ciclo de vida dos dados para evitar distorções de fragmentos.

Primeiro, configure uma política de ISM. Por exemplo, consulte the section called “Políticas de exemplo”. Em seguida, anexe a política a um ou mais índices. Se você incluir um campo de modelo ISM na política, o OpenSearch Service aplicará automaticamente a política a qualquer índice que corresponda ao padrão especificado.

Remover índices não utilizados

Revise regularmente os índices em seu cluster e identifique os que não estão em uso. Obtenha um snapshot desses índices para que sejam armazenados no S3 e depois exclua-os. Ao remover os índices não utilizados, você reduz a contagem de fragmentos e permite uma distribuição mais equilibrada de armazenamento e utilização de recursos entre os nós. Mesmo quando estão ociosos, os índices consomem alguns recursos durante as atividades internas de manutenção do índice.

Em vez de excluir manualmente os índices não utilizados, você pode usar o ISM para obter automaticamente um snapshot e excluir índices após um determinado período.

Usar vários domínios para alta disponibilidade

Para alcançar alta disponibilidade além de 99,9% de tempo de atividade em várias regiões, considere o uso de dois domínios. Para conjuntos de dados pequenos ou de alteração lenta, você pode configurar a replicação entre clusters para manter um modelo ativo-passivo. Nesse modelo, apenas o domínio principal é gravado, mas é possível ler qualquer domínio. Para conjuntos de dados maiores e dados de alteração rápida, configure a entrega dupla em seu pipeline de ingestão para que todos os dados sejam gravados independentemente em ambos os domínios utilizando um modelo ativo-ativo.

Projete seus aplicativos de envio e recebimento de informação com o failover em mente. Certifique-se de testar o processo de failover junto com outros processos de recuperação de desastres.

desempenho

As práticas recomendadas a seguir se aplicam ao ajuste de seus domínios para obter uma performance ideal.

Otimizar o tamanho e a compactação de solicitações em massa

O dimensionamento em massa depende dos dados, da análise e da configuração do cluster, mas um bom ponto de partida é de 3 a 5 MiB por solicitação em massa.

Envie solicitações e receba respostas de seus OpenSearch domínios usando a compressão gzip para reduzir o tamanho da carga útil das solicitações e respostas. Você pode usar a compactação gzip com o cliente OpenSearch Python ou incluir os seguintes cabeçalhos do lado do cliente:

- `'Accept-Encoding': 'gzip'`
- `'Content-Encoding': 'gzip'`

Para otimizar os tamanhos das solicitações em massa, comece com um tamanho de solicitação em massa de 3 MiB. Em seguida, aumente aos poucos o tamanho da solicitação até que a performance da indexação deixe de melhorar.

> **Note**
>
> Para habilitar a compactação gzip em domínios que executam o Elasticsearch versão 6.x, você deve definir `http_compression.enabled` no nível do cluster. Essa configuração é verdadeira por padrão nas versões 7.x do Elasticsearch e em todas as versões do. OpenSearch

Reduzir o tamanho das respostas de solicitações em massa

Para reduzir o tamanho das OpenSearch respostas, exclua campos desnecessários com o `filter_path` parâmetro. Verifique se não filtrou os campos necessários para identificar ou repetir as solicitações com falha. Para ter mais informações e exemplos, consulte the section called “Redução do tamanho da resposta”.

Ajustar os intervalos de atualização

OpenSearch os índices têm uma eventual consistência de leitura. Uma operação de atualização disponibiliza todas as atualizações executadas em um índice para pesquisa. O intervalo de atualização padrão é de um segundo, o que significa que ele OpenSearch executa uma atualização a cada segundo enquanto um índice está sendo gravado.

Quanto menor a frequência com que você atualizar um índice (maior intervalo de atualização), melhor será a performance geral da indexação. A desvantagem de aumentar o intervalo de atualização é que há um atraso maior entre uma atualização de índice e quando os novos dados estão disponíveis para pesquisa. Defina o intervalo de atualização mais alto possível para melhorar a performance geral.

Recomendamos definir o parâmetro `refresh_interval` de todos os seus índices para 30 segundos ou mais.

Habilitar o Auto-Tune

O Auto-Tune usa métricas de desempenho e uso do seu OpenSearch cluster para sugerir alterações nos tamanhos das filas, nos tamanhos do cache e nas configurações da máquina virtual Java (JVM) nos seus nós. Essas alterações opcionais melhoram a velocidade e a estabilidade do cluster. Você pode voltar às configurações padrão do OpenSearch Serviço a qualquer momento. O Auto-Tune é habilitado por padrão em novos domínios, a menos que você o desabilite explicitamente.

Recomendamos habilitar o Auto-Tune em todos os domínios e definir uma janela de manutenção recorrente ou revisar periodicamente as recomendações.

Segurança

As práticas recomendadas a seguir se aplicam à proteção de seus domínios.

Habilite o controle de acesso detalhado

O controle de acesso refinado permite controlar quem pode acessar determinados dados em um OpenSearch domínio do Serviço. Comparado ao controle de acesso generalizado, o controle de acesso detalhado fornece a cada cluster, índice, documento e campo sua própria política de acesso especificada. Os critérios de acesso podem ser baseados em vários fatores, como o perfil da pessoa que solicita o acesso e a ação que ela pretende realizar nos dados. Por exemplo, você

pode conceder a um usuário acesso para gravar em um índice, e outro usuário pode receber acesso apenas para ler os dados no índice sem fazer alterações.

O controle de acesso detalhado permite que dados com diferentes requisitos de acesso existam no mesmo espaço de armazenamento sem problemas de segurança ou conformidade.

Recomendamos habilitar o controle de acesso detalhado em seus domínios.

Implantar domínios em uma VPC

Colocar seu domínio de OpenSearch serviço em uma nuvem privada virtual (VPC) ajuda a permitir a comunicação segura entre o OpenSearch serviço e outros serviços dentro da VPC, sem a necessidade de um gateway de internet, dispositivo NAT ou conexão VPN. Todo o tráfego permanece seguro na nuvem. AWS Devido ao seu isolamento lógico, os domínios que residem em uma VPC contam com uma camada adicional de segurança se comparados aos domínios que utilizam endpoints públicos.

Recomendamos que você crie seus domínios em uma VPC.

Aplicar uma política de acesso restritiva

Mesmo que seu domínio esteja implantado em uma VPC, uma prática recomendada é implementar a segurança em camadas. Certifique-se de verificar a configuração de suas políticas de acesso atuais.

Aplique uma política restritiva de acesso baseada em recursos aos seus domínios e siga o princípio do menor privilégio ao conceder acesso à API de configuração e às operações da API. OpenSearch Como regra geral, evite usar o código `"Principal": {"AWS": "*" }` da entidade principal do usuário anônimo em suas políticas de acesso.

No entanto, há algumas situações em que é aceitável usar uma política de acesso aberta, como quando você habilita o controle de acesso detalhado. Uma política de acesso aberta pode permitir que você acesse o domínio nos casos em que a assinatura da solicitação é difícil ou impossível, como de determinados clientes e ferramentas.

Habilite a criptografia em repouso

OpenSearch Os domínios de serviço oferecem criptografia de dados em repouso para ajudar a impedir o acesso não autorizado aos seus dados. A criptografia em repouso usa AWS Key Management Service (AWS KMS) para armazenar e gerenciar suas chaves de criptografia e o algoritmo Advanced Encryption Standard com chaves de 256 bits (AES-256) para realizar a criptografia.

Se seu domínio armazena dados confidenciais, habilite a criptografia de dados em repouso.

Ativar node-to-node criptografia

Node-to-node a criptografia fornece uma camada adicional de segurança além dos recursos de segurança padrão do OpenSearch Serviço. Ele implementa o Transport Layer Security (TLS) para todas as comunicações entre os nós que são provisionados nele. OpenSearch Node-to-nodecriptografia, todos os dados enviados ao seu domínio de OpenSearch serviço por HTTPS permanecem criptografados em trânsito enquanto são distribuídos e replicados entre os nós.

Se o seu domínio armazenar dados confidenciais, ative a node-to-node criptografia.

Monitor com AWS Security Hub CSPM

Monitore seu uso do OpenSearch Serviço no que se refere às melhores práticas de segurança usando AWS Security Hub CSPM. O CSPM do Security Hub usa controles de segurança para avaliar configurações de recursos e padrões de segurança que ajudam você a atender a vários frameworks de conformidade. Para obter mais informações sobre o uso do Security Hub CSPM para avaliar os recursos do OpenSearch Serviço, consulte Amazon OpenSearch Service os controles no Guia do AWS Security Hub Usuário.

Otimização de custos

As melhores práticas a seguir se aplicam à otimização e economia em seus custos OpenSearch de serviço.

Use os tipos de instâncias de última geração

OpenSearch O serviço está sempre adotando novos tipos de instâncias do Amazon EC2 que oferecem melhor desempenho a um custo menor. Recomendamos sempre usar as instâncias de última geração.

Evite usar instâncias T2 ou `t3.small` para domínios de produção, porque elas podem se tornar instáveis sob cargas pesadas sustentadas. As instâncias `r6g.large` são uma opção para pequenas workloads de produção (tanto como nós de dados quanto como nós principais dedicados).

Usar os volumes gp3 do Amazon EBS

OpenSearch os nós de dados exigem armazenamento de baixa latência e alta taxa de transferência para fornecer indexação e consulta rápidas. Ao usar os volumes gp3 do Amazon EBS, você obtém

maior desempenho básico (IOPS e throughput) a um custo 9,6% menor do que com o tipo de volume Amazon EBS gp2 oferecido anteriormente. É possível provisionar IOPS e throughput adicionais, independentemente do tamanho do volume, usando gp3. Esses volumes também são mais estáveis do que os volumes da geração anterior, pois não usam créditos intermitentes. O tipo de volume gp3 também dobra os limites de tamanho do per-data-node volume do tipo de volume gp2. Com esses volumes maiores, você pode reduzir o custo dos dados passivos, aumentando a quantidade de armazenamento por nó de dados.

Uso UltraWarm e armazenamento refrigerado para dados de registro de séries temporais

Se você estiver usando OpenSearch para análise de registros, mova seus dados para UltraWarm um armazenamento refrigerado para reduzir custos. Use o Index State Management (ISM) para migrar dados entre camadas de armazenamento e gerenciar a retenção de dados.

UltraWarmfornece uma maneira econômica de armazenar grandes quantidades de dados somente para leitura no Service. OpenSearch UltraWarm usa o Amazon S3 para armazenamento, o que significa que os dados são imutáveis e somente uma cópia é necessária. Você paga apenas pelo armazenamento que é equivalente ao tamanho dos fragmentos primários dos índices. As latências UltraWarm das consultas aumentam com a quantidade de dados do S3 necessária para atender à consulta. Depois que os dados são armazenados em cache nos nós, as consultas aos UltraWarm índices têm um desempenho semelhante às consultas aos índices ativos.

O armazenamento frio também é apoiado pelo S3. Quando precisar consultar dados frios, você pode anexá-los seletivamente aos UltraWarm nós existentes. Os dados frios incorrem no mesmo custo de armazenamento gerenciado UltraWarm, mas os objetos no armazenamento frio não consomem recursos UltraWarm do nó. Portanto, o armazenamento a frio fornece uma quantidade significativa de capacidade de armazenamento sem afetar o tamanho ou a contagem de UltraWarm nós.

UltraWarm torna-se econômico quando você tem aproximadamente 2,5 TiB de dados para migrar do armazenamento dinâmico. Monitore sua taxa de preenchimento e planeje transferir os índices para UltraWarm antes de atingir esse volume de dados.

Revisar as recomendações para instâncias reservadas

Considere comprar Instâncias Reservadas (RIs) depois de ter uma boa base sobre seu desempenho e consumo de computação. Os descontos começam em torno de 30% para reservas não antecipadas de um ano e podem aumentar em até 50% para todas as reservas antecipadas de três anos.

Depois de observar uma operação estável por pelo menos 14 dias, consulte Acessar recomendações de reserva no Guia do usuário do AWS Cost Management . O título do Amazon OpenSearch Service exibe recomendações específicas de compra de RI e economias projetadas.

CloudWatch Alarmes recomendados para o Amazon Service OpenSearch

CloudWatch os alarmes realizam uma ação quando uma CloudWatch métrica excede um valor especificado por algum tempo. Por exemplo, talvez você queira AWS enviar um e-mail se o status de integridade do seu cluster `red` for superior a um minuto. Esta seção inclui alguns alarmes recomendados para o Amazon OpenSearch Service e como responder a eles.

Você pode implantar automaticamente esses alarmes usando CloudFormation. Para ver uma pilha de amostra, consulte o GitHubrepositório relacionado.

> **Note**
>
> Se você implantar a CloudFormation pilha, os `KMSKeyInaccessible` alarmes `KMSKeyError` e existirão em um `Insufficient Data` estado porque essas métricas só aparecerão se um domínio encontrar um problema com sua chave de criptografia.

Para obter mais informações sobre a configuração de alarmes, consulte Criação de CloudWatchalarmes da Amazon no Guia do usuário da Amazon CloudWatch .

Alarme	Problema
Máximo de `ClusterStatus.red` é >= 1 por 1 minuto, 1 período consecutivo	Pelo menos um fragmento principal e suas réplicas não estão alocados para um nó. Consulte the section called “Status de cluster vermelho”.
O máximo de `ClusterStatus.yellow` é >= um por um minuto,	Pelo menos um fragmento de réplica não está alocado para um nó. Consulte the section called “Status de cluster amarelo”.

Alarme	Problema
cinco vezes consecutivas	
Mínimo de `FreeStorageSpace` é <= 20480 por 1 minuto, 1 período consecutivo	Um nó no seu cluster tem 20 GiB de espaço de armazenamento livre. Consulte the section called “Falta de espaço de armazenamento disponível”. Esse valor é em MiB; portanto, em vez de 20.480, recomendamos defini-lo como 25% do espaço de armazenamento para cada nó.
`ClusterIndexWritesBlocked` é >= 1 por 5 minutos, 1 período consecutivo	O cluster está bloqueando solicitações de gravação. Consulte the section called “ClusterBlockException”.
Mínimo de `Nodes` é < x por 1 dia, 1 período consecutivo	x é o número de nós em seu cluster. Esse alarme indica que pelo menos um nó no cluster permaneceu inacessível por um dia. Consulte the section called “Nós de cluster com falha”.

Máximo de `AutomatedSnapshotFailure` é >= 1 por 1 minuto, 1 período consecutivo	Ocorreu falha em um snapshot automatizado. Essa falha normalmente é o resultado de um status de integridade vermelho do cluster. Consulte the section called “Status de cluster vermelho”.
	
	Para obter um resumo de todos os snapshots automatizados e algumas informações sobre falhas, experimente uma das seguintes solicitações:
	
	`GET domain_endpoint /_snapshot/cs-automated/_all`
	`GET domain_endpoint /_snapshot/cs-automated-enc/_all`

O máximo de `CPUUtilization` ou `WarmCPUUtilization` é >= 80% por 15 minutos, 3 períodos consecutivos	Às vezes, pode ocorrer 100% de utilização da CPU, mas o alto uso sustentado é um problema. Considere o uso de tipos de instância maiores ou a adição de instâncias.

Alarme	Problema
O máximo de `JVMMemory Pressure` é >= 95% por um minuto, três vezes consecutivas	O cluster poderá apresentar erros de memória insuficiente se o uso aumentar. Considere escalar verticalmente. OpenSearch O serviço usa metade da RAM de uma instância para o heap Java, até um tamanho de heap de 32 GiB. Você pode dimensionar instâncias verticalmente até 64 GiB de RAM, sendo que nesse ponto você poderá dimensionar horizontalmente adicionando instâncias.
O máximo de `OldGenJVM MemoryPressure` é >= 80% por um minuto, três vezes consecutivas	
O máximo de `MasterCPU Utilization` é >= 50% por 15 minutos, 3 períodos consecutivos	Considere o uso de tipos de instância maiores para os nós principais dedicados. Devido à sua função na estabilidade do cluster e implantações azuis/verdes, os nós principais dedicados devem ter um uso de CPU menor do que os nós de dados.
O máximo de `MasterJVM MemoryPressure` é >= 95% por um minuto, três vezes consecutivas	
O máximo de `MasterOld GenJVMMem oryPressure` é >= 80% por um minuto, três vezes consecuti vas	

Alarme	Problema
`KMSKeyError` é >= 1 por 1 minuto, 1 período consecutivo	A chave de AWS KMS criptografia usada para criptografar dados em repouso no seu domínio está desativada. Reative-a para restaurar as operações normais. Para saber mais, consulte the section called "Criptografia em repouso".
`KMSKeyInaccessible` é >= 1 por 1 minuto, 1 período consecutivo	A chave de AWS KMS criptografia usada para criptografar dados em repouso em seu domínio foi excluída ou revogou suas concessões ao OpenSearch Serviço. Você não pode recuperar os domínios que estejam nesse estado. Porém, se tiver um snapshot manual, você poderá usá-lo para migrar para um novo domínio. Para saber mais, consulte the section called "Criptografia em repouso".
`shards.active` é >= 30.000 por 1 minuto, 1 período consecutivo	O número total de fragmentos ativos primários e de réplica é maior que 30.000. Talvez você esteja alternando seus índices com muita frequência. Considere usar o ISM para remover índices quando atingirem um período de validade específico.
Alarmes 5*xx* >= 10% de `OpenSearchRequests`	Um ou mais nós de dados podem estar sobrecarregados ou as solicitações não são concluídas dentro do período de tempo limite ocioso. Considere alternar para tipos de instância maiores ou adicionar mais nós ao cluster. Confirme se você está seguindo as práticas recomendadas para arquitetura de fragmentos e clusters.
Máximo de `MasterReachableFromNode` é < 1 por 5 minutos, 1 período consecutivo	Esse alarme indica que o nó principal foi interrompido ou está fora do alcance. Essas falhas geralmente são o resultado de um problema de conectividade de rede ou de AWS dependência.
A média de `ThreadpoolWriteQueue` é >= 100 por 1 minuto, 1 período consecutivo	O cluster está passando por alta simultaneidade de indexação. Revise e controle as solicitações de indexação ou aumente os recursos do cluster.

Alarme	Problema
A média de `Threadpool SearchQueue` é >= 500 por 1 minuto, 1 período consecutivo	O cluster está passando por alta simultaneidade de pesquisa. Avalie a possibilidade de escalar seu cluster. Você também pode aumentar o tamanho da fila de pesquisa, mas aumentá-la excessivamente pode causar erros de falta de memória.
O máximo de `Threadpool SearchQueue` é >= 5.000 por 1 minuto, 1 período consecutivo	
O aumento na SOMA de `Threadpool SearchRejected` é >=1{ math expression DIFF ( )} por um minuto, um período consecutivo	Esses alarmes notificam você sobre problemas de domínio que podem afetar a performance e a estabilidade.
O aumento na SOMA de `Threadpool WriteRejected` é >=1{ math expression DIFF ( )} por um minuto, um período consecutivo	

> **Note**
>
> Se você só desejar visualizar métricas, consulte the section called “Monitoramento de métricas de cluster”.

Outros alarmes que você pode considerar

Considere configurar os seguintes alarmes, dependendo dos recursos do OpenSearch Serviço que você usa regularmente.

Alarme	Problema
`WarmFreeStorageSpace` é >= 10%	Você atingiu 10% do seu total de armazenamento quente gratuito. `WarmFreeStorageSpace` mede a soma do seu espaço de armazenamento quente livre em MiB. UltraWarm usa o Amazon S3 em vez de discos conectados.
`HotToWarmMigrationQueueSize` é >= 20 por 1 minuto, 3 períodos consecutivos	Um grande número de índices está migrando simultaneamente do sistema ativo para o UltraWarm armazenamento. Avalie a possibilidade de escalar seu cluster.
`HotToWarmMigrationSuccessLatency` é >= 1 dia, 1 período consecutivo	Configure este alarme para que você seja notificado se `HotToWarmMigrationSuccessCount` x latência for maior que 24 horas, caso você esteja tentando alterar índices diários.
O máximo de `WarmJVMMemoryPressure` é >= 95% por um minuto, três vezes consecutivas	O cluster poderá apresentar erros de memória insuficiente se o uso aumentar. Considere a escalabilidade vertical. OpenSearch O serviço usa metade da RAM de uma instância para o heap Java, até um tamanho de heap de 32 GiB. Você pode dimensionar instâncias verticalmente até 64 GiB de RAM, sendo que nesse ponto você poderá dimensionar horizontalmente adicionando instâncias.
O máximo de `WarmOldGenJVMMemoryPressure` é >= 80% por um minuto, três vezes consecutivas	

Alarme	Problema
`WarmToColdMigrationQueueSize` é >= 20 por 1 minuto, 3 períodos consecutivos	Um grande número de índices está migrando simultaneamente UltraWarm para o armazenamento refrigerado. Avalie a possibilidade de escalar seu cluster.
`HotToWarmMigrationFailureCount` é >= 1 por 1 minuto, 1 período consecutivo	As migrações podem falhar durante os snapshots, as realocações de fragmentos ou as uniões de força. As falhas durante os snapshots ou as realocações de fragmentos geralmente ocorrem devido a falhas de nós ou a problemas de conectividade do S3. A falta de espaço em disco geralmente é a causa subjacente das falhas de união de força.
`WarmToColdMigrationFailureCount` é >= 1 por 1 minuto, 1 período consecutivo	As migrações geralmente falham quando as tentativas de migrar metadados de índice para o armazenamento frio falham. Também podem ocorrer falhas quando o estado do cluster de índice quente estiver sendo removido.
`WarmToColdMigrationLatency` é >= 1 dia, 1 período consecutivo	Configure este alarme para que você seja notificado se `WarmToColdMigrationSuccessCount` x latência for maior que 24 horas, caso você esteja tentando alterar índices diários.
`AlertingDegraded` é >= 1 por 1 minuto, 1 período consecutivo	O índice de alerta é vermelho ou um ou mais nós não estão na programação.
`ADPluginUnhealthy` é >= 1 por 1 minuto, 1 período consecutivo	O plug-in de detecção de anomalias não está funcionando corretamente, seja por causa de altas taxas de falhas, seja por um dos índices que está sendo usado estar vermelho.

Alarme	Problema
`AsynchronousSearchFailureRate` é >= 1 por 1 minuto, 1 período consecutivo	Pelo menos uma pesquisa assíncrona falhou no último minuto, o que provavelmente significa que o nó coordenador falhou. O ciclo de vida de uma solicitação de pesquisa assíncrona é gerenciado exclusivamente no nó do coordenador, portanto, se o coordenador ficar inativo, a solicitação falhará.
`AsynchronousSearchStoreHealth` é >= 1 por 1 minuto, 1 período consecutivo	A integridade do armazenamento de respostas de pesquisa assíncrona no índice persistido é vermelha. Você pode estar armazenando grandes respostas assíncronas, que podem desestabilizar um cluster. Tente limitar suas respostas de pesquisa assíncronas a 10 MB ou menos.
`SQLUnhealthy` é >= 1 por 1 minuto, 3 períodos consecutivos	O plug-in SQL está retornando 5 códigos de resposta xx ou passando uma consulta DSL inválida para. OpenSearch Solucione o problema das solicitações que os clientes estão fazendo ao plug-in.
`LTRStatus.red` é >= 1 por 1 minuto, 1 período consecutivo	Pelo menos um dos índices necessários para executar o plug-in Learning to Rank tem fragmentos primários ausentes e não está funcional.

Dimensionamento de domínios do Amazon OpenSearch Service

Não existe um método perfeito para dimensionar os domínios do Amazon OpenSearch Service. No entanto, começando com uma compreensão de suas necessidades de armazenamento, do serviço e de OpenSearch si mesmo, você pode fazer uma estimativa inicial fundamentada sobre suas necessidades de hardware. Essa estimativa pode servir como um ponto de partida útil para a maioria dos aspectos mais importantes do dimensionamento de domínios: testá-los com workloads e monitorar sua performance.

Tópicos

- Cálculo de requisitos de armazenamento
- Como escolher o número de fragmentos
- Escolha dos tipos de instância e testes

Cálculo de requisitos de armazenamento

A maioria das OpenSearch cargas de trabalho se enquadra em uma das duas grandes categorias:

- Índice de longa duração: você escreve um código que processa dados em um ou mais OpenSearch índices e, em seguida, atualiza esses índices periodicamente à medida que os dados de origem são alterados. Alguns exemplos comuns são pesquisa de sites, documentos e comércio eletrônico.
- Índices contínuos: os dados fluem de modo contínuo para um conjunto de índices temporários, com um período de indexação e uma janela de retenção (como um conjunto de índices diários que é retido por duas semanas). Alguns exemplos comuns são de analytics de log, processamento de séries temporais e analytics de cliques.

Para workloads de índice de longa duração, você pode examinar os dados de origem no disco e determinar facilmente a quantidade de espaço de armazenamento que eles consomem. Se os dados vierem de várias fontes, basta adicionar essas fontes.

Para índices contínuos, você pode multiplicar o volume de dados gerados durante um período representativo pelo período de retenção. Por exemplo, se você gerar 200 MiB de dados de log por hora, são 4,7 GiB por dia, que é 66 GiB de dados em um determinado momento, se você tiver um período de retenção de duas semanas.

O tamanho de seus dados de origem, no entanto, é apenas um aspecto dos seus requisitos de armazenamento. Também é necessário considerar o seguinte:

- Número de réplicas: cada réplica é uma cópia completa do fragmento primário. O tamanho do armazenamento do índice mostra o tamanho do fragmento primário e da réplica. Por padrão, cada OpenSearch índice tem uma réplica. Recomendamos pelo menos uma para evitar a perda de dados. As réplicas também melhoram a performance da pesquisa, portanto, é aconselhável ter mais réplicas se você tiver uma workload com uso intensivo de leitura. Use `PUT /my-index/_settings` para atualizar a configuração `number_of_replicas` para o seu índice.
- OpenSearch sobrecarga de indexação: o tamanho em disco de um índice varia. O tamanho total dos dados de origem mais o índice geralmente é de 110% da origem, com o índice de até 10% dos dados de origem. Após indexar os dados, é possível usar a API `_cat/indices?v` e o valor `pri.store.size` para calcular a sobrecarga exata. `_cat/allocation?v` também fornece um resumo útil.

- Espaço reservado para o sistema operacional: por padrão, o Linux reserva 5% do sistema de arquivos para o usuário `root` para processos críticos, recuperação do sistema e para se proteger contra problemas ocasionados pela fragmentação do disco.
- OpenSearch Sobrecarga de OpenSearch serviço: o serviço reserva 20% do espaço de armazenamento de cada instância (até 20 GiB) para mesclagens de segmentos, registros e outras operações internas.

  Por causa desse máximo de 20 GiB, a quantidade total de espaço reservado pode variar muito, dependendo do número de instâncias em seu domínio. Por exemplo, um domínio pode ter três instâncias `m6g.xlarge.search`, cada uma com 500 GiB de espaço de armazenamento, para um total de 1,46 TiB. Nesse caso, o total de espaço reservado é apenas 60 GiB. Outro domínio pode ter 10 instâncias `m3.medium.search`, cada uma com 100 GiB de espaço de armazenamento, para um total de 0,98 TiB. Aqui, o total de espaço reservado é 200 GiB, embora o primeiro domínio seja 50% maior.

  Na fórmula a seguir, aplicamos uma estimativa sobre a “pior das hipóteses” para sobrecarga. Essa estimativa inclui espaço livre adicional para ajudar a minimizar o impacto das falhas nos nós e das interrupções da zona de disponibilidade.

Em resumo, se em determinado momento você tiver 66 GiB de dados e quiser uma réplica, seu requisito de armazenamento mínimo será mais próximo de 66 * 2 * 1,1 / 0,95 / 0,8 = 191 GiB. Você pode generalizar esse cálculo da seguinte maneira:

Dados de origem * (1 + número de réplicas) * (1 + sobrecarga de indexação)/(1 - espaço reservado do Linux)/(1 - sobrecarga do OpenSearch serviço) = requisito mínimo de armazenamento

Ou você pode usar esta versão simplificada:

Dados da origem * (1 + número de réplicas) * 1,45 = requisito de armazenamento mínimo

Espaço de armazenamento insuficiente é uma das causas mais comuns da instabilidade do cluster. Portanto, é necessário verificar os números ao escolher tipos de instância, as contagens de instâncias e os volumes de armazenamento.

Existem outras considerações de armazenamento:

- Se o requisito mínimo de armazenamento ultrapassar 1 PB, consulte the section called “Escala de petabytes”.

- Se você tiver índices contínuos e quiser usar uma arquitetura quente-morna, consulte the section called “UltraWarm armazenamento”.

Como escolher o número de fragmentos

Depois de entender os requisitos de armazenamento, você poderá investigar a sua estratégia de indexação. Por padrão, no OpenSearch Serviço, cada índice é dividido em cinco fragmentos principais e uma réplica (total de 10 fragmentos). Esse comportamento difere do código aberto OpenSearch, cujo padrão é um fragmento primário e uma réplica. Como você não pode alterar facilmente o número de fragmentos principais para um índice existente, decida sobre a contagem de fragmentos antes de indexar seu primeiro documento.

O objetivo geral de escolher um número de fragmentos é distribuir um índice de forma uniforme por todos os nós de dados no cluster. No entanto, esses fragmentos não devem ser muito grandes nem muito numerosos. Uma orientação geral é buscar manter o tamanho do fragmento entre 10 e 30 GiB, para workloads em que a latência de pesquisa é um dos principais objetivos de performance, e entre 30 e 50 GiB, para workloads em que há gravação intensa, como analytics de log.

Fragmentos grandes podem dificultar OpenSearch a recuperação de falhas, mas como cada fragmento usa uma certa quantidade de CPU e memória, ter muitos fragmentos pequenos pode causar problemas de desempenho e erros de falta de memória. Em outras palavras, os fragmentos devem ser pequenos o suficiente para que a instância de OpenSearch serviço subjacente possa lidar com eles, mas não tão pequenos que sobrecarreguem desnecessariamente o hardware.

Por exemplo, suponha que você tenha 66 GiB de dados. Você não espera que esse número aumente ao longo do tempo e deseja manter seus fragmentos em torno de 30 GiB cada um. Seu número de fragmentos, portanto, deve ser aproximadamente 66 * 1,1/30 = 3. Você pode generalizar esse cálculo da seguinte maneira:

(Dados da origem + espaço para crescer) * (1 + sobrecarga de indexação) / tamanho desejado do fragmento = número aproximado de fragmentos principais

Essa equação ajuda a compensar o crescimento dos dados ao longo do tempo. Se você espera que os mesmos 66 GiB de dados quadrupliquem nos próximos anos, o número aproximado de fragmentos será (66 + 198) * 1,1/30 = 10. No entanto, lembre-se de que você ainda não tem esses 198 GiB de dados extras. Verifique se essa preparação para o futuro não cria desnecessariamente fragmentos muito pequenos que consomem enormes quantidades de CPU e memória. Nesse caso, 66 * 1,1/10 fragmentos = 7,26 GiB por fragmento, o que consumirá recursos adicionais e está abaixo do intervalo de tamanho recomendado. Você pode considerar a middle-of-the-road abordagem mais

ampla de seis fragmentos, o que deixa você com fragmentos de 12 GiB hoje e fragmentos de 48 GiB no futuro. Em seguida, novamente, você pode preferir começar com três fragmentos e reindexar seus dados quando os fragmentos ultrapassarem 50 GiB.

Um problema muito menos comum envolve limitar o número de fragmentos por nó. Se você dimensionar seus fragmentos adequadamente, normalmente ficará sem espaço em disco muito antes de atingir esse limite. Por exemplo, uma instância `m6g.large.search` tem um tamanho máximo de disco de 512 GiB. Se você ficar abaixo de 80% do uso do disco e dimensionar seus fragmentos em 20 GiB, ela poderá acomodar aproximadamente 20 fragmentos. Elasticsearch 7. x e versões posteriores, e todas as versões de OpenSearch até 2.15, têm um limite de 1.000 fragmentos por nó. Para ajustar o máximo de fragmentos por nó, ajuste a configuração `cluster.max_shards_per_node`. Para a OpenSearch versão 2.17 e versões posteriores, o OpenSearch Service oferece suporte a 1.000 fragmentos para cada 16 GB de memória heap da JVM, até um máximo de 4.000 fragmentos por nó. Para ver um exemplo, consulte Configurações de cluster. Para saber mais sobre número de fragmentos, consulte the section called “Cotas de número de fragmentos”.

Se dimensionar os fragmentos adequadamente, você quase sempre se manterá abaixo desse limite, mas também é possível considerar o número de fragmentos para cada GiB de heap Java. Em um determinado nó, não tenha mais de 25 fragmentos por GiB de heap de Java. Por exemplo, uma instância `m5.large.search` tem um heap de 4 GiB, de modo que cada nó não deva ter mais de 100 fragmentos. Nessa contagem de fragmentos, cada fragmento tem aproximadamente 5 GiB de tamanho, o que está bem abaixo da nossa recomendação.

Escolha dos tipos de instância e testes

Depois de calcular os requisitos de armazenamento e escolher o número de fragmentos de que precisa, você pode começar a tomar decisões quanto ao hardware. Os requisitos de hardware variam drasticamente por workload, mas ainda podemos fazer algumas recomendações básicas.

Em geral, os limites de armazenamento para cada tipo de instância são mapeados para a quantidade de CPU e memória que pode ser necessária para workloads leves. Por exemplo, uma instância `m6g.large.search` tem um tamanho máximo de volume do EBS de 512 GiB, 2 núcleos de vCPUs e 8 GiB de memória. Se o seu cluster tiver muitos fragmentos, executar agregações desgastantes, atualizar os documentos com frequência ou processar um grande número de consultas, esses recursos poderão ser insuficientes para suas necessidades. Se o cluster estiver em uma dessas categorias, tente começar com uma configuração mais próxima de dois núcleos de vCPU e 8 GiB de memória para cada 100 GiB de seu requisito de armazenamento.

> **Tip**
>
> Para obter um resumo dos recursos de hardware alocados para cada tipo de instância, consulte os preços do Amazon OpenSearch Service.

No entanto, até mesmo esses recursos podem ser insuficientes. Alguns OpenSearch usuários relatam que muitas vezes precisam desses recursos para atender às suas necessidades. Para localizar o hardware certo para sua workload, é necessário fazer uma estimativa inicial embasada, testar workloads representativas, ajustar e testar novamente.

Etapa 1: Fazer uma estimativa inicial

Para começar, recomendamos um mínimo de três nós para evitar possíveis OpenSearch problemas, como um estado cerebral dividido (quando um lapso na comunicação leva a um cluster com dois nós principais). Se você tiver três nós principais dedicados, ainda recomendamos um mínimo de dois nós de dados para replicação.

Etapa 2: Calcular os requisitos de armazenamento por nó

Se você tiver um requisito de 184 GiB de armazenamento e o número mínimo recomendado for de três nós, use a equação 184/3 = 61 GiB para determinar a quantidade de armazenamento necessária para cada nó. Nesse exemplo, é possível selecionar três instâncias `m6g.large.search`, em que cada uma usa um volume de armazenamento do EBS de 90 GiB, para que você tenha uma rede de segurança e espaço para crescimento ao longo do tempo. Essa configuração fornece 6 núcleos de vCPU e 24 GiB de memória, portanto, é adequada para workloads mais leves.

Para obter um exemplo mais substancial, considere um requisito de armazenamento de 14 TiB (14.336 GiB) e uma workload pesada. Nesse caso, você pode optar por iniciar testes com 2 * 144 = 288 núcleos de vCPU e 8 * 144 = 1.152 GiB de memória. Esses números funcionam para aproximadamente 18 instâncias do `i3.4xlarge.search`. Se você não precisar do armazenamento local rápido, também poderá testar 18 instâncias `r6g.4xlarge.search`, cada uma usando um volume de armazenamento do EBS de 1 TiB.

Se o cluster incluir centenas de terabytes de dados, consulte the section called “Escala de petabytes”.

Etapa 3: Executar o teste de representatividade

Depois de configurar o cluster, você pode adicionar seus índices usando o número de fragmentos calculados anteriormente, realizar alguns testes representativos do cliente usando um conjunto de dados realista e monitorar CloudWatch métricas para ver como o cluster lida com a carga de trabalho.

Etapa 4: Sucesso ou iteração

Se o desempenho satisfizer suas necessidades, os testes forem bem-sucedidos e CloudWatch as métricas estiverem normais, o cluster estará pronto para uso. Lembre-se de definir CloudWatch alarmes para detectar o uso insalubre de recursos.

Se a performance não for aceitável, os testes falharem ou `CPUUtilization` ou `JVMMemoryPressure` estiverem altas, poderá ser necessário escolher um tipo de instância diferente (ou adicionar instâncias) e continuar o teste. Conforme você adiciona instâncias, reequilibra OpenSearch automaticamente a distribuição dos fragmentos em todo o cluster.

Como é mais fácil medir a capacidade em excesso de um cluster sobrecarregado do que o déficit de um cluster não sobrecarregado, recomendamos começar com um cluster maior do que você imagina ser necessário. Depois, teste e reduza para um cluster eficiente que tenha os recursos adicionais a fim de garantir operações estáveis durante períodos de maior atividade.

Os clusters de produção ou os clusters com estados complexos se beneficiam de nós principais dedicados, que melhoram a performance e a confiabilidade do cluster.

Escala de petabytes no Amazon Service OpenSearch

Os domínios do Amazon OpenSearch Service oferecem armazenamento anexado de até 10 PB. Você pode configurar um domínio com 1.000 tipos de `OR1.16xlarge.search` instância, cada um com 36 TB de armazenamento. Devido à grande diferença em escala, as recomendações para domínios desse tamanho diferem de nossas recomendações gerais. Esta seção descreve as considerações para a criação de domínios, custos, armazenamento e tamanho de fragmento.

Embora esta seção frequentemente faça referência aos tipos de `i3.16xlarge.search` instância, você pode usar vários outros tipos de instância para alcançar 10 PB do armazenamento total do domínio.

Criar domínios

Domínios deste tamanho excedem o limite padrão de 80 instâncias por domínio. Para solicitar um aumento do limite de serviço de até 1.000 instâncias por domínio, abra um caso no AWS Support Center.

Preços

Antes de criar um domínio desse tamanho, verifique a página de preços do Amazon OpenSearch Service para garantir que os custos associados correspondam às suas expectativas. Examine the section called “UltraWarm armazenamento” para ver se uma arquitetura de atividade muito alta é adequada ao seu caso de uso.

Armazenamento

Os tipos de `i3` instância foram projetados para fornecer armazenamento expresso (NVMe) de memória local rápido e não volátil. Como esse armazenamento local tende a oferecer benefícios de desempenho quando comparado ao Amazon Elastic Block Store, os volumes do EBS não são uma opção quando você seleciona esses tipos de instância no OpenSearch Service. Se você preferir o armazenamento do EBS, use outro tipo de instância, como `r6.12xlarge.search`.

Tamanho e contagem de fragmentos

Uma OpenSearch diretriz comum é não exceder 50 GB por fragmento. Considerando o número de fragmentos necessários para acomodar grandes domínios e os recursos disponíveis para instâncias `i3.16xlarge.search`, recomendamos um tamanho de fragmento de 100 GB.

Por exemplo, se você tiver 450 TB de dados de origem e quiser uma réplica, seu requisito de armazenamento mínimo será mais próximo de 450 TB * 2 * 1,1/0.95 = 1.04 PB. Para obter uma explicação sobre esse cálculo, consulte the section called “Cálculo de requisitos de armazenamento”. Embora 1,04 PB / 15 TB = 70 instâncias, você pode selecionar 90 ou mais instâncias `i3.16xlarge.search` para obter uma rede de segurança de armazenamento, lidar com falhas de nós e lidar com alguma variação na quantidade de dados ao longo do tempo. Cada instância adiciona outros 20 GiB ao seu requisito de armazenamento mínimo, ainda que para discos deste tamanho, esses 20 GiB sejam quase insignificantes.

Controlar o número de fragmentos é complicado. OpenSearch os usuários geralmente alternam os índices diariamente e retêm os dados por uma ou duas semanas. Nesta situação, pode ser útil distinguir entre fragmentos “ativos” e “inativos”. fragmentos ativos estão sendo gravados ou lidos ativamente. Fragmentos inativos podem servir para uma solicitação de leitura ocasional, mas são essencialmente ociosos. Em geral, você deve manter o número de fragmentos ativos abaixo de

alguns milhares. À medida que o número de fragmentos ativos se aproxima de 10.000, riscos de performance e de estabilidade consideráveis podem surgir.

Para calcular o número de fragmentos principais, use esta fórmula: 450.000 GB * 1,1/100 GB por fragmento = 4,950 fragmentos. Ao dobrar esse número para contabilizar réplicas, temos 9.900 fragmentos, o que representa uma grande preocupação se todos os fragmentos estão ativos. No entanto, se você alternar índices e apenas 1/7 ou 1/14 dos fragmentos estiver ativo em um determinado dia (1.414 ou 707 fragmentos, respectivamente), o cluster poderá funcionar normalmente. Como sempre, a etapa mais importante do dimensionamento e da configuração do domínio é executar testes de cliente representativos usando um conjunto de dados realista.

Nódulos de coordenação dedicados no Amazon OpenSearch Service

Os nós de coordenação dedicados no Amazon OpenSearch Service são nós especializados que descarregam tarefas de coordenação dos nós de dados. Essas tarefas incluem gerenciar solicitações de pesquisa e hospedar OpenSearch painéis. Separando essas funções, os nós coordenadores dedicados reduzem a carga nos nós de dados, o que permite que eles se concentrem em armazenamento de dados, indexação e operações de pesquisa. Isso melhora a performance do cluster e a utilização de recursos em geral.

Além disso, os nós coordenadores dedicados ajudam a reduzir o número de endereços IP privados necessários para as configurações da VPC, o que resulta em um gerenciamento de rede mais eficiente. Essa configuração pode resultar em um throughput de indexação até 15% melhor e uma performance de consulta 20% melhor, dependendo das características da workload.

Quando usar nós coordenadores dedicados

Os nós coordenadores dedicados são mais benéficos nos cenários a seguir.

- Clusters grandes: em ambientes com alto volume de dados ou consultas complexas, descarregar as tarefas de coordenação para nós dedicados pode melhorar a performance do cluster.
- Consultas frequentes: as workloads que envolvem consultas ou agregações de pesquisa frequentes, especialmente as que têm histogramas de datas complexos ou várias agregações, se beneficiam com o processamento de consultas mais rápido.

- Uso intenso de painéis — OpenSearch Os painéis podem consumir muitos recursos. Descarregar essa responsabilidade para nós coordenadores dedicados reduz a pressão sobre os nós de dados.

Arquitetura e comportamento

Em um OpenSearch cluster, nós coordenadores dedicados lidam com duas responsabilidades principais.

- Tratamento de solicitações: esses nós recebem solicitações de pesquisa e as encaminham para os nós de dados apropriados, que armazenam os dados relevantes. Eles então consolidam os resultados de vários nós de dados em um único conjunto global de resultados, que é devolvido ao cliente.
- Hospedagem de painéis — os nós coordenadores gerenciam os OpenSearch painéis, o que alivia os nós de dados da carga adicional de hospedar OpenSearch painéis e lidar com o tráfego relacionado.

Nos domínios VPC, os nós coordenadores dedicados recebem interfaces de rede elástica (ENIs) em vez de nós de dados. Esse arranjo ajuda a reduzir o número de endereços IP privados necessários VPCs, o que melhora a eficiência da rede. Os nós coordenadores dedicados normalmente representam cerca de 10% do total de nós de dados.

Requisitos e limitações

Os nós coordenadores dedicados têm os seguintes requisitos e limitações.

- Os nós coordenadores dedicados são suportados em todas as OpenSearch versões e nas versões 6.8 a 7.10 do Elasticsearch.
- Para habilitar nós coordenadores dedicados, o domínio deve ter nós principais dedicados habilitados. Para saber mais, consulte the section called “Nós principais dedicados”.
- O provisionamento de nós coordenadores dedicados pode gerar custos adicionais. Porém, a maior eficiência dos recursos e a performance aprimorada justificam o investimento, especialmente para clusters grandes ou complexos.

Provisionar nós coordenadores dedicados

Realize as etapas a seguir para provisionar nós coordenadores dedicados em um domínio existente. Certifique-se de que seu domínio tenha nós principais dedicados habilitados antes de provisionar os nós coordenadores.

Console

Para provisionar nós de coordenadores dedicados no Console de gerenciamento da AWS

1. Faça login no console do Amazon OpenSearch Service em https://console.aws.amazon.com/aos/casa.
2. Escolha Domínios e selecione o domínio que deseja modificar.
3. Na seção Configuração do cluster, selecione Editar.
4. Escolha Habilitar nós coordenadores dedicados.
5. Selecione o tipo de instância e o número de nós coordenadores a serem provisionados.
6. Escolha Salvar alterações. A atualização do domínio pode levar vários minutos.

AWS CLI

Para provisionar nós coordenadores dedicados usando o AWS CLI, use o update-domain-configcomando. O exemplo a seguir provisiona três nós coordenadores `r6g.large.search` em um domínio.

```
aws opensearch update-domain-config \
  --domain-name my-opensearch-domain \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedCoordinatorCount=3,ZoneAwarenessEnabled
```

Esse comando habilita os nós coordenadores dedicados, define o tipo e a quantidade de instâncias para os nós coordenadores, e habilita o reconhecimento de zonas para garantir maior disponibilidade.

Práticas recomendadas

Considere as seguintes práticas recomendadas ao usar nós coordenadores dedicados.

- Use instâncias de uso geral para a maioria dos casos de uso. Elas oferecem uma abordagem equilibrada entre custo e performance. As instâncias otimizadas para memória são ideais para

workloads que exigem recursos substanciais de memória, como as que envolvem agregações complexas ou pesquisas em grande escala.

- Um bom ponto de partida é provisionar de 5% a 10% dos nós de dados como nós coordenadores dedicados. Por exemplo, se o domínio tiver 90 nós de dados de um determinado tipo de instância, considere provisionar de 5 a 9 nós coordenadores do mesmo tipo de instância.

> **Note**
>
> A disponibilidade de tipos de instância pode variar de acordo com a região. Ao selecionar tipos de instância para nós coordenadores, verifique se o tipo de instância escolhido está disponível na região de destino. Você pode verificar a disponibilidade do tipo de instância no console de OpenSearch serviço ao criar ou modificar seu domínio.

- Para minimizar o risco de um ponto de falha único, provisione pelo menos dois nós coordenadores dedicados. Isso garante que o cluster permanecerá operacional mesmo se houver falha em um nó.
- Se for usar pesquisa inter-regional, é recomendável provisionar nós coordenadores dedicados nos domínios de destino. Os domínios de origem normalmente não lidam com tarefas de coordenação.
- Para ambientes com muita indexação, considere o uso de instâncias otimizadas para CPU que correspondam ao tamanho de instância dos nós de dados para garantir a performance ideal.
- Para workloads com uso intenso de memória, use um tipo de instância um pouco maior para os nós coordenadores dedicados para ajudar a gerenciar o aumento das demandas de memória.
- Acompanhe a CloudWatch métrica `CoordinatorCPUUtilization` da Amazon. Se ela exceder 80% consistentemente, isso pode indicar que você precisa de nós coordenadores maiores ou adicionais para lidar com a carga.
- Dimensione os nós coordenadores dedicados para que correspondam aos nós de dados. Por exemplo, comece com nós coordenadores de uso geral 4xlarge quando estiver usando nós de dados 4xlarge.
- Use várias instâncias menores em vez de menos instâncias maiores para nós coordenadores, a menos que suas solicitações ou respostas individuais precisem de memória (in GBs) extremamente alta. Por exemplo, escolha 12 instâncias 4xl em vez de 6 instâncias de uso geral 8xlarge.

Recomendações de nós por tamanho de cluster

Use as diretrizes a seguir como ponto de partida para provisionar nós coordenadores dedicados com base no tamanho do cluster. Ajuste o número e o tipo de nós com base nas características da workload e nas métricas de performance.

Tamanho do cluster	Nós coordenadores recomendados	Tipo de instância
Pequeno (até 50 nós)	3 a 5 nós	Uso geral
Médio (50 a 100 nós)	5 a 9 nós	Otimizado para memória
Grande (mais de 100 nós)	10 a 15 nós	Otimizado para memória

Nodes mestres dedicados no Amazon OpenSearch Service

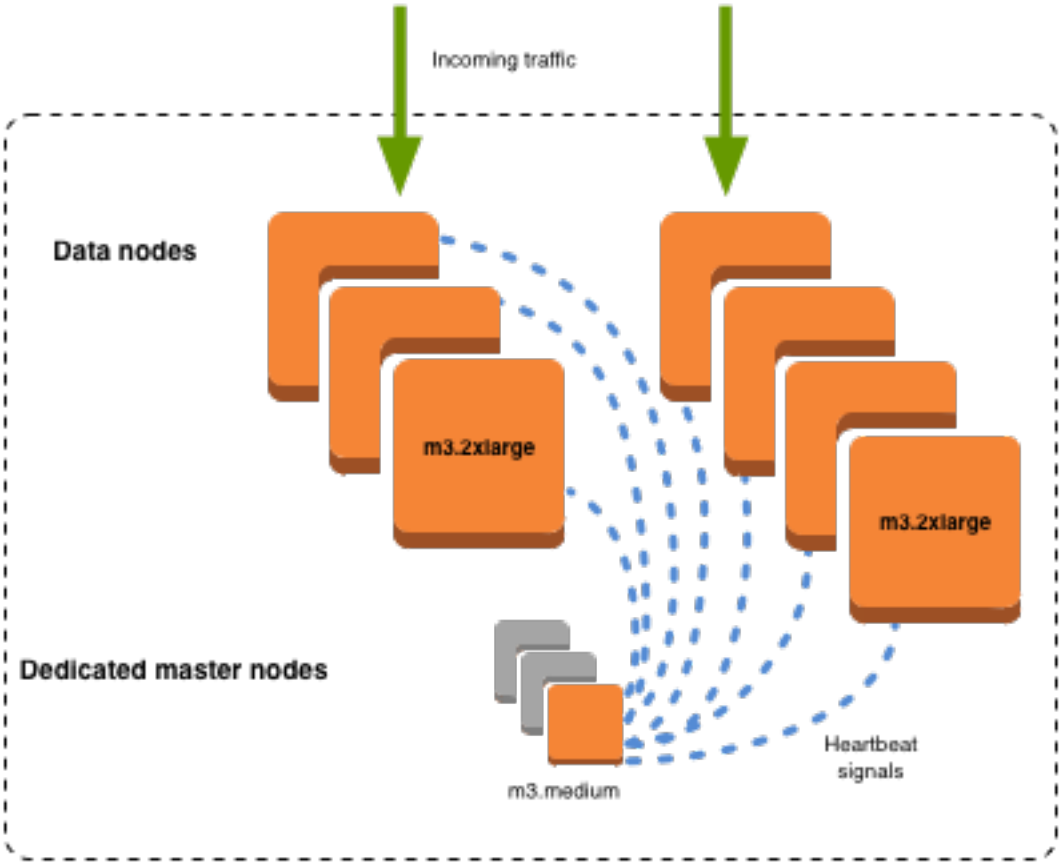
O Amazon OpenSearch Service usa nós mestres dedicados para aumentar a estabilidade do cluster. Um nó principal dedicado executa tarefas de gerenciamento de cluster, mas não mantém dados nem responde a solicitações de carregamento de dados. Essa transferência de tarefas de gerenciamento de cluster aumenta a estabilidade do seu domínio. Assim como todos os outros tipos de nó, você paga uma taxa por hora para cada nó principal dedicado.

Nós principais dedicados executam as seguintes tarefas de gerenciamento de cluster:

- Rastreiam todos os nós no cluster.
- Rastreiam o número de índices no cluster.
- Rastreiam o número de fragmentos pertencentes a cada índice.
- Mantêm informações de roteamento para nós no cluster.
- Atualizam o estado do cluster após alterações de estado, como criação de um índice e adição ou remoção de nós no cluster.
- Replicam alterações no estado do cluster em todos os nós no cluster.

- Monitoram a integridade de todos os nós do cluster, enviando sinais de pulsação, sinais periódicos que monitoram a disponibilidade de nós de dados no cluster.

A ilustração a seguir mostra um domínio OpenSearch de serviço com 10 instâncias. Sete das instâncias são nós de dados e três são nós principais dedicados. Somente um dos nós principais dedicados está ativo. Os dois nós principais dedicados de cor cinza aguardam como backup em caso de falha do nó principal dedicado ativo. Todas as solicitações de carregamento de dados são atendidas por sete nós de dados, e todas as tarefas de gerenciamento de cluster são transferidas para o nó principal dedicado ativo.



Como escolher o número de nós principais dedicados

Recomendamos que você use o Multi-AZ com Standby, que adiciona três nós mestres dedicados a cada domínio do OpenSearch Serviço de produção. Se você implantar com multi-AZ sem modo de espera ou com single-AZ (uma única zona de disponibilidade), ainda recomendamos três nós

principais dedicados. Nunca escolha um número par de nós principais dedicados. Considere o seguinte ao escolher o número de nós principais dedicados:

- Um nó principal dedicado é explicitamente proibido pelo OpenSearch Serviço porque você não tem backup no caso de uma falha. Você receberá uma exceção de validação se tentar criar um domínio com apenas um nó principal dedicado.
- Se você tiver dois nós principais dedicados, seu cluster não terá o quórum necessário de nós para escolher um novo nó principal em caso de falha.

  Um quórum é o número de nós principais dedicados/2+1 (arredondado para o número inteiro mais próximo). Neste caso, 2/2 + 1 = 2. Como um nó principal dedicado falhou e existe apenas um backup, o cluster não tem um quórum e não pode selecionar um novo principal.
- Três nós principais dedicados, o número recomendado, fornecem dois nós de backup em caso de falha de um nó principal e o quórum necessário (2) para selecionar um novo principal.
- Ter quatro nós principais dedicados não é melhor do que ter três, e isso poderá causar problemas se você usar várias zonas de disponibilidade.
  - Se um nó principal falhar, você tem o quórum (3) para escolher um novo principal. Se dois nós falharem, você perderá esse quórum, da mesma forma com três nós principais dedicados.
  - Em uma configuração de três zonas de disponibilidade, duas AZs têm um nó principal dedicado e uma AZ tem dois. Se esse AZ sofrer uma interrupção, os dois restantes AZs não terão o quórum necessário (3) para eleger um novo mestre.
- Ter cinco nós principais dedicados funciona tão bem quanto ter três e permite que você perca dois nós enquanto mantém um quórum. No entanto, como apenas um nó principal dedicado está ativo a qualquer momento, essa configuração significa que você pagará por quatro nós ociosos. Muitos usuários acham esse nível de proteção de failover excessivo.

Se um cluster tiver um número par de nós elegíveis como mestre, OpenSearch e as versões 7 do Elasticsearch. x e posteriores ignoram um nó para que a configuração de votação seja sempre um número ímpar. Nesse caso, quatro nós principais dedicados são essencialmente equivalentes a três (e dois a um).

> **Note**
>
> Se o cluster não tiver o quórum necessário para escolher um novo nó principal, ocorrerão falhas nas solicitações de gravação e leitura para o cluster. Esse comportamento é diferente do OpenSearch padrão.

Escolher tipos de instâncias para nós principais dedicados

OpenSearch Cotas de domínio e instância do serviço

Embora os nós principais dedicados não processem solicitações de pesquisa e consulta, seu tamanho está amplamente correlacionado ao tamanho da instância e ao número de instâncias, índices e fragmentos que podem gerenciar. Para clusters de produção, recomendamos pelo menos os seguintes tipos de instâncias para nós principais dedicados.

Essas recomendações se baseiam em workloads usuais e podem variar de acordo com suas necessidades. Clusters com muitos fragmentos ou mapeamentos de campo podem se beneficiar de tipos de instância maiores. Para obter mais informações, consulte CloudWatch Alarmes recomendados para o Amazon OpenSearch Service para determinar se você precisa usar um tipo de instância maior.

RAM	Suporte Max Node para Elasticsearch e OpenSearch Service 1.x a 2.15	Suporte Max Shard para Elasticsearch e OpenSearch Service 1.x a 2.15	Max Node Support for OpenSearch Service 2.17 e superior	Max Shard Support for OpenSearch Service 2.17 e superior
2 GB	Não aplicável	Não aplicável	10	1 K
4 GB	Não aplicável	Não aplicável	10	5 K
8 GB	10	10 mil	30	15 mil
16 GB	30	30 mil	60	30 mil
32 GB	75	40K	120	60 K

RAM	Suporte Max Node para Elasticsearch e OpenSearch Service 1.x a 2.15	Suporte Max Shard para Elasticsearch e OpenSearch Service 1.x a 2.15	Max Node Support for OpenSearch Service 2.17 e superior	Max Shard Support for OpenSearch Service 2.17 e superior
64 GB	125	75 mil	240	120 K
128 GB	200	75 mil	480	240 K
256 GB	Não aplicável	Não aplicável	1.002	500 mil

Referência geral para Amazon OpenSearch Service

O Amazon OpenSearch Service oferece suporte a uma variedade de instâncias, operações, plug-ins e outros recursos.

Tópicos

- Tipos de instância compatíveis no Amazon OpenSearch Service
- Recursos por versão do mecanismo no Amazon OpenSearch Service
- Plugins por versão do mecanismo no Amazon OpenSearch Service
- Operações suportadas no Amazon OpenSearch Service
- Cotas OpenSearch do Amazon Service
- Instâncias reservadas no Amazon OpenSearch Service
- Técnicas de otimização de custos para o Amazon OpenSearch Service
- Outros recursos suportados no Amazon OpenSearch Service

Tipos de instância compatíveis no Amazon OpenSearch Service

O Amazon OpenSearch Service oferece suporte aos seguintes tipos de instância. Nem todas as regiões são compatíveis com todos os tipos de instância. Para obter detalhes de disponibilidade, consulte os preços OpenSearch do Amazon Service.

Para obter informações sobre qual tipo de instância é apropriado para seu caso de uso, consulte the section called “Dimensionamento de domínios”, the section called “Limites de tamanhos de volume do EBS” e the section called “Limites de rede”.

Observe que as seguintes famílias de instâncias não oferecem suporte aos controles de criptografia de VPC: T2, T3, R3, I3, C4, M4, R4, C5, M5, R5, C6g, M6g, R6g e R6gd. O VPC Encryption Controls é um recurso de segurança e conformidade que oferece controle autoritativo centralizado para monitorar o status de criptografia de seus fluxos de tráfego, ajuda a identificar recursos que permitem a comunicação em texto não criptografado e, eventualmente, fornece mecanismos para aplicar a criptografia em trânsito dentro e fora de sua região. VPCs Para saber mais detalhes, acesse a documentação do VPC Encryption Controls.

Tipos de instâncias da geração atual

Para obter o melhor desempenho, recomendamos que você use os seguintes tipos de instância ao criar novos domínios OpenSearch de serviço.

Tipo de instância	Instâncias	Restrições
i4i	i4i.large.search i4i.xlarge.search i4i.2xlarge.search i4i.4xlarge.search i4i.8xlarge.search i4i.12xlarge.search i4i.16xlarge.search i4i.24xlarge.search i4i.32xlarge.search	Os tipos de instância i4i exigem o Elasticsearch 5.1 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte ao armazenamento de volume do EBS.
i7i	i7i.large.search i7i.xlarge.search	Os tipos de instância i7i exigem o Elasticsearch 5.1 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte ao armazenamento de volume do EBS.

Tipo de instância	Instâncias	Restrições
	i7i.2xlarge.search i7i.4xlarge.search i7i.8xlarge.search i7i.12xlarge.search i7i.16xlarge.search	
8g	i8g.large.search i8g.xlarge.search i8g.2xlarge.search i8g.4xlarge.search i8g.8xlarge.search i8g.12xlarge.search i8g.16xlarge.search	Os tipos de instância i8g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte aos volumes de armazenamento do EBS.

Tipo de instância	Instâncias	Restrições
Graviton3	c7g.large.search c7g.xlarge.search c7g.2xlarge.search c7g.4xlarge.search c7g.8xlarge.search c7g.12xlarge.search c7g.16xlarge.search m7g.medium.search m7g.large.search m7g.xlarge.search m7g.2xlarge.search m7g.4xlarge.search	O Graviton3 suporta apenas armazenamento gp3. As instâncias Graviton3 (c7g, m7g, r7g) oferecem suporte a todas OpenSearch as versões e às versões 7.9 e 7.10 do Elasticsearch.

Tipo de instância	Instâncias	Restrições
	m7g.8xlarge.search m7g.12xlarge.search m7g.16xlarge.search r7g.medium.search r7g.large.search r7g.xlarge.search r7g.2xlarge.search r7g.4xlarge.search r7g.8xlarge.search r7g.12xlarge.search r7g.16xlarge.search r7gd.large.search r7gd.xlarge.search	

Tipo de instância	Instâncias	Restrições
	r7gd.2xlarge.search r7gd.4xlarge.search r7gd.8xlarge.search r7gd.12xlarge.search r7gd.16xlarge.search	

Tipo de instância	Instâncias	Restrições

OR1	`or1.medium.search`
	`or1.large.search`
	`or1.xlarge.search`
	`or1.2xlarge.search`
	`or1.4xlarge.search`
	`or1.8xlarge.search`
	`or1.12xlarge.search`
	`or1.16xlarge.search`
		• Os tipos de OR1 instância exigem a OpenSearch versão 2.11 ou posterior.
		• OR1 as instâncias são compatíveis apenas com outros nós mestres de tipos de instância do Graviton (C6g, M6g, R6g).

Tipo de instância	Instâncias	Restrições
OR2	or2.medium.search or2.large.search or2.xlarge.search or2.2xlarge.search or2.4xlarge.search or2.8xlarge.search or2.12xlarge.search or2.16xlarge.search	• OR2 e os tipos de OM2 instância exigem a OpenSearch versão 2.11 ou posterior e são compatíveis somente com os tipos de instância de nó mestre baseados em Graviton (C6g, M6g, R6g).

Tipo de instância	Instâncias	Restrições
OM2	om2.large.search om2.xlarge.search om2.2xlarge.search om2.4xlarge.search om2.8xlarge.search om2.12xlarge.search om2.16xlarge.search	• OR2 e os tipos de OM2 instância exigem a OpenSearch versão 2.11 ou posterior e são compatíveis somente com os tipos de instância de nó mestre baseados em Graviton (C6g, M6g, R6g).

Tipo de instância	Instâncias	Restrições

OI2	`oi2.large.search`
	`oi2.xlarge.search`
	`oi2.2xlarge.search`
	`oi2.4xlarge.search`
	`oi2.8xlarge.search`
	`oi2.12xlarge.search`
	`oi2.16xlarge.search`
	`oi2.24xlarge.search`
	`oi2.32xlarge.search`
		• Os tipos de OI2 instância exigem a OpenSearch versão 2.11 ou posterior.
		• Essas instâncias são provisionadas com NVMe armazenamento e não exigem armazenamento separado do EBS.
		• Essas instâncias (até 8xlarge) também podem ser usadas como nós quentes na versão OpenSearch 3.3 ou posterior.

Tipo de instância	Instâncias	Restrições

Im4gn	`im4gn.large.search`
	`im4gn.xlarge.search`
	`im4gn.2xlarge.search`
	`im4gn.4xlarge.search`
	`im4gn.8xlarge.search`
	`im4gn.16xlarge.search`
		• Os tipos de instância IM4gn exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte aos volumes de armazenamento do EBS.
		• As instâncias Im4gn só são compatíveis com outros tipos de instância Graviton (C6g, M6g, R6g, R6gd). Você não pode combinar instâncias do Graviton e não Graviton no mesmo cluster.

T3	`t3.small.search` `t3.medium.search`	• Os tipos de instância T3 exigem o Elasticsearch 5.6 ou posterior ou qualquer versão do. OpenSearch • Você pode usar tipos de instância T3 somente se seu domínio for provisionado sem espera. Para saber mais, consulte the section called “Multi-AZ sem modo de espera”. • Você só poderá usar os tipos de instância T3 se a contagem de instâncias de seu domínio for 10 ou menos. • Os tipos de instância T3 não oferecem suporte a UltraWarm armazenamento, armazenamento frio ou Auto-Tune.

Tipo de instância	Instâncias	Restrições

c7i	`c7i.large.search`
	`c7i.xlarge.search`
	`c7i.2xlarge.search`
	`c7i.4xlarge.search`
	`c7i.8xlarge.search`
	`c7i.12xlarge.search`
	`c7i.16xlarge.search`
		• A instância c7i requer o Elasticsearch 5.1 ou posterior ou qualquer versão do OpenSearch, e só oferece suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições

m7i	`m7i.large.search`
	`m7i.xlarge.search`
	`m7i.2xlarge.search`
	`m7i.4xlarge.search`
	`m7i.8xlarge.search`
	`m7i.12xlarge.search`
	`m7i.16xlarge.search`
		• A instância m7i requer o Elasticsearch 5.1 ou posterior ou qualquer versão do OpenSearch, e só oferece suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições

r7i	`r7i.large.search`
	`r7i.xlarge.search`
	`r7i.2xlarge.search`
	`r7i.4xlarge.search`
	`r7i.8xlarge.search`
	`r7i.12xlarge.search`
	`r7i.16xlarge.search`
		• A instância r7i requer o Elasticsearch 5.1 ou posterior ou qualquer versão do OpenSearch, e só oferece suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições

c8g	`c8g.large.search`
	`c8g.xlarge.search`
	`c8g.2xlarge.search`
	`c8g.4xlarge.search`
	`c8g.8xlarge.search`
	`c8g.12xlarge.search`
	`c8g.16xlarge.search`
		• Os tipos de instância c8g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch e só oferecem suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições

m8g	`m8g.medium.search`
	`m8g.large.search`
	`m8g.xlarge.search`
	`m8g.2xlarge.search`
	`m8g.4xlarge.search`
	`m8g.8xlarge.search`
	`m8g.12xlarge.search`
	`m8g.16xlarge.search`
		• Os tipos de instância m8g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch e só oferecem suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições

r8g	`r8g.medium.search`
	`r8g.large.search`
	`r8g.xlarge.search`
	`r8g.2xlarge.search`
	`r8g.4xlarge.search`
	`r8g.8xlarge.search`
	`r8g.12xlarge.search`
	`r8g.16xlarge.search`
		• Os tipos de instância r8g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch e só oferecem suporte a volumes de armazenamento. GP3

Tipo de instância	Instâncias	Restrições
r8gd	r8gd.medium.search r8gd.large.search r8gd.2xlarge.search r8gd.4xlarge.search r8gd.8xlarge.search r8gd.12xlarge.search r8gd.16xlarge.search	• Os tipos de instância r8gd exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch e não oferecem suporte aos volumes de armazenamento do EBS.

Tipos de instância da geração anterior

OpenSearch O serviço oferece tipos de instância da geração anterior para usuários que otimizaram seus aplicativos e ainda precisam fazer o upgrade. Recomendamos usar os tipos de instância da geração atual para usufruir da melhor performance, mas continuamos a oferecer suporte aos tipos de instância da geração anterior a seguir.

Tipo de instância	Instâncias	Restrições

C4	`c4.large.search`
	`c4.xlarge.search`
	`c4.2xlarge.search`
	`c4.4xlarge.search`
	`c4.8xlarge.search`	

I2	`i2.xlarge.search` `i2.2xlarge.search`	

M3	`m3.medium.search`
	`m3.large.search`
	`m3.xlarge.search`
	`m3.2xlarge.search`
		• Os tipos de instância M3 não oferecem suporte à criptografia de dados em repouso, ao controle de acesso refinado ou à pesquisa entre clusters.
		• Os tipos de instância M3 têm restrições adicionais por OpenSearch versão. Para saber mais, consulte the section called “Tipo de instância M3 inválido”.

M4	`m4.large.search`	

Tipo de instância	Instâncias	Restrições

	`m4.xlarge.search`
	`m4.2xlarge.search`
	`m4.4xlarge.search`
	`m4.10xlarge.search`	

R3	`r3.large.search`
	`r3.xlarge.search`
	`r3.2xlarge.search`
	`r3.4xlarge.search`
	`r3.8xlarge.search`
		Os tipos de instância R3 não oferecem suporte à criptografia de dados em repouso ou ao controle de acesso refinado.

Tipo de instância	Instâncias	Restrições

R4	`r4.large.search`
	`r4.xlarge.search`
	`r4.2xlarge.search`
	`r4.4xlarge.search`
	`r4.8xlarge.search`
	`r4.16xlarge.search`	

i4g	i4g.large.search i4g.xlarge.search i4g.2xlarge.search i4g.4xlarge.search i4g.8xlarge.search i4g.16xlarge.search	Os tipos de instância i4g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte aos volumes de armazenamento do EBS.

Tipo de instância	Instâncias	Restrições

C5	`c5.large.search`
	`c5.xlarge.search`
	`c5.2xlarge.search`
	`c5.4xlarge.search`
	`c5.9xlarge.search`
	`c5.18xlarge.search`
		Os tipos de instância C5 exigem o Elasticsearch 5.1 ou posterior ou qualquer versão do. OpenSearch

Tipo de instância	Instâncias	Restrições

C6g	`c6g.large.search`
	`c6g.xlarge.search`
	`c6g.2xlarge.search`
	`c6g.4xlarge.search`
	`c6g.8xlarge.search`
	`c6g.12xlarge.search`
		• Os tipos de instância C6g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do. OpenSearch
		• As instâncias C6g só são compatíveis com outros tipos de instância Graviton (Im4gn, M6g, R6g, R6gd). Você não pode combinar instâncias do Graviton e não Graviton no mesmo cluster.

Tipo de instância	Instâncias	Restrições

I3	`i3.large.search`
	`i3.xlarge.search`
	`i3.2xlarge.search`
	`i3.4xlarge.search`
	`i3.8xlarge.search`
	`i3.16xlarge.search`	

M5	`m5.large.search`
	`m5.xlarge.search`
	`m5.2xlarge.search`
	`m5.4xlarge.search`
	`m5.12xlarge.search`
		Os tipos de instância M5 exigem o Elasticsearch 5.1 ou posterior ou qualquer versão do. OpenSearch

Tipo de instância	Instâncias	Restrições

M6g	`m6g.large.search`
	`m6g.xlarge.search`
	`m6g.2xlarge.search`
	`m6g.4xlarge.search`
	`m6g.8xlarge.search`
	`m6g.12xlarge.search`
		• Os tipos de instância M6g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do. OpenSearch
		• As instâncias M6g só são compatíveis com outros tipos de instância Graviton (Im4gn, C6g, R6g, R6gd). Você não pode combinar instâncias do Graviton e não Graviton no mesmo cluster.

Tipo de instância	Instâncias	Restrições

R5	`r5.large.search`
	`r5.xlarge.search`
	`r5.2xlarge.search`
	`r5.4xlarge.search`
	`r5.12xlarge.search`
		Os tipos de instância R5 exigem o Elasticsearch 5.1 ou posterior ou qualquer versão do. OpenSearch

Tipo de instância	Instâncias	Restrições

R6g	`r6g.large.search`
	`r6g.xlarge.search`
	`r6g.2xlarge.search`
	`r6g.4xlarge.search`
	`r6g.8xlarge.search`
	`r6g.12xlarge.search`
		• Os tipos de instância R6g exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do. OpenSearch
		• As instâncias R6g só são compatíveis com outros tipos de instância Graviton (Im4gn, C6g, M6g, R6gd). Você não pode combinar instâncias do Graviton e não Graviton no mesmo cluster.

Tipo de instância	Instâncias	Restrições

R6gd	`r6gd.large.search`
	`r6gd.xlarge.search`
	`r6gd.2xlarge.search`
	`r6gd.4xlarge.search`
	`r6gd.8xlarge.search`
	`r6gd.12xlarge.search`
	`r6gd.16xlarge.search`
		• Os tipos de instância R6gd exigem o Elasticsearch 7.9 ou posterior ou qualquer versão do OpenSearch, e não oferecem suporte aos volumes de armazenamento do EBS.
		• As instâncias R6gd só são compatíveis com outros tipos de instância Graviton (Im4gn, C6g, M6g, R6g). Você não pode combinar instâncias do Graviton e não Graviton no mesmo cluster.

T2	`t2.micro.search`
	`t2.small.search`
	`t2.medium.search`
		• Você só poderá usar os tipos de instância T2 se a contagem de instâncias de seu domínio for 10 ou menos.
		• O tipo de instância `t2.micro.search` só oferece suporte ao Elasticsearch 1.5 e 2.3.
		• Os tipos de instância T2 não oferecem suporte à criptografia de dados em repouso, ao controle de acesso refinado, ao armazenamento, ao UltraWarm armazenamento frio, à pesquisa entre clusters ou ao Auto-Tune.

> **Tip**
>
> Muitas vezes, recomendamos usar tipos de instâncias diferentes para nós principais dedicados e nós de dados.

Recursos por versão do mecanismo no Amazon OpenSearch Service

Muitos recursos do OpenSearch Service têm um requisito mínimo de OpenSearch versão ou um requisito de versão antiga do Elasticsearch OSS. Se você atender à versão mínima de um recurso, mas o recurso não estiver disponível em seu domínio, atualize o software de serviço do seu domínio.

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Arquitetura de vários níveis	3.3	Não incluído
Fonte derivada	3.1	Não incluído
Star-tree index	2,17	Não incluído
Suporte ao Amazon Q	2,17	Não incluído
Plugins personalizados	2.15	Não incluído
Pesquisa simultânea de segmento	2.13	Não incluído

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Geração de consulta em linguagem natural	2.13	Não incluído
Pesquisa semântica multimodal	2.11	Não incluído
Fontes de dados de consulta direta	2.11	Não incluído
Conectores de Machine Learning	2.9	Não incluído
Pesquisa semântica	2.9	Não incluído
Pesquisa pontual	2,5	Não incluído
Notificações	2.3	Não incluído
ML Commons	1.3	Não incluído
Replicação entre clusters	1.1	7.10
Transformações de índices	1,0	Não incluído

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Nó coordenador dedicado	1,0	6.8
Suporte à VPC	1.0	1,0
Exigir HTTPS para todo o tráfego para o domínio		
Suporte Multi-AZ		
Nós principais dedicados		
Pacotes personalizados		
Endpoints personalizados		
Publicação de logs lentos		
Publicação de logs de erros	1,0	5.1

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Criptografia de dados em repouso		
Autenticação Cognito para painéis OpenSearch		
Atualizações no local		
Snapshots automatizados por hora	1,0	5.3
Node-to-node criptografia	1,0	6.0
Suporte a clientes REST de alto nível do Java		
Solicitação HTTP e compactação da resposta		

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Geração de alertas	1,0	6.2
SQL	1,0	6.5
Pesquisa entre clusters	1,0	6.7
Controle de acesso refinado		
Autenticação SAML para painéis OpenSearch		
Auto-Tune		
Reindexação remota		
UltraWarm	1,0	6.8
Gerenciamento de estados de índice		
k-NN por distância euclidiana	1,0	7.1

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Detecção de anomalias	1,0	7.4
k-NN por similaridade de cosseno	1,0	7.7
Learning to Rank		
Piped Processing Language	1,0	7.9
OpenSearch Relatórios de painéis		
OpenSearch Painéis e análises de rastreamento		
Instâncias do Graviton baseadas em ARM		
Armazenamento de baixa atividade		

Recurso	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Distância de Hamming, distância L1 Norm e desenvolvimento de scripts Painless para K-NN	1,0	7.10
Pesquisa assíncrona		
Suporte ao curador	Não incluído	5.1

Para obter informações sobre plug-ins, que habilitam alguns desses recursos e funcionalidades adicionais, consulte the section called “Plug-ins por versão do mecanismo”. Para obter informações sobre a API do OpenSearch para cada versão, consulte the section called “Operações compatíveis”.

Plugins por versão do mecanismo no Amazon OpenSearch Service

Os domínios OpenSearch do Amazon Service vêm pré-embalados com plug-ins da comunidade. OpenSearch O serviço implanta e gerencia automaticamente plug-ins para você, mas implanta plug-ins diferentes dependendo da versão OpenSearch ou do OSS Elasticsearch legado que você escolher para seu domínio.

A tabela a seguir lista os plug-ins por OpenSearch versão, bem como as versões compatíveis do Elasticsearch OSS legado. Ele inclui apenas plug-ins com os quais você pode interagir — não é abrangente. OpenSearch O serviço usa plug-ins adicionais para habilitar a funcionalidade principal do serviço, como o plug-in S3 Repository para instantâneos e o plug-in OpenSearchPerformance Analyzer para otimização e monitoramento. Para obter uma lista completa de todos os plug-ins em execução no seu domínio, faça a seguinte solicitação:

```
GET _cat/plugins?v
```

Plug-in	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
HanLP	2.11	Não compatível
Análise hebraica	2.11	Não compatível
Classificação de pesquisa do Amazon Personalize	2.9	Não compatível
Pesquisa neural	2.9	Não compatível
Security Analytics	2,5	Não compatível
OpenSearch notifications	2.3	Não compatível
ML Commons	1.3	Não compatível
Análise Sudachi (recomendada para japonês)	1.3	Não compatível
STConvert	1.3	Não compatível

Plug-in	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Análise Pinyin	1.3	Não compatível
Análise Nori	1.3	Não compatível
OpenSearch observabilidade	1.2	Não compatível
OpenSearch replicação entre clusters	1.1	7.10
OpenSearch pesquisa assíncrona	1,0	7.10
Análise IK (Chinês)	1,0	7.7
Análise em vietnamita		
Análise em tailandês		
Learning to Rank		
OpenSearch detecção de anomalias	1,0	7.4

Plug-in	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
OpenSearch k-NN	1,0	7.1
OpenSearch Gerenciamento de estados de índice	1,0	6.8
OpenSearch segurança	1,0	6.7
OpenSearch SQL	1,0	6.5
OpenSearch alertando	1,0	6.2
Ukrainian Analysis	1,0	5.3
Mapper Size	1,0	5.3
Mapper Murmur3	1,0	5.1
Ingest User Agent Processor	1,0	5.1
Ingest Attachment Processor	1,0	5.1

Plug-in	OpenSearch Versão mínima exigida	Versão mínima necessária do Elasticsearch
Stempel Polish Analysis	1,0	5.1
Smart Chinese Analysis	1,0	5.1
Análise da Seunjeon Korean	1,0	5.1
Phonetic Analysis	1,0	2.3
Japanese (kuromoji) Analysis	1,0	Incluído em todos os domínios
ICU Analysis	1,0	Incluído em todos os domínios

Plug-ins opcionais

Além dos plug-ins padrão que vêm pré-instalados, o Amazon OpenSearch Service oferece suporte a vários plug-ins opcionais de análise de linguagem. Você pode usar o Console de gerenciamento da AWS e AWS CLI para associar um plug-in a um domínio, desassociar um plug-in de um domínio e listar todos os plug-ins. Um pacote de plug-in opcional é compatível com uma OpenSearch versão específica e só pode ser associado a domínios com essa versão.

Observe que, quando você reassocia um arquivo de dicionário do plug-in Sudachi, ele não reflete imediatamente no domínio. O dicionário é atualizado quando a próxima blue/green implantação é executada no domínio como parte de uma alteração de configuração ou outra atualização. Como alternativa, você pode criar um novo pacote com os dados atualizados, criar um novo índice usando esse novo pacote, reindexar o índice existente ao novo e, em seguida, excluir o índice antigo. Se

preferir usar a abordagem de reindexação, use um alias de índice para que não haja interrupções no tráfego.

Os plug-ins opcionais usam o tipo de pacote `ZIP-PLUGIN`. Para saber mais sobre plug-ins opcionais, consulte the section called “Pacotes”.

Operações suportadas no Amazon OpenSearch Service

OpenSearch O serviço oferece suporte a várias versões OpenSearch e ao antigo Elasticsearch OSS. As seções a seguir mostram as operações que o OpenSearch Serviço suporta em cada versão.

Tópicos

- Diferenças notáveis de API

Diferenças notáveis de API

Nova lista APIs

Para oferecer suporte a grandes clusters com grande número de índices e fragmentos, introduzimos uma nova Lista APIs com suporte à paginação, como _. list/indices and _list/shards A API List recupera estatísticas sobre índices e fragmentos em um formato paginado. Isso simplifica a tarefa de processar respostas que incluem muitos índices.

- `_list/indices`: _list/indices
- `_list/shards`: _list/shards

Alterações nas existentes APIs

Para compatibilidade com clusters grandes, adicionamos à API `_cluster/stats` compatibilidade com a adição de filtros por métricas para permitir a recuperação apenas de respostas estatísticas relevantes, por exemplo `_cluster/stats/<metric>/nodes/<node-filters>` e `_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Para obter detalhes, consulte _cluster/stats.

Adicionamos à API `_cat/shards` compatibilidade com o cancelamento de tarefas por meio da especificação de um parâmetro de solicitação `cancel_after_time_interval`. Para obter detalhes, consulte _cat/shards.

Limitar o tamanho da resposta para a API _cat

Para compatibilidade com clusters grandes com um total de mais de 200 de instâncias entre nós de dados e nós quentes, temos um limite de 10 mil para o número de índices retornados pelo `_cat/segments API`. Se o número de índices na resposta exceder esse limite, a API retornará um erro 429. Para evitar isso, você pode especificar um filtro de padrão de índices em sua consulta, como `_cat/segments/<index-pattern>`.

Configurações e estatísticas

OpenSearch O serviço só aceita solicitações PUT para a `_cluster/settings` API que usam o formulário de configurações “simples”. Ele rejeita solicitações que usam o formulário de configurações expandidas.

```
// Accepted
PUT _cluster/settings
{
  "persistent" : {
    "action.auto_create_index" : false
  }
}

// Rejected
PUT _cluster/settings
{
  "persistent": {
    "action": {
      "auto_create_index": false
    }
  }
}
```

O cliente Java REST de alto nível usa o formulário expandido, portanto, se for necessário enviar solicitações de configurações, use o cliente de baixo nível.

Antes do Elasticsearch 5.3, a `_cluster/settings` API em domínios de OpenSearch serviço suportava somente o `PUT` método HTTP, não o método. `GET` OpenSearch e versões posteriores do Elasticsearch oferecem suporte ao `GET` método, conforme mostrado no exemplo a seguir:

```
GET https://domain-name.region.es.amazonaws.com/_cluster/settings?pretty
```

Veja um exemplo de retorno:

```
{
  "persistent": {
    "cluster": {
      "routing": {
        "allocation": {
          "cluster_concurrent_rebalance": "2",
          "node_concurrent_recoveries": "2",
          "disk": {
            "watermark": {
              "low": "1.35gb",
              "flood_stage": "0.45gb",
              "high": "0.9gb"
            }
          },
          "node_initial_primarirecoveries": "4"
        }
      }
    },
    "indices": {
      "recovery": {
        "max_bytper_sec": "40mb"
      }
    }
  }
}
```

Se você comparar as respostas de um OpenSearch cluster de código aberto e OpenSearch de um serviço para determinadas configurações e estatísticas APIs, poderá notar campos ausentes. OpenSearch O serviço redige determinadas informações que expõem os componentes internos do serviço, como o caminho de dados do sistema de arquivos `_nodes/stats` ou o nome e a versão do sistema operacional de. `_nodes`

Shrink

A API `_shrink` pode causar falhas em atualizações, alterações de configuração e exclusões de domínio. Não recomendamos usá-la em domínios que executam o Elasticsearch versões 5.3 ou 5.1. Essas versões têm um erro que pode causar falha na restauração de snapshots de índices reduzidos.

Se você usa a `_shrink` API em outras OpenSearch versões do Elasticsearch, faça a seguinte solicitação antes de iniciar a operação de redução:

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": "name-of-the-node-to-shrink-to",
    "index.blocks.read_only": true
  }
}
```

Depois, faça a seguinte solicitação após concluir a operação de redução:

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}

PUT https://domain-name.region.es.amazonaws.com/shrunken-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}
```

Nova lista APIs

Para oferecer suporte a grandes clusters com um grande número de índices e fragmentos, introduzimos uma nova lista APIs com suporte à paginação, ou seja, e. `_list/indices` `_list/shards` A API List recupera estatísticas sobre índices e fragmentos em um formato paginado. Isso simplifica a tarefa de processar respostas que incluem muitos índices. Para saber mais sobre `_list/indices`, consulte List indices. Para saber mais sobre `_list/shards`, consulte List shards.

Alterações nas existentes APIs

Para compatibilidade com clusters grandes, adicionamos compatibilidade em `_cluster/stats/<metric>/nodes/<node-filters>` e `_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Para saber mais sobre `_cluster/stats`, consulte Cluster stats.

Limitando o tamanho da resposta para _cat APIs

Para compatibilidade com clusters grandes com um total de mais de 200 de instâncias entre nós de dados e nós quentes, temos um limite para o número de índices retornados pela API _cat/segments de 10 mil. Se o número de índices na resposta exceder esse limite, a API exibirá um erro 429. Para evitar isso, você pode especificar um filtro de padrão de índices em sua consulta (por exemplo, `_cat/segments/<index-pattern>`).

Além disso, a compatibilidade com cancelamento de tarefas agora está disponível para a API `_cat/shards` para cancelamento de tarefas, por meio da especificação do parâmetro de solicitação `cancel_after_time_interval`. Para saber mais sobre isso, consulte CAT shards.

Escolher tipos de instância para nós principais dedicados

A tabela a seguir fornece recomendações sobre a escolha dos tipos de instância adequados para os nós principais dedicados:

RAM	Máximo de nós compatível	Máximo de fragmentos compatível
2 GB	10	1.000
4 GB	10	5.000
8 GB	30	15.000
16 GB	60	30.000
32 GB	120	60.000
64 GB	240	120.000
128 GB	480	240.000

RAM	Máximo de nós compatível	Máximo de fragmentos compatível
256 GB	1.002	500.000

OpenSearch versão 2.19

Para obter informações sobre as operações da OpenSearch versão 2.19, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico. Para obter mais detalhes sobre as alterações nesta versão, consulte as Notas da versão 2.19.

OpenSearch versão 2.17

Para a OpenSearch versão 2.17, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

> **Note**
>
> A partir da OpenSearch versão 2.17, a `cluster.max_shards_per_node` configuração não pode ser modificada. Para a OpenSearch versão 2.17 e versões posteriores, o OpenSearch Service oferece suporte a 1.000 fragmentos para cada 16 GB de memória heap da JVM, até um máximo de 4.000 fragmentos por nó.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_list`
- `/_ltr`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`

- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
  - cluster.search.request.slowlog.threshold.trace
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

  - `search.phase_took_enabled`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Consulte o método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e outras.
5. Consulte the section called “Shrink”.

> **Note**
>
> Atualmente, a alteração da funcionalidade da configuração `cluster.max_shards_per_node` não está habilitada para clientes com Multi-AZ (zona de disponibilidade) em espera.

OpenSearch versão 2.15

Para a OpenSearch versão 2.15, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* */_forcemerge* , */index-name* */update/id* e */index-name* */_close*)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.search.request.slowlog.level
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/pipeline
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

    - `cluster.search.request.slowlog.threshold.warn`
    - `cluster.search.request.slowlog.threshold.info`
    - `cluster.search.request.slowlog.threshold.debug`
    - `cluster.search.request.slowlog.threshold.trace`
    - `search.phase_took_enabled`
  - `/_cluster/state`
  - `/_cluster/stats`
  - `/_count`
  - `/_dashboards`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.13

Para a OpenSearch versão 2.13, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- indices.breaker.total.limit
- cluster.max_shards_per_node
- cluster.search.request.slowlog.level
- cluster.search.request.slowlog.threshold.warn
- cluster.search.request.slowlog.threshold.info
- cluster.search.request.slowlog.threshold.debug
- cluster.search.request.slowlog.threshold.trace
- search.phase_took_enabled

- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método PUT. Para obter informações sobre o método GET, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.11

Para a OpenSearch versão 2.11, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`

- action.auto_create_index
- action.search.shard_count.limit
- indices.breaker.fielddata.limit
- indices.breaker.request.limit
- indices.breaker.total.limit
- cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.9

Para a OpenSearch versão 2.9, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_security_analytics`
- `/_plugins/_sm`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.7

Para OpenSearch 2.7, o OpenSearch Service suporta as seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-*
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_refresh`
- `/_reindex` [1]
- `/_render`

name /update/*id* e /*index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.5

Para OpenSearch 2.5, o OpenSearch Service suporta as seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]

- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_security_analytics`
- `/_plugins/_sm`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.

4. Refere-se ao método PUT. Para obter informações sobre o método GET, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 2.3

Para OpenSearch 2.3, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* /_forcemerge , */index-name* /update/*id* e */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- _plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_percolate
- /_rank_eval

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 1.3

Para OpenSearch 1.3, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* */_forcemerge* , */index-name* */update/id* e */index-name* */_close*)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_count`
- `/_dashboards`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 1.2

Para OpenSearch 1.2, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`

- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho

por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 1.1

Para OpenSearch 1.1, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`

- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_plugins/_transforms`
- `/_percolate`
- `/_rank_eval`
- `/_update_by_query` [1]
- `/_validate`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

OpenSearch versão 1.0

Para OpenSearch 1.0, o OpenSearch Service oferece suporte às seguintes operações. Para obter informações sobre a maioria das operações, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_plugins/_transforms`
- `/_percolate`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- indices.breaker.total.limit
- cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_rank_eval

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em scroll_id valores, use o corpo da solicitação, não a string de consulta, para passar scroll_id valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método PUT. Para obter informações sobre o método GET, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 7.10

Para o Elasticsearch 7.10, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* /_forcemerge , */index-name* /update/*id* e */index-name* /_close)
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover

- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_index_template [6]
- /_ingest/pipeline
- /_index_template
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_asynchronous_search
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_ppl
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_plugins/_replication
- /_rank_eval
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.

2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.
6. Modelos de índice herdados (`_template`) foram substituídos por modelos que podem ser compostos (`_index_template`) começando com o Elasticsearch 7.8. Os modelos que podem ser compostos têm precedência sobre os modelos legados. Se nenhum modelo que pode ser composto corresponder a um determinado índice, um modelo legado ainda poderá corresponder e ser aplicado. A `_template` operação ainda funciona nas OpenSearch versões posteriores do Elasticsearch OSS, mas as chamadas GET para os dois tipos de modelo retornam resultados diferentes.

Elasticsearch versão 7.9

Para o Elasticsearch 7.9, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_index_template` [6]
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]

- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_ppl
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em scroll_id valores, use o corpo da solicitação, não a string de consulta, para passar scroll_id valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.

4. Refere-se ao método PUT. Para obter informações sobre o método GET, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às OpenSearch operações genéricas que o OpenSearch Serviço suporta e não inclui operações suportadas específicas do plug-in para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.
6. Modelos de índice herdados (`_template`) foram substituídos por modelos que podem ser compostos (`_index_template`) começando com o Elasticsearch 7.8. Os modelos que podem ser compostos têm precedência sobre os modelos legados. Se nenhum modelo que pode ser composto corresponder a um determinado índice, um modelo legado ainda poderá corresponder e ser aplicado. A `_template` operação ainda funciona nas OpenSearch versões posteriores do Elasticsearch OSS, mas as chamadas GET para os dois tipos de modelo retornam resultados diferentes.

Elasticsearch versão 7.8

Para o Elasticsearch 7.8, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_index_template` [6]
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template` [6]
- `/_update_by_query` [1]

- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_validate`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.
6. Modelos de índice herdados (`_template`) foram substituídos por modelos que podem ser compostos (`_index_template`) começando com o Elasticsearch 7.8. Os modelos que podem ser compostos têm precedência sobre os modelos legados. Se nenhum modelo que pode ser composto corresponder a um determinado índice, um modelo legado ainda poderá corresponder

e ser aplicado. A `_template` operação ainda funciona nas OpenSearch versões posteriores do Elasticsearch OSS, mas as chamadas GET para os dois tipos de modelo retornam resultados diferentes.

Elasticsearch versão 7.7

Para o Elasticsearch 7.7, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_anomaly_detection`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `/_plugin/kibana`
- `/_rank_eval`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 7.4

Para o Elasticsearch 7.4, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` , */index-name* `/update/`*id* e */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`

- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em scroll_id valores, use o corpo da solicitação, não a string de consulta, para passar scroll_id valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.

4. Refere-se ao método PUT. Para obter informações sobre o método GET, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 7.1

Para o Elasticsearch 7.1, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`

- `/_rank_eval`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.8

Para o Elasticsearch 6.8, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]

- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.blocks.read_only
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_ism
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho

por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.7

Para o Elasticsearch 6.7, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_security`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`

- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.5

Para o Elasticsearch 6.5, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.

2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.4

Para o Elasticsearch 6.4, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_validate`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.3

Para o Elasticsearch 6.3, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-*
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_refresh`
- `/_reindex` [1]
- `/_render`

name /update/*id*) exceto /*index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho

por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.2

Para o Elasticsearch 6.2, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`

- `/_rank_eval`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 6.0

Para o Elasticsearch 6.0, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-*
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_render`
- `/_rollover`
- `/_scripts` [3]

name /update/*id*) exceto */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho

por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 5.6

Para o Elasticsearch 5.6, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* */_forcemerge* e */index-name* */update/id*) exceto */index-name* */_close*
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[4]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 5.5

Para o Elasticsearch 5.5, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-*
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_render`
- `/_rollover`
- `/_scripts` [3]

name /update/*id*) exceto */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades[4]:
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho

por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a string de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.

3. Para saber as considerações sobre o uso de scripts, consulte the section called “Outros recursos compatíveis”.
4. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
5. Consulte the section called “Shrink”.

Elasticsearch versão 5.3

Para o Elasticsearch 5.3, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* */_forcemerge* e */index-name* */update/id*) exceto */index-name* */_close*
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` para várias propriedades[3]:
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[4]
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação `/_tasks` com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para `/_search/scroll` com um corpo de mensagem deve especificar `"Content-Length"` no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em `scroll_id` valores, use o corpo da solicitação, não a cadeia de caracteres de consulta, para passar `scroll_id` valores para o OpenSearch Serviço.
3. Refere-se ao método `PUT`. Para obter informações sobre o método `GET`, consulte the section called “Diferenças notáveis de API”. Essa lista se refere apenas às operações genéricas do Elasticsearch que o OpenSearch Service suporta e não inclui operações suportadas específicas de plug-ins para detecção de anomalias, ISM e assim por diante.
4. Consulte the section called “Shrink”.

Elasticsearch versão 5.1

Para o Elasticsearch 5.1, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como */index-name* `/_forcemerge` e */index-name* `/update/`*id*) exceto */index-name* `/_close`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_render`
- `/_rollover`
- `/_search` [2]
- `/_search profile`

- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (exceto /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings para várias propriedades (somente PUT):
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_shard_stores
- /_shrink[3]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Essas operações podem ser interrompidas por alterações na configuração de cluster. É recomendável usar a operação /_tasks com essas operações para verificar se as solicitações foram concluídas com êxito.
2. Solicitações de DELETE para /_search/scroll com um corpo de mensagem deve especificar "Content-Length" no cabeçalho HTTP. A maioria dos clientes adicionam esse cabeçalho por padrão. Para evitar problemas com = caracteres em scroll_id valores, use o corpo da solicitação, não a cadeia de caracteres de consulta, para passar scroll_id valores para o OpenSearch Serviço.

3. Consulte the section called “Shrink”.

Elasticsearch versão 2.3

Para o Elasticsearch 2.3, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice (como `/index-name /_forcemerge` e `/index-name /_recovery`) exceto `/index-name /_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cache/clear` (somente Índice)
- `/_cat` (exceto `/_cat/nodeattrs`)
- `/_cluster/health`
- `/_cluster/settings` para várias propriedades (somente PUT):
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `threadpool.get.queue_size`
  - `threadpool.bulk.queue_size`
  - `threadpool.index.queue_size`
  - `threadpool.percolate.queue_size`
  - `threadpool.search.queue_size`
  - `threadpool.suggest.queue_size`
- `/_cluster/stats`
- `/_count`
- `/_flush`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_render`
- `/_search`
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_template`

Elasticsearch versão 1.5

Para o Elasticsearch 1.5, o OpenSearch Service oferece suporte às seguintes operações.

- Todas as operações no caminho do índice, como */index-name* /_optimize e */index-name* /_warmer, exceto */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat
- /_cluster/health
- /_cluster/settings para várias propriedades (somente PUT):
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - threadpool.get.queue_size
  - threadpool.bulk.queue_size
  - threadpool.index.queue_size
  - threadpool.percolate.queue_size
  - threadpool.search.queue_size
  - threadpool.suggest.queue_size
- /_cluster/stats
- /_count
- /_flush
- /_mapping
- /_mget
- /_msearch
- /_nodes
- /_percolate
- /_plugin/kibana
- /_plugin/kibana3
- /_plugin/migration
- /_refresh
- /_search
- /_snapshot
- /_stats
- /_status
- /_template

Cotas OpenSearch do Amazon Service

Sua AWS conta tem cotas padrão, anteriormente chamadas de limites, para cada AWS serviço. A menos que especificado de outra forma, cada cota é específica para a região.

Para ver as cotas para domínios e instâncias do OpenSearch Serviço, Amazon OpenSearch Serverless e Amazon OpenSearch Ingestion, consulte as cotas do OpenSearch Amazon Service no. Referência geral da AWS

Para ver as cotas de OpenSearch serviço no Console de gerenciamento da AWS, abra o console Service Quotas. No painel de navegação, escolha AWS serviços e selecione Amazon OpenSearch Service. Para solicitar um aumento da cota, consulte Requesting a quota increase no Guia do usuário do Service Quotas.

Cotas de armazenamento de nós quentes

A tabela a seguir lista os tipos de instância de node quente e a quantidade máxima de armazenamento que cada tipo pode usar. Para OI2 instâncias OpenSearch otimizadas, 80% do armazenamento local está disponível como cache, e o armazenamento quente endereçável máximo é 5 vezes o tamanho do armazenamento em cache.

Por exemplo, se uma OI2 instância tiver 468 GB de armazenamento local:

- Armazenamento em cache = 375 GB (80% do armazenamento local)
- Armazenamento quente endereçável máximo = 1875 GB (5 x 375 GB de cache)

Tipo de instância	Armazenamento da instância (GB)	Tamanho do cache (GB)	Armazenamento quente endereçável máximo (GB)
oi2.large.search	468	375	1875
oi2.xlarge.search	937	750	3750
oi2.2xlarge.search	1875	1500	7500
oi2.4xlarge.search	3750	3000	15000
oi2.8xlarge.search	7500	6000	30000

UltraWarm cotas de armazenamento

A tabela a seguir lista os tipos de UltraWarm instância e a quantidade máxima de armazenamento que cada tipo pode usar. Para obter mais informações sobre UltraWarm, consultethe section called "UltraWarm armazenamento".

Tipo de instância	Armazenamento máximo
ultrawarm1.medium.search	1,5 TiB
ultrawarm1.large.search	20 TiB

Número de nós de dados por zona de disponibilidade

A tabela a seguir lista o número máximo total de nós de dados de implantação em zona de disponibilidade. O limite geral significa o número de nós de dados de acordo com o limite, incluindo a quantidade de nós quentes e mornos que cada tipo pode usar.

Configuração de zona de disponibilidade	Limite da quantidade de nós quentes	Limite da quantidade de nós mornos	Limite geral (quentes e mornos)
1: AZ	334	250	334
2: AZ	668	500	668
3: AZ	1.002	750	1.002

Limite total de nós por família de instâncias

A tabela a seguir lista o limite total de nós por família de instâncias.

Família de instâncias	ElasticSearch OpenSearch até 2,15	OpenSearch 2.17 e superior	Limite padrão
T2	10	10	10

Família de instâncias	ElasticSearch OpenSearch até 2,15	OpenSearch 2.17 e superior	Limite padrão
T3	10	10	10
M3, C4, M4, R4, C5, M5, R5, I2, I3	10	200	80
Gráviton 2, Gráviton 3, Gráviton 4	200	400	80
C7i, R7i, M7i, i4i, i4i, i8g, i7i	200	400	80
OR1.medium.search OR1.large.search OR2.medium.search OR2.large.search OM2.large.search OI2.large.search	200	400	80
OR1.xlarge.search e superior OR2.xlarge.search e superior OM2.xlarge.search e superior OI2.xlarge.search e superior	200	1.002	80
Ultrawarm1	150	750	150

Limites de tamanhos de volume do EBS

A tabela a seguir mostra os tamanhos mínimo e máximo dos volumes do EBS para cada tipo de instância compatível com o OpenSearch Service. Para obter informações sobre quais tipos de instância incluem armazenamento de instâncias e detalhes adicionais de hardware, consulte os preços do Amazon OpenSearch Service.

- Se você escolher armazenamento magnético em Tipo de volume do EBS ao criar o domínio, o tamanho máximo do volume será 100 GiB para todos os tipos de instância, exceto t2.small e t2.medium, e todas as instâncias Graviton (M6g, C6g, R6g e R6gd), que não são compatíveis com armazenamento magnético. Para os tamanhos máximos listados na tabela a seguir, escolha uma das opções de SSD.
- Alguns tipos de instância de gerações anteriores incluem o armazenamento de instâncias, mas também oferecem suporte ao armazenamento do EBS. Se você escolher o armazenamento do EBS para um desses tipos de instância, os volumes de armazenamento não serão aditivos. Você pode usar um volume do EBS ou o armazenamento de instâncias, mas não ambos.
- Para volumes gp3 do EBS, os limites máximos suportados são 80.000 IOPS e 2.000 taxas de transferência. MB/s Observe que os limites efetivos do seu domínio podem ser menores dependendo do tipo de instância. Para ver os limites máximos específicos da instância, consulte os limites do tipo de instância.

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
t2.micro.search	10 GiB	35 GiB	N/D
t2.small.search	10 GiB	35 GiB	N/D
t2.medium.search	10 GiB	35 GiB	N/D
t3.small.search	10 GiB	100 GiB	100 GiB
t3.medium.search	10 GiB	200 GiB	200 GiB
m3.medium.search	10 GiB	100 GiB	N/D
m3.large.search	10 GiB	512 GiB	N/D
m3.xlarge.search	10 GiB	512 GiB	N/D
m3.2xlarge.search	10 GiB	512 GiB	N/D
m4.large.search	10 GiB	512 GiB	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
m4.xlarge.search	10 GiB	1 TiB	N/D
m4.2xlarge.search	10 GiB	1,5 TiB	N/D
m4.4xlarge.search	10 GiB	1,5 TiB	N/D
m4.10xlarge.search	10 GiB	1,5 TiB	N/D
m5.large.search	10 GiB	512 GiB	1 TiB
m5.xlarge.search	10 GiB	1 TiB	2 TiB
m5.2xlarge.search	10 GiB	1,5 TiB	3 TiB
m5.4xlarge.search	10 GiB	3 TiB	6 TiB
m5.12xlarge.search	10 GiB	9 TiB	18 TiB
m6g.large.search	10 GiB	512 GiB	1 TiB
m6g.xlarge.search	10 GiB	1 TiB	2 TiB
m6g.2xlarge.search	10 GiB	1,5 TiB	3 TiB
m6g.4xlarge.search	10 GiB	3 TiB	6 TiB
m6g.8xlarge.search	10 GiB	6 TiB	12 TiB
m6g.12xlarge.search	10 GiB	9 TiB	18 TiB
c4.large.search	10 GiB	100 GiB	N/D
c4.xlarge.search	10 GiB	512 GiB	N/D
c4.2xlarge.search	10 GiB	1 TiB	N/D
c4.4xlarge.search	10 GiB	1,5 TiB	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
c4.8xlarge.search	10 GiB	1,5 TiB	N/D
c5.large.search	10 GiB	256 GiB	256 GiB
c5.xlarge.search	10 GiB	512 GiB	512 GiB
c5.2xlarge.search	10 GiB	1 TiB	1 TiB
c5.4xlarge.search	10 GiB	1,5 TiB	1,5 TiB
c5.9xlarge.search	10 GiB	3,5 TiB	3,5 TiB
c5.18xlarge.search	10 GiB	7 TiB	7 TiB
c6g.large.search	10 GiB	256 GiB	256 GiB
c6g.xlarge.search	10 GiB	512 GiB	512 GiB
c6g.2xlarge.search	10 GiB	1 TiB	1 TiB
c6g.4xlarge.search	10 GiB	1,5 TiB	1,5 TiB
c6g.8xlarge.search	10 GiB	3 TiB	3 TiB
c6g.12xlarge.search	10 GiB	4,5 TiB	4,5 TiB
r3.large.search	10 GiB	512 GiB	N/D
r3.xlarge.search	10 GiB	512 GiB	N/D
r3.2xlarge.search	10 GiB	512 GiB	N/D
r3.4xlarge.search	10 GiB	512 GiB	N/D
r3.8xlarge.search	10 GiB	512 GiB	N/D
r4.large.search	10 GiB	1 TiB	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
r4.xlarge.search	10 GiB	1,5 TiB	N/D
r4.2xlarge.search	10 GiB	1,5 TiB	N/D
r4.4xlarge.search	10 GiB	1,5 TiB	N/D
r4.8xlarge.search	10 GiB	1,5 TiB	N/D
r4.16xlarge.search	10 GiB	1,5 TiB	N/D
r5.large.search	10 GiB	1 TiB	2 TiB
r5.xlarge.search	10 GiB	1,5 TiB	3 TiB
r5.2xlarge.search	10 GiB	3 TiB	6 TiB
r5.4xlarge.search	10 GiB	6 TiB	12 TiB
r5.12xlarge.search	10 GiB	12 TiB	24 TiB
r6g.large.search	10 GiB	1 TiB	2 TiB
r6g.xlarge.search	10 GiB	1,5 TiB	3 TiB
r6g.2xlarge.search	10 GiB	3 TiB	6 TiB
r6g.4xlarge.search	10 GiB	6 TiB	12 TiB
r6g.8xlarge.search	10 GiB	8 TiB	16 TiB
r6g.12xlarge.search	10 GiB	12 TiB	24 TiB
r6gd.large.search	N/D	N/D	N/D
r6gd.xlarge.search	N/D	N/D	N/D
r6gd.2xlarge.search	N/D	N/D	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
r6gd.4xlarge.search	N/D	N/D	N/D
r6gd.8xlarge.search	N/D	N/D	N/D
r6gd.12xlarge.search	N/D	N/D	N/D
r6gd.16xlarge.search	N/D	N/D	N/D
i2.xlarge.search	10 GiB	512 GiB	N/D
i2.2xlarge.search	10 GiB	512 GiB	N/D
i3.large.search	N/D	N/D	N/D
i3.xlarge.search	N/D	N/D	N/D
i3.2xlarge.search	N/D	N/D	N/D
i3.4xlarge.search	N/D	N/D	N/D
i3.8xlarge.search	N/D	N/D	N/D
i3.16xlarge.search	N/D	N/D	N/D
or1.medium.search	20 GiB	N/D	768 GiB
or1.large.search	20 GiB	N/D	1.532 GiB
or1.xlarge.search	20 GiB	N/D	3 TiB
or1.2xlarge.search	20 GiB	N/D	6 TiB
or1.4xlarge.search	20 GiB	N/D	12 TiB
or1.8xlarge.search	20 GiB	N/D	16 TiB
or1.12xlarge.search	20 GiB	N/D	24 TiB

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
or1.16xlarge.search	20 GiB	N/D	36 TiB
or2.medium.search	20 GiB	N/D	768 GiB
or2.large.search	20 GiB	N/D	1.532 GiB
or2.xlarge.search	20 GiB	N/D	3 TiB
or2.2xlarge.search	20 GiB	N/D	6 TiB
or2.4xlarge.search	20 GiB	N/D	12 TiB
or2.8xlarge.search	20 GiB	N/D	16 TiB
or2.12xlarge.search	20 GiB	N/D	24 TiB
or2.16xlarge.search	20 GiB	N/D	36 TiB
om2.large.search	20 GiB	N/D	768 GiB
om2.xlarge.search	20 GiB	N/D	2 Tib
om2.2xlarge.search	20 GiB	N/D	3 Tib
om2.4xlarge.search	20 GiB	N/D	6 Tib
om2.8xlarge.search	20 GiB	N/D	12 Tib
om2.12xlarge.search	20 GiB	N/D	18 TiB
om2.16xlarge.search	20 GiB	N/D	24 Tib
im4gn.large.search	N/D	N/D	N/D
im4gn.xlarge.search	N/D	N/D	N/D
im4gn.2xlarge.search	N/D	N/D	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
im4gn.4xlarge.search	N/D	N/D	N/D
im4gn.8xlarge.search	N/D	N/D	N/D
im4gn.16xlarge.search	N/D	N/D	N/D
C7g.large.search	10 GiB	N/D	256 GiB
C7g.xlarge.search	10 GiB	N/D	512 GiB
C7g.2xlarge.search	10 GiB	N/D	1 TiB
C7g.4xlarge.search	10 GiB	N/D	1,5 TiB
C7g.8xlarge.search	10 GiB	N/D	3 TiB
C7g.12xlarge.search	10 GiB	N/D	4,5 TiB
C7g.16xlarge.search	10 GiB	N/D	6 TiB
M7g.medium.search	10 GiB	N/D	512 GiB
M7g.large.search	10 GiB	N/D	768 GiB
M7g.xlarge.search	10 GiB	N/D	2 TiB
M7g.2xlarge.search	10 GiB	N/D	3 TiB
M7g.4xlarge.search	10 GiB	N/D	6 TiB
M7g.8xlarge.search	10 GiB	N/D	12 TiB
M7g.12xlarge.search	10 GiB	N/D	18 TiB
M7g.16xlarge.search	10 GiB	N/D	24 TiB
R7g.medium.search	10 GiB	N/D	768 GiB

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
R7g.large.search	10 GiB	N/D	1,5 TiB
R7g.xlarge.search	10 GiB	N/D	3 TiB
R7g.2xlarge.search	10 GiB	N/D	6 TiB
R7g.4xlarge.search	10 GiB	N/D	12 TiB
R7g.8xlarge.search	10 GiB	N/D	16 TiB
R7g.12xlarge.search	10 GiB	N/D	24 TiB
R7g.16xlarge.search	10 GiB	N/D	36 TiB
R7gd.large.search	N/D	N/D	N/D
R7gd.xlarge.search	N/D	N/D	N/D
R7gd.2xlarge.search	N/D	N/D	N/D
R7gd.4xlarge.search	N/D	N/D	N/D
R7gd.8xlarge.search	N/D	N/D	N/D
R7gd.12xlarge.search	N/D	N/D	N/D
R7gd.16xlarge.search	N/D	N/D	N/D
i4i.large.search	N/D	N/D	N/D
i4i.xlarge.search	N/D	N/D	N/D
i4i.2xlarge.search	N/D	N/D	N/D
i4i.4xlarge.search	N/D	N/D	N/D
i4i.8xlarge.search	N/D	N/D	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
i4i.12xlarge.search	N/D	N/D	N/D
i4i.16xlarge.search	N/D	N/D	N/D
i4i.24xlarge.search	N/D	N/D	N/D
i4i.32xlarge.search	N/D	N/D	N/D
i4g.large.search	N/D	N/D	N/D
i4g.xlarge.search	N/D	N/D	N/D
i4g.2xlarge.search	N/D	N/D	N/D
i4g.4xlarge.search	N/D	N/D	N/D
i4g.8xlarge.search	N/D	N/D	N/D
i4g.16xlarge.search	N/D	N/D	N/D
i8g.large.search	N/D	N/D	N/D
i8g.xlarge.search	N/D	N/D	N/D
i8g.2xlarge.search	N/D	N/D	N/D
i8g.4xlarge.search	N/D	N/D	N/D
i8g.8xlarge.search	N/D	N/D	N/D
i8g.12xlarge.search	N/D	N/D	N/D
i8g.16xlarge.search	N/D	N/D	N/D
i7i.large.search	N/D	N/D	N/D
i7i.xlarge.search	N/D	N/D	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
i7i.2xlarge.search	N/D	N/D	N/D
i7i.4xlarge.search	N/D	N/D	N/D
i7i.8xlarge.search	N/D	N/D	N/D
i7i.12xlarge.search	N/D	N/D	N/D
i7i.16xlarge.search	N/D	N/D	N/D
c7i.large.search	10 GiB	N/D	256 GiB
c7i.xlarge.search	10 GiB	N/D	512 GiB
c7i.2xlarge.search	10 GiB	N/D	1 TiB
c7i.4xlarge.search	10 GiB	N/D	1,5 TiB
c7i.8xlarge.search	10 GiB	N/D	3 TiB
c7i.12xlarge.search	10 GiB	N/D	4,5 TiB
c7i.16xlarge.search	10 GiB	N/D	6 TiB
m7i.large.search	10 GiB	N/D	768 GiB
m7i.xlarge.search	10 GiB	N/D	2 TiB
m7i.2xlarge.search	10 GiB	N/D	3 TiB
m7i.4xlarge.search	10 GiB	N/D	6 TiB
m7i.8xlarge.search	10 GiB	N/D	12 TiB
m7i.12xlarge.search	10 GiB	N/D	18 TiB
m7i.16xlarge.search	10 GiB	N/D	24 TiB

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
r7i.large.search	10 GiB	N/D	1,5 TiB
r7i.xlarge.search	10 GiB	N/D	3 TiB
r7i.2xlarge.search	10 GiB	N/D	6 TiB
r7i.4xlarge.search	10 GiB	N/D	12 TiB
r7i.8xlarge.search	10 GiB	N/D	16 TiB
r7i.12xlarge.search	10 GiB	N/D	24 TiB
r7i.12xlarge.search	10 GiB	N/D	36 TiB
c8g.large	10 GiB	N/D	256 GiB
c8g.xlarge	10 GiB	N/D	512 GiB
c8g.2xlarge	10 GiB	N/D	1 TiB
c8g.4xlarge	10 GiB	N/D	1,5 TiB
c8g.8xlarge	10 GiB	N/D	3 TiB
c8g.12xlarge	10 GiB	N/D	4,5 TiB
c8g.16xlarge	10 GiB	N/D	6 TiB
m8g.medium	10 GiB	N/D	512 GiB
m8g.large	10 GiB	N/D	768 GiB
m8g.xlarge	10 GiB	N/D	2 TiB
m8g.2xlarge	10 GiB	N/D	3 TiB
m8g.4xlarge	10 GiB	N/D	6 TiB

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
m8g.8xlarge	10 GiB	N/D	12 TiB
m8g.12xlarge	10 GiB	N/D	18 TiB
m8g.16xlarge	10 GiB	N/D	24 TiB
r8g.medium	10 GiB	N/D	768 GiB
r8g.large	10 GiB	N/D	1.532 GiB
r8g.xlarge	10 GiB	N/D	3 TiB
r8g.2xlarge	10 GiB	N/D	6 TiB
r8g.4xlarge	10 GiB	N/D	12 TiB
r8g.8xlarge	10 GiB	N/D	16 TiB
r8g.12xlarge	10 GiB	N/D	24 TiB
r8g.16xlarge	10 GiB	N/D	36 TiB
r8gd.medium	N/D	N/D	N/D
r8gd.large	N/D	N/D	N/D
r8gd.xlarge	N/D	N/D	N/D
r8gd.2xlarge	N/D	N/D	N/D
r8gd.4xlarge	N/D	N/D	N/D
r8gd.8xlarge	N/D	N/D	N/D
r8gd.12xlarge	N/D	N/D	N/D
r8gd.16xlarge	N/D	N/D	N/D

Tipo de instância	Tamanho mínimo do EBS	Tamanho máximo do EBS (gp2)	Tamanho máximo do EBS (gp3)
oi2.large.search	N/D	N/D	N/D
oi2.xlarge.search	N/D	N/D	N/D
oi2.2xlarge.search	N/D	N/D	N/D
oi2.4xlarge.search	N/D	N/D	N/D
oi2.8xlarge.search	N/D	N/D	N/D
oi2.12xlarge.search	N/D	N/D	N/D
oi2.16xlarge.search	N/D	N/D	N/D

Limites de rede

A tabela a seguir mostra o tamanho máximo de cargas de solicitação HTTP.

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
t2.micro.search	10 MiB
t2.small.search	10 MiB
t2.medium.search	10 MiB
t3.small.search	10 MiB
t3.medium.search	10 MiB
m3.medium.search	10 MiB
m3.large.search	10 MiB
m3.xlarge.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
m3.2xlarge.search	100 MiB
m4.large.search	10 MiB
m4.xlarge.search	100 MiB
m4.2xlarge.search	100 MiB
m4.4xlarge.search	100 MiB
m4.10xlarge.search	100 MiB
m5.large.search	10 MiB
m5.xlarge.search	100 MiB
m5.2xlarge.search	100 MiB
m5.4xlarge.search	100 MiB
m5.12xlarge.search	100 MiB
m6g.large.search	10 MiB
m6g.xlarge.search	100 MiB
m6g.2xlarge.search	100 MiB
m6g.4xlarge.search	100 MiB
m6g.8xlarge.search	100 MiB
m6g.12xlarge.search	100 MiB
c4.large.search	10 MiB
c4.xlarge.search	100 MiB
c4.2xlarge.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
c4.4xlarge.search	100 MiB
c4.8xlarge.search	100 MiB
c5.large.search	10 MiB
c5.xlarge.search	100 MiB
c5.2xlarge.search	100 MiB
c5.4xlarge.search	100 MiB
c5.9xlarge.search	100 MiB
c5.18xlarge.search	100 MiB
c6g.large.search	10 MiB
c6g.xlarge.search	100 MiB
c6g.2xlarge.search	100 MiB
c6g.4xlarge.search	100 MiB
c6g.8xlarge.search	100 MiB
c6g.12xlarge.search	100 MiB
r3.large.search	10 MiB
r3.xlarge.search	100 MiB
r3.2xlarge.search	100 MiB
r3.4xlarge.search	100 MiB
r3.8xlarge.search	100 MiB
r4.large.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
r4.xlarge.search	100 MiB
r4.2xlarge.search	100 MiB
r4.4xlarge.search	100 MiB
r4.8xlarge.search	100 MiB
r4.16xlarge.search	100 MiB
r5.large.search	100 MiB
r5.xlarge.search	100 MiB
r5.2xlarge.search	100 MiB
r5.4xlarge.search	100 MiB
r5.12xlarge.search	100 MiB
r6g.large.search	100 MiB
r6g.xlarge.search	100 MiB
r6g.2xlarge.search	100 MiB
r6g.4xlarge.search	100 MiB
r6g.8xlarge.search	100 MiB
r6g.12xlarge.search	100 MiB
r6gd.large.search	100 MiB
r6gd.xlarge.search	100 MiB
r6gd.2xlarge.search	100 MiB
r6gd.4xlarge.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
r6gd.8xlarge.search	100 MiB
r6gd.12xlarge.search	100 MiB
r6gd.16xlarge.search	100 MiB
i2.xlarge.search	100 MiB
i2.2xlarge.search	100 MiB
i3.large.search	100 MiB
i3.xlarge.search	100 MiB
i3.2xlarge.search	100 MiB
i3.4xlarge.search	100 MiB
i3.8xlarge.search	100 MiB
i3.16xlarge.search	100 MiB
or1.medium.search	10 MiB
or1.large.search	100 MiB
or1.xlarge.search	100 MiB
or1.2xlarge.search	100 MiB
or1.4xlarge.search	100 MiB
or1.8xlarge.search	100 MiB
or1.12xlarge.search	100 MiB
or1.16xlarge.search	100 MiB
or2.medium.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
or2.large.search	100 MiB
or2.xlarge.search	100 MiB
or2.2xlarge.search	100 MiB
or2.4xlarge.search	100 MiB
or2.8xlarge.search	100 MiB
or2.12xlarge.search	100 MiB
or2.16xlarge.search	100 MiB
om2.large.search	10 MiB
om2.xlarge.search	100 MiB
om2.2xlarge.search	100 MiB
om2.4xlarge.search	100 MiB
om2.8xlarge.search	100 MiB
om2.12xlarge.search	100 MiB
om2.16xlarge.search	100 MiB
im4gn.large.search	100 MiB
im4gn.xlarge.search	100 MiB
im4gn.2xlarge.search	100 MiB
im4gn.4xlarge.search	100 MiB
im4gn.8xlarge.search	100 MiB
im4gn.16xlarge.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
i4i.large.search	N/D
i4i.xlarge.search	N/D
i4i.2xlarge.search	N/D
i4i.4xlarge.search	N/D
i4i.8xlarge.search	N/D
i4i.12xlarge.search	N/D
i4i.16xlarge.search	N/D
i4i.24xlarge.search	N/D
i4i.32xlarge.search	N/D
i4g.large.search	N/D
i4g.xlarge.search	N/D
i4g.2xlarge.search	N/D
i4g.4xlarge.search	N/D
i4g.8xlarge.search	N/D
i4g.16xlarge.search	N/D
i8g.large.search	N/D
i8g.xlarge.search	N/D
i8g.2xlarge.search	N/D
i8g.4xlarge.search	N/D
i8g.8xlarge.search	N/D

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
i8g.12xlarge.search	N/D
i8g.16xlarge.search	N/D
i7i.large.search	100 MiB
i7i.xlarge.search	100 MiB
i7i.2xlarge.search	100 MiB
i7i.4xlarge.search	100 MiB
i7i.8xlarge.search	100 MiB
i7i.12xlarge.search	100 MiB
i7i.16xlarge.search	100 MiB
c7i.large.search	100 MiB
c7i.xlarge.search	100 MiB
c7i.2xlarge.search	100 MiB
c7i.4xlarge.search	100 MiB
c7i.8xlarge.search	100 MiB
c7i.12xlarge.search	100 MiB
c7i.16xlarge.search	100 MiB
m7i.large.search	100 MiB
m7i.xlarge.search	100 MiB
m7i.2xlarge.search	100 MiB
m7i.4xlarge.search	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
m7i.8xlarge.search	100 MiB
m7i.12xlarge.search	100 MiB
m7i.16xlarge.search	100 MiB
r7i.large.search	100 MiB
r7i.xlarge.search	100 MiB
r7i.2xlarge.search	100 MiB
r7i.4xlarge.search	100 MiB
r7i.8xlarge.search	100 MiB
r7i.12xlarge.search	100 MiB
r7i.16xlarge.search	100 MiB
c8g.large	100 MiB
c8g.xlarge	100 MiB
c8g.2xlarge	100 MiB
c8g.4xlarge	100 MiB
c8g.8xlarge	100 MiB
c8g.12xlarge	100 MiB
c8g.16xlarge	100 MiB
m8g.medium	100 MiB
m8g.large	100 MiB
m8g.xlarge	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
m8g.2xlarge	100 MiB
m8g.4xlarge	100 MiB
m8g.8xlarge	100 MiB
m8g.12xlarge	100 MiB
m8g.16xlarge	100 MiB
r8g.medium	100 MiB
r8g.large	100 MiB
r8g.xlarge	100 MiB
r8g.2xlarge	100 MiB
r8g.4xlarge	100 MiB
r8g.8xlarge	100 MiB
r8g.12xlarge	100 MiB
r8g.16xlarge	100 MiB
r8gd.medium	100 MiB
r8gd.large	100 MiB
r8gd.xlarge	100 MiB
r8gd.2xlarge	100 MiB
r8gd.4xlarge	100 MiB
r8gd.8xlarge	100 MiB
r8gd.12xlarge	100 MiB

Tipo de instância	Tamanho máximo das cargas úteis das solicitações HTTP
r8gd.16xlarge	100 MiB
oi2.large.search	100 MiB
oi2.xlarge.search	100 MiB
oi2.2xlarge.search	100 MiB
oi2.4xlarge.search	100 MiB
oi2.8xlarge.search	100 MiB
oi2.12xlarge.search	100 MiB
oi2.16xlarge.search	100 MiB

Cotas de tamanhos de fragmentos

A seção a seguir lista os tamanhos máximos dos fragmentos para várias famílias de instâncias.

Tipo de instância	Multi-AZ sem modo de espera	Multi-AZ com modo de espera
R5, C5, M5, C7i, M7i, R7i	N/D	65 GiB
I3, i4i, i4g, i8g, i7i	N/D	65 GiB
R6g, C6g, M6g, R6gd, C7g,M7g, R7g, R7gd, C8g, M8g, R8g, R8gd	N/D	65 GiB
OR1, OR2, OM2, OI2	N/D	N/D
Im4gn	N/D	65 GiB

Para solicitar um aumento na cota, entre em contato com o AWS Support.

Cotas de número de fragmentos

A seção a seguir lista a contagem máxima de fragmentos para OpenSearch as versões.

Versão do mecanismo	Limite	Observações
Elasticsearch 1.5 a 6.x	Não há limite padrão	
Elasticsearch 7.x	1000	O limite padrão pode ser alterado por meio da configuração do cluster max_shards_per_node.
OpenSearch 1.x a 2,15	1000	O limite padrão pode ser alterado por meio da configuração do cluster max_shards_per_node.
OpenSearch 2.17 e superior	1.000 por cada 16 GB de pilha até no máximo 4.000	O limite padrão pode ser alterado.

Limites dos processos Java

OpenSearch O serviço limita a memória de processos Java a 50% da memória total disponível (máximo de até 32 GiB). O limite superior de 32 GB não se aplica às instâncias r7g e OpenSearch otimizadas.

Para solicitar o aumento do limite, entre em contato com o AWS Support.

Usuários avançados também podem especificar a porcentagem da pilha usada para dados de campo. Para obter mais informações, consulte Configurações avançadas de cluster e JVM OutOfMemoryError.

Limites das políticas de domínio

OpenSearch O serviço limita as políticas de acesso em domínios a 100 KiB.

Instâncias reservadas no Amazon OpenSearch Service

As instâncias reservadas (RIs) no Amazon OpenSearch Service oferecem descontos significativos em comparação com as instâncias sob demanda padrão. As instâncias em si são idênticas; RIs são apenas um desconto de cobrança aplicado às instâncias sob demanda em sua conta. Para aplicativos de longa duração com uso previsível, RIs pode proporcionar economias consideráveis ao longo do tempo.

OpenSearch O serviço RIs exige prazos de um ou três anos e tem três opções de pagamento que afetam a taxa de desconto:

- Sem pagamento adiantado: você não paga adiantado. Você paga uma taxa por hora com desconto a cada hora dentro do prazo.
- Adiantamento parcial: você paga uma parte do custo inicial e paga uma taxa por hora com desconto para cada hora dentro do termo.
- Adiantamento total: você paga todos os custos iniciais. Você não paga uma taxa por hora no prazo.

De modo geral, um maior pagamento adiantado significa um desconto maior. Você não pode cancelar instâncias reservadas: ao reservá-las, você se compromete em pagar pelo termo completo, e os pagamentos adiantados não são reembolsáveis.

RIs não são flexíveis; elas se aplicam somente ao tipo exato de instância que você reserva. Por exemplo, uma reserva para oito instâncias `c5.2xlarge.search` não se aplica a dezesseis instâncias `c5.xlarge.search` ou quatro instâncias `c5.4xlarge.search`. No entanto, contas vinculadas que fazem parte de uma organização AWS Organizations podem se beneficiar de qualquer solicitação de desconto não utilizada da conta proprietária da RI, desde que os tipos de instância, região, família e tamanho correspondam. Para obter mais informações, consulte os preços e as perguntas frequentes do Amazon OpenSearch Service.

Compra de instâncias reservadas (console)

O console permite que você exiba as instâncias reservadas existentes e compre novas.

Para comprar uma reserva

1. Vá para https://aws.amazon.com, e escolha Sign In to the Console (Faça login no Console).
2. Em Analytics, escolha Amazon OpenSearch Service.

3. Escolha Reserved Instanace Leases (Locações de instância reservada) no painel de navegação.

   Nesta página, você pode exibir as reservas existentes. Se tiver muitas reservas, você poderá filtrá-las para identificar mais facilmente e exibir uma determinada reserva.

   > **Tip**
   >
   > Se você não encontrar o link Reserved Instance Leases (Locações de instância reservada), crie um domínio na Região da AWS.

4. Escolha Order Reserved Instance (Encomendar instância reservada).
5. Forneça um nome exclusivo e descritivo.
6. Escolha um tipo de instância e o número de instâncias. Para obter orientações, consulte the section called “Dimensionamento de domínios”.
7. Escolha um prazo e uma opção de pagamento. Examine os detalhes de pagamento atentamente.
8. Escolha Próximo.
9. Examine o resumo da compra com atenção. As compras de instâncias reservadas não são reembolsáveis.
10. Escolha Order (Solicitar).

Compra de instâncias reservadas (AWS CLI)

AWS CLI Tem comandos para visualizar ofertas, comprar uma reserva e visualizar suas reservas. O comando e o exemplo de resposta a seguir mostram as ofertas de um determinado Região da AWS:

```
aws opensearch describe-reserved-instance-offerings --region us-east-1
{
  "ReservedInstanceOfferings": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
          "RecurringChargeFrequency": "Hourly"
        }
```

```
      ],
      "UsagePrice": 0.0,
      "PaymentOption": "PARTIAL_UPFRONT",
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
  ]
}
```

Para obter uma explicação de cada valor de retorno, consulte a tabela a seguir.

Campo	Description
`FixedPrice`	O custo inicial da reserva.
`ReservedInstanceOfferingId`	O ID da oferta. Anote esse valor caso você queira reservar a oferta.
`RecurringCharges`	A taxa por hora da reserva.
`UsagePrice`	Um campo herdado. Para OpenSearch Serviço, esse valor é sempre 0.
`PaymentOption`	Sem adiantamento, adiantamento parcial ou adiantamento total.
`Duration`	Extensão do prazo em segundos: • 31.536.000 segundos são um ano. • 94.608.000 segundos são três anos.
`InstanceType`	O tipo de instância da reserva. Para obter informações sobre os recursos de hardware que são alocados para cada tipo de instância , consulte os preços do Amazon OpenSearch Service.
`CurrencyCode`	A moeda de `FixedPrice` e `RecurringChargeAmount` .

Este próximo exemplo compra uma reserva:

```
aws opensearch purchase-reserved-instance-offering --reserved-instance-offering-
id 1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a --reservation-name my-reservation --instance-
count 3 --region us-east-1
{
  "ReservationName": "my-reservation",
  "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a"
}
```

Por fim, você pode listar as reservas para uma determinada Região usando o seguinte exemplo:

```
aws opensearch describe-reserved-instances --region us-east-1
{
  "ReservedInstances": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "ReservationName": "my-reservation",
      "PaymentOption": "PARTIAL_UPFRONT",
      "UsagePrice": 0.0,
      "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "State": "payment-pending",
      "StartTime": 1522872571.229,
      "InstanceCount": 3,
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
  ]
}
```

> **Note**
>
> `StartTime` é tempo epoch Unix, que é o número de segundos decorridos desde a meia-noite UTC de 1° de janeiro de 1970. Por exemplo, o tempo epoch 1522872571 20:09:31 UTC é de 4 de abril de 2018. Você pode usar conversores online.

Para saber mais sobre os comandos usados nos exemplos anteriores, consulte a Referência de comandos da AWS CLI.

Comprando instâncias reservadas (AWS SDKs)

O AWS SDKs (exceto o Android e o iOS SDKs) oferece suporte a todas as operações definidas na Amazon OpenSearch Service API Reference, incluindo as seguintes:

- `DescribeReservedInstanceOfferings`
- `PurchaseReservedInstanceOffering`
- `DescribeReservedInstances`

Esse exemplo de script usa o cliente Python OpenSearchServicede baixo nível da para comprar AWS SDK para Python (Boto3) instâncias reservadas. Você deve fornecer um valor para `instance_type`:

```
import boto3
from botocore.config import Config

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-east-1'
)

client = boto3.client('opensearch', config=my_config)

instance_type = '' # e.g. m4.2xlarge.search
```

```
def describe_RI_offerings(client):
    """Gets the Reserved Instance offerings for this account"""

    response = client.describe_reserved_instance_offerings()
    offerings = (response['ReservedInstanceOfferings'])
    return offerings


def check_instance(offering):
    """Returns True if instance type is the one you specified above"""

    if offering['InstanceType'] == instance_type:
        return True

    return False


def get_instance_id():
    """Iterates through the available offerings to find the ID of the one you
 specified"""

    instance_type_iterator = filter(
        check_instance, describe_RI_offerings(client))
    offering = list(instance_type_iterator)
    id = offering[0]['ReservedInstanceOfferingId']
    return id


def purchase_RI_offering(client):
    """Purchase Reserved Instances"""

    response = client.purchase_reserved_instance_offering(
        ReservedInstanceOfferingId = get_instance_id(),
        ReservationName = 'my-reservation',
        InstanceCount = 1
    )
    print('Purchased reserved instance offering of type ' + instance_type)
    print(response)


def main():
    """Purchase Reserved Instances"""
    purchase_RI_offering(client)
```

Para obter mais informações sobre como instalar e usar o AWS SDKs, consulte Kits AWS de desenvolvimento de software.

Verificação dos custos

Cost Explorer é uma ferramenta gratuita que você pode usar para exibir os dados de gastos nos últimos 13 meses. A análise desses dados ajuda você a identificar tendências e entender se é RIs adequado ao seu caso de uso. Se você já tem RIs, você pode agrupar por opção de compra e mostrar os custos amortizados para comparar esses gastos com seus gastos com instâncias sob demanda. Você também pode definir orçamentos de uso para garantir que está aproveitando suas reservas. Para saber mais, consulte Análise dos custos com o Cost Explorer no Manual do usuário do AWS Billing .

Técnicas de otimização de custos para o Amazon OpenSearch Service

A seguir estão algumas das técnicas mais usadas para otimizar custos ao usar o Amazon OpenSearch Service — tanto clusters gerenciados quanto sem servidor. Como cada carga de trabalho é única, avalie essas estratégias em relação aos seus padrões de uso específicos e valide-as em um ambiente de teste antes de aplicá-las à produção.

Tópicos

- Otimização de custos para clusters gerenciados do Amazon OpenSearch Service
- Otimização de custos para Amazon OpenSearch Service Serverless

Otimização de custos para clusters gerenciados do Amazon OpenSearch Service

Fonte derivada — Ignore o armazenamento do campo _source

A fonte derivada é um recurso de otimização de armazenamento que elimina a sobrecarga de armazenar o `_source` campo:

- OpenSearch armazena cada documento ingerido duas vezes: uma vez no `_source` campo (documento bruto) e uma vez como campos indexados para pesquisa.
- O `_source` campo sozinho pode consumir um espaço de armazenamento significativo — geralmente de 30 a 50% do armazenamento total do índice.

- Com o Derived Source, você pula o armazenamento `_source` e, em vez disso, o reconstrói dinamicamente a partir de campos indexados quando necessário (durante operações de pesquisa, obtenção, mget, reindexação ou atualização).
- Isso é opcional, ativado na criação do índice usando configurações de índice composto.
- Disponível em todas as regiões em que a OpenSearch versão 3.1 ou posterior é suportada.

Ideal para: cargas de trabalho analíticas e de registro nas quais você não precisa recuperar o documento original bruto com frequência, mas ainda precisa pesquisar e agregar campos.

Para obter mais informações, consulte a documentação de código aberto e o anúncio do What's New.

OR1 / OR2 / OM2 instances — OpenSearch -famílias de instâncias otimizadas

OR1 e as OM2 instâncias mais recentes OR2 usam o Amazon S3 para armazenamento de réplicas por meio da replicação de segmentos:

- OR2: Taxa de transferência de indexação até 26% maior do que as OR1 instâncias R7g.
- OM2: taxa de transferência de indexação até 15% maior do que as OR1 instâncias M7g.
- Ambos usam a mesma arquitetura: EBS local para armazenamento primário e S3 para durabilidade.
- Elimina o custo de armazenamento de réplicas — réplicas armazenadas no S3 (durabilidade de 11 a nove) em vez de volumes caros do EBS.
- Melhoria de até 30% no preço-desempenho em relação às instâncias da geração anterior.
- Suporta instantâneos superficiais v2 — instantâneos quase instantâneos sem sobrecarga. I/O

Ideal para: análises operacionais pesadas e cargas de trabalho de análise de registros.

Para obter mais informações, consulte o anúncio OR2 e OM2 o tópico O que há de novo e o guia do the section called “OpenSearch instâncias otimizadas” desenvolvedor.

Rollups de índices — dados agregados de séries temporais históricas

Os pacotes cumulativos de índices resumem e compactam dados de séries temporais mais antigos em intervalos de tempo mais aproximados, reduzindo drasticamente o volume de armazenamento:

- Dados de IoT/sensores: mantenha os dados por segundo em um armazenamento dinâmico dos últimos períodos; faça resumos horários ou diários dos dados mais antigos.

- Métricas do sistema: mantenha métricas detalhadas dos últimos 30 dias; agregue dados antigos em resumos horários ou diários.
- Dados de registro: preserve todos os detalhes da janela ativa de solução de problemas (por exemplo, 1 semana); mantenha padrões de erro resumidos para períodos anteriores.
- Combine com as políticas do ISM para automatizar a migração cumulativa e hierárquica em uma única política de ciclo de vida.
- Maior economia ao agregar de segundos a horas versus segundos a minutos.

Para obter mais informações, consulte a postagem do blog Index Rollups.

Gerenciamento do estado do índice — automatize o ciclo de vida completo dos dados

As políticas do ISM automatizam a movimentação de índices por meio de níveis de armazenamento e ações de ciclo de vida:

- Migre índices automaticamente: de quente para frio UltraWarm para excluir, com base na idade, tamanho ou contagem de documentos.
- Acione os rollups antes das transições de nível para reduzir o volume de dados.
- Defina políticas de rollover (por exemplo, quando um índice atinge 50 GB ou tem 30 dias) para controlar o crescimento do índice.
- Automatize a mesclagem forçada em índices somente para leitura para recuperar o armazenamento de documentos excluídos.
- Combine com pacotes cumulativos para obter o máximo de economia em grandes conjuntos de dados de séries temporais.

Instâncias reservadas — comprometa-se com descontos previsíveis

Para cargas de trabalho analíticas estáveis e previsíveis, as instâncias reservadas oferecem descontos significativos em relação aos preços sob demanda:

- Termos de compromisso de 1 ou 3 anos sem opções de pagamento adiantado, adiantado parcial ou totalmente adiantado.
- Ideal para nós de dados de camada ativa e nós mestres dedicados que funcionam continuamente.
- Use a Calculadora AWS de Preços para estimar a economia antes de se comprometer.
- As instâncias reservadas são um desconto de cobrança aplicado às instâncias sob demanda, sem necessidade de alterações na infraestrutura.

Tipos e contagem de instâncias do tamanho certo

Principais orientações do OpenSearch Well-Architected Lens e as melhores práticas de dimensionamento correto:

- Sempre use a última geração de instâncias (por exemplo, as instâncias Graviton3 oferecem desempenho até 25% melhor do que as instâncias baseadas em Graviton2).
- Use volumes EBS gp3 em vez de gp2 — melhor desempenho a um custo menor sem custo adicional.
- Combine o tipo de instância com a carga de trabalho: otimizado para memória para pesquisas pesadas, otimizado para computação para indexação pesada.
- Avalie os nós dedicados do gerenciador de cluster: necessários apenas para 3 ou mais nós de dados; evite provisionar em excesso o tamanho do nó principal.
- Monitore CloudWatch métricas para detectar o excesso de provisionamento: CPU sustentada abaixo de 40%, JVM abaixo de 50% e armazenamento abaixo de 50% são sinais de desperdício.
- Intervalos ideais: CPU de 60 a 80%, JVM de 65 a 85%, armazenamento de 70 a 85% para cargas de trabalho sustentadas.

Para obter mais informações, consulte a postagem do blog sobre as melhores práticas de dimensionamento correto.

Otimização de fragmentos — Evite fragmentação excessiva

A fragmentação excessiva é um fator de custo oculto — muitos fragmentos pequenos desperdiçam CPU, memória e pilha de JVM:

- Tamanhos de fragmento recomendados: 10—50 GiB por fragmento, dependendo da carga de trabalho.
- Não mais do que 25 fragmentos por GiB de heap Java, não mais do que 1.000 fragmentos por nó de dados.
- Use as políticas de rollover do ISM para controlar o crescimento do índice e evitar a proliferação ilimitada de fragmentos.
- Reduza o número de réplicas onde a durabilidade permitir (OR1/OR2 as instâncias eliminam totalmente a necessidade de réplicas).
- Use a mesclagem forçada em índices somente de leitura para reduzir a contagem de segmentos e recuperar o armazenamento.

Para obter mais informações, consulte Quantos fragmentos eu preciso?

Zero-ETL//Consulta direta com o Amazon S3

Para dados que raramente são consultados, mas que precisam permanecer acessíveis, o Direct Query (sem ETL com S3) permite consultar dados do S3 diretamente sem ingeri-los: OpenSearch

- Sem custo de ingestão — os dados permanecem no S3.
- Sem custo de armazenamento de alto nível para dados de arquivamento.
- Pay-per-query modelo computacional.
- Suporta OpenSearch painéis para visualização.
- A latência de segundos ou minutos é aceitável — não para casos de uso em tempo real.

Amostragem e compressão na ingestão

Reduza os custos antes mesmo que os dados cheguem a OpenSearch:

- Amostragem: consuma somente um subconjunto representativo de fluxos de registros de alto volume (por exemplo, 10% dos registros de depuração).
- Compressão de índice: ative o melhor codec de compressão para reduzir o espaço de armazenamento.
- Filtragem de campos: elimine campos de alta cardinalidade e baixo valor antes da indexação (por exemplo, rastreamentos de pilha brutos para registros antigos).
- Políticas de retenção: defina janelas máximas de retenção alinhadas aos requisitos de conformidade — nunca armazene dados indefinidamente.

Evite custos de suporte estendidos — mantenha-se atualizado sobre as versões do motor

O Amazon OpenSearch Service cobra uma taxa fixa por hora de instância normalizada para as versões do mecanismo no Extended Support:

- Permanecer em versões mais antigas e sem suporte gera cobranças adicionais além dos custos da instância.
- Atualize para as versões atuais suportadas para evitar taxas de Extended Support.

Etiquetas e monitoramento de alocação de custos CloudWatch

A governança proativa de custos evita o desperdício:

- Aplique etiquetas de alocação de custos aos OpenSearch domínios para monitorar detalhadamente os custos por equipe ou carga de trabalho.
- Defina CloudWatch alarmes para utilização do armazenamento, pressão da JVM e CPU para detectar o excesso de provisionamento mais cedo.
- Use o AWS Cost Explorer para identificar domínios com baixa utilização consistente.
- Avalie o ajuste automático — ajusta automaticamente o tamanho da pilha da JVM e outras configurações para melhorar o desempenho e reduzir o desperdício de recursos.

Otimização de custos para Amazon OpenSearch Service Serverless

Pesquisa vetorial otimizada em disco (coleções de vetores)

A pesquisa vetorial otimizada em disco é uma das técnicas mais poderosas de redução de custos para cargas de trabalho vetoriais. Ele executa a pesquisa vetorial por uma fração do custo do modo na memória, mantendo somente vetores compactados na RAM e armazenando vetores de precisão total no disco.

Como funciona:

- No modo standard (`in_memory`), o gráfico HNSW completo é carregado na RAM, o que se torna proibitivamente caro em grande escala.
- No `on_disk` modo, somente vetores compactados (quantizados) são mantidos na memória para geração de candidatos; vetores de precisão total são recuperados do disco somente para a fase final de repontuação (pesquisa em duas fases).
- Isso reduz drasticamente os requisitos de RAM e, ao mesmo tempo, mantém a alta qualidade da pesquisa.
- O `on_disk` modo padrão usa quantização binária de 32x — reduzindo os requisitos de memória em 97% em relação ao modo na memória.
- Suporta níveis de compressão: 2x (FP16), 4x (byte), 8x, 16x, 32x (binário).
- Latência P90 na faixa de 100 a 200 ms — adequada para cargas de trabalho que não exigem tempos de resposta de um dígito em milissegundos.

Benchmarks de economia de custos:

Conjunto de dados	Relembre @100	Latência P90	Redução de custos
Coerência TREC-RAG	0,94	104 ms	83%
Tabelas B-1M	0,96	7 ms	67%
Marco-1M	0,99	7 ms	67%

Ideal para: pipelines de RAG, pesquisa semântica, recuperação de documentos e qualquer carga de trabalho vetorial em que a latência P90 de 100 a 200 ms seja aceitável e a redução de custos seja uma prioridade.

> **Note**
>
> Para aplicar essa alteração aos dados indexados existentes, você precisa reindexar. Você pode usar uma ferramenta de pipeline externa, como reindexar dados em um novo índice.

Para obter mais informações, consulte a postagem do blog Disk-Optimized Vector Search, a postagem no blog Quantization Techniques e. the section called “Trabalho com coleções de pesquisa vetorial”

Otimização automática de vetores (coleções de vetores)

A otimização automática avalia automaticamente as configurações do índice vetorial e recomenda a melhor compensação entre qualidade de pesquisa, latência e custo de memória, sem exigir experiência em vetores:

- Oferece recomendações de otimização em menos de uma hora.
- Integrado com tubulações de ingestão de vetores.
- Pode ser combinado com a indexação acelerada por GPU para bancos de dados vetoriais em escala de bilhões.

Para obter mais informações, consulte a postagem do blog Auto-Optimize.

Indexação vetorial acelerada por GPU (coleções de vetores)

A aceleração da GPU transfere a indexação vetorial HNSW para a tecnologia sem servidor GPUs, reduzindo drasticamente o tempo e o custo de OCU para criar grandes índices vetoriais:

- Tempos de criação de índices de 6,4 a 13,8 vezes mais rápidos em comparação com a indexação somente com CPU.
- Custo de OCU de indexação até 75% menor para cargas de trabalho vetoriais com muita gravação.
- GPUs são conectados dinamicamente — você só paga OCUs quando a aceleração da GPU está ativa.
- Permite que bancos de dados vetoriais em escala de bilhões sejam criados em menos de uma hora.
- Carregado separadamente como aceleração vetorial OCUs.

Ideal para: ingestão inicial de vetores em grande escala ou cenários frequentes de retreinamento de modelos em que a reconstrução de índices é cara.

Para obter mais informações, consulte a postagem no blog sobre aceleração de GPU.

Defina limites máximos de OCU (todos os tipos de coleção)

O Amazon OpenSearch Service Serverless escala automaticamente OCUs com base na demanda. Sem um limite, os custos podem aumentar inesperadamente. Defina um limite máximo de OCU no nível da conta ou por grupo de coleta para evitar uma escalabilidade descontrolada. O sistema se expande até esse limite durante picos de carga, mas não o excederá.

Valores permitidos:

- Nível da conta: qualquer valor de até 1.700 OCUs (não restrito a múltiplos de 16).
- Grupos de coleta: 1, 2, 4, 8, 16 e múltiplos de 16 até OCUs 1.696.

Monitore CloudWatch métricas (`OCUUtilization`) para dimensionar corretamente sua configuração máxima de OCU ao longo do tempo.

> **Note**
>
> Se a utilização atingir o limite máximo da OCU, o desempenho poderá diminuir significativamente, mesmo que os custos estejam contidos. Atingir o limite não resolve a causa subjacente do pico da OCU. Para coleções de vetores, a causa raiz geralmente são os vetores na memória, que devem ser tratados diretamente otimizando a indexação vetorial, reduzindo o tamanho do índice ou ajustando as compensações de recuperação e precisão.

> **Tip**
>
> Comece com uma OCU máxima conservadora e aumente somente quando CloudWatch mostrar uma utilização sustentada perto do limite. Se você atingir o limite de forma consistente, investigue a causa raiz — particularmente o uso de vetores na memória para coleções de vetores — em vez de simplesmente aumentar o limite.

Para obter mais informações, consulte the section called “Gerenciamento de limites de capacidade”.

Otimize o período de retenção (coleções de séries temporais)

As políticas de ciclo de vida dos dados excluem automaticamente os dados das coleções de séries temporais após um período de retenção especificado, evitando o crescimento ilimitado do armazenamento. Somente as coleções de séries temporais dão suporte às políticas de ciclo de vida dos dados; as coleções de buscas e vetoriais não.

A contagem de OCU para coleções de séries temporais é diretamente determinada pela quantidade de dados recentes que devem ser mantidos no armazenamento local. As coleções de séries temporais mantêm somente a parte mais recente dos dados no armazenamento local da OCU; os dados mais antigos são transferidos para o S3 e o número de dados é dimensionado de acordo: OCUs

OCUs obrigatório = máximo (mínimo OCUs, OCUs necessário para manter os dados dentro de sua janela de retenção)

Configurando políticas de ciclo de vida de dados:

- Defina períodos de retenção de 24 horas a 3.650 dias por índice ou padrão de índice.

- O Amazon OpenSearch Service Serverless exclui dados automaticamente com base no melhor esforço (normalmente dentro de 48 horas ou 10% do período de retenção).
- As regras podem ser aplicadas no nível da coleção, no nível do padrão do índice ou no nível do índice individual — regras mais específicas têm precedência.

Exemplo de dimensionamento:

- 1 TiB/day ingestão com retenção de 30 dias = aproximadamente 1 TiB de dados ativos = 20 OCUs para indexação + 20 para pesquisa. OCUs
- Reduzir para uma retenção de 7 dias = aproximadamente 233 GiB de dados ativos = aproximadamente 4 OCUs para indexação + 4 para pesquisa. OCUs

Retenção mais curta significa menos dados ativos no armazenamento local, menos OCUs necessidade e menor custo computacional. Alinhe os períodos de retenção aos requisitos reais de negócios e conformidade — por padrão, não retenha dados indefinidamente.

Para obter mais informações, consulte the section called “Usar políticas de ciclo de vida de dados”.

Evite armazenar dados desnecessários (todos os tipos de coleta)

Reduzir o volume de dados ingeridos reduz diretamente os custos de computação (OCUs) e armazenamento:

- Filtrar campos na ingestão: use pipelines para eliminar campos de baixo valor antes que eles cheguem à coleção.
- Evite ingerir dados duplicados ou redundantes: desduplique no nível do pipeline.
- Use mapeamentos de índice apropriados: desative a indexação em campos que são armazenados, mas nunca pesquisados (). `"index": false`
- Para coleções de pesquisa: evite armazenar grandes bolhas binárias ou texto bruto que aumente o armazenamento sem valor de pesquisa.

Grupos de coleta para cargas de trabalho de vários locatários (todos os tipos de coleção)

Os grupos de coleções permitem que várias coleções com chaves KMS diferentes compartilhem recursos da OCU dentro do mesmo limite de segurança, reduzindo drasticamente os custos de

arquiteturas multilocatárias. Aplicável para clientes que usam várias chaves KMS por inquilino ou por coleção:

- Anteriormente, cada chave KMS exclusiva exigia uma chave dedicada OCUs , tornando o isolamento por inquilino proibitivamente caro.
- Com grupos de coleta, inquilinos com chaves de criptografia separadas podem compartilhar a capacidade da OCU.
- Economia de custos de até 90% para grandes quantidades de cargas de trabalho de locatários menores.
- Suporta tanto o mínimo OCUs (linha de base garantida, sem partidas a frio) quanto o máximo OCUs (limite de custo).
- Coleções com diferentes tipos de acesso à rede (pública e VPC) podem coexistir no mesmo grupo.
- CloudWatch as métricas fornecem visibilidade por grupo sobre o consumo e a latência de recursos.

Ideal para: provedores de SaaS, plataformas multilocatárias ou qualquer carga de trabalho com muitas coleções pequenas, cada uma exigindo sua própria chave KMS.

Para obter mais informações, consulte a postagem no blog do Collection Groups, o anúncio do que há de novo the section called “Grupos de coleções” e.

Outros recursos suportados no Amazon OpenSearch Service

Este tópico descreve recursos adicionais aos quais o Amazon OpenSearch Service oferece suporte.

bootstrap.memory_lock

OpenSearch O serviço é `bootstrap.memory_lock` ativado`opensearch.yml`, o que bloqueia a memória JVM e impede que o sistema operacional a troque por disco. Isso se aplica a todos os tipos de instância compatíveis, exceto os seguintes:

- `t2.micro.search`
- `t2.small.search`
- `t2.medium.search`
- `t3.small.search`
- `t3.medium.search`

Módulo de scripting

OpenSearch O serviço oferece suporte a scripts para o Elasticsearch 5. domínios x e posteriores. Esse serviço não oferece suporte ao desenvolvimento de scripts para as versões 1.5 ou 2.3.

As opções de script compatíveis incluem as seguintes:

- Painless
- Lucene Expressions
- Mustache

Para domínios do Elasticsearch 5.5 e versões posteriores, e para todos os OpenSearch domínios, o OpenSearch Service oferece suporte a scripts armazenados usando o endpoint. `_scripts` Os domínios do Elasticsearch 5.3 e 5.1 oferecem suporte somente a scripts em linha.

Transporte com TLS

OpenSearch O serviço oferece suporte a HTTP na porta 80 e HTTPS pela porta 443, mas não oferece suporte ao transporte TLS.

Tutoriais OpenSearch do Amazon Service

Este capítulo inclui vários start-to-finish tutoriais para trabalhar com o Amazon OpenSearch Service, incluindo como migrar para o serviço, criar um aplicativo de pesquisa simples e criar uma visualização em painéis. OpenSearch

Tópicos

- Tutorial: Criação e pesquisa de documentos no Amazon OpenSearch Service
- Tutorial: Migração para o Amazon Service OpenSearch
- Tutorial: Criando um aplicativo de pesquisa com o Amazon OpenSearch Service
- Tutorial: Visualizando chamadas de suporte ao cliente com OpenSearch serviços e painéis OpenSearch

Tutorial: Criação e pesquisa de documentos no Amazon OpenSearch Service

Neste tutorial, você aprende a criar e pesquisar um documento no Amazon OpenSearch Service. Você adiciona dados a um índice na forma de um documento JSON. OpenSearch O serviço cria um índice em torno do primeiro documento que você adiciona.

Este tutorial explica como fazer solicitações HTTP para criar documentos, gerar automaticamente um ID para um documento e realizar pesquisas básicas e avançadas em seus documentos.

> **Note**
>
> Este tutorial usa um domínio com acesso aberto. Para obter o mais alto nível de segurança, recomendamos colocar o domínio em uma nuvem privada virtual (VPC).

Pré-requisitos

Este tutorial tem os seguintes pré-requisitos:

- Você deve ter um Conta da AWS.
- Você deve ter um domínio OpenSearch de serviço ativo.

Adicionar um documento a um índice

Para adicionar um documento a um índice, você pode usar qualquer ferramenta HTTP, como Postman, cURL ou OpenSearch o console Dashboards. Esses exemplos pressupõem que você esteja usando o console do desenvolvedor em OpenSearch painéis. Se você estiver usando uma ferramenta diferente, ajuste adequadamente fornecendo o URL completo e as credenciais, se necessário.

Para adicionar um documento a um índice

1. Navegue até o URL dos OpenSearch painéis do seu domínio. Você pode encontrar o URL no painel do domínio no console OpenSearch de serviço. O URL segue este formato:

   ```
   domain-endpoint/_dashboards/
   ```

2. Faça login usando o nome de usuário principal e a senha.
3. Abra o painel de navegação esquerdo e escolha Ferramentas de desenvolvimento.
4. O verbo HTTP para criar um novo recurso é PUT. É ele que deve ser usado para criar um novo documento e um índice. Insira o seguinte comando no console:

   ```
   PUT fruit/_doc/1
   {
     "name":"strawberry",
     "color":"red"
   }
   ```

   A solicitação PUT cria um índice chamado fruit e adiciona um único documento ao índice com um ID de 1. Ele produz a seguinte resposta:

   ```
   {
     "_index" : "fruit",
     "_type" : "_doc",
     "_id" : "1",
     "_version" : 1,
     "result" : "created",
     "_shards" : {
       "total" : 2,
       "successful" : 2,
       "failed" : 0
     },
   ```

```
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Criação gerada automaticamente IDs

OpenSearch O serviço pode gerar automaticamente uma identificação para seus documentos. O comando a ser gerado IDs usa uma solicitação POST em vez de uma solicitação PUT e não requer ID de documento (em comparação com a solicitação anterior).

Insira a seguinte solicitação no console do desenvolvedor:

```
POST veggies/_doc
{
  "name":"beet",
  "color":"red",
  "classification":"root"
}
```

Essa solicitação cria um índice chamado veggies e adiciona o documento ao índice. Ele produz a seguinte resposta:

```
{
  "_index" : "veggies",
  "_type" : "_doc",
  "_id" : "3WgyS4IB5DLqbRIvLxtF",
  "_version" : 1,
  "result" : "created",
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Observe o campo adicional `_id` na resposta, que indica que um ID foi criado automaticamente.

> **Note**
>
> Você não acrescenta nada depois de `_doc` no URL, onde o ID normalmente é adicionado. Como está criando um documento com um ID gerado, você ainda não fornece um. Isso está reservado para atualizações.

Atualizar um documento com um comando POST

Para atualizar um documento, use um comando HTTP `POST` com o número do ID.

Primeiro, crie um documento com ID `42`:

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow"
}
```

Em seguida, use esse ID para atualizar o documento:

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow",
  "classification":"berries"
}
```

Esse comando atualiza o documento com o novo campo `classification`. Ele produz a seguinte resposta:

```
{
  "_index" : "fruits",
  "_type" : "_doc",
  "_id" : "42",
  "_version" : 2,
  "result" : "updated",
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "failed" : 0
```

```
  },
  "_seq_no" : 1,
  "_primary_term" : 1
}
```

> **Note**
>
> Se você tentar atualizar um documento que não existe, o OpenSearch Service cria o documento.

Executar ações em massa

Você pode usar a operação da API `POST _bulk` para executar várias ações em um ou mais índices em uma solicitação. Os comandos de ação em massa têm o seguinte formato:

```
POST /_bulk
<action_meta>\n
<action_data>\n
<action_meta>\n
<action_data>\n
```

Cada ação requer duas linhas de JSON. Primeiro, é necessário fornecer a descrição ou os metadados da ação. Na próxima linha, você deve fornecer os dados. Cada parte é separada por uma nova linha (\n). Uma descrição de ação de uma inserção pode ser semelhante a esta:

```
{ "create" : { "_index" : "veggies", "_type" : "_doc", "_id" : "7" } }
```

E a próxima linha contendo os dados pode ter a seguinte aparência:

```
{ "name":"kale", "color":"green", "classification":"leafy-green" }
```

Juntos, os metadados e os dados representam uma única ação em uma operação em massa. Você pode realizar várias operações em uma solicitação, como esta:

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "35" } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "36" } }
```

```
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "37" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "38" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "39" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
{ "delete" : { "_index" : "vegetables", "_id" : "1" } }
```

Observe que a última ação é `delete`. Não há dados seguindo a ação `delete`.

Pesquisando documentos

Agora que os dados existem no seu cluster, você pode procurá-los. Por exemplo, talvez você queira pesquisar todos os tubérculos ou obter uma contagem de todas as folhas verdes ou encontrar o número de erros registrados por hora.

Pesquisas básicas

Uma pesquisa básica é semelhante a esta:

```
GET veggies/_search?q=name:l*
```

A solicitação produz uma resposta JSON que contém o documento lettuce.

Pesquisas avançadas

É possível realizar pesquisas mais avançadas fornecendo as opções de consulta como JSON no corpo da solicitação:

```
GET veggies/_search
{
  "query": {
    "term": {
      "name": "lettuce"
    }
  }
}
```

Este exemplo também produz uma resposta JSON com o documento lettuce.

Classificar

É possível executar mais desse tipo de consulta usando a classificação. Primeiro, é necessário recriar o índice, porque o mapeamento automático de campo escolheu tipos que não podem ser classificados por padrão. Envie as seguintes solicitações para excluir e recriar o índice:

```
DELETE /veggies

PUT /veggies
{
   "mappings":{
      "properties":{
         "name":{
            "type":"keyword"
         },
         "color":{
            "type":"keyword"
         },
         "classification":{
            "type":"keyword"
         }
      }
   }
}
```

Em seguida, preencha novamente o índice com dados:

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "7"  } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "8" } }
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "9" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "10" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "11" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
```

Agora você pode pesquisar com uma classificação. Esta solicitação adiciona uma classificação crescente:

```
GET veggies/_search
{
```

```
  "query" : {
    "term": { "color": "green" }
  },
  "sort" : [
      "classification"
  ]
}
```

Recursos relacionados

Para saber mais, consulte os seguintes recursos:

- Introdução
- Indexação de dados
- Pesquisa de dados

Tutorial: Migração para o Amazon Service OpenSearch

Os instantâneos de índice são uma forma popular de migrar de um cluster Elasticsearch autogerenciado OpenSearch ou legado para o Amazon Service. OpenSearch Em termos gerais, o processo consiste nas seguintes etapas:

1. Faça um snapshot do cluster existente e carregue do snapshot para um bucket do Amazon S3.
2. Crie um domínio OpenSearch de serviço.
3. Conceda ao OpenSearch serviço permissões para acessar o bucket e garanta que você tenha permissões para trabalhar com snapshots.
4. Restaure o instantâneo no domínio do OpenSearch Serviço.

Esta demonstração fornece etapas mais detalhadas e opções alternativas, quando aplicável.

Obter e carregar do snapshot

Embora você possa usar o plug-in repository-s3 para tirar instantâneos diretamente no S3, você precisa instalar o plug-in em cada nó, ajustar `opensearch.yml` (ou `elasticsearch.yml` se estiver usando um cluster do Elasticsearch), reiniciar cada nó, adicionar suas credenciais e, finalmente, tirar o instantâneo. AWS O plug-in é uma ótima opção para uso contínuo ou para migrar clusters maiores.

Para clusters menores, uma abordagem única é tirar um instantâneo do sistema de arquivos compartilhado e, em seguida, usá-lo AWS CLI para carregá-lo no S3. Se você já tiver um snapshot, avance para a etapa 4.

Para obter um snapshot e carregar no Amazon S3

1. Adicione a configuração `path.repo` ao `opensearch.yml` (ou ao `Elasticsearch.yml`) em todos os nós e, em seguida, reinicie cada nó.

```
path.repo: ["/my/shared/directory/snapshots"]
```

2. Registre um repositório de snapshots, o que é obrigatório para poder tirar um snapshot. Um repositório é apenas um local de armazenamento: um sistema de arquivos compartilhados, o Amazon S3, o Sistema de Arquivos Distribuído do Hadoop (HDFS) etc. Nesse caso, usaremos um sistema de arquivos compartilhados (“fs”):

```
PUT _snapshot/my-snapshot-repo-name
{
  "type": "fs",
  "settings": {
    "location": "/my/shared/directory/snapshots"
  }
}
```

3. Faça o snapshot:

```
PUT _snapshot/my-snapshot-repo-name/my-snapshot-name
{
  "indices": "migration-index1,migration-index2,other-indices-*",
  "include_global_state": false
}
```

4. Instale a AWS CLI, e execute `aws configure` para adicionar suas credenciais.

5. Navegue até o diretório de snapshots. Depois disso, execute os seguintes comandos para criar um novo bucket do S3 e carregar do conteúdo do diretório de snapshots para esse bucket:

```
aws s3 mb s3://amzn-s3-demo-bucket --region us-west-2
aws s3 sync . s3://amzn-s3-demo-bucket --sse AES256
```

Dependendo do tamanho do snapshot e da velocidade da sua conexão com a Internet, essa operação pode demorar um pouco.

Criar um domínio

Embora o console seja a maneira mais fácil de criar um domínio, nesse caso, você já tem o terminal aberto e AWS CLI instalado. Modifique o seguinte comando para criar um domínio que atenda às suas necessidades:

```
aws opensearch create-domain \
  --domain-name migration-domain \
  --engine-version OpenSearch_1.0 \
  --cluster-config InstanceType=c5.large.search,InstanceCount=2 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-user-password}' \
  --access-policies '{"Version": "2012-10-17",       "Statement":
[{"Effect":"Allow","Principal": {"AWS": "arn:aws:iam::aws-
region:user/UserName"},"Action":["es:ESHttp*"],"Resource":"arn:aws:es:aws-
region:111122223333:domain/migration-domain/*"}]}' \
  --region aws-region
```

Da maneira em que se encontra, o comando cria um domínio acessível à Internet com dois nós de dados, cada um com 100 GiB de armazenamento. Ele também habilita o controle de acesso refinado com autenticação básica de HTTP e todas as configurações de criptografia. Use o console OpenSearch de serviço se precisar de uma configuração de segurança mais avançada, como uma VPC.

Antes de emitir o comando, altere o nome do domínio, as credenciais do usuário mestre e o número da conta. Especifique o mesmo Região da AWS que você usou para o bucket do S3 e uma OpenSearch/Elasticsearch versão compatível com seu snapshot.

> **Important**
>
> Os snapshots são compatíveis somente com versões posteriores e somente com uma versão principal. Por exemplo, você não pode restaurar um snapshot de um OpenSearch 1. cluster x em um Elasticsearch 7. cluster x, somente OpenSearch 1. x ou 2. cluster x. A versão secundária também é importante. Você não pode restaurar um snapshot de um cluster 5.3.3 autogerenciado em um domínio de serviço OpenSearch 5.3.2. Recomendamos escolher a versão mais recente OpenSearch ou o Elasticsearch compatível com seu snapshot. Para obter uma tabela de versões compatíveis, consulte the section called “Como usar um snapshot para migrar dados”.

Conceder permissões para o bucket do S3

No console AWS Identity and Access Management (IAM), crie uma função com as seguintes permissões e relação de confiança. Ao criar a função, escolha S3 como o Serviço da AWS . Nomeie a função como `OpenSearchSnapshotRole` para que ela seja fácil de encontrar.

Permissões

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket"
      ]
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Effect": "Allow",
```

```
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ]
    }
  ]
}
```

Relação de confiança

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Em seguida, dê ao seu perfil do IAM pessoal permissões para assumir `OpenSearchSnapshotRole`. Crie a seguinte política e anexe-a à sua identidade:

Permissões

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
    }
  ]
}
```

Mapeie a função de snapshot em OpenSearch painéis (se estiver usando controle de acesso refinado)

Se você habilitou o controle de acesso detalhado, mesmo se usar a autenticação básica HTTP para todos os outros fins, precisará mapear o perfil do `manage_snapshots` para o seu perfil do IAM para poder trabalhar com snapshots.

Para conceder à sua identidade permissões para trabalhar com snapshots

1. Faça login nos painéis usando as credenciais de usuário mestre que você especificou ao criar o domínio do OpenSearch Serviço. Você pode encontrar o URL dos painéis no console de OpenSearch serviço. Ele segue o formato `https://domain-endpoint/_dashboards/`.
2. No menu principal, escolha Segurança, Funções e selecione a função manage_snapshots.
3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Adicione o ARN do domínio do seu perfil do IAM pessoal no campo apropriado. O ARN assume um dos seguintes formatos:

   ```
   arn:aws:iam::123456789123:user/user-name
   ```

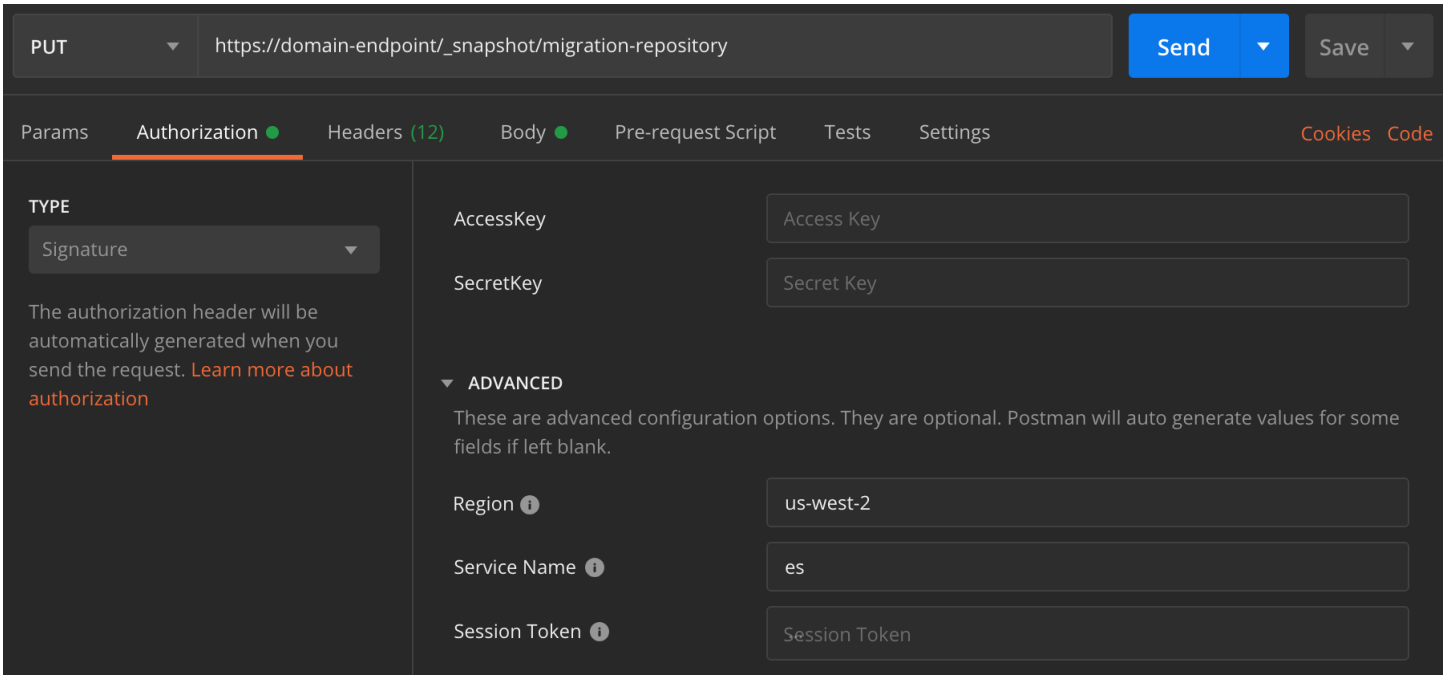
   ```
   arn:aws:iam::123456789123:role/role-name
   ```

5. Selecione Mapa e confirme se o perfil aparece em Usuários mapeados.

Restaure o snapshot

Neste momento, você tem duas maneiras de acessar seu domínio de OpenSearch serviço: autenticação básica HTTP com suas credenciais de usuário mestre ou AWS autenticação usando suas credenciais do IAM. Como os snapshots usam o Amazon S3, que não tem nenhum conceito do usuário principal, você deve usar suas credenciais do IAM para registrar o repositório de snapshots com seu domínio de serviço. OpenSearch

A maioria das linguagens de programação tem bibliotecas para ajudar com a assinatura de solicitações, mas a abordagem mais simples é usar uma ferramenta como o Postman e colocar suas credenciais do IAM na seção Autorização .



Como restaurar o snapshot

1. Independentemente de como você optar por assinar suas solicitações, a primeira etapa é registrar o repositório:

```
PUT _snapshot/my-snapshot-repo-name
{
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "region": "us-west-2",
    "role_arn": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
  }
}
```

2. Depois disso, liste os snapshots no repositório e encontre o que deseja restaurar. Neste momento, é possível continuar usando o Postman ou alternar para uma ferramenta como o curl.

   Abreviatura

   ```
   GET _snapshot/my-snapshot-repo-name/_all
   ```

   curl

```
curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/
_snapshot/my-snapshot-repo-name/_all
```

3. Restaure o snapshot.

   Abreviatura

   ```
   POST _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore
   {
     "indices": "migration-index1,migration-index2,other-indices-*",
     "include_global_state": false
   }
   ```

   curl

   ```
   curl -XPOST -u 'master-user:master-user-password' https://domain-endpoint/
   _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore \
     -H 'Content-Type: application/json' \
     -d '{"indices":"migration-index1,migration-index2,other-indices-
   *","include_global_state":false}'
   ```

4. Por fim, verifique se seus índices foram restaurados conforme o esperado.

   Abreviatura

   ```
   GET _cat/indices?v
   ```

   curl

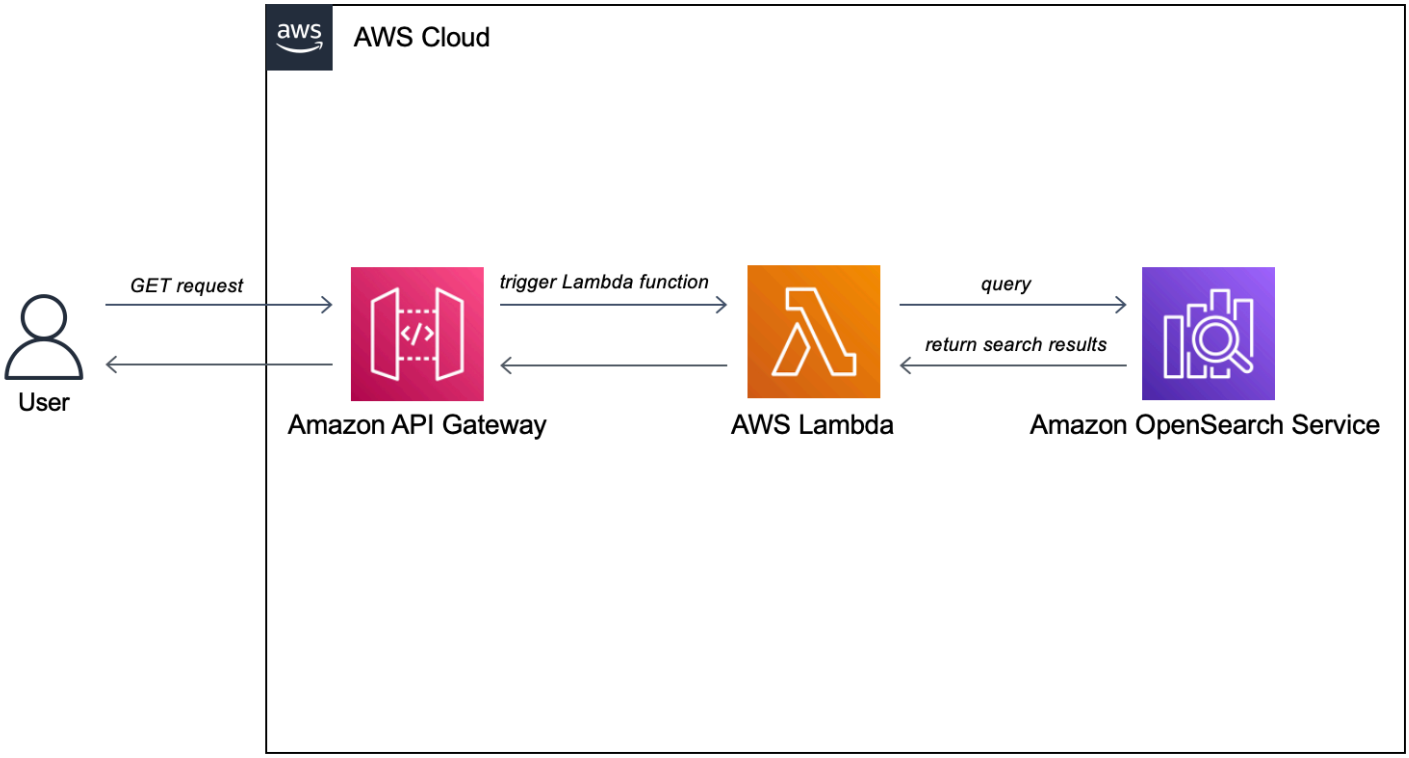
   ```
   curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/_cat/
   indices?v
   ```

Neste momento, a migração está concluída. Você pode configurar seus clientes para usar o novo endpoint de OpenSearch serviço, redimensionar o domínio de acordo com sua carga de trabalho, verificar a contagem de fragmentos de seus índices, mudar para um usuário mestre do IAM ou começar a criar visualizações em painéis. OpenSearch

Tutorial: Criando um aplicativo de pesquisa com o Amazon OpenSearch Service

Uma forma comum de criar um aplicativo de pesquisa com o Amazon OpenSearch Service é usar formulários da web para enviar consultas de usuários a um servidor. Em seguida, você pode autorizar o servidor a ligar OpenSearch APIs diretamente para o e fazer com que o servidor envie solicitações ao OpenSearch Serviço. No entanto, se desejar escrever um código do lado do cliente que não dependa de um servidor, é necessário compensar os riscos de segurança e performance. Não é aconselhável permitir o acesso público não assinado ao OpenSearch APIs . Os usuários podem acessar endpoints não seguros ou afetar a performance do cluster por meio de consultas excessivamente amplas (ou muitas consultas).

Este capítulo apresenta uma solução: use o Amazon API Gateway para restringir os usuários a um subconjunto do OpenSearch APIs e AWS Lambda assinar solicitações do API Gateway para o OpenSearch Service.



> **Note**
>
> Os preços padrão do API Gateway e do Lambda se aplicam, mas dentro do uso limitado desse tutorial, os custos devem ser insignificantes.

Pré-requisitos

Um pré-requisito para este tutorial é um domínio de OpenSearch serviço. Se você ainda não tiver um, siga as etapas em Criar um domínio OpenSearch de serviço para criar um.

Etapa 1: Indexar dados de exemplo

Faça download de sample-movies.zip e use a operação da API _bulk para adicionar os 5.000 documentos ao índice `movies`:

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/_bulk
{ "index": { "_index": "movies", "_id": "tt1979320" } }
{"directors":["Ron
 Howard"],"release_date":"2013-09-02T00:00:00Z","rating":8.3,"genres":
["Action","Biography","Drama","Sport"],"image_url":"http://ia.media-imdb.com/images/
M/MV5BMTQyMDE0MTY0OV5BMl5BanBnXkFtZTcwMjI2OTI0OQ@@._V1_SX400_.jpg","plot":"A re-
creation of the merciless 1970s rivalry between Formula One rivals James Hunt and
 Niki Lauda.","title":"Rush","rank":2,"running_time_secs":7380,"actors":["Daniel
 Brühl","Chris Hemsworth","Olivia Wilde"],"year":2013,"id":"tt1979320","type":"add"}
{ "index": { "_index": "movies", "_id": "tt1951264" } }
{"directors":["Francis Lawrence"],"release_date":"2013-11-11T00:00:00Z","genres":
["Action","Adventure","Sci-Fi","Thriller"],"image_url":"http://ia.media-imdb.com/
images/M/
MV5BMTAyMjQ3OTAxMzNeQTJeQWpwZ15BbWU4MDU0NzA1MzAx._V1_SX400_.jpg","plot":"Katniss
 Everdeen and Peeta Mellark become targets of the Capitol after
 their victory in the 74th Hunger Games sparks a rebellion in
 the Districts of Panem.","title":"The Hunger Games: Catching
 Fire","rank":4,"running_time_secs":8760,"actors":["Jennifer Lawrence","Josh
 Hutcherson","Liam Hemsworth"],"year":2013,"id":"tt1951264","type":"add"}
...
```

Observe que o exemplo acima é um comando com um pequeno subconjunto dos dados disponíveis. Para executar a operação `_bulk`, você precisa copiar e colar todo o conteúdo do arquivo `sample-movies`. Para obter instruções adicionais, consulte the section called “Opção 2: carregar vários documentos”.

Também é possível usar o seguinte comando do curl para obter o mesmo resultado:

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/_bulk' --data-binary
 @bulk_movies.json -H 'Content-Type: application/json'
```

Etapa 2: criar e implantar a função do Lambda

Antes de criar sua API no API Gateway, crie a função do Lambda para a qual ela passará as solicitações.

Criar a função do Lambda

Nessa solução, o API Gateway passa solicitações para uma função Lambda, que consulta o OpenSearch Serviço e retorna os resultados. Como essa função de exemplo usa bibliotecas externas, é necessário criar um pacote de implantação e carregar para o Lambda.

Para criar o pacote de implantação

1. Abra um prompt de comando e crie um diretório de projeto do `my-opensearch-function`. Por exemplo, no macOS:

   ```
   mkdir my-opensearch-function
   ```

2. Navegue até o diretório de projeto do `my-sourcecode-function`.

   ```
   cd my-opensearch-function
   ```

3. Copie o conteúdo do seguinte código Python de exemplo e salve-o em um novo arquivo chamado `opensearch-lambda.py`. Adicione sua região e o endpoint do host ao arquivo.

   ```
   import boto3
   import json
   import requests
   from requests_aws4auth import AWS4Auth

   region = '' # For example, us-west-1
   service = 'es'
   credentials = boto3.Session().get_credentials()
   awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
    session_token=credentials.token)
   ```

```
host = '' # The OpenSearch domain endpoint with https:// and without a trailing
 slash
index = 'movies'
url = host + '/' + index + '/_search'

# Lambda execution starts here
def lambda_handler(event, context):

    # Put the user query into the query DSL for more accurate search results.
    # Note that certain fields are boosted (^).
    query = {
        "size": 25,
        "query": {
            "multi_match": {
                "query": event['queryStringParameters']['q'],
                "fields": ["title^4", "plot^2", "actors", "directors"]
            }
        }
    }

    # Elasticsearch 6.x requires an explicit Content-Type header
    headers = { "Content-Type": "application/json" }

    # Make the signed HTTP request
    r = requests.get(url, auth=awsauth, headers=headers, data=json.dumps(query))

    # Create the response and add some extra content to support CORS
    response = {
        "statusCode": 200,
        "headers": {
            "Access-Control-Allow-Origin": '*'
        },
        "isBase64Encoded": False
    }

    # Add the search results to the response
    response['body'] = r.text
    return response
```

4. Instale a biblioteca externa em um novo diretório de `package`.

```
pip3 install --target ./package boto3
pip3 install --target ./package requests
```

```
pip3 install --target ./package requests_aws4auth
```

5. Crie um pacote de implantação com a biblioteca instalada na raiz. O seguinte comando gera um arquivo `my-deployment-package.zip` no diretório do projeto.

```
cd package
zip -r ../my-deployment-package.zip .
```

6. Adicione o arquivo `opensearch-lambda.py` à raiz do arquivo zip.

```
cd ..
zip my-deployment-package.zip opensearch-lambda.py
```

Para saber mais sobre a criação de funções do Lambda e pacotes de implantação, consulte Implantar funções do Lambda em Python com arquivos .zip no Guia do desenvolvedor do AWS Lambda e the section called “Criar o pacote de implantação do Lambda” neste guia.

Para criar sua função usando o console do Lambda

1. Navegue até o console Lambda em casahttps://console.aws.amazon.com/lambda/. No painel de navegação à esquerda, escolha Funções.
2. Selecione Criar função.
3. Configure os campos a seguir.
   - Nome da função: opensearch-function
   - Runtime: Python 3.9
   - Arquitetura: x86_64

   Mantenha todas as outras opções padrão e escolha Criar função.
4. Na seção Fonte do código da página de resumo da função, escolha Carregar no menu suspenso e selecione .zip. file. Localize o arquivo `my-deployment-package.zip` que você criou e escolha Salvar.
5. O manipulador é o método no código da sua função que processa eventos. Em Configurações do Runtime, escolha Editar e altere o nome do manipulador de acordo com o nome do arquivo no pacote de implantação onde a função do Lambda está localizada Como seu arquivo se chama `opensearch-lambda.py`, renomeie o manipulador para *`opensearch-`*

`lambda`.lambda_handler. Para saber mais, consulte Manipulador de função do Lambda em Python.

Etapa 3: Criar a API no Gateway da API

O uso do API Gateway permite criar uma API mais limitada e simplifica o processo de interação com a OpenSearch `_search` API. O API Gateway também permite ativar recursos de segurança, como a autenticação do Amazon Cognito e a limitação de solicitações. Execute as seguintes etapas para criar e implantar uma API:

Criar e configurar a API

Para criar sua API usando o console do API Gateway

1. Navegue até o console do API Gateway em https://console.aws.amazon.com/apigateway/casa. No painel de navegação esquerdo, escolha APIs.
2. Localize a API REST (não privada) e escolha Compilar.
3. Na página seguinte, localize a seção Criar nova API e verifique se a opção Nova API está selecionada.
4. Configure os campos a seguir.
   - Nome da API: opensearch-api
   - Descrição: API pública para pesquisar um domínio do Amazon OpenSearch Service
   - Tipo do endpoint: Regional
5. Selecione Criar API.
6. Escolha Ações e Criar método.
7. Select GET no menu suspenso e clique na marca de seleção para confirmar.
8. Defina as seguintes configurações e escolha Salvar:

Configuração	Valor
Tipo de integração	Função do Lambda
Usar a integração de proxy do Lambda	Sim

Configuração	Valor
Região do Lambda	*us-west-1*
Função do Lambda	opensearch-lambda
Usar o tempo limite padrão	Sim

Configurar a solicitação de método

Escolha Solicitação de métodot e defina as seguintes configurações:

Configuração	Valor
Autorização	NONE
Validador da solicitação	Validar parâmetros e cabeçalhos da string de consulta
Chave da API necessária	false

Em Parâmetros da string de consulta do URL), escolha Adicionar string de consulta e configure o seguinte parâmetro:

Configuração	Valor
Nome	q
Obrigatório	Sim

Implante a API e configure um estágio

O console do API Gateway permite que você implante uma API criando uma implantação e associando-a a um estágio novo ou existente.

1. Escolha Ações e Implantar API.

2. Para Estágio da implantação), escolha Novo estágio e atribua o nome `opensearch-api-test` ao estágio.
3. Escolha Implantar.
4. Defina as seguintes configurações no editor de estágios e, em seguida, escolha Salvar alterações:

Configuração	Valor
Habilitar controle de utilização	Sim
Taxa	1000
Intermitência	500

Essas definições configuram uma API que possui apenas um método: uma solicitação `GET` para a raiz do endpoint (`https://`*`some-id`*`.execute-api.`*`us-west-1`*`.amazonaws.com/search-es-api-test`). A solicitação requer um único parâmetro (q), a string de consulta a ser pesquisada. Quando chamado, o método passa a solicitação para o Lambda, que executa a função `opensearch-lambda`. Para saber mais, consulte Criação de uma API no Amazon API Gateway e Implantação de uma API REST no Amazon API Gateway.

Etapa 4: (opcional) modificar a política de acesso ao domínio

Seu domínio OpenSearch de serviço deve permitir que a função Lambda faça `GET` solicitações ao `movies` índice. Se o domínio tiver uma política de acesso aberto com controle de acesso refinado habilitado, você pode deixar como está:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
```

```
      "Action": "es:*",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
  ]
}
```

Ou você pode escolher tornar a política de acesso ao domínio mais granular. Por exemplo, a política mínima a seguir fornece à `opensearch-lambda-role` (criada por meio do Lambda) acesso de leitura ao índice `movies`. Para obter o nome exato da função que o Lambda cria automaticamente, acesse o console AWS Identity and Access Management (IAM), escolha Roles e pesquise por "lambda".

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/service-role/opensearch-lambda-role-1abcdefg"
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/movies/_search"
    }
  ]
}
```

> **Important**
>
> Se você tiver um controle de acesso refinado habilitado para o domínio, também precisará mapear a função para um usuário nos OpenSearch painéis, caso contrário, você verá erros de permissão.

Configurar as permissões do perfil de execução do Lambda

Além de configurar a política de acesso ao domínio, você também deve garantir que a função de execução do Lambda tenha as permissões de IAM necessárias para acessar OpenSearch seu domínio de serviço. A função Lambda requer permissões específicas, dependendo se você está usando um domínio gerenciado ou uma coleção OpenSearch Service Serverless.

Para domínios OpenSearch de serviço gerenciados:

Anexe a seguinte política do IAM à sua função de execução do Lambda para permitir que ela faça solicitações ao seu domínio de OpenSearch serviço:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPost"
      ],
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
  ]
}
```

Para coleções OpenSearch Service Serverless:

Se você estiver usando o OpenSearch Service Serverless, anexe a seguinte política do IAM à sua função de execução do Lambda:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Action": "aoss:*",
      "Resource": "arn:aws:aoss:us-west-1:123456789012:collection/collection-id"
    }
  ]
}
```

Para anexar essas políticas ao perfil de execução do Lambda:

1. Navegue até o console do IAM em https://console.aws.amazon.com/iam/.
2. Escolha Perfis e pesquise o perfil de execução do Lambda (normalmente denominado `opensearch-lambda-role-xxxxxxxx`).
3. Selecione Adicionar permissões e depois Criar política em linha.
4. Escolha a guia JSON e cole a política apropriada acima, substituindo os valores do espaço reservado pelo seu recurso real. ARNs
5. Escolha Revisar política, forneça um nome como `0penSearchAccess` e escolha Criar política.

> **Note**
>
> Sem essas permissões do IAM, sua função Lambda receberá erros de “Acesso negado” ao tentar consultar seu domínio de OpenSearch serviço, mesmo que a política de acesso ao domínio permita as solicitações.

Para saber mais sobre políticas de acesso, consulte the section called “Configuração de políticas de acesso”.

Mapeamento da função do Lambda (se estiver usando um controle de acesso minucioso)

O controle de acesso minucioso introduz uma etapa adicional antes de testar a aplicação. Mesmo se você usar a autenticação básica do HTTP para todos os outros fins, será necessário mapear a função do Lambda para um usuário. Caso contrário, você receberá erros de permissões.

1. Navegue até o URL dos OpenSearch painéis do domínio.
2. No menu principal, escolha Segurança, Funções e selecione o link para `all_access`, a função para a qual precisa mapear a função do Lambda.

3. Escolha Usuários mapeados e Gerenciar mapeamento.
4. Em Funções de backend, adicione o nome do recurso da Amazon (ARN) da função do Lambda. O ARN deve assumir a forma de `arn:aws:iam::123456789123:role/service-role/opensearch-lambda-role-1abcdefg`.
5. Selecione Mapa e confirme se o usuário ou função aparece em Usuários mapeados.

Etapa 5: Testar a aplicação Web

Para testar o aplicativo web

1. Faça download do sample-site.zip, descompacte-o e abra `scripts/search.js` em seu editor de texto de preferência.
2. Atualize a variável `apigatewayendpoint` para apontar para o endpoint do API Gateway Você pode encontrar rapidamente o endpoint no API Gateway escolhendo Estágios e selecionando o nome da API. A variável `apigatewayendpoint` deve assumir a forma de `https://some-id.execute-api.us-west-1.amazonaws.com/opensearch-api-test`.
3. Abra `index.html` e tente executar pesquisas para thor, casa e alguns outros termos.

Movie Search

thor

Found 7 results.



Thor

2011 — The powerful but arrogant god Thor is cast out of Asgard to live amongst humans in Midgard (Earth), where he soon becomes one of their finest defenders.



Thor: The Dark World

2013 — Faced with an enemy that even Odin and Asgard cannot withstand, Thor must embark on his most perilous and personal journey yet, one that will reunite him with Jane Foster and force him to sacrifice everything to save us all.



Vikingdom

2013 — A forgotten king, Eirick, is tasked with the impossible odds to defeat Thor, the God of Thunder.

Solucionar erros CORS

Mesmo que a função do Lambda inclua conteúdo na resposta para ser compatível com o CORS, você ainda pode ver o seguinte erro:

```
Access to XMLHttpRequest at '<api-gateway-endpoint>' from origin 'null' has been
 blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present in the
 requested resource.
```

Se isso acontecer, tente o seguinte:

1. Habilite o CORS no recurso GET. Em Avançado, defina Access-Control-Allow-Credentials como `'true'`.
2. Reimplante a API no API Gateway [Ações, Implantar API].
3. Exclua e torne a adicionar o acionador da função do Lambda. Adicione readicionar, escolha Adicionar acionador e crie o endpoint HTTP que invoca sua função. O acionador deve ter a seguinte configuração:

Trigger	solicitações de	Estágio de implantação	Segurança
API Gateway	opensearch-api	opensearch-api-test	Abra o

Próximas etapas

Este capítulo é apenas um ponto de partida para demonstrar um conceito. Você pode considerar as seguintes modificações:

- Adicione seus próprios dados ao domínio do OpenSearch Serviço.
- Adicionar métodos à API.
- Na função do Lambda, modifique a consulta de pesquisa ou incremente campos diferentes.
- Estilize os resultados de maneira diferente ou modifique `search.js` para exibir campos diferentes para o usuário.

Tutorial: Visualizando chamadas de suporte ao cliente com OpenSearch serviços e painéis OpenSearch

Este capítulo é uma descrição completa da seguinte situação: uma empresa recebe um determinado número de chamadas de suporte ao cliente e quer analisá-las. O que é o assunto de cada chamada? Quantas eram positivas? Quantas eram negativas? Como os gerentes podem pesquisar ou revisar as transcrições dessas chamadas?

Um fluxo de trabalho manual pode envolver funcionários ouvindo gravações, anotando o assunto de cada chamada e decidindo se a interação do cliente foi positiva.

Esse processo seria extremamente trabalhoso. Supondo um tempo médio de 10 minutos por chamada, cada funcionário escutaria apenas 48 chamadas por dia. Independentemente do viés humano, os dados que eles geram seriam altamente precisos, mas a quantidade de dados seria mínima: apenas o assunto da chamada e um booliano para saber se o cliente estava ou não satisfeito. Qualquer coisa mais complexa, como uma transcrição completa, tomaria uma quantidade imensa de tempo.

Usando o Amazon S3, o Amazon Transcribe, o Amazon Comprehend e o Amazon Service OpenSearch , você pode automatizar um processo similar com muito pouco código e acabar com muito mais dados. Por exemplo, você pode obter uma transcrição completa da chamada, as palavras-chave da transcrição e um “sentimento” global da chamada (positivo, negativo, neutro ou misto). Em seguida, você pode usar OpenSearch OpenSearch painéis para pesquisar e visualizar os dados.

Embora você possa usar esse passo a passo no estado em que se encontra, a intenção é gerar ideias sobre como enriquecer seus documentos JSON antes de indexá-los no Service. OpenSearch

Custos estimados

Em geral, executar as etapas desta demonstração devem custar menos de US$ 2. A demonstração usa os seguintes recursos:

- Bucket do S3 com menos de 100 MB transferidos e armazenados

  Para saber mais, consulte Definição de preços do Amazon S3.
- OpenSearch Domínio de serviço com uma `t2.medium` instância e 10 GiB de armazenamento EBS por várias horas

  Para saber mais, consulte Amazon OpenSearch Service Pricing.

- Várias chamadas para o Amazon Transcribe

  Para saber mais, consulte Preços do Amazon Transcribe.
- Várias chamadas de processamento de linguagem natural para o Amazon Comprehend

  Para saber mais, consulte Preços do Amazon Comprehend.

Tópicos

- Etapa 1: Configurar os pré-requisitos
- Etapa 2: Copiar código de exemplo
- (Opcional) Etapa 3: Indexar dados de exemplo
- Etapa 4: Analisar e visualizar seus dados
- Etapa 5: Limpar recursos e próximas etapas

Etapa 1: Configurar os pré-requisitos

Para continuar, você deve ter os recursos a seguir.

Pré-requisito	Description
Bucket do Amazon S3.	Para saber mais, consulte Creating a Bucket (Criar um bucket) no Manual do usuário do Amazon Simple Storage Service.
OpenSearch Domínio do serviço	O destino dos dados. Para obter mais informações, consulte Criação OpenSearch de domínios de serviço.

Se você ainda não tiver esses recursos, poderá criá-los usando os seguintes comandos do AWS CLI :

```
aws s3 mb s3://my-transcribe-test --region us-west-2
```

```
aws opensearch create-domain --domain-name my-transcribe-test --engine-version
 OpenSearch_1.0 --cluster-config  InstanceType=t2.medium.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 --access-policies
 '{"Version": "2012-10-17",       "Statement":[{"Effect":"Allow","Principal":
```

```
{"AWS":"arn:aws:iam::123456789012:root"},"Action":"es:*","Resource":"arn:aws:es:us-
west-2:123456789012:domain/my-transcribe-test/*"}]}' --region us-west-2
```

> **Note**
>
> Esses comandos usam a região `us-west-2`, mas você pode usar qualquer região compatível com o Amazon Comprehend. Para saber mais, consulte o Referência geral da AWS.

Etapa 2: Copiar código de exemplo

1. Copie e cole o código de exemplo Python 3 a seguir em um novo arquivo chamado `call-center.py`:

```
import boto3
import datetime
import json
import requests
from requests_aws4auth import AWS4Auth
import time
import urllib.request

# Variables to update
audio_file_name = '' # For example, 000001.mp3
bucket_name = '' # For example, my-transcribe-test
domain = '' # For example, https://search-my-transcribe-test-12345.us-
west-2.es.amazonaws.com
index = 'support-calls'
type = '_doc'
region = 'us-west-2'

# Upload audio file to S3.
s3_client = boto3.client('s3')

audio_file = open(audio_file_name, 'rb')

print('Uploading ' + audio_file_name + '...')
response = s3_client.put_object(
    Body=audio_file,
    Bucket=bucket_name,
```

```
    Key=audio_file_name
)

# # Build the URL to the audio file on S3.
# # Only for the us-east-1 region.
# mp3_uri = 'https://' + bucket_name + '.s3.amazonaws.com/' + audio_file_name

# Get the necessary details and build the URL to the audio file on S3.
# For all other regions.
response = s3_client.get_bucket_location(
    Bucket=bucket_name
)
bucket_region = response['LocationConstraint']
mp3_uri = 'https://' + bucket_name + '.s3-' + bucket_region + '.amazonaws.com/' +
 audio_file_name

# Start transcription job.
transcribe_client = boto3.client('transcribe')

print('Starting transcription job...')
response = transcribe_client.start_transcription_job(
    TranscriptionJobName=audio_file_name,
    LanguageCode='en-US',
    MediaFormat='mp3',
    Media={
        'MediaFileUri': mp3_uri
    },
    Settings={
        'ShowSpeakerLabels': True,
        'MaxSpeakerLabels': 2 # assumes two people on a phone call
    }
)

# Wait for the transcription job to finish.
print('Waiting for job to complete...')
while True:
    response =
 transcribe_client.get_transcription_job(TranscriptionJobName=audio_file_name)
    if response['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
 'FAILED']:
        break
    else:
        print('Still waiting...')
    time.sleep(10)
```

```
transcript_uri = response['TranscriptionJob']['Transcript']['TranscriptFileUri']

# Open the JSON file, read it, and get the transcript.
response = urllib.request.urlopen(transcript_uri)
raw_json = response.read()
loaded_json = json.loads(raw_json)
transcript = loaded_json['results']['transcripts'][0]['transcript']

# Send transcript to Comprehend for key phrases and sentiment.
comprehend_client = boto3.client('comprehend')

# If necessary, trim the transcript.
# If the transcript is more than 5 KB, the Comprehend calls fail.
if len(transcript) > 5000:
    trimmed_transcript = transcript[:5000]
else:
    trimmed_transcript = transcript

print('Detecting key phrases...')
response = comprehend_client.detect_key_phrases(
    Text=trimmed_transcript,
    LanguageCode='en'
)

keywords = []
for keyword in response['KeyPhrases']:
    keywords.append(keyword['Text'])

print('Detecting sentiment...')
response = comprehend_client.detect_sentiment(
    Text=trimmed_transcript,
    LanguageCode='en'
)

sentiment = response['Sentiment']

# Build the Amazon OpenSearch Service URL.
id = audio_file_name.strip('.mp3')
url = domain + '/' + index + '/' + type + '/' + id

# Create the JSON document.
json_document = {'transcript': transcript, 'keywords': keywords, 'sentiment':
 sentiment, 'timestamp': datetime.datetime.now().isoformat()}
```

```
# Provide all details necessary to sign the indexing request.
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region,
 'opensearchservice', session_token=credentials.token)

# Index the document.
print('Indexing document...')
response = requests.put(url, auth=awsauth, json=json_document, headers=headers)

print(response)
print(response.json())
```

2. Atualize as primeiras seis variáveis.
3. Instale os pacotes exigidos usando os seguintes comandos:

```
pip install boto3
pip install requests
pip install requests_aws4auth
```

4. Coloque seu MP3 no mesmo diretório `call-center.py` e execute o script. Uma saída de exemplo se segue:

```
$ python call-center.py
Uploading 000001.mp3...
Starting transcription job...
Waiting for job to complete...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Detecting key phrases...
Detecting sentiment...
Indexing document...
<Response [201]>
{u'_type': u'call', u'_seq_no': 0, u'_shards': {u'successful': 1, u'failed': 0,
 u'total': 2}, u'_index': u'support-calls4', u'_version': 1, u'_primary_term': 1,
 u'result': u'created', u'_id': u'000001'}
```

`call-center.py` executa uma série de operações:

1. O script carrega um arquivo de áudio (nesse caso, um MP3, mas o Amazon Transcribe suporta vários formatos) para seu bucket do S3.
2. Ele envia o URL do arquivo de áudio para o Amazon Transcribe e aguarda até que o trabalho de transcrição termine.

   O tempo para concluir o trabalho de transcrição depende do tamanho do arquivo de áudio. Considere minutos, não segundos.

   > **Tip**
   >
   > Para melhorar a qualidade da transcrição, você pode configurar um vocabulário personalizado para o Amazon Transcribe.

3. Depois que o trabalho de transcrição for concluído, o script extrairá a transcrição, a deixará com 5.000 caracteres e a enviará para o Amazon Comprehend para uma análise de palavras-chave e sentimento.
4. Por fim, o script adiciona a transcrição completa, as palavras-chave, o sentimento e o registro de data e hora atual a um documento JSON e o indexa no Service. OpenSearch

> **Tip**
>
> LibriVoxtem audiolivros de domínio público que você pode usar para testes.

(Opcional) Etapa 3: Indexar dados de exemplo

Se você não tiver várias gravações de chamadas à disposição — e quem tem? — poderá indexar os documentos de exemplo em sample-calls.zip, os quais são comparáveis àqueles produzidos pelo `call-center.py`.

1. Crie um arquivo chamado `bulk-helper.py`:

```
import boto3
from opensearchpy import OpenSearch, RequestsHttpConnection
import json
from requests_aws4auth import AWS4Auth
```

```
host = '' # For example, my-test-domain.us-west-2.es.amazonaws.com
region = '' # For example, us-west-2
service = 'es'

bulk_file = open('sample-calls.bulk', 'r').read()

credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    connection_class = RequestsHttpConnection
)

response = search.bulk(bulk_file)
print(json.dumps(response, indent=2, sort_keys=True))
```

2. Atualize as primeiras duas variáveis para `host` e `region`.
3. Instale o pacote exigido usando o seguinte comando:

```
pip install opensearch-py
```

4. Faça download e descompacte sample-calls.zip.
5. Coloque `sample-calls.bulk` no mesmo diretório que `bulk-helper.py` e execute o auxiliar. Uma saída de exemplo se segue:

```
$ python bulk-helper.py
{
  "errors": false,
  "items": [
    {
      "index": {
        "_id": "1",
        "_index": "support-calls",
        "_primary_term": 1,
        "_seq_no": 42,
        "_shards": {
```

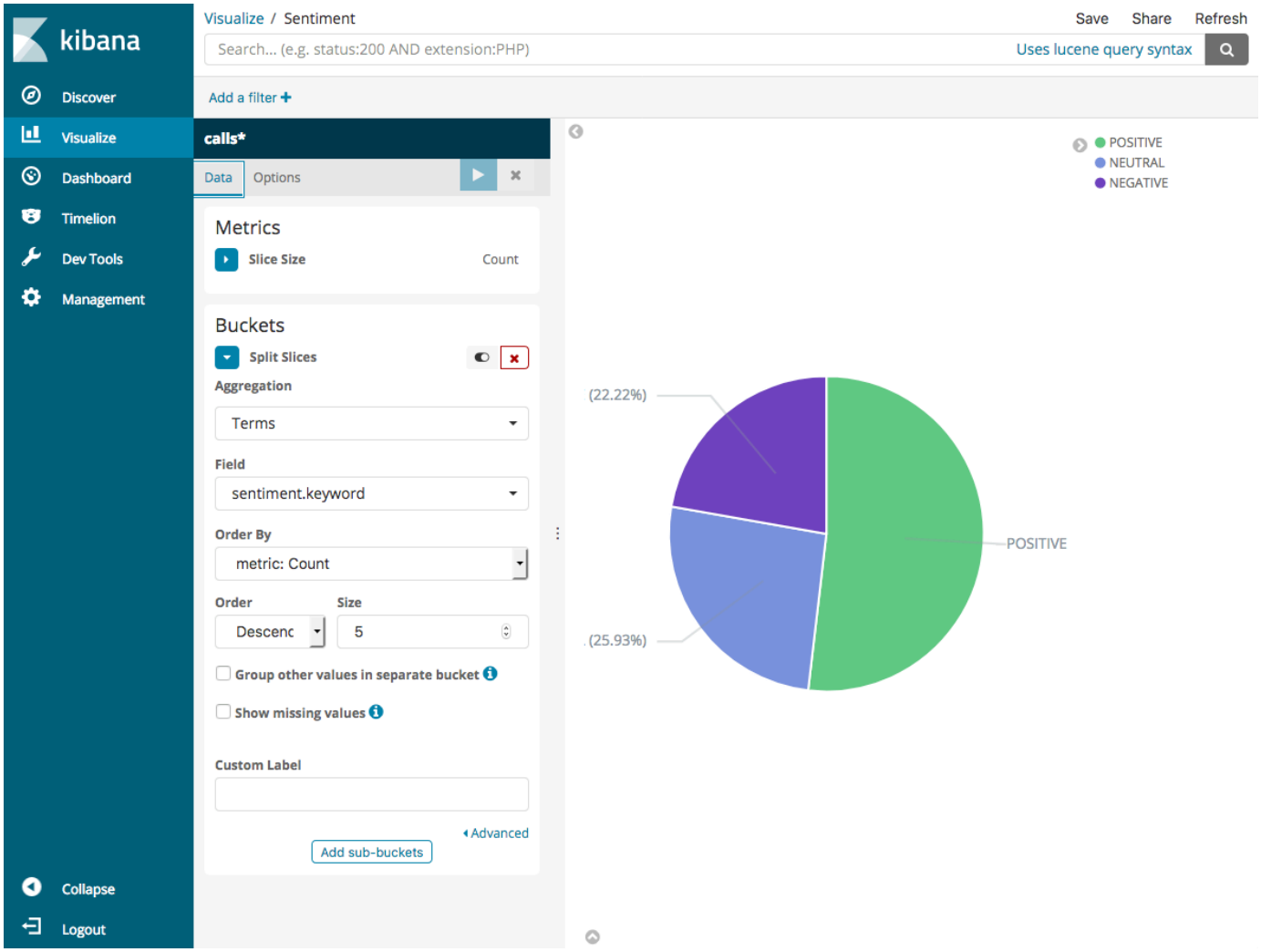
```
          "failed": 0,
          "successful": 1,
          "total": 2
        },
        "_type": "_doc",
        "_version": 9,
        "result": "updated",
        "status": 200
      }
    },
    ...
  ],
  "took": 27
}
```

Etapa 4: Analisar e visualizar seus dados

Agora que você tem alguns dados no OpenSearch Service, você pode visualizá-los usando OpenSearch painéis.

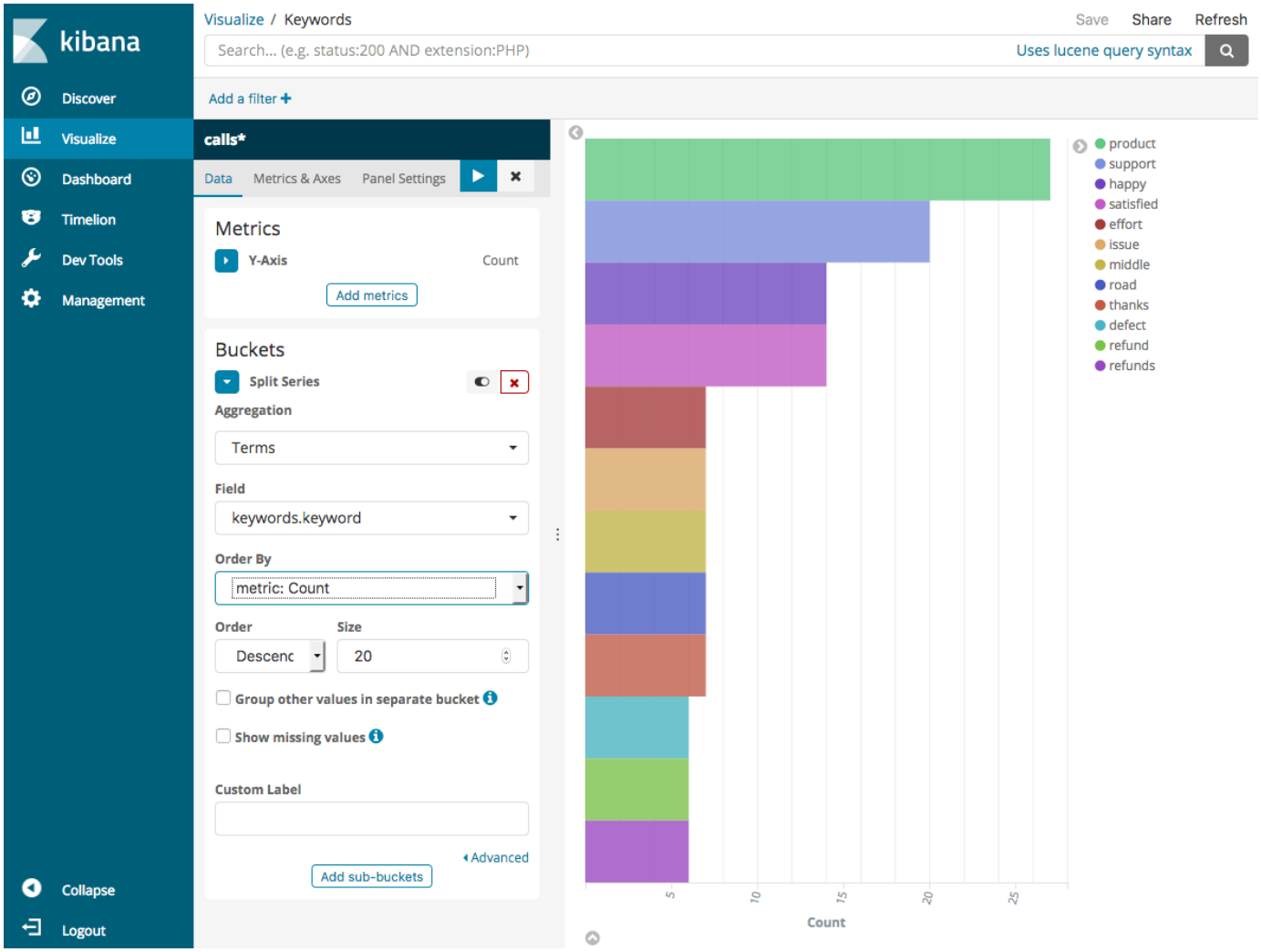
1. Acesse `https://search-`*`domain`*`.`*`region`*`.es.amazonaws.com/_dashboards`.
2. Antes de usar os OpenSearch painéis, você precisa de um padrão de índice. O Dashboard usa padrões de índice para restringir sua análise a um ou mais índices. Para corresponder ao índice `support-calls` criado por `call-center.py`, vá para Stack Management (Gerenciamento de pilhas), Index Patterns (Padrões de índice) e definir um padrão de índice de `support*`. Em seguida, escolha Next step (Próxima etapa).
3. Para o nome de campo Filtro de tempo, escolha timestamp.
4. Agora, você pode começar a criar visualizações. Escolha Visualizar e, em seguida, adicione uma nova visualização.
5. Escolha o gráfico de pizza e o padrão de índice `support*`.
6. A visualização padrão é básica. Portanto, escolha Dividir fatias para criar uma visualização mais interessante.

   Em Aggregation, escolha Terms. Em Campo, escolha sentiment.keyword. Em seguida, escolha Aplicar alterações e Salvar.

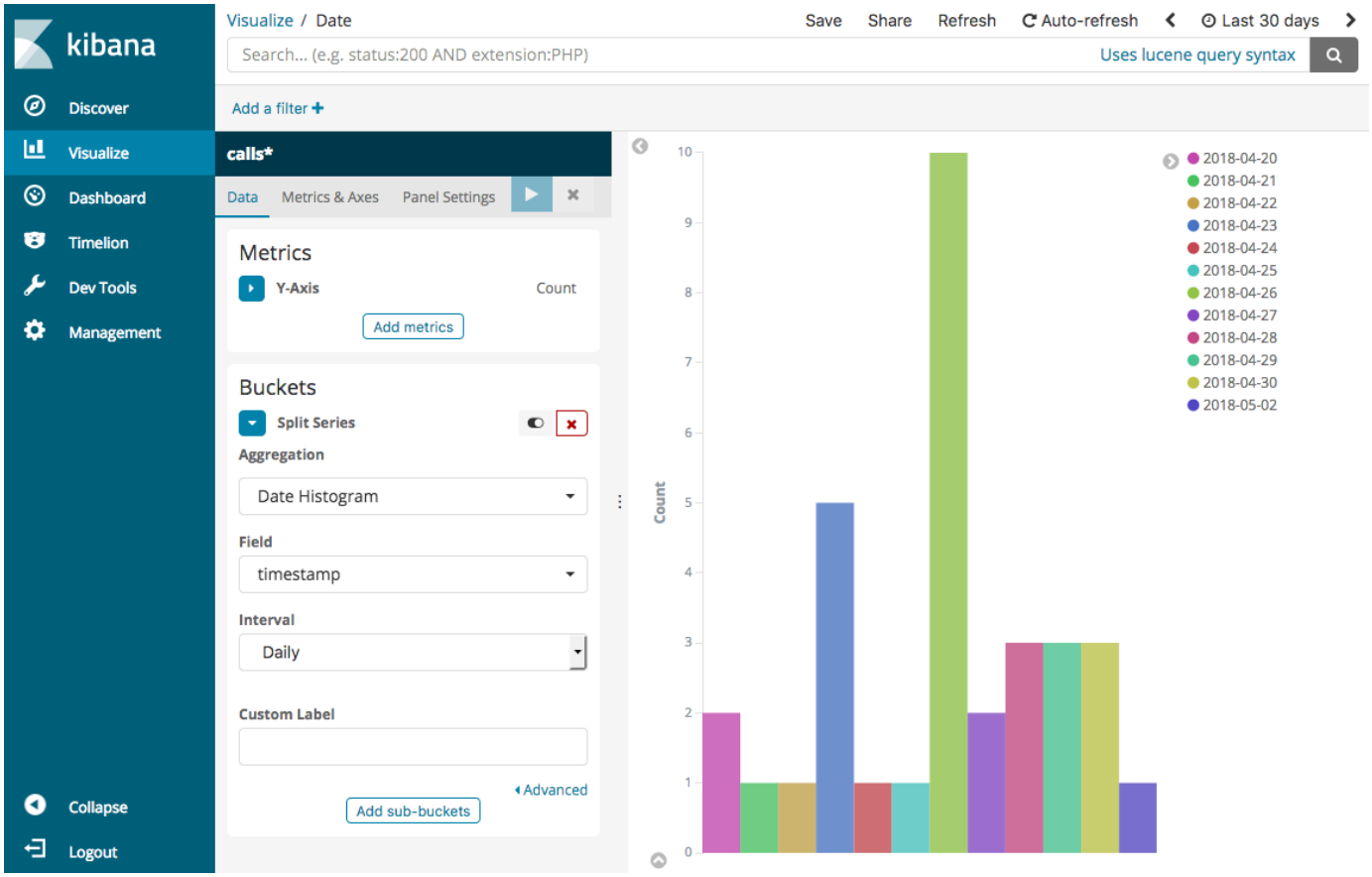


7. Volte para a página Visualizar e adicione outra visualização. Dessa vez, escolha o gráfico de barras horizontais.
8. Selecione Dividir séries.

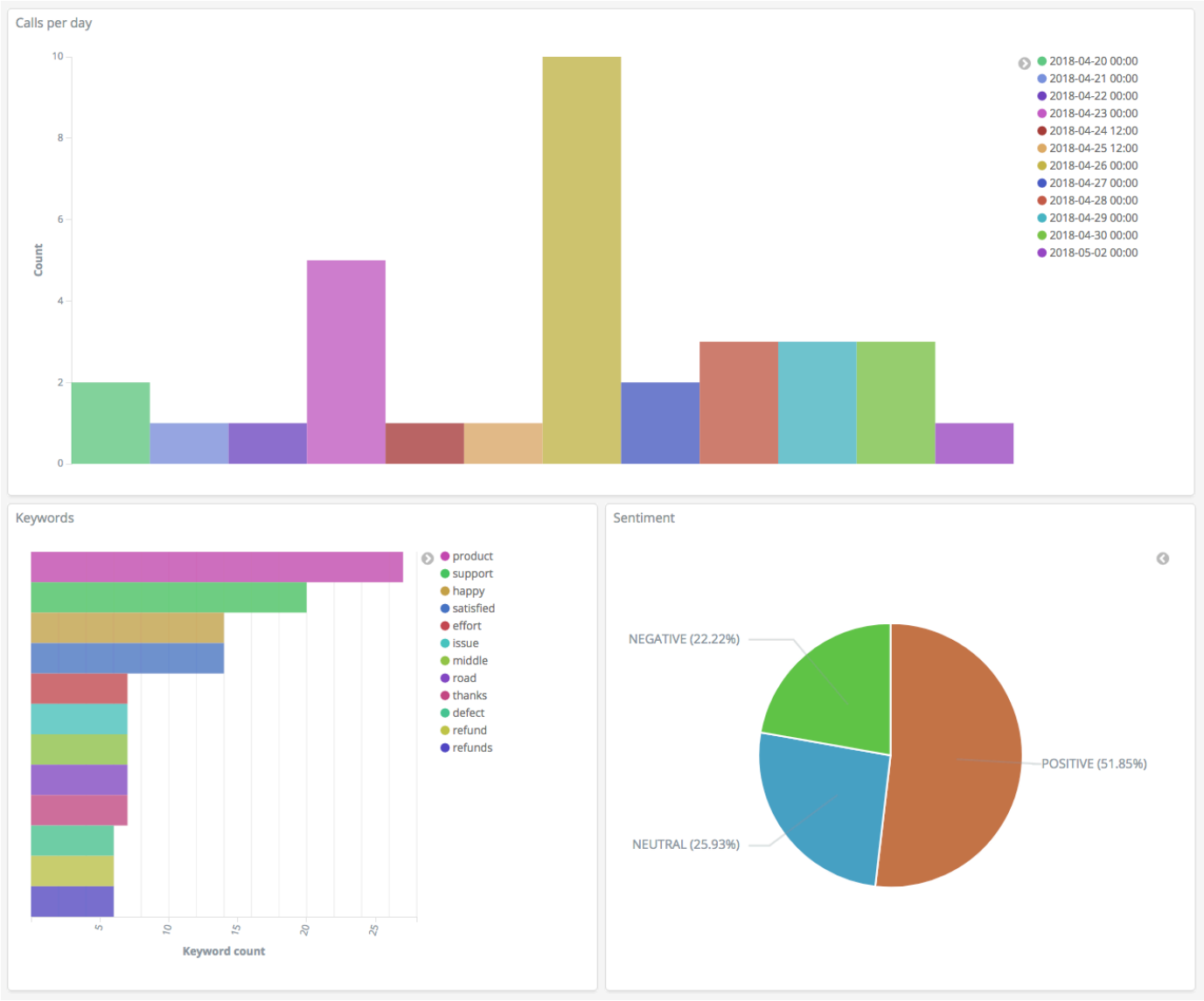
   Em Aggregation, escolha Terms. Em Campo, escolha keywords.keyword e altere Tamanho para 20. Em seguida, escolha Aplicar alterações e Salvar.



9. Volte para a página Visualizar e adicione uma visualização final, um gráfico de barras verticais.
10. Selecione Dividir séries. Em Agregação, escolha Histograma de data. Em Campo, escolha timestamp e altere Intervalo para Diariamente.
11. Escolha Métricas e eixos e altere Modo para normal.
12. Escolha Aplicar alterações e Salvar.



13. Agora que você tem três visualizações, poderá adicioná-las a uma visualização do Dashboards. Escolha Painel, crie um painel e adicione suas visualizações.



Etapa 5: Limpar recursos e próximas etapas

Para evitar cobranças desnecessárias, exclua o bucket do S3 e o domínio do OpenSearch serviço. Para saber mais, consulte Excluir um bucket no Guia do usuário do Amazon Simple Storage Service e Excluir um domínio de OpenSearch serviço neste guia.

As transcrições exigem muito menos espaço em disco do que MP3 os arquivos. Talvez você consiga reduzir sua janela de MP3 retenção — por exemplo, de três meses de gravações de chamadas para um mês — reter anos de transcrições e ainda economizar nos custos de armazenamento.

Você também pode automatizar o processo de transcrição usando o AWS Step Functions Lambda, adicionar metadados adicionais antes da indexação ou criar visualizações mais complexas para se adequar ao seu caso de uso exato.

Renomeação do Amazon OpenSearch Service - Resumo das alterações

Em 8 de setembro de 2021, nosso pacote de pesquisa e análise foi renomeado para Amazon OpenSearch Service. OpenSearch Suporte de serviço OpenSearch , bem como OSS legado do Elasticsearch. As seções a seguir descrevem as diferentes partes do serviço que foram alteradas com a renomeação do serviço e quais ações você precisa adotar para garantir que seus domínios continuem a funcionar corretamente.

Algumas dessas mudanças só se aplicam quando você atualiza seus domínios do Elasticsearch para o. OpenSearch Em outros casos, como no console de Billing and Cost Management, a experiência muda imediatamente.

Essa lista não é exaustiva. Ao mesmo tempo que outras partes do produto também mudaram, essas atualizações são as mais relevantes.

Nova versão de API

A nova versão da API de configuração do OpenSearch Service (01/01/2021) funciona com o Elasticsearch OpenSearch OSS legado. 21 operações de API foram substituídas por nomes mais concisos e independentes do mecanismo (por exemplo, `CreateElasticsearchDomain` alterados para`CreateDomain`), mas o OpenSearch Service continua oferecendo suporte às duas versões da API.

Recomendamos utilizar as novas operações de API para criar e gerenciar domínios no futuro. Observe que, ao usar as novas operações de API para criar um domínio, você precisará especificar o parâmetro `EngineVersion` no formato `Elasticsearch_X.Y` ou `OpenSearch_X.Y`, em vez de apenas o número da versão. Se você não especificar uma versão, o padrão será a versão mais recente do. OpenSearch

Atualize seu AWS CLI para a versão 1.20.40 ou posterior para usá-lo `aws opensearch ...` para criar e gerenciar seus domínios. Para o novo formato da CLI, consulte a referência da CLIOpenSearch .

Tipos de instâncias renomeados

Os tipos de instância no Amazon OpenSearch Service agora estão no formato `<type>.<size>.search` — por exemplo, `m6g.large.search` em vez de`m6g.large.elasticsearch`. Medida a ser tomada Os domínios existentes começarão a se referir automaticamente aos novos tipos de instâncias dentro da API e no console do Billing and Cost Management.

Se você tiver instâncias reservadas (RIs), seu contrato não será afetado pela alteração. A versão antiga da API de configuração ainda é compatível com o formato de nomenclatura antigo, mas se desejar usar a nova versão da API, você precisará usar o novo formato.

Alterações na política de acesso

As seções a seguir descrevem quais ações você precisará executar para atualizar suas políticas de acesso.

Políticas do IAM

Recomendamos atualizar suas políticas do IAM para usar as operações de API renomeadas. No entanto, o OpenSearch Serviço continuará a respeitar as políticas existentes ao replicar internamente as permissões antigas da API. Por exemplo, se você tiver permissão para executar a operação `CreateElasticsearchDomain`, agora você poderá fazer chamadas tanto para `CreateElasticsearchDomain` (operação da API antiga) quanto `CreateDomain` (operação da API nova). O mesmo se aplica às negações explícitas. Para obter uma lista das operações de API atualizadas, consulte a referência de elementos das políticas.

Políticas de SCP

As políticas de controle de serviço (SCPs) introduzem uma camada adicional de complexidade em comparação com o IAM padrão. Para evitar que suas políticas de SCP falhem, você deverá adicionar as operações de API antigas e novas a cada uma de suas políticas de SCP. Por exemplo, se um usuário tem permissões para `CreateElasticsearchDomain`, você também precisa conceder a eles permissões para `CreateDomain` para que eles possam manter a capacidade de criar domínios. O mesmo se aplica às negações explícitas.

Por exemplo:

```
"Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:CreateElasticsearchDomain",
        "es:CreateDomain"
         ...
      ],
    },
      "Effect": "Deny",
      "Action:" [
        "es:DeleteElasticsearchDomain",
        "es:DeleteDomain"
         ...
```

Novos tipos de recursos

OpenSearch O serviço apresenta os seguintes novos tipos de recursos:

Recurso	Description
`AWS::OpenSearchService::Domain`	Representa um domínio OpenSearch do Amazon Service. Esse recurso existe no nível de serviço e não é específico do software em execução no domínio. Aplica-se a serviços como CloudFormation e AWS Resource Groups (Grupos de recursos) nos quais você cria e gerencia recursos para o serviço como um todo. Para obter instruções sobre como atualizar domínios definidos no CloudFormation Elasticsearch para OpenSearch, consulte Comentários no Guia do Usuário. CloudForm ation
`AWS::OpenSearch::Domain`	Representa o OpenSearch/Elasticsearch software executado em um domínio. Esse recurso se aplica a serviços como AWS

Recurso	Description
	CloudTraile AWS Config, que fazem referênci a ao software executado no domínio e não ao OpenSearch Serviço como um todo. Esses serviços agora contêm tipos de recursos separados para domínios que executam Elasticsearch (`AWS::Elasticsearch ::Domain`) versus domínios executando (). OpenSearch `AWS::OpenSearch::Domain`

> **Note**
>
> Em AWS Config, você continuará vendo seus dados sob o tipo de `AWS::Elasticsearch::Domain` recurso existente por várias semanas, mesmo se atualizar um ou mais domínios para o. OpenSearch

Kibana renomeado para Painéis OpenSearch

OpenSearch Os painéis, a AWS alternativa ao Kibana, são uma ferramenta de visualização de código aberto projetada para trabalhar com eles. OpenSearch Depois de atualizar um domínio do Elasticsearch para OpenSearch, o `/_plugin/kibana` endpoint muda para. `/_dashboards` OpenSearch O serviço redirecionará todas as solicitações para o novo endpoint, mas se você usar o endpoint Kibana em qualquer uma das suas políticas do IAM, atualize essas políticas para incluir também o novo endpoint. `/_dashboards`

Se você estiver usandothe section called “Autenticação SAML para painéis OpenSearch”, antes de atualizar seu domínio para OpenSearch, você precisa alterar todo o Kibana URLs configurado em seu provedor de identidade (IdP) de para. `/_plugin/kibana` `/_dashboards` Os mais comuns URLs são assertion consumer service (ACS) URLs e destinatário. URLs

A `kibana_read_only` função padrão dos OpenSearch Painéis foi renomeada para`opensearch_dashboards_read_only`, e a `kibana_user` função foi renomeada para. `opensearch_dashboards_user` A alteração se aplica a todos os recém-criados 1 OpenSearch . domínios x executando o software de serviço R20211203 ou posterior. Se você atualizar um domínio existente para o software de serviço R20211203, os nomes de funções permanecem os mesmos.

Métricas renomeadas CloudWatch

Várias CloudWatch métricas mudam nos domínios em execução OpenSearch. Quando você atualiza um domínio para OpenSearch, as métricas mudam automaticamente e seus CloudWatch alarmes atuais são interrompidos. Antes de atualizar seu cluster de uma versão do Elasticsearch para uma OpenSearch versão, certifique-se de atualizar seus CloudWatch alarmes para usar as novas métricas.

As seguintes métricas foram alteradas:

Nome da métrica original	Novo nome
`KibanaHealthyNodes`	`OpenSearchDashboardsHealthyNodes`
`KibanaConcurrentConnections`	`OpenSearchDashboardsConcurrentConnections`
`KibanaHeapTotal`	`OpenSearchDashboardsHeapTotal`
`KibanaHeapUsed`	`OpenSearchDashboardsHeapUsed`
`KibanaHeapUtilization`	`OpenSearchDashboardsHeapUtilization`
`KibanaOS1MinuteLoad`	`OpenSearchDashboardsOS1MinuteLoad`
`KibanaRequestTotal`	`OpenSearchDashboardsRequestTotal`
`KibanaResponseTimesMaxInMillis`	`OpenSearchDashboardsResponseTimesMaxInMillis`
`ESReportingFailedRequestSysErrCount`	`KibanaReportingFailedRequestSysErrCount`
`ESReportingRequestCount`	`KibanaReportingRequestCount`
`ESReportingFailedRequestUserErrCount`	`KibanaReportingFailedRequestUserErrCount`

Nome da métrica original	Novo nome
`ESReportingSuccessCount`	`KibanaReportingSuccessCount`
`ElasticsearchRequests`	`OpenSearchRequests`

Para obter uma lista completa das métricas que o OpenSearch Serviço envia para a Amazon CloudWatch, consultethe section called “Monitoramento de métricas de cluster”.

Abra o console do Billing and Cost Management.

Os dados históricos no console do Billing and Cost Management e nos relatórios de custos e uso continuarão usando o nome antigo do serviço, então você precisa começar a usar filtros tanto para o OpenSearch Amazon Service quanto para o nome antigo do Elasticsearch ao pesquisar dados. Se você já tiver relatórios salvos, atualize os filtros para garantir que eles também incluam OpenSearch Serviço. Inicialmente, você pode receber um alerta quando seu uso do Elasticsearch diminui e aumenta OpenSearch, mas ele desaparece em alguns dias.

Além do nome do serviço, os campos a seguir serão alterados para todos os relatórios, listas e operações de API de lista de preços:

Campo	Formato antigo	Formato de linha
Tipo de instância	`m5.large.elasticsearch`	`m5.large.search`
Família de produtos	Instância do Elasticsearch Volume do Elasticsearch	Instância OpenSearch de serviço da Amazon Volume OpenSearch de serviços da Amazon
Descrição dos preços	5,098 USD por hora de instância c5.18xlarge.elasticsearch (ou hora parcial), UE	5,098 USD por hora de instância c5.18xlarge.search (ou hora parcial), UE
Família de instâncias	`ultrawarm.elasticsearch`	`ultrawarm.search`

Novo formato dos eventos

O formato dos eventos que o OpenSearch Serviço envia para a Amazon EventBridge e a Amazon CloudWatch mudou, especificamente o `detail-type` campo. O campo de origem (`aws.es`) permanece o mesmo. Para obter o formato completo de cada tipo de evento, consulte the section called “Monitoramento de eventos”. Se você tiver regras de evento existentes que dependem do formato antigo, atualize-as para que estejam em conformidade com o novo formato.

O que não mudou?

Os seguintes recursos e funcionalidades, entre outros não listados, permanecerão os mesmos:

- Entidade principal do serviço (`es.amazonaws.com`)
- Código do fornecedor
- Domínio ARNs
- Endpoints de domínio

Comece: atualize seus domínios para 1.x OpenSearch

OpenSearch 1. x suporta atualizações das versões 6.8 e 7 do Elasticsearch. x. Para obter instruções sobre como atualizar seu domínio, consulte the section called “Atualizar um domínio (console)”. Se você estiver usando a API de configuração AWS CLI ou para atualizar seu domínio, precisará especificar `TargetVersion` as`OpenSearch_1.x`.

OpenSearch 1. x introduz uma configuração de domínio adicional chamada Ativar modo de compatibilidade. Como certos clientes e plug-ins do Elasticsearch OSS verificam a versão do cluster antes de se conectarem, o modo de compatibilidade é configurado OpenSearch para relatar sua versão como 7.10 para que esses clientes continuem trabalhando.

Você pode ativar o modo de compatibilidade ao criar OpenSearch domínios pela primeira vez ou ao fazer o upgrade para uma versão OpenSearch do Elasticsearch. Se não estiver definido, o parâmetro assumirá como padrão o valor `false` quando você criar um domínio e `true` quando você atualizar um domínio.

Para habilitar o modo de compatibilidade usando a API de configuração, defina `override_main_response_version` como `true`:

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/upgradeDomain
{
  "DomainName": "domain-name",
  "TargetVersion": "OpenSearch_1.0",
  "AdvancedOptions": {
    "override_main_response_version": "true"
   }
}
```

Para ativar ou desativar o modo de compatibilidade em OpenSearch domínios existentes, você precisa usar a operação da API OpenSearch _cluster/settings:

```
PUT /_cluster/settings
{
  "persistent" : {
    "compatibility.override_main_response_version" : true
  }
}
```

Solução de problemas do Amazon OpenSearch Service

Este tópico descreve como identificar e resolver problemas comuns do Amazon OpenSearch Service. Consulte as informações nesta seção antes de entrar em contato com o AWS Support.

Não consigo acessar os OpenSearch painéis

O endpoint do OpenSearch Dashboards não é compatível com solicitações assinadas. Se a política de controle de acesso do seu domínio apenas concede acesso a determinados perfis do IAM e, se você não tiver configurado a autenticação do Amazon Cognito, poderá receber a seguinte mensagem de erro ao tentar acessar o Dashboards:

```
"User: anonymous is not authorized to perform: es:ESHttpGet"
```

Se seu domínio OpenSearch de serviço usa acesso VPC, você pode não receber esse erro, mas a solicitação pode expirar. Para saber mais sobre como corrigir esse problema e as várias opções de configuração disponíveis, consulte the section called “Controlar acesso ao Dashboards”, the section called “Sobre políticas de acesso em domínios da VPC” e the section called “Gerenciamento de Identidade e Acesso”.

Não é possível acessar o domínio da VPC

Consulte the section called “Sobre políticas de acesso em domínios da VPC” e the section called “Teste dos domínios da VPC”.

Cluster no estado somente leitura

Em comparação com as versões anteriores do Elasticsearch OpenSearch e do Elasticsearch 7. x use um sistema diferente para coordenação de clusters. Nesse novo sistema, quando o cluster perde quorum, o cluster fica indisponível até que você execute uma ação. A perda de quorum pode assumir duas formas:

- Se o cluster usar nós principais dedicados, a perda de quorum ocorrerá quando a metade ou mais estiverem indisponíveis.
- Se o cluster não usar nós principais dedicados, a perda de quorum ocorrerá quando a metade ou mais dos seus nós de dados estiverem indisponíveis.

Se ocorrer perda de quorum e seu cluster tiver mais de um nó, o OpenSearch Service restaurará o quorum e colocará o cluster em um estado somente para leitura. Você tem duas opções:

- Remover o estado somente leitura e usar o cluster no estado em que se encontra.
- Restaure o cluster ou os índices individuais de um snapshot.

Se você preferir usar o cluster no estado em que se encontra, verifique se a integridade do cluster está verde usando a seguinte solicitação:

```
GET _cat/health?v
```

Se a integridade do cluster for vermelha, recomendamos restaurar o cluster a partir de um snapshot. Você também poderá consultar the section called “Status de cluster vermelho” para ver as etapas de solução de problemas. Se a integridade do cluster estiver verde, verifique se todos os índices esperados estão presentes usando a seguinte solicitação:

```
GET _cat/indices?v
```

Execute algumas pesquisas para verificar se os dados esperados estão presentes. Se estiverem, você poderá remover o estado somente leitura usando a seguinte solicitação:

```
PUT _cluster/settings
{
  "persistent": {
    "cluster.blocks.read_only": false
  }
}
```

Se ocorrer perda de quorum e seu cluster tiver apenas um nó, o OpenSearch Service substituirá o nó e não colocará o cluster em um estado somente para leitura. Caso contrário, suas opções serão as mesmas: usar o cluster no estado em que se encontra ou restaurar a partir de um snapshot.

Em ambas as situações, o OpenSearch Serviço envia dois eventos para o seu Health Dashboard. O primeiro informa sobre a perda de quorum. A segunda ocorre depois que o OpenSearch Serviço restaura com sucesso o quórum. Para obter mais informações sobre como usar o Health Dashboard, consulte o Guia AWS Health do usuário.

Status de cluster vermelho

Um status de cluster vermelho significa que pelo menos um fragmento primário e suas réplicas não estão alocados em um nó. OpenSearch O serviço continua tentando tirar instantâneos automatizados de todos os índices, independentemente de seu status, mas os instantâneos falham enquanto o status vermelho do cluster persiste.

As causas mais comuns do status vermelho para o cluster são nós de cluster que apresentam falha e o travamento do processamento do OpenSearch devido a uma carga contínua de processos pesados.

> **Note**
>
> OpenSearch O serviço armazena instantâneos automatizados por 14 dias, independentemente do status do cluster. Portanto, se o status de cluster vermelho persistir por mais de duas semanas, o último snapshot automatizado saudável será excluído e você poderá perder permanentemente os dados do seu cluster. Se o seu domínio de OpenSearch serviço entrar em um status de cluster vermelho, Suporte poderá entrar em contato com você para perguntar se você deseja resolver o problema sozinho ou se deseja que a equipe de suporte ajude. Você pode definir um CloudWatch alarme para notificá-lo quando ocorrer um status de cluster vermelho.

Em última análise, fragmentos vermelhos resultam em clusters vermelhos e índices vermelhos, em fragmentos vermelhos. Para identificar os índices que causam o status do cluster vermelho, OpenSearch é útil APIs.

- `GET /_cluster/allocation/explain` escolhe o primeiro fragmento sem atribuição que ele encontrar e explica por que ele não pode ser alocado para um nó:

```
{
    "index": "test4",
    "shard": 0,
    "primary": true,
    "current_state": "unassigned",
    "can_allocate": "no",
    "allocate_explanation": "cannot allocate because allocation is not permitted to
 any of the nodes"
}
```

- `GET /_cat/indices?v` mostra o status de integridade, o número de documentos e o uso do disco para cada índice:

```
health status index            uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   test1            30h1EiMvS5uAFr2t5CEVoQ   5   0        820            0
       14mb           14mb
green  open   test2            sdIxs_WDT56afFGu5KPbFQ   1   0          0            0
       233b           233b
green  open   test3            GGRZp_TBRZuSaZpAGk2pmw   1   1          2            0
     14.7kb          7.3kb
red    open   test4            BJxfAErbTtu5HBjIXJV_7A   1   0
green  open   test5            _8C6MIXOSxCqVYicH3jsEA   1   0          7            0
     24.3kb         24.3kb
```

A exclusão de índices vermelhos é a maneira mais rápida de corrigir um status de cluster vermelho. Dependendo do motivo do status do cluster vermelho, você pode então escalar seu domínio de OpenSearch serviço para usar tipos de instância maiores, mais instâncias ou mais armazenamento baseado em EBS e tentar recriar os índices problemáticos.

Se a exclusão de um índice problemático não for viável, você pode restaurar um snapshot, excluir documentos do índice, alterar as configurações de índice, reduzir o número de réplicas ou excluir outros índices para liberar espaço em disco. A etapa importante é resolver o status do cluster vermelho antes de reconfigurar seu domínio de OpenSearch serviço. A reconfiguração de um domínio com um status de cluster vermelho pode agravar o problema e fazer com que o domínio fique preso em um estado de configuração Em processamento até que você resolva o status.

Correção automática de clusters vermelhos

Se o status do seu cluster ficar vermelho continuamente por mais de uma hora, o OpenSearch Service tentará corrigi-lo automaticamente redirecionando fragmentos não alocados ou restaurando a partir de snapshots anteriores.

Se não conseguir corrigir um ou mais índices vermelhos e o status do cluster permanecer vermelho por um total de 14 dias, o OpenSearch Serviço tomará medidas adicionais somente se o cluster atender a pelo menos um dos seguintes critérios:

- Tem apenas uma zona de disponibilidade
- Nós principais dedicados

- Contém tipos de instância intermitentes (T2 ou T3)

No momento, se seu cluster atender a um desses critérios, o OpenSearch Service enviará notificações diárias nos próximos 7 dias, explicando que, se você não corrigir esses índices, todos os fragmentos não atribuídos serão excluídos. Se o status do seu cluster ainda estiver vermelho após 21 dias, o OpenSearch Service excluirá os fragmentos não atribuídos (armazenamento e computação) em todos os índices vermelhos. Você recebe notificações no painel Notificações do console de OpenSearch serviço para cada um desses eventos. Para obter mais informações, consulte the section called “Eventos de integridade do cluster”.

Recuperação de uma carga contínua de processamento pesado

Para determinar se um status de cluster vermelho deve-se a uma carga contínua de processamento pesado em um nó de dados, monitore as métricas de cluster a seguir.

Métrica relevante	Description	Recuperação
JVMMemoryPressão	Especifica a porcentagem do heap do Java usada para todos os nós de dados em um cluster. Visualize a estatística Máximo para essa métrica e procure quedas ainda menores na pressão de memória enquanto o coletor de lixo Java falhar na recuperação de memória suficiente. Esse padrão provavelmente se deve a consultas complexas ou a campos de dados grandes. Os tipos de instância x86 usam o coletor de lixo Concurrent Mark Sweep (CMS), que é executado junto com os threads da aplicação para manter as pausas curtas. Se o CMS não conseguir recuperar memória suficiente durante suas coletas normais, ele acionará uma	Defina disjuntores de memória para JVM. Para saber mais, consulte the section called “JVM OutOfMemoryError”. Se o problema persistir, exclua índices desnecess ários, reduza o número ou a complexidade das solicitaç ões para o domínio, adicione instâncias ou use tipos de instância maiores.

Métrica relevante	Description	Recuperação
	coleta de resíduos completa, o que pode levar a longas pausas na aplicação e afetar a estabilidade do cluster. Os tipos de instância Graviton baseados em ARM usam o coletor de lixo Garbage-First (G1), que é semelhante ao CMS, mas usa pausas curtas adicionais e desfragmentação de pilha para reduzir ainda mais a necessidade de coleções de lixo completas. Em ambos os casos, se o uso de memória continuar crescendo além do que o coletor de lixo pode recuperar durante a coleta de lixo completa, ocorrerá uma OpenSearch falha com um erro de falta de memória. Em todos os tipos de instância, uma boa regra prática é manter o uso abaixo de 80%. A API `_nodes/stats/jvm` oferece um resumo útil das estatísticas do JVM, do uso do grupo de memórias e das informações sobre coleta de lixo: `GET domain-endpoint /_nodes/stats/jvm?pretty`	

Métrica relevante	Description	Recuperação
CPUUtilization	Especifica a porcentagem de recursos da CPU usados para nós de dados em um cluster. Visualize a estatística Maximum para essa métrica e procure um padrão contínuo de uso intenso.	Adicione nós de dados ou aumente o tamanho dos tipos de instância dos nós de dados existentes.
Nós	Especifica o número de nós em um cluster. Visualize a estatísti ca Mínimo para essa métrica. Esse valor oscila quando o serviço implanta uma nova frota de instância s para um cluster.	Adicione nós de dados.

Status de cluster amarelo

O status de cluster amarelo significa que os fragmentos principais de todos os índices estão alocados a nós em um cluster, mas os fragmentos de réplica de pelo menos um índice não. Os clusters de nó único sempre são inicializados com um status de cluster amarelo porque não há outro nó ao qual o OpenSearch Serviço possa atribuir uma réplica. Para obter o status de cluster verde, aumente a contagem de nós. Para saber mais, consulte the section called “Dimensionamento de domínios”.

Clusters de vários nós podem ter brevemente um status de cluster amarelo após a criação de um novo índice ou após uma falha de nó. Esse status se resolve automaticamente à medida que os dados são OpenSearch replicados em todo o cluster. A falta de espaço em disco também pode causar status de cluster amarelo; o cluster só poderá distribuir fragmentos de réplica se os nós tiverem espaço em disco para acomodá-los.

ClusterBlockException

Você pode receber um erro `ClusterBlockException` pelos motivos a seguir.

Falta de espaço de armazenamento disponível

Se um ou mais nós em seu cluster tiverem espaço de armazenamento menor que o valor mínimo de 1) 20% do espaço de armazenamento disponível ou 2) 20 GiB de espaço de armazenamento, operações básicas de gravação, como adição de documentos e criação de índices, podem começar a falhar. the section called “Cálculo de requisitos de armazenamento”fornece um resumo de como o OpenSearch Serviço usa o espaço em disco.

Para evitar problemas, monitore a `FreeStorageSpace` métrica no console OpenSearch de serviço e crie CloudWatch alarmes para acionar quando `FreeStorageSpace` cair abaixo de um determinado limite. `GET /_cat/allocation?v`também fornece um resumo útil da alocação de fragmentos e do uso do disco. Para resolver problemas associados à falta de espaço de armazenamento, escale seu domínio de OpenSearch serviço para usar tipos de instância maiores, mais instâncias ou mais armazenamento baseado em EBS.

Alta pressão da memória da JVM

Quando a métrica de JVMMemorypressão excede 92% por 30 minutos, o OpenSearch Service aciona um mecanismo de proteção e bloqueia todas as operações de gravação para evitar que o cluster alcance o status vermelho. Quando a proteção é ativada, as operações de gravação falham devido ao erro `ClusterBlockException`, novos índices não podem ser criados e o erro `IndexCreateBlockException` é lançado.

Quando a métrica de JVMMemorypressão retorna para 88% ou menos por cinco minutos, a proteção é desativada e as operações de gravação no cluster são desbloqueadas.

A alta pressão da memória da JVM pode ser causada por picos no número de solicitações ao cluster, alocações de fragmentos não equilibradas entre os nós, excesso de fragmentos em um cluster, explosões de mapeamento de índices ou dados de campo ou tipos de instâncias que não conseguem administrar as cargas recebidas. Também pode ser causada pelo uso de agregações, curingas ou amplos intervalos de tempo nas consultas.

Para reduzir o tráfego para o cluster e resolver problemas de alta pressão da memória da JVM, experimente uma ou mais destas opções:

- Escale o domínio para que o tamanho máximo da pilha por nó seja de 32 GB.
- Reduza o número de fragmentos excluindo índices antigos ou não utilizados.
- Limpe o cache de dados com a operação da API `POST` *`index-name`*`/_cache/clear?fielddata=true`. Limpar o cache poderá interromper as consultas em andamento.

Em geral, para evitar a alta pressão da memória da JVM futuramente, siga estas práticas recomendadas:

- Evite agregações em campos de texto ou altere o tipo de mapeamento de seus índices para `keyword`.
- Otimize as solicitações de pesquisa e indexação escolhendo o número correto de fragmentos.
- Configure as políticas do Index State Management (ISM) para remover regularmente os índices não utilizados.

Erro ao migrar para multi-AZ com modo de espera

Os seguintes problemas podem ocorrer quando você migra um domínio existente para o Multi-AZ com modo de espera.

Criação de um índice, modelo de índice ou política do ISM durante a migração de domínios sem espera para domínios com modo de espera

Se você criar um índice ao migrar um domínio do Multi-AZ sem modo de espera para com modo de espera e o modelo de índice ou a política do ISM não seguir as diretrizes recomendadas de cópia de dados, isso pode causar uma inconsistência de dados e a migração pode falhar. Para evitar essa situação, crie o novo índice com uma contagem de cópias de dados (incluindo nós primários e réplicas) que seja múltipla de três. Você pode verificar o progresso da migração usando a API `DescribeDomainChangeProgress`. Se você encontrar um erro de contagem de réplicas, corrija o erro e entre em contato com o Suporte da AWS para tentar a migração novamente.

Número incorreto de cópias de dados

Se você não tiver o número certo de cópias de dados em seu domínio, a migração para o multi-AZ com modo de espera falhará.

JVM OutOfMemoryError

Normalmente, `OutOfMemoryError` em JVM significa que um dos seguintes disjuntores para JVM foi atingido.

Disjuntor	Description	Propriedade de configuração de cluster
Disjuntor principal	Porcentagem total de memória do heap de JVM permitida para todos os disjuntores. O valor padrão é 95%.	`indices.breaker.total.limit`
Disjuntor de dados de campo	Porcentagem de memória do heap de JVM com permissão para carregar um único campo de dados na memória. O valor padrão é 40%. Se você carregar dados com campos grandes, talvez precise aumentar esse limite.	`indices.breaker.fielddata.limit`
Disjuntor de solicitações	Porcentagem de memória do heap de JVM permitida para estruturas de dados usados para responder a uma solicitação de serviço. O valor padrão é 60%. Se suas solicitações de serviço envolverem o cálculo de agregações, recomenda-se aumentar esse limite.	`indices.breaker.request.limit`

Nós de cluster com falha

As instâncias do Amazon EC2 podem experimentar encerramentos e reinicializações inesperados. Normalmente, o OpenSearch Service reinicia os nós para você. No entanto, é possível que um ou mais nós em um cluster do OpenSearch permaneçam em condição de falha.

Para verificar essa condição, abra o painel do seu domínio no console OpenSearch de serviço. Escolha a guia Integridade do cluster e, em seguida, a métrica Total de nós. Veja se o número de nós relatado é inferior ao número que você configurou para seu cluster. Se a métrica mostrar que um ou mais nós estão inativos por mais de um dia, entre em contato com o AWS Support.

Você também pode definir um CloudWatch alarme para notificá-lo quando esse problema ocorrer.

> **Note**
>
> A métrica Total de nós não é precisa durante as alterações na configuração do cluster e durante a manutenção de rotina do serviço. Esse comportamento é esperado. A métrica logo informará o número correto de nós do cluster. Para saber mais, consulte the section called "Alterações de configuração".

Para proteger seus clusters contra terminações e reinicializações inesperadas de nós, crie pelo menos uma réplica para cada índice em seu OpenSearch domínio de serviço.

Limite máximo de fragmentos excedido

OpenSearch bem como 7. As versões x do Elasticsearch têm uma configuração padrão de no máximo 1.000 fragmentos por nó. OpenSearch/Elasticsearch gera um erro se uma solicitação, como a criação de um novo índice, fizer com que você exceda esse limite. Se você encontrar esse erro, terá várias opções:

- Adicionar mais nós de dados ao cluster.
- Aumentar a configuração `_cluster/settings/cluster.max_shards_per_node`.
- Usar a API _shrink para reduzir o número de fragmentos no nó.

Domínio paralisado no estado de processamento

Seu domínio OpenSearch de serviço entra no estado "Processamento" quando está no meio de uma alteração na configuração. Quando você inicia uma alteração na configuração, o status do domínio muda para "Processando" enquanto o OpenSearch Serviço cria um novo ambiente. No novo ambiente, o OpenSearch Service lança um novo conjunto de nós aplicáveis (como dados, mestre ou UltraWarm). Após a conclusão da migração, os nós mais antigos são encerrados.

O cluster pode ficar paralisado no estado “Processing” (Processamento) caso alguma destas situações ocorra:

- Um novo conjunto de nós de dados não possa ser iniciado.
- A migração de fragmentos para o novo conjunto de nós de dados não seja bem-sucedida.
- Ocorreu uma falha na verificação de validação com erros.

Para obter etapas detalhadas de resolução em cada uma dessas situações, consulte Por que meu domínio do Amazon OpenSearch Service está preso no estado “Processando”? .

O saldo de intermitência do EBS está baixo

OpenSearch O serviço envia uma notificação ao console quando o saldo máximo do EBS em um de seus volumes de uso geral (SSD) está abaixo de 70% e uma notificação de acompanhamento se o saldo cair abaixo de 20%. Para corrigir esse problema, você pode aumentar a escala verticalmente do cluster ou reduzir as IOPS de leitura e gravação para que o saldo de intermitência possa ser creditado. O saldo intermitente permanece em 0 para domínios com tipos de volume gp3 e domínios com volume gp2 cujo tamanho de volume seja superior a 1000 GiB. Para saber mais, consulte Volumes de Finalidade geral (SSD) (gp2). Você pode monitorar o equilíbrio de intermitência do EBS com a `BurstBalance` CloudWatch métrica.

A métrica do EBS aumenta durante o redimensionamento de volume

Ao modificar os tamanhos dos volumes do Amazon Elastic Block Store, você pode observar aumentos temporários em várias métricas do EBS, como, `Write Throughput`, `Write Throughput Micro bursting`, `Disk Queue Depth` e `IOPS`. Esse é o comportamento esperado durante a operação de redimensionamento e normalmente dura alguns segundos. Porém, a duração pode variar com base no tamanho do volume que está sendo modificado.

Para evitar problemas de latência e rejeições de solicitações, realize redimensionamentos de volume do EBS somente quando o cluster estiver íntegro e o tráfego de rede for pouco.

Não é possível habilitar logs de auditoria

Você pode encontrar o seguinte erro ao tentar habilitar a publicação do registro de auditoria usando o console OpenSearch de serviço:

A política de acesso a recursos especificada para o grupo de CloudWatch registros de registros não concede permissões suficientes para que o Amazon OpenSearch Service crie um fluxo de registros. Verifique a Política de acesso a recursos.

Se você encontrar esse erro, verifique se o elemento `resource` da sua política inclui o ARN do grupo de logs correto. Se isso acontecer, faça o seguinte:

1. Espere vários minutos.
2. Atualize a página em seu navegador da Web.
3. Escolha Selecionar grupo existente.
4. Em Grupo de logs existente, escolha o grupo de logs que você criou antes de receber a mensagem de erro.
5. Na seção de política de acesso, escolha Selecionar política existente.
6. Em Política existente, escolha a política que você criou antes de receber a mensagem de erro.
7. Escolha Habilitar.

Se o erro persistir após o processo ser repetido várias vezes, entre em contato com o AWS Support.

Não é possível fechar o índice

OpenSearch O serviço oferece suporte à `_close`API somente para as versões 7.4 OpenSearch e posteriores do Elasticsearch. Se você estiver usando uma versão anterior e restaurando um índice de um snapshot, poderá excluir o índice existente (antes ou depois de reindexá-lo).

Verificações de licenças do cliente

As distribuições padrão do Logstash e do Beats incluem uma verificação de licença proprietária e não conseguem se conectar à versão de código aberto do. OpenSearch Certifique-se de usar as distribuições Apache 2.0 (OSS) desses clientes com o Service. OpenSearch

Controle de utilização de solicitações

Se você receber erros `403 Request throttled due to too many requests` ou `429 Too Many Requests` persistentes, considere a escalabilidade vertical. O Amazon OpenSearch Service limitará as solicitações se a carga útil fizer com que o uso de memória exceda o tamanho máximo do heap Java.

Não é possível executar o SSH no nó

Você não pode usar o SSH para acessar nenhum dos nós do seu OpenSearch cluster e não pode `opensearch.yml` modificá-lo diretamente. Em vez disso, use o console ou SDKs para configurar seu domínio. AWS CLI Você também pode especificar algumas configurações em nível de cluster usando o OpenSearch REST APIs. Para saber mais, consulte a Referência da API do Amazon OpenSearch Service the section called “Operações compatíveis” e.

Se você precisar de mais informações sobre o desempenho do cluster, poderá publicar logs de erro e logs lentos no CloudWatch.

Erro de snapshot “Not Valid for the Object's Storage Class” (Inválido para a classe de armazenamento do objeto)

OpenSearch Os snapshots de serviço não são compatíveis com a classe de armazenamento Amazon Glacier. Você poderá encontrar esse erro ao tentar listar os snapshots se o bucket do S3 incluir uma regra de ciclo de vida que faça a transição de objetos para a classe de armazenamento do Amazon Glacier.

Se você precisar restaurar um snapshot armazenado no bucket, restaure os objetos do Amazon Glacier, copie-os em um novo bucket e registre o novo bucket como um repositório de snapshots.

Cabeçalho de host inválido

OpenSearch O serviço exige que os clientes especifiquem `Host` nos cabeçalhos da solicitação. Um valor `Host` válido é o endpoint do domínio sem `https://`, como:

```
Host: search-my-sample-domain-ih2lhn2ew2scurji.us-west-2.es.amazonaws.com
```

Se você receber um `Invalid Host Header` erro ao fazer uma solicitação, verifique se seu cliente ou proxy inclui o endpoint do domínio OpenSearch Service (e não, por exemplo, seu endereço IP) no `Host` cabeçalho.

Tipo de instância M3 inválido

OpenSearch O serviço não oferece suporte à adição ou modificação de instâncias M3 em domínios existentes em execução OpenSearch ou nas versões 6.7 e posteriores do Elasticsearch. Você pode continuar a usar instâncias M3 com o Elasticsearch 6.5 e anteriores.

Recomendamos escolher um tipo de instância mais recente. Para domínios que executam o OpenSearch Elasticsearch 6.7 ou posterior, a seguinte restrição se aplica:

- Se o domínio existente não usar instâncias M3, você não poderá mais alterar para elas.
- Se você alterar um domínio existente de um tipo de instância M3 para outro tipo de instância, não será possível alternar novamente.

As consultas quentes param de funcionar após a ativação UltraWarm

Quando você ativa UltraWarm em um domínio, se não houver substituições preexistentes na `search.max_buckets` configuração, o OpenSearch Service define automaticamente o valor como para evitar que consultas com muita memória `10000` saturem os nós quentes. Se suas hot queries estiverem usando mais de 10.000 buckets, elas poderão parar de funcionar quando você ativar. UltraWarm

Como você não pode modificar essa configuração devido à natureza gerenciada do Amazon OpenSearch Service, você precisa abrir um caso de suporte para aumentar o limite. Os aumentos de limite não exigem uma assinatura do Premium Support.

Não é possível reverter para a versão anterior após a atualização.

As atualizações no local são irreversíveis, mas se você entrar em contato com o AWS Support, eles poderão ajudar a restaurar o snapshot automático anterior à atualização em um novo domínio. Por exemplo, se você atualizar um domínio do Elasticsearch 5.6 para 6.4, o AWS Support poderá ajudá-lo a restaurar o snapshot de pré-atualização em um novo domínio do Elasticsearch 5.6. Se você tirou um snapshot manual do domínio original, pode realizar essa etapa por conta própria.

Precisa de um resumo dos domínios para todos Regiões da AWS

O script a seguir usa o AWS CLI comando describe-regions do Amazon EC2 para criar uma lista de todas as regiões nas quais OpenSearch o serviço pode estar disponível. Em seguida, exige list-domain-namesque cada região:

```
for region in `aws ec2 describe-regions --output text | cut -f4`
do
    echo "\nListing domains in region '$region':"
    aws opensearch list-domain-names --region $region --query 'DomainNames'
done
```

Você recebe a seguinte saída para cada região:

```
Listing domains in region:'us-west-2'...
[
  {
    "DomainName": "sample-domain"
  }
]
```

As regiões nas quais o OpenSearch serviço não está disponível retornam “Não foi possível conectar-se ao URL do endpoint”.

Erro do navegador ao usar OpenSearch painéis

Seu navegador agrupa mensagens de erro de serviço em objetos de resposta HTTP quando você usa painéis para visualizar dados em seu domínio de OpenSearch serviço. Você pode usar ferramentas de desenvolvedor normalmente disponíveis em navegadores da web, como Modo de Desenvolvedor no Chrome, para visualizar erros de serviço subjacentes e auxiliar suas operações de depuração.

Para visualizar erros de serviço no Chrome

1. Na barra de menu superior do Chrome, escolha Visualizar, Desenvolvedor , Ferramentas do desenvolvedor.
2. Escolha a guia Redes.
3. Na coluna Status, escolha qualquer sessão HTTP com status 500.

Para visualizar erros de serviço no Firefox

1. No menu, escolha Tools, Web Developer, Network.
2. Escolha qualquer sessão HTTP com status 500.
3. Escolha a guia Response para visualizar a resposta do serviço.

Distorção de armazenamento e de fragmentos do nó

A distorção de fragmentos de nós ocorre quando um ou mais nós em um cluster têm significativamente mais fragmentos do que os outros nós. A distorção de armazenamento de nós ocorre quando um ou mais nós em um cluster têm significativamente mais armazenamento (`disk.indices`) do que os outros nós. Embora essas duas condições possam ocorrer temporariamente, como quando um domínio substituiu um nó e ainda está alocando fragmentos a ele, você deve resolvê-las se elas persistirem.

Para identificar os dois tipos de distorção, execute a operação _cat/allocation da API e compare as entradas `shards` e `disk.indices` na resposta:

```
shards    | disk.indices  | disk.used  | disk.avail   | disk.total   | disk.percent |
host     | ip       | node
   264    |      465.3mb  |   229.9mb  |      1.4tb   |      1.5tb   |            0 |
 x.x.x.x  | x.x.x.x  | node1
   115    |        7.9mb  |    83.7mb  |     49.1gb   |     49.2gb   |            0 |
x.x.x.x  | x.x.x.x  | node2
   264    |      465.3mb  |   235.3mb  |      1.4tb   |      1.5tb   |            0 |
 x.x.x.x  | x.x.x.x  | node3
   116    |        7.9mb  |    82.8mb  |     49.1gb   |     49.2gb   |            0 |
x.x.x.x  | x.x.x.x  | node4
   115    |        8.4mb  |      85mb  |     49.1gb   |     49.2gb   |            0 |
x.x.x.x  | x.x.x.x  | node5
```

Embora alguma distorção de armazenamento seja normal, qualquer coisa 10% acima da média é significativa. Quando a distribuição de fragmentos é distorcida, o uso da CPU, da rede e da largura de banda do disco também pode ficar distorcido. Como mais dados geralmente significam mais operações de indexação e pesquisa, os nós mais pesados também tendem a ser os nós com mais recursos, enquanto os nós mais leves representam capacidade subutilizada.

Correção: use contagens de fragmentos que sejam múltiplos da contagem de nós de dados para garantir que cada índice seja distribuído uniformemente entre os nós de dados.

Distorção de armazenamento e de fragmentos de índices

A distorção de fragmentos de índices ocorre quando um ou mais nós retêm mais fragmentos de um índice do que os outros nós. A distorção de armazenamento de índices ocorre quando um ou mais nós retêm uma quantidade desproporcionalmente grande do armazenamento total de um índice.

A distorção de índices é mais difícil de identificar do que a distorção de nós porque requer alguma manipulação da saída da API _cat/shards. Investigue a distorção de índices se houver alguma indicação de distorção nas métricas do cluster ou do nó. Estas são indicações comuns de distorção de índices:

- Erros HTTP 429 que ocorrem em um subconjunto de nós de dados
- Índice desigual ou enfileiramento de operações de pesquisa nos nós de dados
- Utilização desigual da and/or CPU do heap JVM em todos os nós de dados

Correção: use contagens de fragmentos que sejam múltiplos da contagem de nós de dados para garantir que cada índice seja distribuído uniformemente entre os nós de dados. Se você ainda vê armazenamento de índice ou distorção de fragmentos, talvez seja necessário forçar uma realocação de fragmentos, o que ocorre com cada implantação azul/verde do seu domínio de serviço. OpenSearch

Operação não autorizada após a seleção do acesso via VPC

Ao criar um novo domínio usando o console OpenSearch de serviço, você tem a opção de selecionar VPC ou acesso público. Se você selecionar o acesso à VPC, o OpenSearch serviço consultará as informações da VPC e falhará se você não tiver as permissões adequadas:

```
You are not authorized to perform this operation. (Service: AmazonEC2; Status Code:
 403; Error Code: UnauthorizedOperation
```

Para habilitar esta consulta, você deve ter acesso às operações `ec2:DescribeVpcs`, `ec2:DescribeSubnets` e `ec2:DescribeSecurityGroups`. Esse requisito se aplica somente ao console. Se você usa a AWS CLI para criar e configurar um domínio com um VPC endpoint, não precisará acessar essas operações.

Preso no carregamento após a criação do domínio da VPC

Depois de criar um novo domínio que usa o acesso da VPC, o Estado de configuração do domínio pode ficar travado em Carregando. Se esse problema ocorrer, você provavelmente desativou AWS Security Token Service (AWS STS) para sua região.

Para adicionar VPC endpoints à sua VPC, o OpenSearch serviço precisa assumir a função. `AWSServiceRoleForAmazonOpenSearchService` Portanto, AWS STS deve estar habilitado para criar novos domínios que usem o acesso VPC em uma determinada região. Para saber mais sobre como ativar e desativar AWS STS, consulte o Guia do usuário do IAM.

Solicitações negadas para a OpenSearch API

Com a introdução do controle de acesso baseado em tags para a OpenSearch API, você pode começar a ver erros de acesso negado onde não via antes. Isso pode ocorrer porque uma ou mais de suas políticas de acesso contém `Deny` usando a condição `ResourceTag`, e essas condições agora estão sendo aplicadas.

Por exemplo, a política antigamente só negava acesso à ação `CreateDomain` da API de configuração quando o domínio tinha a tag`environment=production`. Mesmo que a lista de ações também incluísse `ESHttpPut`, a declaração de negação não se aplicava a essa ação ou a qualquer outras ações `ESHttp*`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [{
    "Action": [
      "es:CreateDomain",
      "es:ESHttpPut"
    ],
    "Effect": "Deny",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
```

```
      }
    }
  }]
}
```

Com o suporte adicional de tags para métodos OpenSearch HTTP, uma política baseada em identidade do IAM, como a descrita acima, resultará na negação do acesso à ação do usuário vinculado. `ESHttpPut` Anteriormente, na ausência de validação de tags, o usuário anexado ainda teria conseguido enviar solicitações PUT.

Se você começar a encontrar erros de acesso negado após atualizar os domínios para o software de serviço R20220323 ou posterior, verifique as políticas de acesso baseadas em identidade para ver se o caso descrito aqui está ocorrendo e atualize-as, se necessário, para permitir o acesso.

Não é possível conectar via Alpine Linux

O Alpine Linux limita o tamanho da resposta DNS a 512 bytes. Se você tentar se conectar ao seu domínio de OpenSearch serviço a partir da versão 3.18.0 ou inferior do Alpine Linux, a resolução de DNS poderá falhar se o domínio estiver em uma VPC e tiver mais de 20 nós. Se você usa uma versão Alpine Linux superior à 3.18.0, você deve ser capaz de resolver mais de 20 hosts. Para saber mais, consulte notas de lançamento do Alpine Linux 3.18.0.

Se seu domínio estiver em uma VPC, recomendamos usar outras distribuições Linux, como Debian, Ubuntu, CentOS, Red Hat Enterprise Linux ou Amazon Linux 2, para conectar a ele.

Muitas solicitações de pesquisa de contrapressão

O controle de admissão baseado em CPU é um mecanismo de controle que limita proativamente o número de solicitações a um nó com base em sua capacidade atual, tanto para aumentos orgânicos quanto para picos de tráfego. Solicitações excessivas retornam um código de status HTTP 429 “Muitas solicitações” após a rejeição. Esses erros indicam recursos de cluster insuficientes, solicitações de pesquisa que consomem muitos recursos ou um aumento não intencional na workload.

A contrapressão de pesquisa fornece o motivo da rejeição, o que pode ajudar a ajustar solicitações de pesquisa que consomem muitos recursos. Para picos de tráfego, recomendamos novas tentativas do lado do cliente com recuo e instabilidade exponenciais.

Erro de certificado ao usar o SDK

Como você AWS SDKs usa os certificados CA do seu computador, as alterações nos certificados nos AWS servidores podem causar falhas de conexão quando você tenta usar um SDK. As mensagens de erro variam, mas geralmente contêm o seguinte texto:

```
Failed to query OpenSearch
...
SSL3_GET_SERVER_CERTIFICATE:certificate verify failed
```

Você pode evitar essas falhas mantendo os certificados CA e o sistema operacional do seu computador up-to-date. Se encontrar esse problema em um ambiente corporativo e não gerenciar seu computador, talvez tenha de pedir a um administrador para auxiliá-lo no processo de atualização.

A lista a seguir mostra as versões mínimas de sistema operacional e Java:

- As versões do Microsoft Windows que têm atualizações de janeiro de 2005 ou posteriores instaladas contêm pelo menos um dos requisitos CAs em sua lista de confiança.
- O Mac OS X 10.4 com Java para Mac OS X 10.4 Release 5 (fevereiro de 2007), Mac OS X 10.5 (outubro de 2007) e versões posteriores contêm pelo menos um dos requisitos CAs em sua lista de confiança.
- O Red Hat Enterprise Linux 5 (março de 2007), 6 e 7 e o CentOS 5, 6 e 7 contêm pelo menos um dos requisitos CAs em sua lista de CAs confiáveis padrão.
- O Java 1.4.2_12 (maio de 2006), 5 Update 2 (março de 2005) e todas as versões posteriores, incluindo Java 6 (dezembro de 2006), 7 e 8, contêm pelo menos um dos requisitos CAs em sua lista de CAs confiáveis padrão.

As três autoridades de certificação são:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority – G2
- Starfield Class 2 Certification Authority

Os certificados raiz das duas primeiras autoridades estão disponíveis no Amazon Trust Services, mas manter seu computador up-to-date é a solução mais simples. Para saber mais sobre os certificados fornecidos pelo ACM, consulte. AWS Certificate Manager FAQs

> **Note**
>
> Atualmente, os domínios OpenSearch de serviço na região us-east-1 usam certificados de uma autoridade diferente. Pretendemos atualizar a região para usar essas novas autoridades de certificação em um futuro próximo.

Ocorre falha na instalação do plug-in personalizado devido a incompatibilidade entre versões

Problema: A instalação do plug-in falhou devido a uma incompatibilidade de versão entre o plug-in e a OpenSearch instância em execução. O sistema retorna o seguinte erro:

```
PluginValidationFailureReason : The provided plugin could not be loaded.
```

Causa: O plug-in foi compilado para OpenSearch ${*MAJOR*}. ${*MINOR*}.{*PATCH*}, mas seu ambiente está executando OpenSearch ${*MAJOR*}. $ {*MINOR*} 0,0. OpenSearch requer uma correspondência exata de versões entre os plug-ins e a OpenSearch instalação principal por motivos de estabilidade e segurança.

Possível correção: Crie o plug-in com a OpenSearch versão ${*MAJOR*}. $ {*MINOR*} .0 para corresponder à versão do seu cluster.

Para verificar e atualizar a versão do OpenSearch

1. Use a API ou o painel do cluster para executar o comando a seguir. Substitua *default placeholder values* por suas próprias informações.

   Solicitação de API:

   ```
   curl -X GET your-opensearch-endpoint/
   ```

   Console de Ferramentas de Desenvolvimento no painel:

   ```
   GET /
   ```

   O comando retorna informações no formato a seguir.

```
{
  "name": "node-id",
  "cluster_name": "account-id:domain-name",
  "cluster_uuid": "cluster-uuid",
  "version": {
    "distribution": "opensearch",
    "number": "2.17.0",
    "build_type": "tar",
    "build_hash": "unknown",
    "build_date": "2024-12-17T11:00:09.799828091Z",
    "build_snapshot": false,
    "lucene_version": "9.11.1",
    "minimum_wire_compatibility_version": "7.10.0",
    "minimum_index_compatibility_version": "7.0.0"
  },
  "tagline": "The OpenSearch Project: https://opensearch.org/"
}
```

2. Se o número da versão não for ${*MAJOR*}. $ {*MINOR*} .0, reconstrua o plug-in fazendo o seguinte:

   a. Atualize o plug-in `descriptor.properties` para especificar a versão ${*MAJOR*}. $ {*MINOR*} 0,0.

   b. Recompile o plug-in usando o comando para seu tipo de projeto.

   c. Execute o comando update-package usando o arquivo recém-compilado `.zip`.

   d. Execute o comando associate-package para associar a versão mais recente do plug-in, criada quando você executou o `update-package` comando na etapa anterior.

Histórico de documentos do Amazon OpenSearch Service

Este tópico descreve mudanças importantes no Amazon OpenSearch Service. As atualizações de software de serviço adicionam suporte a novos recursos, patches de segurança, correções de bugs e outras melhorias. Para usar novos recursos, talvez seja necessário atualizar o software de serviço em seu domínio. Para obter mais informações, consulte the section called “Atualizações de software de serviço”.

Os recursos do serviço são implementados de forma incremental até Regiões da AWS onde o serviço está disponível. Atualizamos esta documentação apenas para a primeira versão. Não fornecemos informações sobre a disponibilidade da região nem anunciamos lançamentos subsequentes da região. Para obter informações sobre a disponibilidade de recursos do serviço na região e para assinar notificações sobre atualizações, consulte O que há de novo em AWS?

Para receber notificações sobre atualizações, inscreva-se no feed RSS.

> **Note**
>
> Lançamentos de patches: versões de software de serviço que terminam em “-P” e um número, como R20211203-P4, são lançamentos de patches. É provável que os patches incluam melhorias de performance, pequenas correções de bugs e correções de segurança ou melhorias de postura. Como os patches não incluem novos recursos ou as últimas alterações, eles geralmente não têm impacto direto no usuário ou na documentação e, por isso, as especificidades de cada patch não estão incluídas neste histórico de documentos.

Alteração	Descrição	Data
Support para OpenSearch 3.5	O Amazon OpenSearch Service é compatível com as OpenSearch versões 3.4 e 3.5. Para obter mais informações, consulte as notas da versão OpenSearch 3.4 e as notas da versão OpenSearch 3.5. Para obter informações sobre as	17 de março de 2026

	operações OpenSearch 3.4 e 3.5, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico. Para obter uma lista completa das alterações mais importantes, consulte a OpenSearch documentação aqui.	
Novos recursos: aceleração de GPU para índices vetoriais, otimização automática e ingestão de vetores	O Amazon OpenSearch Service agora oferece suporte à aceleração por GPU para índices vetoriais, permitindo operações mais rápidas de pesquisa vetorial. O Amazon OpenSearch Service também introduz a otimização automática para ajustar automaticamente o desempenho da coleta e a ingestão de vetores para simplificar o carregamento de dados. Para obter mais informações, consulte Aceleração de GPU para índices vetoriais, Otimização automática e Ingestão de vetores.	2 de dezembro de 2025

Support para OpenSearch 3.3	O Amazon OpenSearch Service é compatível com as OpenSearch versões 3.2 e 3.3. Para obter mais informações, consulte as notas da versão OpenSearch 3.2 e as notas da versão OpenSearch 3.3. Para obter informações sobre as operações OpenSearch 3.2 e 3.3, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.	19 de novembro de 2025
Compatibilidade com as famílias de instâncias baseadas no Graviton4 (c8g, m8g, r8g r8gd)	O Amazon OpenSearch Service agora oferece suporte a famílias de instâncias baseadas em Graviton4 (c8g, m8g, r8g, r8gd). Para obter mais informações, consulte Tipos de instância compatíveis no Amazon OpenSearch Service.	17 de outubro de 2025

Novo recurso: inferência OpenSearch de lote offline de aprendizado de máquina de ingestão	Os pipelines do Amazon OpenSearch Ingestion (OSI) oferecem suporte ao processamento de inferência em lote offline de aprendizado de máquina (ML) para enriquecer com eficiência grandes volumes de dados a baixo custo. Use inferência em lote offline sempre que tiver grandes conjuntos de dados que possam ser processados de modo assíncrono. A inferência de lotes offline funciona com o Amazon Bedrock e SageMaker os modelos. Esse recurso está disponível em todos os domínios Regiões da AWS compatíveis com OpenSearch Ingestão com o OpenSearch Service 2.17+. Para obter mais informações, consulte Como usar um pipeline OpenSearch de ingestão com inferência de lote offline de aprendizado de máquina.	2 de outubro de 2025

Novo recurso: CloudFormation modelos para configurar inferência remota para realce semântico	O destaque semântico é um atributo de pesquisa avançada que aumenta a relevância dos resultados analisando o significado e o contexto das consultas, em vez de confiar apenas em correspondências exatas de palavras-chave. Esse recurso usa modelos de machine learning para avaliar a semelhança semântica entre as consultas da pesquisa e o conteúdo do documento ao identificar e destacar as frases ou passagens mais contextualmente relevantes dos documentos. O Amazon OpenSearch Service agora inclui um CloudFormation modelo que configura automaticamente os SageMaker modelos da Amazon para realce semântico. Para saber mais, consulte Modelos de inferência remota para destaque semântico.	26 de setembro de 2025
A política AmazonOpenSearchIngestionServiceRolePolicy gerenciada foi atualizada	A atualização dá permissão ao OpenSearch Ingestion para modificar os VPC endpoints criados OpenSearch por para compartilhar pipelines entre eles. VPCs	19 de setembro de 2025

Novo recurso: configurar pipelines OpenSearch de ingestão para ingestão entre contas	Para fontes baseadas em push, como HTTP e OTel, o Amazon OpenSearch Ingestion permite que você compartilhe pipelines entre Contas da AWS uma nuvem privada virtual (VPC) e um endpoint de pipeline em uma VPC separada. Para obter mais informações, consulte Como configurar pipelines de OpenSearch ingestão para ingestão entre contas	19 de setembro de 2025
O Amazon OpenSearch Serverless oferece suporte à pesquisa vetorial baseada em disco	A pesquisa vetorial baseada em disco reduz significativamente os custos operacionais de workloads vetoriais em ambientes com pouca memória. Para saber mais, consulte Pesquisa vetorial baseada em disco.	18 de setembro de 2025

Support para OpenSearch 3.1	O Amazon OpenSearch Service é compatível com OpenSearch a versão 3.1. Para saber mais sobre compatibilidade com atualizações, consulte Caminhos de atualização compatíveis. Para obter mais informações sobre as mudanças nas novas versões, consulte a documentação OpenSearch 3.0 e 3.1. Para obter informações sobre as operações OpenSearch 3.1, consulte a referência da API OpenSearch REST ou a referência da API para o plug-in específico.	9 de setembro de 2025
O Amazon OpenSearch Serverless oferece suporte aos padrões federais de processamento de informações (FIPS) 140-2	O FIPS 140-2 é um padrão do governo americano e do governo canadense que especifica os requisitos de segurança para módulos criptográficos que protegem informações sensíveis. Os endpoints FIPS Amazon OpenSearch Serverless estão disponíveis Regiões da AWS onde o FIPS é compatível. Esses endpoints usam o TLS 1.2 ou posteriores e algoritmos criptográficos validados por FIPS em todas as comunicações.	5 de setembro de 2025

Novo atributo: controle de acesso refinado baseado em atributo SAML	O Amazon OpenSearch Service oferece suporte ao controle de acesso refinado com SAML para mapear dinamicamente usuários e grupos do seu provedor de identidade para usuários e funções de controle de acesso OpenSearch refinados. Você pode definir o escopo dessas funções para OpenSearch domínios específicos e coleções sem servidor, além de definir permissões em nível de índice e segurança em nível de documento. Para saber mais, consulte Configurar o controle de acesso refinado baseado em atributo SAML.	8 de agosto de 2025

Novo tópico: endpoints e OpenSearch cotas de interface do usuário	A Amazon OpenSearch UI expande sua experiência de análise operacional moderniza da para novas Regiões da AWS, incluindo Ásia-Pací fico (Hyderabad), Ásia-Pací fico (Osaka), Ásia-Pacífico (Seul), Europa (Milão), Europa (Zurique), Europa (Espanha) e Oeste dos EUA (Norte da Califórnia), permitindo que você obtenha insights sobre dados que abrangem domínios gerenciados e coleções sem servidor a partir de um único endpoint. OpenSearch A interface do usuário pode se conectar a OpenSearch domínios acima da versão 1.3 e coleções OpenSearch sem servidor. Agora disponível em 22 Regiões da AWS.	7 de agosto de 2025

Novos atributos: enriquecimento semântico automático	O enriquecimento semântico automático simplifica a implementação e os recursos da pesquisa semântica no Amazon Service. OpenSearch A pesquisa semântica retorna resultados de consulta que incorporam não apenas a correspondência às palavras-chave, mas a intenção e o significado contextual da pesquisa do usuário. Para saber mais, consulte Sobre enriquecimento semântico automático. O enriquecimento semântico automático oferece os seguintes benefícios: Implantação simplificada Você não precisa ter experiência em machine learning (ML) ou integrações complexas. Processo automatizado O enriquecimento semântico acontece automaticamente durante a ingestão de dados. Mais relevância de consulta O enriquecimento semântico melhora a qualidade e a precisão	6 de agosto de 2025

	contextual dos resultados da pesquisa.
Escalabilidade	
	O enriquecimento semântico aplica a pesquisa semântica a grandes conjuntos de dados sem intervenção manual.

Novo recurso: Machine Learning no Amazon OpenSearch Serverless	OpenSearch O Serverless agora oferece suporte a recursos de Machine Learning para fluxos de trabalho de IA, como RAG (Retrieval-Augmented Generation) e pesquisa semântica. Use conectores para acessar algoritmos de ML e modelos remotos hospedados em plataformas de terceiros ou em Serviços da AWS. Para obter mais informaçõ es, consulte Configurar o Machine Learning no Amazon OpenSearch Serverless. O Machine Learning on OpenSearch Serverless inclui os seguintes recursos: Pesquisa neural e pesquisa híbrida Execute operações de pesquisa semântica que entendem o contexto e o significado da consulta para conseguir resultados mais relevantes. Combine a pesquisa de palavra-chave e a pesquisa semântica com a pesquisa híbrida para aumentar a relevância. Para obter mais informações, consulte	6 de agosto de 2025

Configurar a pesquisa neural e a pesquisa híbrida OpenSearch sem servidor.

Conectores e modelos

Conecte-se a modelos hospedados em plataformas de ML de terceiros e nos Serviços da AWS. Configure conectores com modelos e implante-os para executar previsões e fluxos de trabalho de IA.

Fluxos de trabalho

Automatize tarefas complexas de configuração e pré-processamento de ML encadeando várias chamadas de API. Os fluxos de trabalho simplificam a criação de aplicativos de IA em. OpenSearch Para obter mais informações, consulte Configurar fluxos de trabalho sem OpenSearch servidor.

Novos recursos: OpenSearch instantâneos sem servidor	Snapshots são point-in-time backups de suas coleções Amazon OpenSearch Serverless que fornecem recursos de recuperação de desastres. OpenSearch O Serverless cria e gerencia automaticamente instantâneos de suas coleções, garantindo a continuidade dos negócios e a proteção dos dados. Benefícios principais • Backups automáticos de hora em hora, sem necessidade de configuração manual • Nenhum custo adicional de manutenção • Nenhum custo adicional de armazenamento • Recuperação rápida em caso de perda acidental de dados • Capacidade de restaurar índices específicos a partir de um snapshot Para saber mais, consulte Fazer backup de coleções usando snapshots.	30 de julho de 2025

OpenSearch A ingestão oferece suporte a duas novas fontes	OpenSearch A ingestão agora é compatível com as seguintes fontes: • Amazon Aurora • Amazon RDS	17 de julho de 2025

Novos recursos: integração OpenSearch de serviços com Amazon S3 Vectors	Essa integração oferece os seguintes recursos para aprimorar consideravelmente a pesquisa vetorial no OpenSearch Service. • O Amazon S3 Vectors disponibiliza o primeiro armazenamento de objetos em nuvem com compatibilidade nativa com armazenamento e consulta de vetores. O S3 Vectors oferece armazenamento vetorial econômico, elástico e durável que pode ser consultado com base em significado semântico e similaridade. Ele oferece tempos de resposta a consulta de menos de um segundo e custos até 90% mais baixos para carregar, armazenar e consultar vetores. • O Amazon OpenSearch Service oferece a capacidad e de usar o Amazon S3 como um mecanismo vetorial para índices vetoriais. Esse atributo permite descarregar dados vetoriais para o Amazon S3 enquanto mantém recursos de pesquisa vetorial em	15 de julho de 2025

	fração de segundos por um preço baixo. Para obter mais informações, consulte (Prévia) Integração OpenSearch de serviços com Amazon S3 Vectors.	
Novo tópico: Integração da Atlassian com o Inestion OpenSearch	Agora você pode usar os pipelines de OpenSearch ingestão da Amazon para processar dados dos serviços Atlassian Jira e Confluenc e. Essa integração permite que você crie uma base de conhecimento pesquisáv el unificada sincronizando projetos completos do Jira e espaços do Confluence em. OpenSearch A integraçã o mantém a relevância em tempo real por meio de monitoramento contínuo e sincronização automática das atualizações, ao mesmo tempo que oferece opções flexíveis de filtragem para projetos, tipos de problemas e conteúdos específicos. Para obter mais informaçõ es, consulte Como usar um pipeline de OpenSearch ingestão com os serviços da Atlassian.	5 de junho de 2025

Gerenciamento personalizado de plug-ins com o AWS CLI	Agora você pode gerenciar plug-ins personalizados para o Amazon OpenSearch Service usando AWS CLI o. Esse atributo permite instalar, atualizar, proteger e desinstal ar plug-ins personalizados por meio de um fluxo de trabalho de linha de comando otimizado. Você também pode proteger seus plug-ins com AWS KMS chaves e gerenciar atualizações de plug-ins ao atualizar para versões mais recentes do. OpenSearch Para obter mais informaçõ es, consulte Gerenciamento de plug-ins personalizados usando a AWS CLIAWS KMS integração de pacotes personalizados do Amazon OpenSearch Service	27 de maio de 2025
Support para OpenSearch 2.19	O Amazon OpenSearch Service é compatível com OpenSearch a versão 2.19. Para obter mais informaçõ es, consulte as notas de versão OpenSearch 2.19. Para obter informações sobre as operações da OpenSearc h versão 2.19, consulte a referência da API OpenSearc h REST ou a referência da API para o plug-in específico.	30 de abril de 2025

Novo suporte para especificar funções de pipeline no Ingestion OpenSearch	Ao criar ou atualizar o pipeline no Amazon OpenSearch Ingestion, você não tem mais uma função específica do pipeline em uma configuração de pipeline no formato YAML. Em vez disso, no console, você especifica o perfil em um novo campo Perfil do pipeline. Se você estiver usando o AWS CLI, use um novo `--pipeline-role-arn` parâmetro. Para saber mais, consulte os seguintes tópicos: • Configurando funções e usuários na Amazon OpenSearch Ingestion • Criação de pipelines OpenSearch de ingestão da Amazon • Tutorial: Ingestão de dados em um domínio usando o Amazon OpenSearch Ingestion • Tutorial: Ingestão de dados em uma coleção usando o Amazon OpenSearch Ingestion • CreatePipeline • UpdatePipeline	22 de abril de 2025

Novo tópico	O tópico de solução de problemas Falha na instalação de plugin personalizado devido à compatibilidade de versão aborda as etapas a serem seguidas quando você encontra o erro `PluginValidationFailureReason : The provided plugin could not be loaded` . Esse erro geralmente é resultado de uma incompatibilidade de versão entre o plug-in e a OpenSearch instância em execução.	21 de abril de 2025
Novo recurso: compatibilidade de OpenSearch interface de usuário com pesquisa entre clusters	OpenSearch A interface do usuário agora é compatível com a pesquisa entre clusters. Isso possibilita que você use a OpenSearch interface de usuário em uma Região da AWS para acessar clusters em uma região diferente. Isso é feito configurando-o como um cluster remoto conectado a um cluster na mesma região. Para saber mais, consulte Acesso a dados entre regiões e entre contas com pesquisa entre clusters.	16 de abril de 2025

Conteúdo revisado: “Usando a OpenSearch interface do usuário no Amazon OpenSearch Service”	Revisamos e expandimos o capítulo Usando a interface do usuário OpenSearch no Amazon OpenSearch Service, incluindo procedimentos expandidos, detalhes adicionais de configuração e um novo histórico de lançamento do suporte do Amazon OpenSearch Service para interface do usuário. OpenSearch	15 de abril de 2025
Desenvolvedor Amazon Q para Amazon OpenSearch Service	A integração do Amazon Q com o Amazon OpenSearch Service oferece os seguintes recursos generativos: • Gerar visualizações usando linguagem natural • Visualizar resumos e insights de alertas • Visualizar os resumos dos resultados das consultas gerados pelo Amazon Q na página Discover • Visualizar os detectores de anomalias recomendados • Acesse o chat do Amazon Q para perguntas OpenSearch relacionadas	31 de março de 2025

A política AmazonOpenSearchServiceRolePolicy gerenciada foi atualizada	A política permite OpenSearch atualizar o escopo de acesso de qualquer Centro de Identidade do AWS IAM aplicativo que seja gerenciado somente pelo OpenSearch.	28 de março de 2025
OR2 e OM2 instâncias para Amazon OpenSearch Service	O Amazon OpenSearch Service agora oferece suporte OR2 e OM2 instâncias. Em benchmarks internos, OR2 mostrou uma taxa de transferência de indexação até 26% melhor do que OR1 e 70% em relação ao R7. OM2 mostrou-se até 15% melhor OR1 e 66% acima do m7g.	25 de março de 2025
Consulta direta para CloudWatch Logs e Security Lake	O Amazon OpenSearch Service agora oferece suporte a consultas diretas para consultar dados no CloudWatch Logs e no Security Lake.	1.º de dezembro de 2024
Índices k-NN	A partir da OpenSearch versão 2.17, você pode mover os índices k-NN para os níveis de armazenamento refrigerado UltraWarm e.	13 de novembro de 2024

OpenSearch Suporte 2.17	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.17. Esta versão inclui todos os recursos que faziam parte das versões 2.16 e 2.17. Para saber mais, consulte as notas de release 2.16 e 2.17.	13 de novembro de 2024
OpenSearch Suporte 2.15	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.15. Esta versão inclui todos os recursos que faziam parte das versões 2.14 e 2.15. Para saber mais, consulte as notas de release 2.14 e 2.15.	11 de outubro de 2024
Nova função vinculada ao serviço	O Amazon OpenSearch Service adiciona uma função vinculada ao serviço chamada`AWSServiceRoleForOpensearchIngestionSelfManagedVpce`, que permite que o Amazon OpenSearch Ingestion envie dados métricos Amazon CloudWatch para pipelines com endpoints VPC autogerenciados.	12 de junho de 2024

Consultas diretas	O Amazon OpenSearch Service agora oferece suporte à execução de consultas diretamente nos dados armazenados no Amazon S3, sem a necessidade de ingerir os dados em um índice. OpenSearch	22 de maio de 2024
OpenSearch 2.13 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.13. Essa versão inclui todos os recursos que faziam parte das versões 2.12 e 2.13. Para saber mais, consulte as notas de lançamento das versões2.12 e 2.13.	21 de maio de 2024
Suporte da Amazon OpenSearch Ingestion para Data Prepper versão 2.7	O Amazon OpenSearch Ingestion adiciona suporte à versão 2.7 do Data Prepper. Para saber mais, consulte as notas de lançamento da versão 2.7.	4 de abril de 2024
AWS service (Serviço da AWS) acesso privado para coleções sem OpenSearch servidor	Agora você pode conceder acesso específico Serviços da AWS, como o Amazon Bedrock, às suas coleções OpenSearch sem servidor dentro de uma política de acesso à rede.	28 de março de 2024

Atualizações do EBS no local	Agora você pode fazer algumas alterações no EBS em seus domínios sem uma implantação azul/verde no Amazon Service. OpenSearch	14 de fevereiro de 2024
Visibilidade de alterações de configuração	Agora você pode acompanha r as alterações na configura ção do domínio no console do Amazon OpenSearch Service e usando a API de configura ção.	6 de fevereiro de 2024
Disponibilidade geral das coleções de pesquisa vetorial	As coleções de pesquisa vetorial do Amazon OpenSearch Serverless agora estão disponíveis ao público em geral. As melhorias a seguir foram feitas durante a fase de pré-visualização: • Coleções de pesquisa vetorial agora oferecem suporte a workloads com bilhões de vetores, cada um com até 128 dimensões. • OpenSearch Os painéis agora oferecem suporte a coleções de pesquisa vetorial.	29 de novembro de 2023
OR1 Instâncias	O Amazon OpenSearch Service agora oferece suporte a tipos de OR1 instância.	29 de novembro de 2023

Consultas diretas com o Amazon S3 (demonstração)	As consultas diretas fornecem uma solução totalmente gerenciada para disponibilizar dados transacionais no Amazon OpenSearch Service em segundos após serem gravados em um bucket do Amazon S3.	29 de novembro de 2023
Capacidade de 10 TiB para coleções de séries temporais	O Amazon OpenSearch Serverless adiciona suporte para até 10 TiB de dados de índice para coleções de séries temporais. Essa versão também suporta uma capacidade máxima permitida de 200 OCUs para todos os tipos de coleções e a capacidade de desativar réplicas em espera ao criar uma coleção.	29 de novembro de 2023
OpenSearch Suporte 2.11	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.11. Essa versão inclui todos os recursos que faziam parte das versões 2.10 e 2.11. Para saber mais, consulte as notas de release 2.10 e 2.11.	17 de novembro de 2023

Suporte da Amazon OpenSearch Ingestion para Data Prepper versão 2.6	O Amazon OpenSearch Ingestion adiciona suporte à versão 2.6 do Data Prepper. Para saber mais, consulte as notas de lançamento da versão 2.6. Além disso, agora você pode especific ar o Amazon DynamoDB como origem de pipeline. Para obter mais informações, consulte Uso de um pipeline de OpenSearch ingestão com o Amazon DynamoDB.	17 de novembro de 2023
Suporte da Amazon OpenSearch Ingestion para Data Prepper versão 2.5	O Amazon OpenSearch Ingestion adiciona suporte para a versão 2.5 do Data Prepper. Para saber mais, consulte as notas de lançamento da versão 2.5. Além disso, agora você pode especificar um domínio OpenSearch de serviço ou uma coleção OpenSearc h sem servidor como fonte de pipeline. Para obter mais informações, consulte o plug-in de OpenSearch origem na documentação do Data Prepper.	17 de novembro de 2023

CloudFormation modelo para inferência remota	Para facilitar a configuração da inferência remota para pesquisa semântica, o Amazon OpenSearch Service fornece um AWS CloudFormation modelo no console que automatiza o processo de provisionamento do modelo para você.	7 de novembro de 2023
Atualizar política de função vinculada ao serviço	Adiciona as permissões necessárias para que a política de função vinculada ao serviço atribua e IPv6 cancele `AmazonOpenSearchServiceRolePolicy` a atribuição de endereços. A política obsoleta do Elasticsearch `AmazonElasticsearchServiceRolePolicy` também foi atualizada para garantir a compatibilidade com versões anteriores.	26 de outubro de 2023

Políticas de ciclo de vida do Amazon OpenSearch Serverless	O Amazon OpenSearch Serverless introduz políticas de ciclo de vida de índices para simplificar o gerenciamento da retenção e exclusão de dados. Agora você pode usar APIs ou uma interface de configuração no console para definir políticas de retenção de dados para coletas de séries temporais, eliminando a necessidade de criar índices diários ou scripts para excluir dados antigos.	25 de outubro de 2023
Compatibilidade com instâncias Im4gn	O Amazon OpenSearch Service agora oferece suporte aos tipos de instância IM4gn. As instâncias Im4gn são otimizadas para cargas de trabalho que gerenciam grandes conjuntos de dados e precisam de alta densidade de armazenamento por vCPU.	20 de outubro de 2023
Opções administrativas	O Amazon OpenSearch Service agora oferece várias opções administrativas que fornecem controle granular se você precisar solucionar problemas com seu domínio. Essas opções incluem a capacidade de reiniciar o OpenSearch processo em um nó de dados e a capacidade de reiniciar um nó de dados.	17 de outubro de 2023

Plug-ins opcionais	O Amazon OpenSearch Service adiciona suporte para quatro novos plug-ins de análise de idiomas: Nori (coreano), Sudachi (japonês) , Pinyin (chinês) e STConvert Analysis (chinês), bem como o plug-in Amazon Personalize Search Ranking.	16 de outubro de 2023
OpenSearch 2.9 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.9. Essa versão inclui todos os recursos que faziam parte das versões 2.8 e 2.9. Para saber mais, consulte as notas de release 2.8 e 2.9.	2 de outubro de 2023
Conectores de ML	O Amazon OpenSearch Service adiciona suporte para conectores de aprendizado de máquina (ML). Os conectores facilitam o acesso a modelos de ML hospedados em outras Serviços da AWS plataformas de aprendizado de máquina (ML) ou em plataformas de aprendizado de máquina (ML) de terceiros.	6 de setembro de 2023

Amazon OpenSearch Ingestion adiciona suporte à versão 2.4 do Data Prepper	O Amazon OpenSearch Ingestion adiciona suporte à versão 2.4 do Data Prepper. Para saber mais, consulte as notas de lançamento da versão 2.4. Além disso, agora você pode especific ar o Amazon Managed Streaming for Apache Kafka (Amazon MSK) como origem do pipeline.	31 de agosto de 2023
Capacidade de 6 TiB para coleções de séries temporais	O Amazon OpenSearch Serverless adiciona suporte para até 6 TiB de dados de índice para coleções de séries temporais. Essa versão também suporta uma capacidade máxima permitida de 100 OCUs para pesquisas e coleções de séries temporais.	15 de agosto de 2023
Coleções de pesquisa vetorial	O Amazon OpenSearch Serverless adiciona a opção de criar uma coleção de pesquisa vetorial, que você pode usar para armazenar incorporações vetoriais para potencializar pesquisas semânticas e de similaridade.	26 de julho de 2023

OpenSearch Suporte 2.7	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.7. Essa versão inclui todos os recursos que faziam parte das versões 2.6 e 2.7. Para saber mais, consulte as notas de release 2.6 e 2.7.	10 de julho de 2023
Suporte à versão 2.3 do Data Prepper	O Amazon OpenSearch Ingestion adiciona suporte ao Data Prepper versão 2.3. Para saber mais, consulte as notas de lançamento da versão 2.3. Além disso, agora você pode especificar o Amazon Security Lake como origem de pipeline.	26 de junho de 2023
Multi-AZ com modo de espera	O Amazon OpenSearch Service adiciona a opção de implantar um domínio em três zonas de disponibilidade (AZs), com cada AZ contendo uma cópia completa dos dados e com os nós em uma delas AZs atuando como standby. A opção de implantação do multi-AZ com modo de espera fornece 99,99% de disponibilidade e desempenho consistente no caso de uma falha na infraestrutura.	3 de maio de 2023

Nova função vinculada ao serviço	O Amazon OpenSearch Service adiciona uma função vinculada ao serviço chamada`AWSServiceRoleForAmazonOpenSearchIngestionService` , que permite que o Amazon OpenSearch Ingestion envie dados métricos para. Amazon CloudWatch	26 de abril de 2023
OpenSearch Ingestão da Amazon	O Amazon OpenSearch Ingestion é um coletor de dados totalmente gerenciado que fornece dados de log e rastreamento em tempo real para domínios de OpenSearch serviço e OpenSearch coleções sem servidor. OpenSearch A ingestão elimina a necessidade de usar soluções de terceiros , como Logstash ou Jaeger, para ingerir dados em seus domínios e coleções.	26 de abril de 2023
OpenSearch 2.5 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.5. Essa versão inclui todos os recursos que faziam parte das versões 2.4 e 2.5. Para saber mais, consulte as notas de release 2.4 e 2.5.	13 de março de 2023

Janelas de manutenção fora do horário de pico	O Amazon OpenSearch Service adiciona períodos fora de pico, que são blocos de tempo diários de 10 horas e baixo tráfego, durante os quais ele pode agendar atualizaç ões de software do serviço e otimizações de ajuste automático que exigem uma implantação. blue/green As atualizações fora do horário de pico ajudam a minimizar a sobrecarga nos nós principai s dedicados de um cluster durante períodos de maior tráfego. Para novos domínios criados após 16 de fevereiro, a janela fora do horário de pico é configurada automaticamente entre 22h e 8h, horário local. Para domínios existentes, você precisa habilitar a janela manualmente.	16 de fevereiro de 2023
Configurar a autenticação SAML durante a criação do domínio	O Amazon OpenSearch Service agora oferece suporte à configuração da autentica ção SAML durante a criação do domínio. Anteriormente, era necessário configurar as opções de SAML após a criação do domínio.	1° de fevereiro de 2023

Reindexação remota para domínios de VPC	O Amazon OpenSearch Service adiciona a opção de uma conexão de VPC endpoint entre dois domínios. Agora, você pode utilizar a reindexação remota para copiar índices de um domínio de VPC para outro sem um proxy reverso. Seus domínios de VPC devem estar executando o software de serviço R20221114 ou posterior para usar esse recurso.	31 de janeiro de 2023

Disponibilidade OpenSearch geral do Amazon Serverless	O Amazon OpenSearch Serverless agora está disponível ao público em geral. As melhorias a seguir foram feitas durante a fase de pré-visualização: • Agora, a capacidade pode ser reduzida para o mínimo configurado OCUs quando há uma diminuição no tráfego no endpoint de coleta. • O máximo permitido OCUs para indexação e pesquisa foi aumentado de 20 para 50. Cada OCU inclui armazenamento efêmero de atividade muito alta que é suficiente para 120 GiB de dados de indexação. • Agora, é possível definir as configurações de acesso aos dados ao criar coleções, em vez de configurá-las em um fluxo de trabalho separado.	25 de janeiro de 2023

Simulação assíncrona	O Amazon OpenSearch Service agora oferece suporte à execução seca assíncron a, que permite realizar uma verificação de validação antes de fazer uma alteração na configuração e notifica se suas alterações causarão uma implantação. blue/green	19 de janeiro de 2023
Nova função vinculada ao serviço	O Amazon OpenSearc h Service adiciona uma função vinculada ao serviço chamada`AWSServic eRoleForAmazonOpen SearchServerless` , que permite que o OpenSearc h Serverless envie dados métricos para. Amazon CloudWatch	29 de novembro de 2022
OpenSearch Prévia do Amazon Serverless	O Amazon OpenSearch Serverless é uma configura ção sob demanda, com escalabilidade automátic a e sem servidor para o Amazon Service. OpenSearc h O Serverless remove as complexidades operacionais de provisionamento, configura ção e ajuste de seus clusters. OpenSearch	29 de novembro de 2022

OpenSearch 2.3 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 2.3. Essa versão inclui todos os recursos que faziam parte das versões 2.0, 2.1 e 2.2. Para saber mais, consulte as notas das versões 2.0, 2.1, 2.2 e 2.3. A versão 2.3 contém uma alteração significativa. Para saber mais, consulte Caminhos de atualização com suporte.	15 de novembro de 2022
Suporte ao plug-in Notifications	O Amazon OpenSearch Service agora oferece suporte ao plug-in Notifications, que oferece uma localização central para todas as notificações dos OpenSearch plug-ins. A partir da versão 2.0, os destinos de alerta foram descontinuados e substituídos por canais de notificação.	15 de novembro de 2022

Suporte ao Kibana 7.1.1	Os domínios do Amazon OpenSearch Service que executam o Elasticsearch 7.1 agora oferecem suporte à versão de patch mais recente do Kibana 7.1.1, que adiciona correções de bugs e melhora a segurança. Quando você atualiza seus domínios 7.1 para o software de serviço R20221114, o OpenSearch Service os atualizará automaticamente para esta versão de patch.	15 de novembro de 2022
Suporte ao Kibana 6.8.13	Os domínios do Amazon OpenSearch Service que executam o Elasticsearch 6.8 agora oferecem suporte à versão de patch mais recente do Kibana 6.8.13, que adiciona correções de bugs e melhora a segurança. Quando você atualiza seus domínios 6.8 para o software de serviço R20221114, o OpenSearch Service os atualizará automaticamente para esta versão de patch.	15 de novembro de 2022

Suporte ao Kibana 6.3.2	Os domínios do Amazon OpenSearch Service que executam o Elasticsearch 6.3 agora oferecem suporte à versão de patch mais recente do Kibana 6.3.2, que adiciona correções de bugs e melhora a segurança. Quando você atualiza seus domínios 6.3 para o software de serviço R20221114, o OpenSearch Service os atualizará automaticamente para esta versão de patch.	15 de novembro de 2022

AWS PrivateLink	Com os endpoints OpenSearch VPC gerenciados pelo Amazon Service, você pode se conectar diretamente aos domínios do Service OpenSearch VPC usando uma interface VPC endpoint em vez de se conectar pela Internet. Um OpenSearch VPC endpoint gerenciado por serviços pode ser acessado somente dentro da VPC em que o endpoint é provisionado ou de qualquer VPC VPCs emparelhado com a VPC em que o endpoint é provisionado, conforme permitido pelas tabelas de rotas e grupos de segurança. Seu domínio da VPC deve estar executando o software de serviço R20220928 ou posterior para se conectar a um endpoint da VPC de interface.	7 de novembro de 2022
Correções de erros e melhorias na performance	O software de serviço R20220928 inclui correções de bugs e aprimoramentos de desempenho, incluindo registro SAML aprimorado. A atualização também altera o inquilino padrão para `Global`, em vez de `Private`.	3 de outubro de 2022

Referência de API aprimorada	O Amazon OpenSearch Service oferece uma referência de API de configuração aprimorada e abrangente. As novas referências contêm todas as ações e os tipos de dados disponíveis, exemplos de sintaxe de solicitação e resposta e links para as referências de SDK correspondentes para todas as linguagens compatíveis.	13 de setembro de 2022
Validação azul/verde	O Amazon OpenSearch Service agora executa uma verificação de validação antes das blue/green implantações e detecta erros de validação se seu domínio não estiver qualificado para uma atualização.	16 de agosto de 2022
OpenSearch 1.3 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 1.3. Para saber mais, consulte as notas de lançamento da versão 1.3.	27 de julho de 2022

Suporte ao plug-in ML Commons	O Amazon OpenSearch Service adiciona suporte ao plug-in ML Commons, que fornece um conjunto de algoritmos comuns de aprendizado de máquina por meio de transporte e chamadas de API REST. Você também pode interagir com o plug-in ML Commons por meio de comandos PPL.	27 de julho de 2022
Suporte ao volume gp3	O Amazon OpenSearch Service adiciona suporte para o tipo de volume SSD de uso geral do gp3 EBS. Você pode especificar IOPS provisionadas e throughput adicionais ao criar ou modificar o domínio.	26 de julho de 2022
Documentação aprimorada de práticas recomendadas	A documentação do Amazon OpenSearch Service fornece melhores práticas operacionais aprimoradas e recomendações gerais para criar e operar domínios OpenSearch de serviços.	6 de julho de 2022
Integração com o Service Quotas	Agora você pode visualizar as cotas do Amazon OpenSearch Service e solicitar aumentos de cotas no console Service Quotas.	29 de junho de 2022

Controle de acesso baseado em tags para a API OpenSearch	Agora você pode usar tags para controlar o acesso ao OpenSearch APIs. Anteriorm ente, só era possível usar tags para controlar o acesso à API de configuração.	16 de junho de 2022
Pesquisa entre clusters e entre regiões	A pesquisa entre clusters agora é suportada Regiões da AWS desde que ambos os domínios estejam executand o a versão 7.10 ou posterior do Elasticsearch, ou qualquer versão do. OpenSearch	14 de junho de 2022
Suporte ao Kibana 5.6	O Amazon OpenSearch Service adiciona suporte para um único Kibana 5.6.16. Com o Kibana 5.6.16, é possível usar o Kibana 5.6 como front-end enquanto se conecta ao Elasticsearch versões 5.1, 5.3, 5.5 e 5.6. Para usar o Kibana 5.6, é necessário estar no software de serviço R20220323 ou superior.	4 de abril de 2022

R20220323-P1	A Amazon OpenSearch Service lançou recenteme nte a atualização de software de serviço R20220323, mas a atualização foi posterior mente revertida devido a um problema. Recomenda mos que você atualize seus domínios para o patch release R20220323-P1 ou posterior, o que corrige o problema.	4 de abril de 2022
OpenSearch 1.2 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 1.2. Para saber mais, consulte as notas de lançamento da versão 1.2.	4 de abril de 2022
Observabilidade	A instalação padrão do OpenSearch Dashboards for Amazon OpenSearch Service inclui o plug-in Observabi lity, que você pode usar para visualizar eventos orientado s por dados usando a Piped Processing Language (PPL) para explorar e consultar seus dados. O plug-in requer OpenSearch 1.2 ou posterior e o software de serviço R20220323 ou posterior.	4 de abril de 2022

Suporte ao Kibana 7.7.1	Os domínios do Amazon OpenSearch Service que executam o Elasticsearch 7.7 agora oferecem suporte à versão de patch mais recente do Kibana 7.7, que adiciona correções de bugs e melhora a segurança. Quando você atualiza seus domínios 7.7 para o software de serviço R20220323 ou posterior, o OpenSearch Service os atualizará automaticamente para esta versão de patch.	4 de abril de 2022
Alterações métricas de pressão de memória JVM	O Amazon OpenSearch Service mudou a lógica das `JVMMemoryPressure` CloudWatch métricas para refletir com mais precisão a utilização da memória. Anteriormente, as métricas consideravam apenas o grupo de memória de geração antiga do heap da JVM. Com essa mudança, a métrica também considera o grupo de memória de geração jovem. Depois de atualizar seu domínio para o software de serviço R20220323, você poderá ver um aumento nas métricas `JVMMemoryPressure` , `MasterJVMMemoryPressure` e/ou `WarmJVMMemoryPressure` .	4 de abril de 2022

Dicionários personalizados com o plug-in Análise IK (Chinês)	O Amazon OpenSearch Service agora oferece suporte ao uso de dicionários personalizados com o plug-in de análise IK (chinês).	4 de abril de 2022
Replicação entre clusters em domínios existentes	O Amazon OpenSearch Service removeu a limitação de que você só pode implementar pesquisa e replicação entre clusters em domínios criados em ou após 3 de junho de 2020. Agora você pode habilitar esses recursos em todos os domínios, independentemente de quando eles foram criados. Ambos os domínios devem estar no software de serviço R20220323 ou posterior.	4 de abril de 2022
Visibilidade da implantação azul/verde	O Amazon OpenSearch Service agora oferece mais visibilidade sobre o progresso das implantações azul/verde. Você pode monitorar esses detalhes no console ou por meio da API de configuração.	27 de janeiro de 2022

Controle de acesso refinado em domínios existentes	Não é possível habilitar o controle de acesso refinado em domínios existentes. Você pode habilitar um período de migração temporária para políticas de acesso aberto/baseado em IP para garantir que os usuários possam continuar acessando o seu domínio enquanto você cria e mapeia funções. Para habilitar o controle de acesso refinado em domínios existentes, é necessário ter o software de serviço R20211203 ou superior.	6 de janeiro de 2022
Funções de OpenSearch painéis renomeadas	Com o software de serviço R20211203, a função `kibana_user` foi renomeada para `opensearch_dashboards_user`, e `kibana_read_only` foi renomeada para `opensearch_dashboards_read_only`. Essa alteração se aplica a todos os 1 recém-criados OpenSearch . domínios x. Para OpenSearch domínios existentes que você atualiza para o software de serviço R20211203, as funções permanecem as mesmas.	4 de janeiro de 2022

OpenSearch 1.1 suporte	O Amazon OpenSearch Service agora oferece suporte à OpenSearch versão 1.1. Para saber mais, consulte as notas de lançamento da versão 1.1.	4 de janeiro de 2022
Editor visual do ISM	A instalação padrão do OpenSearch Dashboards for Amazon OpenSearch Service agora oferece suporte ao editor visual das políticas do ISM. Esse recurso requer a OpenSearch versão 1.1 ou posterior.	4 de janeiro de 2022
Atualização da prevenção contra o problema confused deputy entre serviços	O Amazon OpenSearch Service oferece suporte ao uso das chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e das chaves de contexto nas políticas de recursos do IAM para evitar o problema confuso do substituto. Para usar essas chaves de condição, é necessário estar no software de serviço R20211203 ou superior.	4 de janeiro de 2022

Patch do Log4j	O software de serviço R20211203-P2 atualiza a versão do Log4j usada no Service conforme recomendado pelos avisos em CVE-2021-44228 e OpenSearch CVE-2021-45046. O patch se aplica aos domínios que executam todas as versões do Elasticsearch OpenSearch e do Elasticsearch. OpenSearch O serviço continuará atualizando várias versões do Log4j internamente e elas não estarão necessariamente restritas à versão mais recente do Log4j. A versão do Log4j em seu domínio depende da versão do software que o domínio está executando. No entanto, independentemente da versão do Log4j, desde que você esteja executando a R20211203-P2 ou posterior, seus domínios contêm a atualização do Log4j necessária para tratar o CVE-2021-44228 e o CVE-2021-45046.	15 de dezembro de 2021

Replicação entre clusters	A replicação entre clusters permite replicar índices, mapeamentos e metadados de um domínio de serviço para outro. OpenSearch A replicação entre clusters requer um domínio executando o Elasticsearch 7.10 ou 1.1 ou posterior. OpenSearch	5 de outubro de 2021
Novas AWS políticas gerenciadas	O lançamento do Amazon OpenSearch Service inclui novas políticas AWS gerenciadas e a descontinuação de políticas antigas.	8 de setembro de 2021
Suporte ao Kibana 6.4.3	Os domínios do Amazon OpenSearch Service que executam a versão 6.4 antiga do Elasticsearch agora oferecem suporte à versão de patch mais recente do Kibana 6.4, que adiciona correções de bugs e melhora a segurança. OpenSearch O serviço atualizará automaticamente os domínios para esta versão de patch.	8 de setembro de 2021

Streams de dados	O Amazon OpenSearch Service adiciona suporte para fluxos de dados, o que simplifica o processo de gerenciamento de dados de séries temporais. Seu domínio deve estar executando a OpenSearch versão 1.0 ou posterior para usar fluxos de dados.	8 de setembro de 2021
OpenSearch Serviço Amazon	AWS renomeia o Amazon OpenSearch Service para remover a marca “Elastics earch” antiga. O Amazon OpenSearch Service oferece suporte OpenSearch e lega o Elasticsearch OSS. Ao criar um cluster, você tem a opção de qual mecanismo de pesquisa usar. OpenSearc h O serviço oferece ampla compatibilidade com o Elasticsearch OSS 7.10, a versão final de código aberto do software.	8 de setembro de 2021

Armazenamento de baixa atividade	O armazenamento inativo é um novo nível de armazenamento para dados históricos ou acessados com pouca frequência. Os índices de baixa atividade ocupam apenas o armazenamento S3 e não têm computação anexada a eles. O armazenamento de baixa atividade requer um domínio executando o Elasticsearch 7.9 ou posterior e o software de serviço R20210426 ou posterior.	13 de maio de 2021
Instâncias do Graviton baseadas em ARM	O Amazon OpenSearch Service agora oferece suporte aos tipos de instância Graviton baseados em ARM (M6G, C6G, R6G e R6GD). Os tipos de instância do Graviton estão disponíveis em domínios novos e existentes executando o Elasticsearch 7.9 ou posterior e o software de serviço R20210331 ou posterior.	4 de maio de 2021

Modelos do ISM	O Amazon OpenSearch Service adiciona suporte aos modelos do ISM, que permitem anexar automatic amente uma política do ISM a um índice se o índice corresponder a um padrão definido na política. Os modelos do ISM exigem o software de serviço R20210426 ou posterior. Esta atualização também defasa a configuração `policy_id` , o que significa que você não pode mais usar modelos de índice para aplicar políticas do ISM a índices recém-cri ados. A atualização introduz uma alteração significativa nos CloudFormation modelos existentes usando essa configuração.	27 de abril de 2021
Suporte ao Elasticsearch 7.10	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 7.10. Para saber mais, consulte as notas de lançamento da versão 7.10.	21 de abril de 2021

Pesquisa assíncrona	O Amazon OpenSearch Service agora oferece suporte à pesquisa assíncrona, que permite executar solicitações de pesquisa em segundo plano. A pesquisa assíncrona requer um domínio executando o Elasticsearch 7.10 ou superior e o software de serviço R20210331 ou superior.	21 de abril de 2021
Controle de acesso baseado em tags para a API de configuração	Agora você pode usar AWS tags para controlar o acesso à API de configuração do Amazon ES.	2 de março de 2021
Auto-Tune	O Amazon OpenSearch Service adiciona o Auto-Tune, que usa métricas de desempenho e uso do seu cluster para sugerir alterações nas configurações da JVM em seus nós. O Auto-Tune requer um domínio executando o Elasticsearch 6.7 ou posterior e o software de serviço R20201117 ou posterior.	24 de fevereiro de 2021

Trace Analytics	A instalação padrão do Kibana para Amazon OpenSearch Service agora inclui o plug-in de análise de rastreamento, que permite monitorar dados de rastreamento de seus aplicativos distribuídos. O plug-in requer um domínio executando o Elasticsearch 7.9 ou posterior e o software de serviço R20210201 ou posterior.	17 de fevereiro de 2021
Métricas de fragmentos	O Amazon OpenSearch Service adiciona as seguintes CloudWatch métricas para rastrear o status do fragmento:`Shards.active` ,`Shards.unassigned` ,, `Shards.delayedUnassigned` `Shards.activePrimary` `Shards.initializing` ,`Shards.relocating` . As métricas estão disponíveis em domínios que executam o software de serviço R20210201 ou superior.	17 de fevereiro de 2021

Relatórios do Kibana	A instalação padrão do Kibana para Amazon OpenSearch Service agora oferece suporte a relatórios sob demanda para as páginas Discover, Visualize e Dashboard. Esse recurso requer o Elasticsearch 7.9 ou posterior e o software de serviço R20210201 ou posterior.	17 de fevereiro de 2021
Suporte ao Kibana 5.6.16	Os domínios do Amazon OpenSearch Service que executam o Elasticsearch 5.6 agora oferecem suporte à versão de patch mais recente do Kibana 5.6, que adiciona correções de bugs e melhora a segurança. O Amazon ES atualizará automaticamente os domínios para esta versão de patch.	17 de fevereiro de 2021
Criptografia para domínios existentes	O Amazon OpenSearch Service agora oferece suporte para habilitar a criptografia de dados em repouso e a node-to-node criptografia em domínios existentes que executam o Elasticsearch 6.7 ou posterior. Após habilitar essas configurações, você não poderá desabilitá-las.	27 de janeiro de 2021

Reindexação remota	O Amazon OpenSearch Service agora oferece suporte à reindexação remota, o que permite migrar índices de domínios remotos. Esse recurso exige o software de serviço R20201117 ou posterior.	24 de novembro de 2020
Piped Processing Language	O Amazon OpenSearch Service agora oferece suporte à Piped Processing Language (PPL), uma linguagem de consulta que permite usar a sintaxe pipe (|) para consultar dados armazenados no Elasticsearch. Esse recurso exige o software de serviço R20201117 ou posterior. Para saber mais, consulte .	24 de novembro de 2020
Cadernos do Kibana	O Amazon OpenSearch Service adiciona suporte aos notebooks Kibana, o que permite combinar visualizações ao vivo e texto narrativo em uma única interface. Esse recurso exige o software de serviço R20201117 ou posterior.	24 de novembro de 2020

Gráficos de Gantt	A instalação padrão do Kibana para Amazon OpenSearch Service agora oferece suporte a um novo tipo de visualização, gráficos de Gantt. Esse recurso exige o software de serviço R20201117 ou posterior.	24 de novembro de 2020
Suporte ao Elasticsearch 7.9	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 7.9. Para saber mais, consulte as notas de lançamento da versão 7.9.	24 de novembro de 2020
Atualizações na detecção de anomalias	A detecção de anomalias para o Amazon OpenSearch Service adiciona suporte à alta cardinalidade, o que permite categorizar anomalias com uma dimensão como endereço IP, ID do produto, código do país e assim por diante. Esse recurso exige o software de serviço R20201117 ou posterior.	24 de novembro de 2020

Atualizações do dicionário dinâmico	O Amazon OpenSearch Service agora permite que você atualize seus analisadores de pesquisa sem reindexar. Você pode atualizar os arquivos de dicionário em alguns ou todos os seus domínios, e o Amazon ES rastreia as versões do pacote ao longo do tempo para que você tenha um histórico do que mudou e quando. Esse recurso exige o software de serviço R20201019 ou posterior.	17 de novembro de 2020
Endpoints personalizados	O Amazon OpenSearch Service agora oferece suporte a endpoints personalizados, que permitem que você forneça um novo URL ao seu domínio do Amazon ES. Se você trocar de domínios, poderá manter o mesmo URL. Esse recurso exige o software de serviço R20201019 ou posterior.	5 de novembro de 2020

Novos plug-ins de idiomas	O Amazon OpenSearch Service agora oferece suporte aos plug-ins IK (chinês) Analysis, Vietnamese Analysis e Thai Analysis em domínios que executam o Elasticsearch 7.7 ou posterior com o software de serviço R20201019 ou posterior.	28 de outubro de 2020
Suporte ao Elasticsearch 7.8	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 7.8. Para saber mais, consulte as notas de lançamento da versão 7.8.	28 de outubro de 2020
Autenticação SAML para Kibana	O Amazon OpenSearch Service agora oferece suporte à autenticação SAML para o Kibana, que permite que você use provedores de identidade terceirizados para fazer login no Kibana, gerenciar o controle de acesso refinado, pesquisar seus dados e criar visualizações. Esse recurso exige o software de serviço R20201019 ou posterior.	27 de outubro de 2020
Instâncias T3	O Amazon OpenSearch Service agora oferece suporte `t3.small` aos tipos de `t3.medium` instância e.	23 de setembro de 2020

Logs de auditoria	O Amazon OpenSearch Service agora oferece suporte a registros de auditoria para seus dados, o que permite rastrear tentativas de login malsucedidas, acesso de usuários a índices, documento s e campos e muito mais. Esse recurso exige o software de serviço R20200910 ou posterior.	16 de setembro de 2020
UltraWarm atualizações	UltraWarm for Amazon OpenSearch Service adiciona novas métricas, novas configurações, uma fila de migração maior e uma API de cancelamento. Essas atualizaç ões exigem o software de serviço R20200910 ou posterior. Para saber mais, consulte .	14 de setembro de 2020
Learning to Rank	O Amazon OpenSearch Service agora oferece suporte ao plug-in de código aberto Learning to Rank, que permite usar tecnologias de aprendiza do de máquina para melhorar a relevância da pesquisa. Esse recurso exige o software de serviço R20200721 ou posterior.	27 de julho de 2020

Similaridade de cosseno k-NN	O algoritmo k-vizinhos mais próximos (k-NN) agora permite procurar “vizinhos mais próximos” por similaridade de cossenos, além da distância euclidiana. Esse recurso exige o software de serviço R20200721 ou posterior.	23 de julho de 2020
Compactação gzip	O Amazon OpenSearch Service agora oferece suporte à compressão gzip para a maioria das solicitações e respostas HTTP, o que pode reduzir a latência e conservar a largura de banda. Esse recurso exige o software de serviço R20200721 ou posterior.	23 de julho de 2020
Suporte ao Elasticsearch 7.7	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 7.7. Para saber mais, consulte as notas de lançamento da versão 7.7.	23 de julho de 2020
Serviço de mapas do Kibana	A instalação padrão do Kibana para Amazon OpenSearch Service agora inclui um servidor de mapas WMS, exceto para domínios nas regiões da Índia e da China.	18 de junho de 2020

Melhorias em SQL	O suporte SQL para o Amazon OpenSearch Service agora oferece suporte a muitas novas operações, uma interface de usuário Kibana dedicada para exploração de dados e uma CLI interativa.	3 de junho de 2020
Pesquisa entre clusters	O Amazon OpenSearch Service permite que você realize consultas e agregações entre clusters em vários domínios conectados.	3 de junho de 2020
Detecção de anomalias	O Amazon OpenSearch Service permite que você detecte automaticamente anomalias quase em tempo real.	3 de junho de 2020
UltraWarm	UltraWarm o armazenamento para o Amazon OpenSearch Service saiu da versão prévia pública e agora está disponível ao público em geral. O recurso agora oferece suporte a uma variedade maior de versões Regiões da AWS e. Para obter mais informações, consulte .	5 de maio de 2020

Dicionários personalizados	O Amazon OpenSearch Service permite que você faça upload de arquivos de dicionário personalizados para uso com seu cluster. Esses arquivos melhoram seus resultados de pesquisa instruindo o Elasticsearch a ignorar certas palavras de alta frequência ou para tratar termos como equivalentes.	21 de abril de 2020
Suporte ao Elasticsearch 7.4	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 7.4. Para saber mais, consulte Versões compatíveis.	12 de março de 2020
k-NN	O Amazon OpenSearch Service adiciona suporte à pesquisa K-Nearest Neighbor (k-NN). O k-NN requer o software de serviço R20200302 ou posterior.	3 de março de 2020
Gerenciamento de estados de índice	O Amazon OpenSearch Service adiciona o Index State Management (ISM), que permite automatizar tarefas rotineiras, como excluir índices quando eles atingem uma certa idade. Esse recurso exige o software de serviço R20200302 ou superior.	3 de março de 2020

Suporte ao Elasticsearch 5.6.16	O Amazon OpenSearch Service agora oferece suporte à versão de patch mais recente para a versão 5.6, que adiciona correções de bugs e melhora a segurança . O Amazon ES atualizará automaticamente os domínios 5.6 para esta versão. Observe que essa versão do Elasticsearch informa incorretamente sua versão como 5.6.17.	2 de março de 2020
Controle de acesso refinado	O Amazon OpenSearch Service agora oferece suporte ao controle de acesso refinado, que oferece segurança em nível de índice, documento e campo, multilocação do Kibana e autenticação básica HTTP opcional para seu cluster.	11 de fevereiro de 2020

UltraWarm armazenamento (pré-visualização)	O Amazon OpenSearch Service adiciona UltraWarm um novo nível de armazenam ento aquecido que usa o Amazon S3 e uma solução sofisticada de cache para melhorar o desempenho. Para índices nos quais você não está gravando ativamente e consultando com menos frequência, o UltraWarm armazenamento oferece custos significativamente mais baixos por GiB.	3 de dezembro de 2019
Recursos de criptografia para regiões da China	A criptografia de dados em repouso e a node-to-node criptografia agora estão disponíveis na região da `cn-north-1` China (Pequim) e na região `cn-northwest-1` da China (Ningxia).	20 de novembro de 2019
Exigir HTTPS	Agora você pode exigir que todo o tráfego para os domínios do Amazon ES seja recebido via HTTPS. Ao configurar seu domínio, marque a caixa Exigir HTTPS. Esse recurso exige o software de serviço R20190808 ou superior.	3 de outubro de 2019

Suporte ao Elasticsearch 7.1 e 6.8	O Amazon OpenSearch Service agora oferece suporte às versões 7.1 e 6.8 do Elasticsearch. Para saber mais, consulte Versões compatíveis.	13 de agosto de 2019
Snapshots a cada hora	Em vez de instantâneos diários, o Amazon OpenSearch Service agora tira instantâneos de hora em hora de domínios que executam o Elasticsearch 5.3 e versões posteriores, para que você tenha backups mais frequentes para restaurar seus dados.	8 de julho de 2019
Suporte ao Elasticsearch 6.7	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 6.7. Para saber mais, consulte Versões compatíveis.	29 de maio de 2019
Suporte a SQL	O Amazon OpenSearch Service agora permite que você consulte seus dados usando SQL. O suporte a SQL exige o software de serviço R20190418 ou posterior.	15 de maio de 2019

Tipos de instâncias da série 5	O Amazon OpenSearch Service agora oferece suporte aos tipos de instância M5, C5 e R5. Em comparaçã o aos tipos de instância da geração anterior, esses novos tipos oferecem uma melhor performance a preços mais baixos. Para saber mais, consulte Limites.	24 de abril de 2019
Suporte ao Elasticsearch 6.5	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 6.5.	8 de abril de 2019
Geração de alertas	O alerta para o Amazon OpenSearch Service notifica você quando os dados de um ou mais índices do Amazon ES atendem a determina das condições. Os alertas exigem o software de serviço R20190221 ou posterior.	25 de março de 2019
Suporte a três zonas de disponibilidade	O Amazon OpenSearch Service agora oferece suporte a três zonas de disponibi lidade em várias regiões. Essa versão também inclui uma experiência de console simplificada. Esse recurso multi-AZ exige o software de serviço R20181023 ou posterior.	7 de fevereiro de 2019

Suporte ao Elasticsearch 6.4	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 6.4.	23 de janeiro de 2019
Clusters de 200 nós	O Amazon ES agora permite que você crie clusters com até 200 nós de dados para um total de 3 PB de armazenam ento.	22 de janeiro de 2019
Atualizações de software de serviço	O Amazon ES agora permite que você atualize manualmen te o software de serviço para seu domínio para se beneficia r de novos recursos mais rapidamente ou atualizar em um horário com baixo volume de tráfego. Para saber mais, consulte .	20 de novembro de 2018
Novas CloudWatch métricas	O Amazon ES agora oferece métricas em nível de nó e novas guias Integridade do cluster e Integridade da instância no console do Amazon ES.	20 de novembro de 2018
Suporte à China (Pequim)	O Amazon OpenSearch Service agora está disponíve l na região cn-north-1, onde oferece suporte aos tipos de instância M4, C4 e R4.	17 de outubro de 2018

Node-to-node criptografia	O Amazon OpenSearch Service agora oferece suporte à node-to-node criptografia, que mantém seus dados criptografados à medida que o Amazon ES os distribui por todo o cluster.	18 de setembro de 2018
Atualizações de versão em vigor	O Amazon OpenSearch Service agora oferece suporte a atualizações de versão no local.	14 de agosto de 2018
Suporte ao Elasticsearch 6.3 e 5.6	O Amazon OpenSearch Service agora oferece suporte às versões 6.3 e 5.6 do Elasticsearch.	14 de agosto de 2018
Logs de erro	O Amazon ES agora permite que você publique registros de erros do Elasticsearch na Amazon. CloudWatch	31 de julho de 2018
Instâncias reservadas para China (Ningxia)	O Amazon ES agora oferece Instâncias reservadas para a região da China (Ningxia).	29 de maio de 2018
Instâncias reservadas	O Amazon ES agora oferece suporte a instâncias reservadas.	7 de maio de 2018

Atualizações anteriores

A tabela a seguir descreve alterações importantes no Amazon ES antes de maio de 2018.

Alteração	Descrição	Data
Autenticação do Amazon Cognito para Kibana	O Amazon ES agora oferece proteção da página de login para o Kibana. Para saber mais, consulte the section called “Autenticação do Amazon Cognito para painéis OpenSearch ”.	2 de abril de 2018
Suporte ao Elasticsearch 6.2	O Amazon OpenSearch Service agora oferece suporte ao Elasticsearch versão 6.2.	14 de março de 2018
Plug-in de análise coreana	O Amazon ES agora oferece suporte a uma versão otimizada para memória do plug-in de análise coreana Seunjeon.	13 de março de 2018
Atualizações instantâneas de controle de acesso	As alterações nas políticas de controle de acesso em domínios do Amazon ES agora entram em vigor instantaneamente.	7 de março de 2018
Escala de petabytes	O Amazon ES agora oferece suporte a tipos de instância I3 e armazenamento de domínio total de até 1,5 PB. Para saber mais, consulte the section called “Escala de petabytes”.	19 de dezembro de 2017
Criptografia de dados em repouso	O Amazon ES agora oferece suporte à criptografia de dados em repouso. Para saber mais, consulte the section called “Criptografia em repouso”.	7 de dezembro de 2017
Suporte ao Elasticsearch 6.0	O Amazon ES agora oferece suporte ao Elasticsearch versão 6.0 Para considerações e instruções de migração, consulte the section called “Atualização de domínios”.	6 de dezembro de 2017
Suporte à VPC	O Amazon ES agora permite iniciar domínios em uma Amazon Virtual Private Cloud. O suporte a VPC fornece uma camada adicional de segurança e simplifica a comunicação entre o Amazon ES e outros serviços dentro de uma VPC. Para saber mais, consulte the section called “Suporte à VPC”.	17 de outubro de 2017

Alteração	Descrição	Data
Publicação de logs lentos	O Amazon ES agora oferece suporte à publicação de registros lentos no CloudWatch Logs. Para saber mais, consulte the section called “Monitoramento de logs”.	16 de outubro de 2017
Suporte ao Elasticsearch 5.5	O Amazon ES agora oferece suporte ao Elasticsearch versão 5.5. Agora você pode restaurar snapshots automatizados sem precisar entrar em contato com o Suporte e armazenar scripts por meio da API `_scripts`.	7 de setembro de 2017
Suporte ao Elasticsearch 5.3	O Amazon ES adicionou suporte ao Elasticsearch versão 5.3.	1 de junho de 2017
Mais instâncias e capacidade de EBS por cluster	O Amazon ES agora oferece suporte a até 100 nós e 150 TB de capacidade de EBS por cluster.	5 de abril de 2017
Suporte no Canadá (Central) e na UE (Londres)	O Amazon ES adicionou suporte às seguintes regiões: Canadá (Central), ca-central-1 e UE (Londres), eu-west-2.	20 de março de 2017
Mais instâncias e maiores volumes de EBS	O Amazon ES adicionou suporte a mais instâncias e a volumes do EBS maiores.	21 de fevereiro de 2017
Suporte ao Elasticsearch 5.1	O Amazon ES adicionou suporte ao Elasticsearch versão 5.1.	30 de janeiro de 2017
Compatibilidade com o plug-in Phonetic Analysis	O Amazon ES agora oferece integração incorporada com o plug-in Phonetic Analysis, o qual permite a você realizar consultas “sonoras” em seus dados.	22 de dezembro de 2016
Suporte no Leste dos EUA (Ohio)	O Amazon ES adicionou suporte à seguinte região: Leste dos EUA (Ohio), us-east-2.	17 de outubro de 2016

Alteração	Descrição	Data
Nova métrica de performance	O Amazon ES adicionou uma métrica de performance, `ClusterUsedSpace` .	29 de julho de 2016
Suporte ao Elasticsearch 2.3	O Amazon ES adicionou suporte ao Elasticsearch versão 2.3.	27 de julho de 2016
Suporte na Ásia-Pacífico (Mumbai)	O Amazon ES adicionou suporte à seguinte região: Ásia-Pacífico (Mumbai), ap-south-1.	27 de junho de 2016
Mais instâncias por cluster	O Amazon ES aumentou o número máximo de instâncias (contagem de instâncias) por cluster de 10 para 20.	18 de maio de 2016
Suporte na Ásia-Pacífico (Seul)	O Amazon ES adicionou suporte à seguinte região: Ásia-Pacífico (Seul), ap-northeast-2.	28 de janeiro de 2016
Amazon ES	Versão inicial.	1 de outubro de 2015

AWS Glossário

Para obter a AWS terminologia mais recente, consulte o AWS glossário na Glossário da AWS Referência.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.