Manual do usuário

Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus: Manual do usuário

Table of Contents

O que é o Amazon Managed Service for Prometheus?

O Amazon Managed Service for Prometheus é um serviço de monitoramento sem servidor Prometheus-compatible para métricas de contêineres que facilita o monitoramento seguro de ambientes de contêineres em grande escala. Com o Amazon Managed Service for Prometheus, você pode usar o mesmo modelo de dados e linguagem de consulta de código aberto do Prometheus que você usa atualmente para monitorar o desempenho de suas workloads em contêineres e também desfrutar de maior escalabilidade, disponibilidade e segurança sem precisar gerenciar a infraestrutura subjacente.

O Amazon Managed Service for Prometheus escala automaticamente a ingestão, o armazenamento e a consulta de métricas operacionais à medida que as workloads aumentam e diminuem. Ele se integra aos serviços AWS de segurança para permitir acesso rápido e seguro aos dados.

O Amazon Managed Service for Prometheus foi projetado para ser altamente disponível usando várias implantações de zona Multi-AZ de disponibilidade (). Os dados ingeridos em um espaço de trabalho são replicados em três zonas de disponibilidade na mesma região.

O Amazon Managed Service for Prometheus funciona com clusters de contêineres que são executados no Amazon Elastic Kubernetes Service e em ambientes Kubernetes autogerenciados.

Com o Amazon Managed Service for Prometheus, você usa o mesmo modelo de dados de código aberto do Prometheus e a mesma linguagem de consulta PromQL que você usa com o Prometheus. As equipes de engenharia podem usar o PromQL para filtrar, agregar e alertar sobre métricas e obter visibilidade de desempenho rapidamente sem nenhuma alteração no código. O Amazon Managed Service for Prometheus fornece recursos flexíveis de consulta sem o custo operacional e a complexidade.

As métricas ingeridas em um espaço de trabalho são armazenadas durante 150 dias por padrão e, em seguida, excluídas automaticamente. Você pode ajustar o período de retenção configurando seu espaço de trabalho em um máximo de até 1.095 dias (3 anos). Para obter mais informações, consulte Configurar seu espaço de trabalho.

Regiões aceitas

O Amazon Managed Service for Prometheus atualmente é compatível com as seguintes regiões:

Nome da região	Região	Endpoint	Protocolo
Leste dos EUA (Ohio)	us-east-2	aps.us-east-2.amazonaws.com aps-workspaces.us-east-2.amazonaws.com aps-workspaces-fips.us-east-2.amazonaws.com aps-workspaces-fips.us-east-2.api.aws aps-workspaces.us-east-2.api.aws aps-fips.us-east-2.amazonaws.com aps.us-east-2.api.aws aps-fips.us-east-2.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS
Leste dos EUA (Norte da Virgínia)	us-east-1	aps.us-east-1.amazonaws.com aps-workspaces.us-east-1.amazonaws.com aps-workspaces-fips.us-east-1.amazonaws.com aps-workspaces-fips.us-east-1.api.aws aps-workspaces.us-east-1.api.aws aps-fips.us-east-1.amazonaws.com aps.us-east-1.api.aws aps-fips.us-east-1.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS
Oeste dos EUA (N. da	us-west-1	aps.us-west-1.amazonaws.com aps-workspaces.us-west-1.amazonaws.com	HTTPS HTTPS HTTPS

Nome da região	Região	Endpoint	Protocolo	
Califórnia)		aps-workspaces-fips.us-west-1.amazonaws.com aps-workspaces-fips.us-west-1.api.aws aps-workspaces.us-west-1.api.aws aps-fips.us-west-1.amazonaws.com aps.us-west-1.api.aws aps-fips.us-west-1.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS	
Oeste dos EUA (Oregon)	us-west-2	aps.us-west-2.amazonaws.com aps-workspaces.us-west-2.amazonaws.com aps-workspaces-fips.us-west-2.amazonaws.com aps-workspaces-fips.us-west-2.api.aws aps-workspaces.us-west-2.api.aws aps-fips.us-west-2.amazonaws.com aps.us-west-2.api.aws aps-fips.us-west-2.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS	
África (Cidade do Cabo)	af-south-1	aps.af-south-1.amazonaws.com aps-workspaces.af-south-1.amazonaws.com aps-workspaces.af-south-1.api.aws aps.af-south-1.api.aws	HTTPS HTTPS HTTPS HTTPS	

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Hong Kong)	ap-east-1	aps.ap-east-1.amazonaws.com	HTTPS
		aps-workspaces.ap-east-1.amazonaws.com	HTTPS
		aps-workspaces.ap-east-1.api.aws	HTTPS
		aps.ap-east-1.api.aws	HTTPS
Ásia-Pacífico (Hyderabad)	ap-south-2	aps.ap-south-2.amazonaws.com	HTTPS
		aps-workspaces.ap-south-2.amazonaws.com	HTTPS
		aps-workspaces.ap-south-2.api.aws	HTTPS
		aps.ap-south-2.api.aws	HTTPS
Ásia-Pacífico (Jacarta)	ap-southeast-3	aps.ap-southeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-3.api.aws	HTTPS
		aps.ap-southeast-3.api.aws	HTTPS
Ásia-Pacífico (Malásia)	ap-southeast-5	aps.ap-southeast-5.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-5.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-5.api.aws	HTTPS
		aps.ap-southeast-5.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Melbourne)	ap-southeast-4	aps.ap-southeast-4.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-4.amazonaws.com	HTTPS
		aps-workspaces.ap-southeast-4.api.aws	HTTPS
		aps.ap-southeast-4.api.aws	HTTPS
Ásia-Pacífico (Mumbai)	ap-south-1	aps.ap-south-1.amazonaws.com	HTTPS
		aps-workspaces.ap-south-1.amazonaws.com	HTTPS
		aps-workspaces.ap-south-1.api.aws	HTTPS
		aps.ap-south-1.api.aws	HTTPS
Ásia-Pacífico (Osaka)	ap-northeast-3	aps.ap-northeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-3.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-3.api.aws	HTTPS
		aps.ap-northeast-3.api.aws	HTTPS
Ásia-Pacífico (Seul)	ap-northeast-2	aps.ap-northeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-2.amazonaws.com	HTTPS
		aps-workspaces.ap-northeast-2.api.aws	HTTPS
		aps.ap-northeast-2.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Singapura)	ap-southeast-1	aps.ap-southeast-1.amazonaws.com aps-workspaces.ap-southeast-1.amazonaws.com aps-workspaces.ap-southeast-1.api.aws aps.ap-southeast-1.api.aws	HTTPS HTTPS HTTPS HTTPS
Ásia-Pacífico (Sydney)	ap-southeast-2	aps.ap-southeast-2.amazonaws.com aps-workspaces.ap-southeast-2.amazonaws.com aps-workspaces.ap-southeast-2.api.aws aps.ap-southeast-2.api.aws	HTTPS HTTPS HTTPS HTTPS
Ásia-Pacífico (Taipei)	ap-east-2	aps.ap-east-2.amazonaws.com aps-workspaces.ap-east-2.amazonaws.com aps-workspaces.ap-east-2.api.aws aps.ap-east-2.api.aws	HTTPS HTTPS HTTPS HTTPS
Ásia-Pacífico (Tailândia)	ap-southeast-7	aps.ap-southeast-7.amazonaws.com aps-workspaces.ap-southeast-7.amazonaws.com aps-workspaces.ap-southeast-7.api.aws aps.ap-southeast-7.api.aws	HTTPS HTTPS HTTPS HTTPS

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Tóquio)	ap-northeast-1	aps.ap-northeast-1.amazonaws.com aps-workspaces.ap-northeast-1.amazonaws.com aps-workspaces.ap-northeast-1.api.aws aps.ap-northeast-1.api.aws	HTTPS HTTPS HTTPS HTTPS
Canadá (Central)	ca-central-1	aps.ca-central-1.amazonaws.com aps-workspaces.ca-central-1.amazonaws.com aps-workspaces-fips.ca-central-1.amazonaws.com aps-workspaces-fips.ca-central-1.api.aws aps-workspaces.ca-central-1.api.aws aps-fips.ca-central-1.amazonaws.com aps.ca-central-1.api.aws aps-fips.ca-central-1.api.aws	HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS HTTPS

Nome da região	Região	Endpoint	Protocolo
Oeste do Canadá (Calgary)	ca-west-1	aps.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.ca-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.ca-west-1.api.aws	HTTPS
		aps-workspaces.ca-west-1.api.aws	HTTPS
		aps-fips.ca-west-1.amazonaws.com	HTTPS
		aps.ca-west-1.api.aws	HTTPS
		aps-fips.ca-west-1.api.aws	HTTPS
Europa (Frankfurt)	eu-central-1	aps.eu-central-1.amazonaws.com	HTTPS
		aps-workspaces.eu-central-1.amazonaws.com	HTTPS
		aps-workspaces.eu-central-1.api.aws	HTTPS
		aps.eu-central-1.api.aws	HTTPS
Europa (Irlanda)	eu-west-1	aps.eu-west-1.amazonaws.com	HTTPS
		aps-workspaces.eu-west-1.amazonaws.com	HTTPS
		aps-workspaces.eu-west-1.api.aws	HTTPS
		aps.eu-west-1.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
Europa (Londres)	eu-west-2	aps.eu-west-2.amazonaws.com	HTTPS
		aps-workspaces.eu-west-2.amazonaws.com	HTTPS
		aps-workspaces.eu-west-2.api.aws	HTTPS
		aps.eu-west-2.api.aws	HTTPS
Europa (Milão)	eu-south-1	aps.eu-south-1.amazonaws.com	HTTPS
		aps-workspaces.eu-south-1.amazonaws.com	HTTPS
		aps-workspaces.eu-south-1.api.aws	HTTPS
		aps.eu-south-1.api.aws	HTTPS
Europa (Paris)	eu-west-3	aps.eu-west-3.amazonaws.com	HTTPS
		aps-workspaces.eu-west-3.amazonaws.com	HTTPS
		aps-workspaces.eu-west-3.api.aws	HTTPS
		aps.eu-west-3.api.aws	HTTPS
Europa (Espanha)	eu-south-2	aps.eu-south-2.amazonaws.com	HTTPS
		aps-workspaces.eu-south-2.amazonaws.com	HTTPS
		aps-workspaces.eu-south-2.api.aws	HTTPS
		aps.eu-south-2.api.aws	HTTPS
Europa (Estocolmo)	eu-north-1	aps.eu-north-1.amazonaws.com	HTTPS
		aps-workspaces.eu-north-1.amazonaws.com	HTTPS
		aps-workspaces.eu-north-1.api.aws	HTTPS
		aps.eu-north-1.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
Europa (Zurique)	eu-central-2	aps.eu-central-2.amazonaws.com	HTTPS
		aps-workspaces.eu-central-2.amazonaws.com	HTTPS
		aps-workspaces.eu-central-2.api.aws	HTTPS
		aps.eu-central-2.api.aws	HTTPS
Israel (Tel Aviv)	il-central-1	aps.il-central-1.amazonaws.com	HTTPS
		aps-workspaces.il-central-1.amazonaws.com	HTTPS
		aps-workspaces.il-central-1.api.aws	HTTPS
		aps.il-central-1.api.aws	HTTPS
México (Central)	mx-central-1	aps.mx-central-1.amazonaws.com	HTTPS
		aps-workspaces.mx-central-1.amazonaws.com	HTTPS
		aps-workspaces.mx-central-1.api.aws	HTTPS
		aps.mx-central-1.api.aws	HTTPS
Oriente Médio (Barém)	me-south-1	aps.me-south-1.amazonaws.com	HTTPS
		aps-workspaces.me-south-1.amazonaws.com	HTTPS
		aps-workspaces.me-south-1.api.aws	HTTPS
		aps.me-south-1.api.aws	HTTPS
Oriente Médio (Emirados Árabes Unidos)	me-central-1	aps.me-central-1.amazonaws.com	HTTPS
		aps-workspaces.me-central-1.amazonaws.com	HTTPS
		aps-workspaces.me-central-1.api.aws	HTTPS
		aps.me-central-1.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
América do Sul (São Paulo)	sa-east-1	aps.sa-east-1.amazonaws.com	HTTPS
		aps-workspaces.sa-east-1.amazonaws.com	HTTPS
		aps-workspaces.sa-east-1.api.aws	HTTPS
		aps.sa-east-1.api.aws	HTTPS
AWS GovCloud (US-East)	us-gov-east-1	aps.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-east-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-east-1.api.aws	HTTPS
		aps-workspaces.us-gov-east-1.api.aws	HTTPS
		aps-fips.us-gov-east-1.amazonaws.com	HTTPS
		aps.us-gov-east-1.api.aws	HTTPS
		aps-fips.us-gov-east-1.api.aws	

Nome da região	Região	Endpoint	Protocolo
AWS GovCloud (US-West)	us-gov-west-1	aps.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-west-1.amazonaws.com	HTTPS
		aps-workspaces-fips.us-gov-west-1.api.aws	HTTPS
		aps-workspaces.us-gov-west-1.api.aws	HTTPS
		aps-fips.us-gov-west-1.amazonaws.com	HTTPS
		aps.us-gov-west-1.api.aws	HTTPS
		aps-fips.us-gov-west-1.api.aws	

O Amazon Managed Service for Prometheus inclui endpoints do plano de controle (para realizar tarefas de gerenciamento do espaço de trabalho) e endpoints do plano de dados (para trabalhar Prometheus-compatible com dados em uma instância do espaço de trabalho). Os endpoints do plano de ambiente de gerenciamento começam com `aps.*` e os endpoints do ambiente de dados começam com `aps-workspaces.*`. Os endpoints que terminam em `.amazonaws.com` são compatíveis com IPv4 e os endpoints que terminam em `.api.aws` são compatíveis com IPv4 e IPv6.

Preços

Você incorre em cobranças pela ingestão e armazenamento de métricas. As cobranças de armazenamento são baseadas no tamanho compactado das amostras métricas e dos metadados. Para obter mais informações, consulte Definição de preços do Amazon Managed Service for Prometheus.

Você pode usar os Relatórios de AWS Custo AWS Cost Explorer e Uso para monitorar suas cobranças. Para obter mais informações, consulte Explorando seus dados usando o Cost Explorer e O que são relatórios de AWS custo e uso.

Suporte premium

Se você assinar qualquer nível dos planos de suporte AWS premium, seu suporte premium se aplica ao Amazon Managed Service for Prometheus.

Conceitos básicos do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus é um serviço com tecnologia sem servidor compatível com o Prometheus para monitorar métricas de contêiner que facilita o monitoramento seguro de ambientes de contêiner em escala. Esta seção mostra três áreas principais do uso do Amazon Managed Service para Prometheus:

- Criar um espaço de trabalho: criar um espaço de trabalho do Amazon Managed Service for Prometheus para armazenar e monitorar suas métricas.
- Ingerir métricas: seu espaço de trabalho fica vazio até você ter métricas em seu espaço de trabalho. Você pode enviar métricas ao Amazon Managed Service for Prometheus ou fazer com que o ele extraia métricas automaticamente.
- Consultar métricas: quando tiver métricas como dados em seu espaço de trabalho, você terá tudo pronto para consultar os dados a fim de explorar ou monitorar essas métricas.

Se você é novo no AWS, esta seção também inclui detalhes sobre como configurar um Conta da AWS.

Tópicos

- Configurar AWS
- Criar um espaço de trabalho do Amazon Managed Service for Prometheus
- Ingerir métricas do Prometheus no espaço de trabalho
- Consultar as métricas do Prometheus

Configurar AWS

Conclua as tarefas desta seção para começar a AWS usá-las pela primeira vez. Se você já tem uma AWS conta, vá paraCriar um espaço de trabalho do Amazon Managed Service for Prometheus.

Quando você se inscreve AWS, sua AWS conta tem acesso automático a todos os serviços AWS, incluindo o Amazon Managed Service for Prometheus. Entretanto, você será cobrado apenas pelos serviços que usar.

Tópicos

- Inscreva-se para um Conta da AWS

Inscreva-se para um Conta da AWS

Para começar AWS, você precisa de um Conta da AWS. Para obter informações sobre como criar um Conta da AWS, consulte Introdução a um Conta da AWS no Guia de AWS Gerenciamento de contas referência.

Criar um espaço de trabalho do Amazon Managed Service for Prometheus

Um espaço de trabalho é um espaço lógico dedicado ao armazenamento e à consulta das métricas do Prometheus. Um espaço de trabalho oferece suporte a um controle de acesso refinado para autorizar seu gerenciamento, como atualizar, listar, descrever e excluir, além da ingestão e consulta de métricas. É possível ter um ou mais espaços de trabalho em cada região na sua conta.

Para configurar um espaço de trabalho, siga estas etapas.

> **Note**
>
> Para obter informações detalhadas sobre como criar um espaço de trabalho e as opções disponíveis, consulte Criar um espaço de trabalho do Amazon Managed Service for Prometheus.

Para criar um espaço de trabalho do Amazon Managed Service for Prometheus

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. Em Alias do espaço de trabalho, insira um alias para o novo espaço de trabalho.

   Os aliases do workspace são nomes simplificados, que ajudam a identificar seus workspaces. Eles não precisam ser exclusivos. Dois espaços de trabalho podem ter o mesmo alias, mas todos os espaços de trabalho terão um espaço de trabalho exclusivo IDs, que é gerado pelo Amazon Managed Service for Prometheus.
3. (Opcional) Para adicionar tags ao namespace, selecione Adicionar nova tag.

Em seguida, em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.

Para adicionar outra tag, escolha novamente Adicionar nova tag.

4. Selecione Criar espaço de trabalho.

   A página de detalhes do espaço de trabalho é exibida. Isso exibe informações, incluindo o status, o ARN, o ID do espaço de trabalho e o endpoint desse espaço de trabalho, tanto URLs para gravação remota quanto para consultas.

   Inicialmente, o status provável é CRIANDO. Espere até que o status esteja como ATIVO antes de prosseguir com a configuração da ingestão de métricas.

   Faça anotações do URLs exibido para Endpoint - URL de gravação remota e Endpoint - URL de consulta. Você precisará deles ao configurar seu servidor Prometheus para gravar métricas remotamente nesse workspace e ao consultar essas métricas.

Ingerir métricas do Prometheus no espaço de trabalho

Uma forma de ingerir métricas é usar um agente autônomo do Prometheus (uma instância do Prometheus em execução no modo Agente) para extrair métricas do cluster e encaminhá-las para o Amazon Managed Service for Prometheus para armazenamento e monitoramento. Esta seção explica como configurar a ingestão de métricas no espaço de trabalho do Amazon Managed Service for Prometheus a partir do Amazon EKS configurando uma nova instância do agente do Prometheus usando o Helm.

Para gerar métricas no Amazon EKS, como Kubernetes ou métricas em nível de nó, você pode usar os complementos da comunidade Amazon EKS. Para obter mais informações, consulte Complementos comunitários disponíveis no Guia do usuário do Amazon EKS.

Para obter informações sobre outras formas de ingerir dados no Amazon Managed Service for Prometheus, incluindo como proteger métricas e criar métricas de alta disponibilidade, consulte Ingerir métricas no seu espaço de trabalho do Amazon Managed Service for Prometheus.

> **Note**
>
> As métricas ingeridas em um espaço de trabalho são armazenadas durante 150 dias por padrão e, em seguida, excluídas automaticamente. Você pode ajustar o período de retenção

configurando seu espaço de trabalho em um máximo de até 1.095 dias (3 anos). Para obter mais informações, consulte Configurar seu espaço de trabalho.

As instruções nesta seção permitem que você comece a usar o Amazon Managed Service for Prometheus rapidamente. Elas consideram que você já tenha criado um espaço de trabalho. Nesta seção, você configura um novo servidor do Prometheus em um cluster do Amazon EKS, e o novo servidor usa uma configuração padrão para atuar como agente a fim de enviar métricas ao Amazon Managed Service for Prometheus. Este método tem os seguintes pré-requisitos:

- Você deve ter um cluster do Amazon EKS do qual o novo servidor do Prometheus coletará métricas.
- Seu cluster do Amazon EKS deve ter um driver do Amazon EBS CSI instalado (necessário ao Helm).
- Você deve usar a CLI do Helm 3.0 ou posterior.
- Você deve usar um computador Linux ou MacOS para executar as etapas nas seções a seguir.

Etapa 1: Adicionar novos repositórios de charts do Helm

Insira os comandos a seguir para adicionar novos repositórios de charts do Helm. Para obter mais informações sobre esses comandos, consulte o Repositório do Helm.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo add kube-state-metrics https://kubernetes.github.io/kube-state-metrics
helm repo update
```

Etapa 2: Criar um namespace do Prometheus

Digite o comando a seguir para criar um namespace do Prometheus para o servidor Prometheus e outros componentes de monitoramento. *prometheus-agent-namespace*Substitua pelo nome que você deseja para esse namespace.

```
kubectl create namespace prometheus-agent-namespace
```

Etapa 3: Configurar perfis do IAM para as contas de serviço

Para esse método de ingestão, é necessário usar perfis do IAM para contas de serviço no cluster do Amazon EKS em que o agente do Prometheus está em execução.

Com os perfis do IAM para contas de serviço, é possível associar um perfil do IAM a uma conta de serviço do Kubernetes. Essa conta de serviço pode então fornecer AWS permissões para os contêineres em qualquer pod que use essa conta de serviço. Para obter mais informações, consulte Perfis do IAM para contas de serviço.

Se você ainda não configurou esses perfis, siga as instruções em Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS para configurar os perfis. As instruções nessa seção exigem o uso do `eksctl`. Para obter mais informações, consulte Conceitos básicos do Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Quando você não está usando o EKS ou AWS está usando apenas a chave de acesso e a chave secreta para acessar o Amazon Managed Service para Prometheus, você não pode usar `EKS-IAM-ROLE` o SigV4 baseado.

Etapa 4: Configurar o novo servidor e começar a ingerir métricas

Para instalar o novo agente do Prometheus e enviar métricas para o espaço de trabalho do Amazon Managed Service for Prometheus, siga estas etapas.

Como instalar o novo agente do Prometheus e enviar métricas para o espaço de trabalho do Amazon Managed Service for Prometheus

1. Use um editor de textos para criar um arquivo chamado `my_prometheus_values_yaml` com o conteúdo a seguir.

   - *`IAM_PROXY_PROMETHEUS_ROLE_ARN`*Substitua pelo ARN do amp-iamproxy-ingest-roleque você criou em. Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS
   - *`WORKSPACE_ID`*Substitua pelo ID do seu espaço de trabalho do Amazon Managed Service for Prometheus.

- *REGION*Substitua pela região do seu espaço de trabalho do Amazon Managed Service for Prometheus.

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
## For the rest of prometheus helm chart values see: https://github.com/prometheus-
community/helm-charts/blob/main/charts/prometheus/values.yaml
##
serviceAccounts:
  server:
    name: amp-iamproxy-ingest-service-account
    annotations:
      eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
server:
  remoteWrite:
    - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
      sigv4:
        region: ${REGION}
      queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
```

2. Insira o comando a seguir para criar o servidor Prometheus.

- Substitua *prometheus-chart-name* pelo nome da versão do Prometheus.
- *prometheus-agent-namespace*Substitua pelo nome do seu namespace Prometheus.

```
helm install prometheus-chart-name prometheus-community/prometheus -n prometheus-
agent-namespace \
-f my_prometheus_values_yaml
```

Consultar as métricas do Prometheus

Agora que as métricas estão sendo ingeridas no espaço de trabalho, você pode consultá-las. Uma forma comum de consultar métricas é usar um serviço como o Grafana. Nesta seção, você

aprenderá a usar o Amazon Managed Grafana para consultar métricas do Amazon Managed Service for Prometheus.

> **Note**
>
> Para saber mais sobre outras formas de consultar suas métricas do Amazon Managed Service for Prometheus ou usar o Amazon Managed Service for APIs Prometheus, consulte. Consultar as métricas do Prometheus

Esta seção considera que você já tenha criado um espaço de trabalho e esteja ingerindo métricas nele.

As consultas são realizadas por meio da linguagem de consulta padrão do Prometheus, PromQL. Para obter mais informações sobre o PromQL e sua sintaxe, veja Consultando Prometheus na documentação do Prometheus.

O Amazon Managed Grafana é um serviço totalmente gerenciado para o Grafana de código aberto que simplifica a conexão com ISVs terceirizados de código aberto AWS e serviços para visualizar e analisar suas fontes de dados em grande escala.

O Amazon Managed Service for Prometheus oferece suporte ao uso do Amazon Managed Grafana para consultar métricas em um espaço de trabalho. No console do Amazon Managed Grafana, você pode adicionar um espaço de trabalho do Amazon Managed Service for Prometheus como fonte de dados descobrindo suas contas existentes do Amazon Managed Service for Prometheus. O Amazon Managed Grafana gerencia a configuração das credenciais de autenticação necessárias para acessar o Amazon Managed Service for Prometheus. Para obter instruções detalhadas sobre como criar uma conexão com o Amazon Managed Service for Prometheus a partir do Amazon Managed Grafana, consulte as instruções no Guia do usuário do Amazon Managed Grafana.

Você também pode visualizar seus alertas do Amazon Managed Service for Prometheus no Amazon Managed Grafana. Para obter instruções sobre como configurar a integração com alertas, consulte Integrar alertas com o Amazon Managed Grafana ou o Grafana de código aberto.

> **Note**
>
> Se você configurou o espaço de trabalho do Amazon Managed Grafana para usar uma VPC privada, deve conectar o espaço de trabalho do Amazon Managed Service for Prometheus

à mesma VPC. Para obter mais informações, consulte Conexão com o Amazon Managed Grafana em uma VPC privada.

Gerenciar espaços de trabalho do Amazon Managed Service for Prometheus

Um espaço de trabalho é um espaço lógico dedicado ao armazenamento e à consulta das métricas do Prometheus. Um espaço de trabalho oferece suporte a um controle de acesso refinado para autorizar seu gerenciamento, como atualizar, listar, descrever e excluir, além da ingestão e consulta de métricas. É possível ter um ou mais espaços de trabalho em cada região na sua conta.

Use os procedimentos desta seção para criar e gerenciar seus espaços de trabalho do Amazon Managed Service for Prometheus.

Tópicos

Criar um espaço de trabalho do Amazon Managed Service for Prometheus

Siga estas etapas para criar um espaço de trabalho do Amazon Managed Service for Prometheus. Você pode escolher usar o AWS CLI ou o console do Amazon Managed Service for Prometheus.

> **Note**
>
> Se você estiver executando um cluster do Amazon EKS, também poderá criar um novo espaço de trabalho usando Controladores da AWS para Kubernetes.

Para criar um espaço de trabalho usando o AWS CLI

1. Insira o comando a seguir para criar o espaço de trabalho. Este exemplo cria um espaço de trabalho chamado `my-first-workspace`, mas você pode usar um alias diferente (ou

nenhum), se preferir. Os aliases do espaço de trabalho são nomes simplificados, que ajudam a identificar seus espaços de trabalho. Eles não precisam ser exclusivos. Dois espaços de trabalho podem ter o mesmo alias, mas todos os espaços de trabalho têm um espaço de trabalho exclusivo IDs, que é gerado pelo Amazon Managed Service for Prometheus.

(Opcional) Para usar sua própria chave KMS para criptografar dados armazenados em seu espaço de trabalho, você pode incluir o `kmsKeyArn` parâmetro com a AWS KMS chave a ser usada. Embora o Amazon Managed Service for Prometheus não cobre pelo uso de chaves gerenciadas pelo cliente, pode haver custos associados às chaves de. AWS Key Management Service Para obter mais informações sobre a criptografia de dados no espaço de trabalho do Amazon Managed Service for Prometheus ou sobre como criar, gerenciar e usar sua própria chave gerenciada pelo cliente, consulte Criptografia em repouso.

Os parâmetros entre colchetes (`[]`) são opcionais. Não inclua os colchetes no comando.

```
aws amp create-workspace [--alias my-first-workspace] [--kmsKeyArn arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-abcd-56ef-7890abcd12ef]  [--tags Status=Secret,Team=My-Team]
```

Este comando retorna os seguintes dados:

- `workspaceId` é a ID exclusiva desse espaço de trabalho. Anote essa ID.
- `arn` é o ARN desse espaço de trabalho.
- `status` é o status atual do espaço de trabalho. Imediatamente depois de criar o espaço de trabalho, ele provavelmente será `CREATING`.
- `kmsKeyArn` é a chave gerenciada pelo cliente usada para criptografar os dados do espaço de trabalho, se fornecida.

> **Note**
>
> Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem usar coletores gerenciados pela AWS para ingestão.
> Escolha se deseja usar as chaves gerenciadas pelo cliente ou as chaves AWS próprias com cuidado. Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem ser convertidos para usar chaves AWS próprias posteriormente (e vice-versa).

- `tags` lista as tags do espaço de trabalho, se houver.

2. Se seu comando `create-workspace` retornar um status de `CREATING`, você poderá inserir o comando a seguir para determinar quando o espaço de trabalho estará pronto. *`my-workspace-id`*Substitua pelo valor pelo qual o `create-workspace` comando retornou`workspaceId`.

```
aws amp describe-workspace --workspace-id my-workspace-id
```

   Quando o comando `describe-workspace` retornar `ACTIVE` para o `status`, o espaço de trabalho estará pronto para uso.

Para criar um espaço de trabalho usando o console do Amazon Managed Service for Prometheus

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. Escolha Criar.
3. Em Alias do espaço de trabalho, insira um alias para o novo espaço de trabalho.

   Os aliases do workspace são nomes simplificados, que ajudam a identificar seus workspaces. Eles não precisam ser exclusivos. Dois espaços de trabalho podem ter o mesmo alias, mas todos os espaços de trabalho têm um espaço de trabalho exclusivo IDs, que é gerado pelo Amazon Managed Service for Prometheus.
4. (Opcional) Para usar sua própria chave KMS para criptografar dados armazenados em seu espaço de trabalho, você pode selecionar Personalizar configurações de criptografia e escolher a AWS KMS chave a ser usada (ou criar uma nova). É possível selecionar uma chave na conta a partir da lista suspensa ou inserir o ARN de qualquer chave à qual tenha acesso. Embora o Amazon Managed Service for Prometheus não cobre pelo uso de chaves gerenciadas pelo cliente, pode haver custos associados às chaves de. AWS Key Management Service

   Para obter mais informações sobre a criptografia de dados no espaço de trabalho do Amazon Managed Service for Prometheus ou sobre como criar, gerenciar e usar sua própria chave gerenciada pelo cliente, consulte Criptografia em repouso.

   > **Note**
   >
   > Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem usar coletores gerenciados pela AWS para ingestão.

Escolha se deseja usar as chaves gerenciadas pelo cliente ou as chaves AWS próprias com cuidado. Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem ser convertidos para usar chaves AWS próprias posteriormente (e vice-versa).

5. (Opcional) Para adicionar uma ou mais tags ao espaço de trabalho, selecione Adicionar nova tag. Em seguida, em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.

   Para adicionar outra tag, escolha novamente Adicionar nova tag.

6. Selecione Criar espaço de trabalho.

   A página de detalhes do espaço de trabalho é exibida. Isso exibe informações, incluindo o status, o ARN, o ID do espaço de trabalho e o endpoint desse espaço de trabalho URLs para gravação e consultas remotas.

   O status retorna CREATING até que o espaço de trabalho esteja pronto. Espere até que o status esteja como ATIVO antes de prosseguir com a configuração da ingestão de métricas.

   Anote os URLs que são exibidos para Endpoint - URL de gravação remota e Endpoint - URL de consulta. Você precisará deles ao configurar seu servidor Prometheus para gravar métricas remotamente nesse espaço de trabalho e ao consultar essas métricas.

Para obter informações sobre como ingerir métricas no espaço de trabalho, consulte Ingerir métricas do Prometheus no espaço de trabalho.

Configurar seu espaço de trabalho

Você pode configurar seu espaço de trabalho para o seguinte:

- Defina conjuntos de rótulos e defina limites nas séries temporais ativas que correspondem aos conjuntos de rótulos definidos. Um conjunto de rótulos é um conjunto de um ou mais rótulos, que são name/value pares que ajudam a contextualizar as métricas de séries temporais.

  Ao definir conjuntos de rótulos e definir limites ativos de séries temporais, você pode limitar os picos em um locatário ou fonte para afetar somente esse locatário ou fonte. Por exemplo, se você definir um limite de 1.000.000 de séries temporais ativas no conjunto de rótulos `team=A env=prod`, se o número de séries temporais ingeridas que correspondem a esse conjunto de

rótulos exceder o limite, somente as séries temporais que correspondem ao conjunto de rótulos serão limitadas. Dessa forma, outros inquilinos ou fontes métricas não são afetados.

Para obter mais informações sobre rótulos no Prometheus, consulte Data Model.

- Defina um período de retenção para definir o número de dias para que os dados sejam retidos no espaço de trabalho.

Como configurar seu espaço de trabalho

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os workspaces.
3. Selecione o ID do espaço de trabalho do espaço de trabalho.
4. Escolha a guia Configurações do espaço de trabalho.
5. Para definir o período de retenção do espaço de trabalho, escolha Editar na seção Período de retenção. Em seguida, especifique o novo período de retenção em dias. O máximo é de 1.095 dias (3 anos).
6. Para adicionar ou modificar conjuntos de rótulos e os limites de série ativos deles, escolha Editar na seção Conjuntos de rótulos. Faça o seguinte:

   a. (Opcional) Insira um valor em Limite de bucket padrão para definir um limite para o número máximo de séries temporais ativas que podem ser ingeridas no espaço de trabalho, contando apenas as séries temporais que não correspondem a nenhum conjunto de rótulos definido.

   b. Para definir um conjunto de rótulos, insira um limite de série temporal ativo para o novo conjunto de rótulos em Limite de séries ativas.

      Em seguida, insira um rótulo e um valor para um rótulo que será usado no conjunto de rótulos e escolha Adicionar rótulo.

   c. (Opcional) Para definir outro conjunto de etiquetas, escolha Adicionar outro conjunto de rótulos e repita as etapas anteriores.
7. Ao concluir, escolha Save changes.

Editar um alias de espaço de trabalho

Você pode editar um espaço de trabalho para alterar seu alias. Para alterar o alias do espaço de trabalho usando a AWS CLI, insira o comando a seguir.

```
aws amp update-workspace-alias --workspace-id my-workspace-id --alias "new-alias"
```

Para editar um espaço de trabalho usando o console do Amazon Managed Service for Prometheus

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os espaços de trabalho.
3. Escolha o ID do espaço de trabalho que você deseja editar e, em seguida, selecione Editar.
4. Insira um novo alias para o espaço de trabalho e selecione Salvar.

Encontrar os detalhes do seu espaço de trabalho do Amazon Managed Service for Prometheus, incluindo o ARN

Você pode encontrar os detalhes do seu espaço de trabalho do Amazon Managed Service for Prometheus usando o console da AWS ou o AWS CLI.

Console

Para encontrar os detalhes do seu espaço de trabalho usando o console do Amazon Managed Service for Prometheus

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os workspaces.
3. Selecione o ID do espaço de trabalho do espaço de trabalho. Isso exibirá detalhes sobre seu espaço de trabalho, incluindo:
   - Status atual: o status do seu espaço de trabalho, por exemplo, Ativo, é exibido em Status.
   - ARN: o ARN do espaço de trabalho é exibido ARN.

- ID: o ID do espaço de trabalho é exibido em ID do espaço de trabalho.
- URLs— O console exibe vários URLs para o espaço de trabalho, incluindo aqueles URLs para gravação ou consulta de dados do espaço de trabalho.

> **Note**
>
> Por padrão, os URLs dados são os IPv4 URLs. Você também pode usar o dualstack (IPv4 e IPv6 suportado). URLs Eles são iguais, mas estão no domínio `api.aws` no lugar do `amazonaws.com` padrão. Por exemplo, se você visse o seguinte (um IPv4 URL):
>
> ```
> https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
> ```
>
> Você pode criar uma pilha dupla (incluindo suporte para IPv6), URL da seguinte forma:
>
> ```
> https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/api/v1/remote_write
> ```

Abaixo desta seção, há guias com informações sobre regras, gerenciador de alertas, logs, configuração e tags.

AWS CLI

Para encontrar os detalhes do seu espaço de trabalho usando o AWS CLI

O comando a seguir retorna os detalhes do espaço de trabalho. Você deve *my-workspace-id* substituir pelo ID do espaço de trabalho do qual deseja obter os detalhes.

```
aws amp describe-workspace --workspace-id my-workspace-id
```

Isso retorna detalhes sobre seu espaço de trabalho, incluindo:

- Status atual: o status do seu espaço de trabalho, por exemplo, `ACTIVE`, é retornado na propriedade `statusCode`.
- ARN: o ARN do espaço de trabalho é retornado na propriedade `arn`.

- URLs— AWS CLI Retorna o URL base do espaço de trabalho na `prometheusEndpoint` propriedade.

> **Note**
>
> Por padrão, o URL retornado é o IPv4 URL. Você também pode usar um URL de pilha dupla (IPv4 IPv6 compatível) no domínio em `api.aws` vez do padrão. `amazonaws.com` Por exemplo, se você visse o seguinte (um IPv4 URL):
>
> ```
> https://aps-workspaces.us-east-1.amazonaws.com/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```
>
> Você pode criar uma pilha dupla (incluindo suporte para IPv6), URL da seguinte forma:
>
> ```
> https://aps-workspaces.us-east-1.api.aws/workspaces/ws-abcd1234-ef56-7890-ab12-example/
> ```
>
> Você também pode criar a gravação e a consulta remotas URLs para o espaço de trabalho adicionando `/api/v1/remote_write` ou`/api/v1/query`, respectivamente.

Excluir um espaço de trabalho do Amazon Managed Service for Prometheus

A exclusão de um espaço de trabalho exclui os dados que foram ingeridos nele.

> **Note**
>
> A exclusão de um espaço de trabalho do Amazon Managed Service for Prometheus não exclui automaticamente AWS nenhum coletor gerenciado que esteja coletando métricas e as enviando para o espaço de trabalho. Para obter mais informações, consulte Encontrar e excluir extratores.

Para excluir um espaço de trabalho usando o AWS CLI

Use o seguinte comando:

```
aws amp delete-workspace --workspace-id my-workspace-id
```

Para excluir um espaço de trabalho usando o console do Amazon Managed Service for Prometheus

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os workspaces.
3. Escolha o ID do espaço de trabalho que você deseja excluir e, em seguida, selecione Excluir.
4. Na caixa de confirmação, insira **`delete`** e selecione Excluir.

Ingerir métricas no seu espaço de trabalho do Amazon Managed Service for Prometheus

As métricas devem ser ingeridas no seu espaço de trabalho do Amazon Managed Service for Prometheus antes que você possa consultar ou alertar sobre essas métricas. Esta seção explica como configurar a ingestão de métricas em seu espaço de trabalho.

> **Note**
>
> As métricas ingeridas em um espaço de trabalho são armazenadas durante 150 dias por padrão e, em seguida, excluídas automaticamente. Você pode ajustar o período de retenção configurando seu espaço de trabalho em um máximo de até 1.095 dias (3 anos). Para obter mais informações, consulte Configurar seu espaço de trabalho.

Há dois métodos para ingerir métricas no espaço de trabalho do Amazon Managed Service for Prometheus.

- Usando um coletor AWS gerenciado — o Amazon Managed Service for Prometheus fornece um raspador totalmente gerenciado e sem agentes para extrair automaticamente métricas de seus clusters do Amazon Elastic Kubernetes Service (Amazon EKS). A extração extrai automaticamente as métricas dos endpoints compatíveis com o Prometheus.
- Usar um coletor gerenciado pelo cliente: há muitas opções para gerenciar seu próprio coletor. Dois dos coletores mais comuns de usar são instalar sua própria instância do Prometheus, executar no modo agente ou usar o Distro for. AWS OpenTelemetry Essas etapas são descritas em detalhes nas seções a seguir.

  Os coletores enviam métricas para o Amazon Managed Service for Prometheus usando a funcionalidade de gravação remota do Prometheus. É possível enviar métricas diretamente para o Amazon Managed Service for Prometheus usando a gravação remota do Prometheus em sua própria aplicação. Para obter mais detalhes sobre como usar diretamente a gravação remota e as configurações de gravação remota, consulte remote_write na documentação do Prometheus.

Tópicos

- Métricas de ingestão com coletores AWS gerenciados

- Coletores gerenciados pelo cliente

Métricas de ingestão com coletores AWS gerenciados

Um caso de uso comum do Amazon Managed Service for Prometheus é monitorar clusters do Kubernetes gerenciados pelo Amazon Elastic Kubernetes Service (Amazon EKS). Os clusters do Kubernetes e muitas aplicações executadas no Amazon EKS exportam automaticamente suas métricas para acesso aos extratores compatíveis com o Prometheus.

> **Note**
>
> O Amazon EKS expõe métricas do servidor de API, métricas `kube-controller-manager` e métricas `kube-scheduler` em um cluster. Muitas outras tecnologias e aplicativos executados em ambientes Kubernetes fornecem métricas compatíveis com o Prometheus. Para obter uma lista de exportadores bem-documentados, veja Exportadores e integrações na documentação do Prometheus.

O Amazon Managed Service for Prometheus fornece um extrator ou coletor totalmente gerenciado e sem agentes que descobre e extrai automaticamente métricas compatíveis com o Prometheus. Não é necessário gerenciar, instalar, aplicar patches ou manter agentes ou extratores. Um coletor do Amazon Managed Service for Prometheus fornece uma coleção de métricas confiável, estável, altamente disponível e escalada automaticamente para o cluster do Amazon EKS. Os coletores gerenciados pelo Amazon Managed Service for Prometheus funcionam com clusters do Amazon EKS, incluindo EC2 e Fargate.

Um coletor do Amazon Managed Service for Prometheus cria uma interface de rede elástica (ENI) por sub-rede especificada ao criar o extrator. O coletor coleta as métricas por meio delas ENIs e as usa `remote_write` para enviar os dados para seu espaço de trabalho do Amazon Managed Service for Prometheus usando um VPC endpoint. Os dados extraídos nunca viajam na Internet pública.

Os tópicos a seguir fornecem mais informações sobre como usar um coletor do Amazon Managed Service for Prometheus no cluster do Amazon EKS e sobre as métricas coletadas.

Tópicos

- Configurar coletores gerenciados para o Amazon EKS
- Configure coletores gerenciados do Prometheus para o Amazon MSK

- O que são métricas compatíveis com o Prometheus?
- Monitorar coletores com logs fornecidos

Configurar coletores gerenciados para o Amazon EKS

Para usar um coletor do Amazon Managed Service for Prometheus, crie um extrator que descubra e extraia métricas no cluster do Amazon EKS. Você também pode criar um extrator que se integre ao Amazon Managed Streaming for Apache Kafka. Para obter mais informações, consulte Integrar o Amazon MSK.

- É possível criar um extrator como parte da criação do cluster do Amazon EKS. Para obter mais informações sobre a criação de um cluster do Amazon EKS, incluindo a criação de um extrator, consulte Criar um cluster do Amazon EKS no Guia do usuário do Amazon EKS.
- Você pode criar seu próprio raspador, programaticamente com a AWS API ou usando o. AWS CLI

Um coletor do Amazon Managed Service for Prometheus extrai métricas compatíveis com o Prometheus. Para obter mais informações sobre as métricas compatíveis com o Prometheus, consulte O que são métricas compatíveis com o Prometheus?. Os clusters do Amazon EKS expõem métricas para o servidor da API. Clusters do Amazon EKS na versão `1.28` ou superior do Kubernetes também expõem métricas para o `kube-scheduler` e o `kube-controller-manager`. Para obter mais informações, consulte Obter as métricas brutas do ambiente de gerenciamento no formato do Prometheus no Guia do usuário do Amazon EKS.

> **Note**
>
> A coleta de métricas de um cluster pode gerar cobranças pelo uso da rede. Uma forma de otimizar esses custos é configurar seu endpoint `/metrics` para que compacte as métricas fornecidas (por exemplo, com gzip), reduzindo os dados que devem ser movidos pela rede. O modo de fazer isso depende do aplicativo ou da biblioteca que fornece as métricas. Algumas bibliotecas são gzip por padrão.

Os tópicos a seguir descrevem como criar, gerenciar e configurar extratores.

Tópicos

- Criar um extrator

- Configurar o cluster do Amazon EKS
- Encontrar e excluir extratores
- Configuração do extrator
- Solução de problemas de configuração do extrator
- Limitações do extrator

Criar um extrator

Um coletor do Amazon Managed Service for Prometheus consiste em um extrator que descobre e coleta métricas de um cluster do Amazon EKS. O Amazon Managed Service for Prometheus gerencia o extrator para você, fornecendo a escalabilidade, a segurança e a confiabilidade necessárias, sem que você precise gerenciar instâncias, agentes ou extratores por conta própria.

Existem três maneiras de criar um extrator:

- Um extrator é criado automaticamente para você ao criar um cluster do Amazon EKS pelo Amazon EKS e escolher ativar as métricas do Prometheus.
- É possível criar um extrator no console do Amazon EKS para um cluster existente. Abra o cluster no console do Amazon EKS e, na guia Observabilidade, escolha Adicionar extrator.

  Para obter mais detalhes sobre as configurações disponíveis, consulte Ativar as métricas do Prometheus no Guia do usuário do Amazon EKS.
- Você pode criar um raspador usando a AWS API ou o. AWS CLI

  Essas opções são descritas no procedimento a seguir.

Há alguns pré-requisitos para a criação de um extrator próprio:

- É necessário ter um cluster do Amazon EKS.
- O cluster do Amazon EKS deve ter o controle de acesso ao endpoint do cluster definido para incluir acesso privado. Ele pode incluir o privado e o público, mas deve incluir o privado.
- A Amazon VPC na qual o cluster Amazon EKS reside deve ter o DNS habilitado.

> **Note**
>
> O cluster será associado ao extrator pelo nome do recurso da Amazon (ARN). Se você excluir um cluster e criar um novo com o mesmo nome, o ARN será reutilizado para o novo cluster. Por esse motivo, o extrator tentará coletar métricas para o novo cluster. Você exclui extratores separadamente da exclusão do cluster.

AWS API

Como criar um extrator usando a API da AWS

Use a operação `CreateScraper` da API para criar um raspador com a AWS API. O exemplo a seguir cria um extrator na região `us-west-2`. Você precisa substituir as informações do espaço de trabalho Conta da AWS, da segurança e do cluster do Amazon EKS pelas suas próprias IDs e fornecer a configuração a ser usada para seu raspador.

> **Note**
>
> O grupo de segurança e as sub-redes devem ser definidos como o grupo de segurança e as sub-redes do cluster ao qual você se conectará.
> É necessário incluir, pelo menos, duas sub-redes em, pelo menos, duas zonas de disponibilidade.

`scrapeConfiguration` é um arquivo YAML de configuração do Prometheus codificado em base64. É possível baixar uma configuração de uso geral com a operação `GetDefaultScraperConfiguration` da API. Para obter mais informações sobre o formato do `scrapeConfiguration`, consulte Configuração do extrator.

```
POST /scrapers HTTP/1.1
Content-Length: 415
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: aws-cli/1.18.147 Python/2.7.18 Linux/5.4.58-37.125.amzn2int.x86_64
 botocore/1.18.6

{
    "alias": "myScraper",
```

```
    "destination":  {
        "ampConfiguration": {
            "workspaceArn": "arn:aws:aps:us-west-2:account-id:workspace/
ws-workspace-id"
        }
    },
    "source": {
        "eksConfiguration": {
            "clusterArn": "arn:aws:eks:us-west-2:account-id:cluster/cluster-name",
            "securityGroupIds": ["sg-security-group-id"],
            "subnetIds": ["subnet-subnet-id-1", "subnet-subnet-id-2"]
        }
    },
    "scrapeConfiguration": {
        "configurationBlob": <base64-encoded-blob>
    }
}
```

AWS CLI

Como criar um extrator usando a AWS CLI

Use o comando `create-scraper` para criar um extrator com a AWS CLI. O exemplo a seguir cria um extrator na região `us-west-2`. Você precisa substituir as informações do espaço de trabalho Conta da AWS, da segurança e do cluster do Amazon EKS pelas suas próprias IDs e fornecer a configuração a ser usada para seu raspador.

> **Note**
>
> O grupo de segurança e as sub-redes devem ser definidos como o grupo de segurança e as sub-redes do cluster ao qual você se conectará.
> É necessário incluir, pelo menos, duas sub-redes em, pelo menos, duas zonas de disponibilidade.

`scrape-configuration` é um arquivo YAML de configuração do Prometheus codificado em base64. É possível baixar uma configuração de uso geral com o comando `get-default-scraper-configuration`. Para obter mais informações sobre o formato do `scrape-configuration`, consulte Configuração do extrator.

```
aws amp create-scraper \
```

```
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-
id:cluster/cluster-name', securityGroupIds=['sg-security-group-
id'],subnetIds=['subnet-subnet-id-1', 'subnet-subnet-id-2']}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-
id:workspace/ws-workspace-id'}"
```

Veja a seguir uma lista completa das operações do extrator que você pode usar com a API da AWS :

- Crie um extrator com a operação CreateScraper da API.
- Liste os extratores existentes com a operação ListScrapers da API.
- Atualize o alias, a configuração ou o destino de um raspador com a operação da UpdateScraperAPI.
- Exclua um extrator com a operação DeleteScraper da API.
- Obtenha mais detalhes sobre um extrator com a operação DescribeScraper da API.
- Obtenha uma configuração de uso geral para extratores com a operação GetDefaultScraperConfiguration da API.

> **Note**
>
> O cluster do Amazon EKS que você está extraindo deve ser configurado para permitir que o Amazon Managed Service for Prometheus acesse as métricas. O próximo tópico descreve como configurar o cluster.

Configuração entre contas

Para criar um extrator entre contas quando seu cluster Amazon EKS e o espaço de trabalho do Amazon Managed Service for Prometheus estão em contas diferentes, use o procedimento a seguir. Por exemplo, você tem uma conta de origem `account_id_source` que contém o cluster Amazon EKS e uma conta de destino `account_id_target` que contém o espaço de trabalho do Amazon Managed Service for Prometheus.

Como criar um extrator em uma configuração entre contas

1. Na conta de origem, crie um perfil `arn:aws:iam::`*`account_id`*`_source:role/Source` e adicione a política de confiança a seguir.

```
{
    "Effect": "Allow",
    "Principal": {
    "Service": [
        "scraper.aps.amazonaws.com"
     ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "scraper_ARN"
        },
        "StringEquals": {
            "AWS:SourceAccount": "account_id"
        }
    }
}
```

2. Em cada combinação de origem (cluster Amazon EKS) e destino (espaço de trabalho do Amazon Managed Service for Prometheus), você precisa criar uma `arn:aws:iam::`*`account_id`*`_target:role/Target` função e adicionar a seguinte política de confiança com permissões para. AmazonPrometheusRemoteWriteAccess

```
{
  "Effect": "Allow",
  "Principal": {
     "AWS": "arn:aws:iam::account_id_source:role/Source"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
     "StringEquals": {
        "sts:ExternalId": "scraper_ARN"
      }
  }
}
```

3. Crie um extrator com a opção `--role-configuration`.

```
aws amp create-scraper \
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-
id_source:cluster/xarw,subnetIds=[subnet-subnet-id]}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-
id_target:workspace/ws-workspace-id'}"\
  --role-configuration '{"sourceRoleArn":"arn:aws:iam::account-id_source:role/
Source", "targetRoleArn":"arn:aws:iam::account-id_target:role/Target"}'
```

4. Valide a criação do extrator.

```
aws amp list-scrapers
{
    "scrapers": [
        {
            "scraperId": "scraper-id",
            "arn": "arn:aws:aps:us-west-2:account_id_source:scraper/scraper-id",
            "roleArn": "arn:aws:iam::account_id_source:role/aws-service-role/
scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraperInternal_cc319052-41a3-4",
            "status": {
                "statusCode": "ACTIVE"
            },
            "createdAt": "2024-10-29T16:37:58.789000+00:00",
            "lastModifiedAt": "2024-10-29T16:55:17.085000+00:00",
            "tags": {},
            "source": {
                "eksConfiguration": {
                    "clusterArn": "arn:aws:eks:us-west-2:account_id_source:cluster/
xarw",
                    "securityGroupIds": [
                        "sg-security-group-id",
                        "sg-security-group-id"
                    ],
                    "subnetIds": [
                        "subnet-subnet_id"
                    ]
                }
            },
            "destination": {
                "ampConfiguration": {
```

```
                    "workspaceArn": "arn:aws:aps:us-
west-2:account_id_target:workspace/ws-workspace-id"
                }
            }
        }
    ]
}
```

Alteração entre uma RoleConfiguration função vinculada ao serviço

Quando quiser voltar para um perfil vinculado ao serviço em vez de `RoleConfiguration` para gravar em um espaço de trabalho do Amazon Managed Service for Prometheus, você deve atualizar o `UpdateScraper` e fornecer um espaço de trabalho na mesma conta do extrator sem o `RoleConfiguration`. O `RoleConfiguration` será removido do extrator, e o perfil vinculado ao serviço será usado.

Ao alterar entre os espaços de trabalho na mesma conta como o extrator, para continuar usando o `RoleConfiguration`, será necessário fornecer o `RoleConfiguration` novamente em `UpdateScraper`.

Criar extrator para espaços de trabalho habilitados com chaves gerenciadas pelo cliente

Para criar um extrator para ingerir métricas em um espaço de trabalho do Amazon Managed Service for Prometheus com chaves gerenciadas pelo cliente, use o `--role-configuration` com a origem e o destino definidos na mesma conta.

```
aws amp create-scraper \
  --source eksConfiguration="{clusterArn='arn:aws:eks:us-west-2:account-id:cluster/
xarw,subnetIds=[subnet-subnet_id]}" \
  --scrape-configuration configurationBlob=<base64-encoded-blob> \
  --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-west-2:account-
id:workspace/ws-workspace-id'}"\
  --role-configuration '{"sourceRoleArn":"arn:aws:iam::account_id:role/Source",
 "targetRoleArn":"arn:aws:iam::account_id:role/Target"}'
```

Erros comuns ao criar extratores

Veja a seguir os problemas mais comuns ao tentar criar um novo extrator.

- AWS Os recursos necessários não existem. O grupo de segurança, as sub-redes e o cluster do Amazon EKS especificados devem existir.
- Espaço insuficiente de endereços IP. Você deve ter pelo menos um endereço IP disponível em cada sub-rede que você transmite à API `CreateScraper`.

Configurar o cluster do Amazon EKS

O cluster do Amazon EKS deve ser configurado para permitir que o extrator acesse as métricas. Há duas opções para essa configuração:

- Use as entradas de acesso do Amazon EKS para fornecer automaticamente aos coletores do Amazon Managed Service for Prometheus acesso ao seu cluster.
- Configure manualmente seu cluster do Amazon EKS para extração de métricas gerenciadas.

Os tópicos a seguir descrevem cada uma delas em mais detalhes.

Configurar o Amazon EKS para acesso do extrator com entradas de acesso

Usar entradas de acesso do Amazon EKS é a maneira mais fácil de fornecer ao Amazon Managed Service for Prometheus acesso para extrair métricas do seu cluster.

O cluster do Amazon EKS do qual você faz a extração deve ser configurado de modo que permita a autenticação da API. O modo de autenticação do cluster deve ser definido como `API` ou `API_AND_CONFIG_MAP`. Isso pode ser visualizado no console do Amazon EKS na guia Configuração de acesso dos detalhes do cluster. Para obter mais informações, consulte o Como permitir a perfis do IAM ou usuários acesso ao objeto do Kubernetes no seu cluster do Amazon EKS no Guia do Usuário do Amazon EKS.

Você pode criar o extrator durante ou após a criação do cluster:

- Ao criar um cluster: você pode configurar esse acesso ao criar um cluster do Amazon EKS pelo console do Amazon EKS (siga as instruções para criar um extrator como parte do cluster), e uma política de entrada de acesso será criada automaticamente, concedendo ao Amazon Managed Service for Prometheus acesso às métricas do cluster.

- Adicionar após a criação de um cluster: se o seu cluster do Amazon EKS já existir, defina o modo de autenticação como `API` ou `API_AND_CONFIG_MAP`, e qualquer extrator que você criar pela CLI ou API do Amazon Managed Service for Prometheus ou pelo console do Amazon EKS terá automaticamente a política de entrada de acesso correta criada para você, e os extratores terão acesso ao seu cluster.

Política de entrada de acesso criada

Quando você cria um extrator e permite que o Amazon Managed Service for Prometheus gere uma política de entrada de acesso para você, ele gera a seguinte política. Para obter mais informações sobre entradas de acesso, consulte o Como permitir a perfis do IAM ou usuários acesso ao objeto do Kubernetes no Guia do Usuário do Amazon EKS.

```
{
    "rules": [
        {
            "effect": "allow",
            "apiGroups": [
                ""
            ],
            "resources": [
                "nodes",
                "nodes/proxy",
                "nodes/metrics",
                "services",
                "endpoints",
                "pods",
                "ingresses",
                "configmaps"
            ],
            "verbs": [
                "get",
                "list",
                "watch"
            ]
        },
        {
            "effect": "allow",
            "apiGroups": [
                "extensions",
                "networking.k8s.io"
            ],
```

```
            "resources": [
                "ingresses/status",
                "ingresses"
            ],
            "verbs": [
                "get",
                "list",
                "watch"
            ]
        },
        {
            "effect": "allow",
            "apiGroups": [
                "metrics.eks.amazonaws.com"
            ],
            "resources": [
                "kcm/metrics",
                "ksh/metrics"
            ],
            "verbs": [
                "get"
            ]
        },
        {
            "effect": "allow",
            "nonResourceURLs": [
                "/metrics"
            ],
            "verbs": [
                "get"
            ]
        }
    ]
}
```

Como configurar manualmente o Amazon EKS para acesso do extrator

Se você preferir usar o `aws-auth ConfigMap` para controlar o acesso ao seu cluster do Kubernetes, você ainda poderá conceder aos raspadores do Amazon Managed Service for Prometheus acesso às suas métricas. As etapas a seguir concederão ao Amazon Managed Service for Prometheus acesso para extrair métricas do seu cluster do Amazon EKS.

> **Note**
>
> Para obter mais informações sobre `ConfigMap` e entradas de acesso, consulte Como permitir a perfis do IAM ou usuários acesso ao Kubernetes no Guia do Usuário do Amazon EKS.

Este procedimento usa `kubectl` e a AWS CLI. Para obter informações sobre a instalação do `kubectl`, consulte Instalar o kubectl no Guia do usuário do Amazon EKS.

Para configurar manualmente seu cluster do Amazon EKS para extração de métricas gerenciadas

1. Crie um arquivo denominado `clusterrole-binding.yml` com o seguinte texto:

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: aps-collector-role
rules:
  - apiGroups: [""]
    resources: ["nodes", "nodes/proxy", "nodes/metrics", "services", "endpoints",
 "pods", "ingresses", "configmaps"]
    verbs: ["describe", "get", "list", "watch"]
  - apiGroups: ["extensions", "networking.k8s.io"]
    resources: ["ingresses/status", "ingresses"]
    verbs: ["describe", "get", "list", "watch"]
  - nonResourceURLs: ["/metrics"]
    verbs: ["get"]
  - apiGroups: ["metrics.eks.amazonaws.com"]
    resources: ["kcm/metrics", "ksh/metrics"]
    verbs: ["get"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: aps-collector-user-role-binding
subjects:
- kind: User
  name: aps-collector-user
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: ClusterRole
```

```
  name: aps-collector-role
  apiGroup: rbac.authorization.k8s.io
```

2. Execute o seguinte comando no cluster:

```
kubectl apply -f clusterrole-binding.yml
```

Isso criará a vinculação e a regra do perfil do cluster. Esse exemplo usa `aps-collector-role` como nome do perfil e `aps-collector-user` como nome do usuário.

3. O comando a seguir fornece informações sobre o raspador com o ID*scraper-id*. Esse é o extrator que você criou usando o comando na seção anterior.

```
aws amp describe-scraper --scraper-id scraper-id
```

4. Nos resultados do `describe-scraper`, encontre o `roleArn`. Ele terá o seguinte formato:

```
arn:aws:iam::account-id:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper_unique-id
```

O Amazon EKS requer um formato diferente para esse ARN. É necessário ajustar o formato do ARN retornado para ser usado na próxima etapa. Edite-o para corresponder a este formato:

```
arn:aws:iam::account-id:role/AWSServiceRoleForAmazonPrometheusScraper_unique-id
```

Por exemplo, este ARN:

```
arn:aws:iam::111122223333:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-56ef-7
```

Deve ser reescrito como:

```
arn:aws:iam::111122223333:role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-56ef-7
```

5. Execute o seguinte comando no cluster, usando o `roleArn` modificado da etapa anterior, bem como o nome e a região do cluster:

```
eksctl create iamidentitymapping --cluster cluster-name --region region-id --
arn roleArn --username aps-collector-user
```

Isso permite que o extrator acesse o cluster usando o perfil e o usuário que você criou no arquivo `clusterrole-binding.yml`.

Encontrar e excluir extratores

Você pode usar a AWS API ou a AWS CLI para listar os scrapers em sua conta ou excluí-los.

> **Note**
>
> Verifique se você está usando a versão mais recente do AWS CLI ou SDK. A versão mais recente do SDK fornece os recursos e as funcionalidades mais recentes, além de atualizações de segurança. Como alternativa, use AWS CloudShell, que fornece uma experiência sempre na linha de up-to-date comando, automaticamente.

Para listar todos os extratores na conta, use a operação ListScrapers da API.

Como alternativa, com o AWS CLI, ligue para:

```
aws amp list-scrapers --region aws-region
```

`ListScrapers` retorna todos os extratores da conta, por exemplo:

```
{
    "scrapers": [
        {
            "scraperId": "s-1234abcd-56ef-7890-abcd-1234ef567890",
            "arn": "arn:aws:aps:us-west-2:123456789012:scraper/s-1234abcd-56ef-7890-
abcd-1234ef567890",
            "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-2931",
            "status": {
                "statusCode": "DELETING"
            },
            "createdAt": "2023-10-12T15:22:19.014000-07:00",
            "lastModifiedAt": "2023-10-12T15:55:43.487000-07:00",
```

```
            "tags": {},
            "source": {
                "eksConfiguration": {
                    "clusterArn": "arn:aws:eks:us-west-2:123456789012:cluster/my-
cluster",
                    "securityGroupIds": [
                        "sg-1234abcd5678ef90"
                    ],
                    "subnetIds": [
                        "subnet-abcd1234ef567890",
                        "subnet-1234abcd5678ab90"
                    ]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:us-west-2:123456789012:workspace/
ws-1234abcd-5678-ef90-ab12-cdef3456a78"
                }
            }
        }
    ]
}
```

Para excluir um extrator, localize o `scraperId` do extrator que deseja excluir usando a operação `ListScrapers` e, em seguida, use a operação DeleteScraper para excluí-lo.

Como alternativa, com o AWS CLI, ligue para:

```
aws amp delete-scraper --scraper-id scraperId
```

Configuração do extrator

É possível controlar como o extrator descobre e coleta métricas com uma configuração de extrator compatível com o Prometheus. Por exemplo, é possível alterar o intervalo em que as métricas são enviadas para o espaço de trabalho, além de usar a nova rotulagem para reescrever dinamicamente os rótulos de uma métrica. A configuração do extrator é um arquivo YAML que faz parte da definição do extrator.

Quando um novo extrator é criado, você especifica uma configuração fornecendo um arquivo YAML codificado em base64 na chamada de API. É possível baixar um arquivo de configuração de uso

geral com a operação `GetDefaultScraperConfiguration` na API do Amazon Managed Service for Prometheus.

Para modificar a configuração de um extrator, você pode usar a operação `UpdateScraper`. Se precisar atualizar a fonte das métricas (por exemplo, para um cluster diferente do Amazon EKS), você deve excluir o extrator e recriá-lo com a nova origem.

Configurações aceitas

Para obter mais informações sobre o formato de configuração do extrator, incluindo uma análise detalhada dos valores possíveis, consulte Configuração na documentação do Prometheus. As opções de configuração global e do `<scrape_config>` descrevem as opções mais comumente necessárias.

Como o Amazon EKS é o único serviço compatível, a única configuração de descoberta de serviços (`<*_sd_config>`) aceita é a `<kubernetes_sd_config>`.

A lista completa de seções de configuração permitidas:

- `<global>`
- `<scrape_config>`
- `<static_config>`
- `<relabel_config>`
- `<metric_relabel_configs>`
- `<kubernetes_sd_config>`

As limitações dessas seções são listadas após o arquivo de configuração de amostra.

Arquivo de configuração de exemplo

Veja a seguir um exemplo de arquivo de configuração YAML com um intervalo de extração de 30 segundos. Esse exemplo inclui suporte para as métricas do servidor da API kube, bem como kube-controller-manager para as métricas do kube-scheduler. Para obter mais informações, consulte Obter as métricas brutas do ambiente de gerenciamento no formato do Prometheus no Guia do usuário do Amazon EKS.

```
global:
   scrape_interval: 30s
   external_labels:
```

```
     clusterArn: apiserver-test-2
scrape_configs:
  - job_name: pod_exporter
    kubernetes_sd_configs:
      - role: pod
  - job_name: cadvisor
    scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    kubernetes_sd_configs:
      - role: node
    relabel_configs:
      - action: labelmap
        regex: __meta_kubernetes_node_label_(.+)
      - replacement: kubernetes.default.svc:443
        target_label: __address__
      - source_labels: [__meta_kubernetes_node_name]
        regex: (.+)
        target_label: __metrics_path__
        replacement: /api/v1/nodes/$1/proxy/metrics/cadvisor
  # apiserver metrics
  - scheme: https
    authorization:
      type: Bearer
      credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    job_name: kubernetes-apiservers
    kubernetes_sd_configs:
    - role: endpoints
    relabel_configs:
    - action: keep
      regex: default;kubernetes;https
      source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
  # kube proxy metrics
  - job_name: kube-proxy
    honor_labels: true
    kubernetes_sd_configs:
    - role: pod
    relabel_configs:
    - action: keep
      source_labels:
```

```
      - __meta_kubernetes_namespace
      - __meta_kubernetes_pod_name
      separator: '/'
      regex: 'kube-system/kube-proxy.+'
    - source_labels:
      - __address__
      action: replace
      target_label: __address__
      regex: (.+?)(\\:\\d+)?
      replacement: $1:10249
  # Scheduler metrics
  - job_name: 'ksh-metrics'
    kubernetes_sd_configs:
    - role: endpoints
    metrics_path: /apis/metrics.eks.amazonaws.com/v1/ksh/container/metrics
    scheme: https
    bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    relabel_configs:
    - source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
      action: keep
      regex: default;kubernetes;https
  # Controller Manager metrics
  - job_name: 'kcm-metrics'
    kubernetes_sd_configs:
    - role: endpoints
    metrics_path: /apis/metrics.eks.amazonaws.com/v1/kcm/container/metrics
    scheme: https
    bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
    relabel_configs:
    - source_labels:
      - __meta_kubernetes_namespace
      - __meta_kubernetes_service_name
      - __meta_kubernetes_endpoint_port_name
      action: keep
      regex: default;kubernetes;https
```

A seguir estão as limitações específicas dos coletores AWS gerenciados:

- Intervalo de extração: a configuração do extrator não pode especificar um intervalo de extração inferior a 30 segundos.

- Destinos: os destinos no `static_config` devem ser especificados como endereços IP.
- Resolução de DNS: relacionado ao nome de destino, o único nome de servidor reconhecido nessa configuração é o servidor da API do Kubernetes, `kubernetes.default.svc`. Todos os outros nomes de máquinas devem ser especificados por endereço IP.
- Autorização: omita se nenhuma autorização for necessária. Se for necessária, a autorização deve ser `Bearer` e deve apontar para o arquivo `/var/run/secrets/kubernetes.io/serviceaccount/token`. Em outras palavras, se usada, a seção de autorização deverá ter o seguinte aspecto:

```
authorization:
  type: Bearer
  credentials_file: /var/run/secrets/kubernetes.io/serviceaccount/token
```

> **Note**
>
> `type: Bearer` é o padrão, então pode ser omitido.

Solução de problemas de configuração do extrator

Coletores do Amazon Managed Service for Prometheus descobrem e extraem métricas automaticamente. Mas como você pode solucionar problemas quando não vê uma métrica que espera ver no espaço de trabalho do Amazon Managed Service for Prometheus?

> **Important**
>
> Verifique se o acesso privado ao cluster do Amazon EKS está habilitado. Para obter mais informações, consulte Endpoint privado do cluster no Guia do usuário do Amazon EKS.

A métrica up é uma ferramenta útil. Para cada endpoint que um coletor do Amazon Managed Service for Prometheus descobre, ele vende automaticamente essa métrica. Há três estados dessa métrica que podem ajudar você a solucionar o que está acontecendo no coletor.

- up não está presente: se não houver nenhuma métrica up presente para um endpoint, isso significa que o coletor não conseguiu encontrar o endpoint.

Se você tiver certeza de que o endpoint existe, há vários motivos pelos quais o coletor pode não conseguir encontrá-lo.

- Talvez seja necessário ajustar a configuração de extração. Talvez a descoberta `relabel_config` precise ser ajustada.
- Pode haver um problema com o `role` usado para descoberta.
- A Amazon VPC usada pelo cluster Amazon EKS pode não ter o DNS habilitado, o que impediria que o coletor encontrasse o endpoint.

- `up` está presente, mas é sempre 0: se `up` estiver presente, mas for 0, o coletor poderá descobrir o endpoint, mas não encontrará nenhuma métrica compatível com o Prometheus.

  Nesse caso, você pode tentar usar um comando `curl` diretamente no endpoint. Você pode validar se os detalhes estão corretos, por exemplo, o protocolo (`http` ou `https`), o endpoint ou a porta que você está usando. Você também pode verificar se o endpoint está respondendo com uma resposta `200` válida e segue o formato do Prometheus. Finalmente, o corpo da resposta não pode ser maior do que o tamanho máximo permitido. (Para ver os limites dos coletores AWS gerenciados, consulte a seção a seguir.)
- `up` está presente e é maior que 0: se `up` estiver presente e for maior que 0, as métricas serão enviadas para o Amazon Managed Service for Prometheus.

  Verifique se você está procurando as métricas corretas no Amazon Managed Service for Prometheus (ou no painel alternativo, como Amazon Managed Grafana). É possível usar o curl novamente para verificar os dados esperados no endpoint do `/metrics`. Verifique também se você não excedeu outros limites, como o número de endpoints por extrator. Você pode verificar o número de endpoints de métricas passando pela extração ao verificar a contagem de métricas `up`, usando `count(up)`.

Limitações do extrator

Há poucas limitações nos extratores totalmente gerenciados fornecidos pelo Amazon Managed Service for Prometheus.

- Região: o cluster do EKS, o extrator gerenciado e o espaço de trabalho do Amazon Managed Service for Prometheus devem estar todos na mesma região da AWS .
- Coletores: é possível ter no máximo 10 extratores do Amazon Managed Service for Prometheus por região e por conta.

> **Note**
>
> É possível solicitar um aumento para esse limite solicitando um aumento de cota.

- Resposta de métricas: o corpo de uma resposta de qualquer solicitação de endpoint do `/metrics` não pode ter mais de 50 megabytes (MB).
- Endpoints por extrator: um extrator pode extrair no máximo 30.000 endpoints do `/metrics`.
- Intervalo de extração: a configuração do extrator não pode especificar um intervalo de extração inferior a 30 segundos.

Configure coletores gerenciados do Prometheus para o Amazon MSK

Para usar um coletor do Amazon Managed Service for Prometheus, crie um extrator que descubra e extraia métricas no cluster do Amazon Streaming gerenciado para Apache Kafka. É possível também criar um extrator que se integre ao Amazon Elastic Kubernetes Service. Para obter mais informações, consulte Integrar o Amazon EKS.

Criar um extrator

Um coletor do Amazon Managed Service for Prometheus consiste em um extrator que descobre e coleta métricas de um cluster do Amazon MKS. O Amazon Managed Service for Prometheus gerencia o extrator para você, fornecendo a escalabilidade, a segurança e a confiabilidade necessárias, sem que você precise gerenciar instâncias, agentes ou extratores por conta própria.

Você pode criar um raspador usando a AWS API ou a AWS CLI conforme descrito nos procedimentos a seguir.

Há alguns pré-requisitos para a criação de um extrator próprio:

- É necessário ter um cluster do Amazon MKS.
- Configure o grupo de segurança do seu cluster Amazon MSK para permitir tráfego de entrada nas portas 11001 (JMX Exporter) e 11002 (Node Exporter) em sua Amazon VPC, pois o extrator exige acesso a esses registros DNS para coletar métricas do Prometheus.
- A Amazon VPC na qual o cluster Amazon MKS reside deve ter o DNS habilitado.

> **Note**
>
> O cluster será associado ao extrator pelo nome do recurso da Amazon (ARN). Se você excluir um cluster e criar um novo com o mesmo nome, o ARN será reutilizado para o novo cluster. Por esse motivo, o extrator tentará coletar métricas para o novo cluster. Você exclui extratores separadamente da exclusão do cluster.

To create a scraper using the AWS API

Use a operação `CreateScraper` da API para criar um raspador com a AWS API. O exemplo a seguir cria um extrator na Região Leste dos EUA (Norte da Virgínia). Substitua o *`example`* conteúdo pelas informações do cluster do Amazon MSK e forneça a configuração do seu scraper.

> **Note**
>
> Configure o grupo de segurança e as sub-redes para corresponder ao seu cluster de destino. Inclua pelo menos duas sub-redes em duas zonas de disponibilidade.

```
               POST /scrapers HTTP/1.1
Content-Length: 415
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: aws-cli/1.18.147 Python/2.7.18 Linux/5.4.58-37.125.amzn2int.x86_64
 botocore/1.18.6

{
    "alias": "myScraper",
    "destination":  {
        "ampConfiguration": {
            "workspaceArn": "arn:aws:aps:us-east-1:123456789012:workspace/ws-
workspace-id"
        }
    },
    "source": {
        "vpcConfiguration": {
            "securityGroupIds": ["sg-security-group-id"],
            "subnetIds": ["subnet-subnet-id-1", "subnet-subnet-id-2"]
        }
```

```
    },
    "scrapeConfiguration": {
        "configurationBlob": base64-encoded-blob
    }
}
```

No exemplo, o parâmetro `scrapeConfiguration` exige um arquivo YAML de configuração do Prometheus codificado em base64 que especifique os registros DNS do cluster MSK.

Cada registro DNS representa um endpoint de corretor em uma zona de disponibilidade específica, permitindo que os clientes se conectem a corretores distribuídos por sua escolha AZs para alta disponibilidade.

O número de registros DNS nas propriedades do cluster MSK corresponde ao número de nós de agente e zonas de disponibilidade na configuração do cluster:

- Configuração padrão — 3 nós de intermediário em 3 AZs = 3 registros DNS
- Configuração personalizada — 2 nós de intermediário em 2 AZs = 2 registros DNS

Para obter os registros DNS do seu cluster MSK, abra o console MSK em casa? https://console.aws.amazon.com/msk/ region=us-east-1#/home/. Acesse o cluster do MSK. Selecione Propriedades, Agentes e Endpoints.

Você tem duas opções para configurar o Prometheus para extrair métricas do seu cluster MSK:

1. Resolução de DNS em nível de cluster (recomendada): use o nome DNS base do cluster para descobrir automaticamente todos os agentes. Se o endpoint do seu agente for `b-1.clusterName.xxx.xxx.xxx`, use `clusterName.xxx.xxx.xxx` como registro DNS. Isso permite que o Prometheus extraia automaticamente todos os agentes no cluster.

   Endpoints individuais de agente: especifique cada endpoint do agente individualmente para controle granular. Use os identificadores completos do agente (b-1, b-2) em sua configuração. Por exemplo:

   ```
   dns_sd_configs:
     - names:
       - b-1.clusterName.xxx.xxx.xxx
       - b-2.clusterName.xxx.xxx.xxx
       - b-3.clusterName.xxx.xxx.xxx
   ```

> **Note**
>
> `clusterName.xxx.xxx.xxx`Substitua pelo endpoint real do cluster MSK a partir do AWS console.

Para obter mais informações, consulte <dns_sd_config> na documentação do Prometheus.

Este é um exemplo do arquivo de configuração do extrator:

```
global:
  scrape_interval: 30s
  external_labels:
    clusterArn: msk-test-1

scrape_configs:
  - job_name: msk-jmx
    scheme: http
    metrics_path: /metrics
    scrape_timeout: 10s
    dns_sd_configs:
      - names:
          - dns-record-1
          - dns-record-2
          - dns-record-3
        type: A
        port: 11001
    relabel_configs:
      - source_labels: [__meta_dns_name]
        target_label: broker_dns
      - source_labels: [__address__]
        target_label: instance
        regex: '(.*)'
        replacement: '${1}'

  - job_name: msk-node
    scheme: http
    metrics_path: /metrics
    scrape_timeout: 10s
    dns_sd_configs:
      - names:
          - dns-record-1
          - dns-record-2
```

```
          - dns-record-3
        type: A
        port: 11002
    relabel_configs:
      - source_labels: [__meta_dns_name]
        target_label: broker_dns
      - source_labels: [__address__]
        target_label: instance
        regex: '(.*)'
        replacement: '${1}'
```

Execute um dos comandos a seguir para converter o arquivo YAML em base64. Você também pode usar qualquer conversor base64 online para converter o arquivo.

Example Linux/macOS

```
echo -n scraper config updated with dns records | base64
```

Example Windows PowerShell

```
[Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes(scraper config
 updated with dns records))
```

To create a scraper using the AWS CLI

Use o comando `create-scraper` para criar um extrator usando o AWS Command Line Interface. O exemplo a seguir cria um extrator na Região Leste dos EUA (Norte da Virgínia). Substitua o *example* conteúdo pelas informações do cluster do Amazon MSK e forneça a configuração do seu scraper.

> **Note**
>
> Configure o grupo de segurança e as sub-redes para corresponder ao seu cluster de destino. Inclua pelo menos duas sub-redes em duas zonas de disponibilidade.

```
aws amp create-scraper \
 --source vpcConfiguration="{securityGroupIds=['sg-security-group-
id'],subnetIds=['subnet-subnet-id-1', 'subnet-subnet-id-2']}" \
--scrape-configuration configurationBlob=base64-encoded-blob \
```

```
 --destination ampConfiguration="{workspaceArn='arn:aws:aps:us-
west-2:123456789012:workspace/ws-workspace-id'}"
```

- Veja a seguir uma lista completa das operações do raspador que você pode usar com a AWS API:

  Crie um extrator com a operação CreateScraper da API.
- Liste os extratores existentes com a operação ListScrapers da API.
- Atualize o alias, a configuração ou o destino de um raspador com a operação da UpdateScraperAPI.
- Exclua um extrator com a operação DeleteScraper da API.
- Obtenha mais detalhes sobre um extrator com a operação DescribeScraper da API.

Configuração entre contas

Para criar um extrator em uma configuração entre contas quando o cluster Amazon MSK do qual você deseja coletar métricas estiver em uma conta diferente do coletor Amazon Managed Service for Prometheus, use o procedimento abaixo.

Por exemplo, quando você tem duas contas, a primeira conta de origem `account_id_source` na qual o Amazon MSK está localizado e uma segunda conta de destino `account_id_target` na qual reside o espaço de trabalho do Amazon Managed Service for Prometheus.

Como criar um extrator em uma configuração entre contas

1. Na conta de origem, crie um perfil `arn:aws:iam::111122223333:role/Source` e adicione a política de confiança a seguir.

```
{
    "Effect": "Allow",
    "Principal": {
    "Service": [
        "scraper.aps.amazonaws.com"
     ]
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "arn:aws:aps:aws-region:111122223333:scraper/scraper-
id"
```

```
        },
        "StringEquals": {
            "AWS:SourceAccount": "111122223333"
        }
    }
}
```

2. Em cada combinação de origem (cluster Amazon MSK) e destino (espaço de trabalho do Amazon Managed Service for Prometheus), você precisa criar uma `arn:aws:iam::444455556666:role/Target` função e adicionar a seguinte política de confiança com permissões para. AmazonPrometheusRemoteWriteAccess

```
{
  "Effect": "Allow",
  "Principal": {
     "AWS": "arn:aws:iam::111122223333:role/Source"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
     "StringEquals": {
        "sts:ExternalId": "arn:aws:aps:aws-region:111122223333:scraper/scraper-id"
      }
  }
}
```

3. Crie um extrator com a opção `--role-configuration`.

```
aws amp create-scraper \ --source vpcConfiguration="{subnetIds=[subnet-
subnet-id], "securityGroupIds": ["sg-security-group-id"]}" \ --
scrape-configuration configurationBlob=<base64-encoded-blob> \
 --destination ampConfiguration="{workspaceArn='arn:aws:aps:aws-
region:444455556666:workspace/ws-workspace-id'}"\ --role-configuration
 '{"sourceRoleArn":"arn:aws:iam::111122223333:role/Source",
 "targetRoleArn":"arn:aws:iam::444455556666:role/Target"}'
```

4. Valide a criação do extrator.

```
aws amp list-scrapers
{
```

```
    "scrapers": [
        {
            "scraperId": "s-example123456789abcdef0",
            "arn": "arn:aws:aps:aws-region:111122223333:scraper/s-
example123456789abcdef0": "arn:aws:iam::111122223333:role/Source",
            "status": "ACTIVE",
            "creationTime": "2025-10-27T18:45:00.000Z",
            "lastModificationTime": "2025-10-27T18:50:00.000Z",
            "tags": {},
            "statusReason": "Scraper is running successfully",
            "source": {
                "vpcConfiguration": {
                    "subnetIds": ["subnet-subnet-id"],
                    "securityGroupIds": ["sg-security-group-id"]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:aws-region:444455556666:workspace/
ws-workspace-id'"
                }
            },
            "scrapeConfiguration": {
                "configurationBlob": "<base64-encoded-blob>"
            }
        }
    ]
}
```

Alteração entre uma RoleConfiguration função vinculada ao serviço

Quando quiser voltar para um perfil vinculado ao serviço em vez de `RoleConfiguration` para gravar em um espaço de trabalho do Amazon Managed Service for Prometheus, você deve atualizar o `UpdateScraper` e fornecer um espaço de trabalho na mesma conta do extrator sem o `RoleConfiguration`. O `RoleConfiguration` será removido do extrator, e o perfil vinculado ao serviço será usado.

Ao alterar entre os espaços de trabalho na mesma conta como o extrator, para continuar usando o `RoleConfiguration`, será necessário fornecer o `RoleConfiguration` novamente em `UpdateScraper`.

Encontrar e excluir extratores

Você pode usar a AWS API ou a AWS CLI para listar os scrapers em sua conta ou excluí-los.

> **Note**
>
> Verifique se você está usando a versão mais recente do AWS CLI ou SDK. A versão mais recente do SDK fornece os recursos e as funcionalidades mais recentes, além de atualizações de segurança. Como alternativa, use AWS CloudShell, que fornece uma experiência sempre na linha de up-to-date comando, automaticamente.

Para listar todos os extratores na conta, use a operação ListScrapers da API.

Como alternativa, com o AWS CLI, ligue para:

```
aws amp list-scrapers
```

`ListScrapers` retorna todos os extratores da conta, por exemplo:

```
{
    "scrapers": [
        {
            "scraperId": "s-1234abcd-56ef-7890-abcd-1234ef567890",
            "arn": "arn:aws:aps:aws-region:123456789012:scraper/s-1234abcd-56ef-7890-
abcd-1234ef567890",
            "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
AWSServiceRoleForAmazonPrometheusScraper_1234abcd-2931",
            "status": {
                "statusCode": "DELETING"
            },
            "createdAt": "2023-10-12T15:22:19.014000-07:00",
            "lastModifiedAt": "2023-10-12T15:55:43.487000-07:00",
            "tags": {},
            "source": {
                "vpcConfiguration": {
                   "securityGroupIds": [
```

```
                        "sg-1234abcd5678ef90"
                    ],
                    "subnetIds": [
                        "subnet-abcd1234ef567890",
                        "subnet-1234abcd5678ab90"
                    ]
                }
            },
            "destination": {
                "ampConfiguration": {
                    "workspaceArn": "arn:aws:aps:aws-region:123456789012:workspace/
ws-1234abcd-5678-ef90-ab12-cdef3456a78"
                }
            }
        }
    ]
}
```

Para excluir um extrator, localize o `scraperId` do extrator que deseja excluir usando a operação `ListScrapers` e, em seguida, use a operação DeleteScraper para excluí-lo.

Como alternativa, com o AWS CLI, ligue para:

```
aws amp delete-scraper --scraper-id scraperId
```

Métricas coletadas do Amazon MSK

Quando você faz a integração com o Amazon MSK, o coletor do Amazon Managed Service for Prometheus coleta automaticamente as seguintes métricas:

Métricas: trabalhos jmx_exporter e pod_exporter

Métrica	Descrição e Objetivo
jmx_config_reload_failure_total	Número total de vezes que o exportador JMX falhou ao recarregar seu arquivo de configura ção.
jmx_scrape_duration_seconds	Tempo gasto para coletar métricas do JMX em segundos para o ciclo de coleta atual.

Métrica	Descrição e Objetivo
jmx_scrape_error	Indica se ocorreu um erro durante a coleta da métrica JMX (1 = erro, 0 = sucesso).
Java_lang_memória_ _usado HeapMemoryUsage	Quantidade de memória heap (em bytes) usada atualmente pela JVM.
Java_lang_memória_ _max HeapMemoryUsage	Quantidade máxima de memória heap (em bytes) que pode ser usada para gerenciamento de memória.
Java_lang_memória_ _usado NonHeapMemoryUsage	Quantidade de memória não heap (em bytes) usada atualmente pela JVM.
kafka_cluster_Partition_Value	Estado ou valor atual relacionado às partições de cluster do Kafka, detalhado por ID de partição e tópico.
kafka_consumer_consumer_coordinator_metrics_assigned_partitions	Número de partições atualmente atribuídas a esse consumidor.
kafka_consumer_consumer_coordinator_metrics_commit_latency_avg	Tempo médio necessário para confirmar compensações em milissegundos.
kafka_consumer_consumer_coordinator_metrics_commit_rate	Número de confirmações de deslocamento por segundo.
kafka_consumer_consumer_coordinator_metrics_failed_rebalance_total	Número total de reequilíbrios de grupos de consumidores malsucedidos.
kafka_consumer_consumer_coordinator_metrics_last_heartbeat_seconds_ago	Número de segundos desde que a última pulsação foi enviada ao coordenador.
kafka_consumer_consumer_coordinator_metrics_rebalance_latency_avg	Tempo médio gasto para rebalancear grupos de consumidores em milissegundos.
kafka_consumer_consumer_coordinator_metrics_rebalance_total	Número total de reequilíbrios de grupos de consumidores.

Métrica	Descrição e Objetivo
kafka_consumer_consumer_fetch_manager_metrics_bytes_consumed_rate	Número médio de bytes consumidos por segundo pelo consumidor.
kafka_consumer_consumer_fetch_manager_metrics_fetch_latency_avg	Tempo médio gasto para uma solicitação de busca em milissegundos.
kafka_consumer_consumer_fetch_manager_metrics_fetch_rate	Número de solicitações de busca por segundo.
kafka_consumer_consumer_fetch_manager_metrics_records_consumed_rate	O número médio de registros consumidos por segundo.
kafka_consumer_consumer_fetch_manager_metrics_records_lag_max	O atraso máximo em termos de número de registros para qualquer partição desse consumidor.
kafka_consumer_consumer_metrics_connection_count	Número atual de conexões ativas.
kafka_consumer_consumer_metrics_incoming_byte_rate	Número médio de bytes recebidos por segundo de todos os servidores.
kafka_consumer_consumer_metrics_last_poll_seconds_ago	Número de segundos desde a última chamada do consumidor poll().
kafka_consumer_consumer_metrics_request_rate	Número de solicitações enviadas por segundo.
kafka_consumer_consumer_metrics_response_rate	Número de respostas recebidas por segundo.
kafka_consumer_group_ _Valor ConsumerLagMetrics	Valor atual do atraso do consumidor para um grupo de consumidores, indicando o quanto o consumidor está atrasado.
kafka_controller_ _Valor KafkaController	Estado ou valor atual do controlador Kafka (1 = controlador ativo, 0 = não ativo).

Métrica	Descrição e Objetivo
kafka_controller_ _Contagem ControllerEventManager	Número total de eventos de controlador processados.
kafka_controller_ _Mean ControllerEventManager	Tempo médio gasto para processar eventos do controlador.
controlador_kafka_ _ ControllerStats MeanRate	Taxa média de operações de estatística de controlador por segundo.
kafka_coordinator_group_ _Valor GroupMetadataManager	Estado ou valor atual do gerenciador de metadados do grupo para grupos de consumidores.
kafka_log_ _Contagem LogFlushStats	Número total de operações de descarga de log.
kafka_log_ _Mean LogFlushStats	Tempo médio gasto com operações de descarga de log.
kafka_log_ _ LogFlushStats MeanRate	Taxa média de operações de descarga de log por segundo.
kafka_network_ _Contagem RequestMetrics	Contagem total de solicitações de rede processadas.
kafka_network_ _Mean RequestMetrics	Tempo médio gasto para processar solicitações de rede.
rede_kafka_ _ RequestMetrics MeanRate	Taxa média de solicitações de rede por segundo.
Kafka_Network_Acceptor_ MeanRate	Taxa média de conexões aceitas por segundo.
kafka_server_Fetch_queue_size	Tamanho atual da fila de solicitações de busca.
kafka_server_Produce_queue_size	Tamanho atual da fila de solicitações de produção.
kafka_server_Request_queue_size	Tamanho atual da fila de solicitações gerais.

Métrica	Descrição e Objetivo
kafka_server_ _Contagem BrokerTopicMetrics	Contagem total de operações de tópicos da corretora (mensagensin/out, bytes in/out).
servidor_kafka_ _ BrokerTopicMetrics MeanRate	Taxa média de operações de tópico de agente por segundo.
servidor_kafka_ _ BrokerTopicMetrics OneMinuteRate	Taxa média móvel de um minuto das operações temáticas do agente.
kafka_server_ _Valor DelayedOperationPu rgatory	Número atual de operações atrasadas no purgatório (esperando para serem concluídas).
servidor_kafka_ _ DelayedFetchMetrics MeanRate	Taxa média de operações de busca atrasadas por segundo.
kafka_server_ _Valor FetcherLagMetrics	Valor atual do atraso para threads de busca de réplicas (o quanto está atrás do líder).
servidor_kafka_ _ FetcherStats MeanRate	Taxa média de operações de busca por segundo.
kafka_server_ _Valor ReplicaManager	Estado ou valor atual do gerenciador de réplicas.
servidor_kafka_ _ ReplicaManager MeanRate	Taxa média de operações do gerenciador de réplicas por segundo.
servidor_kafka_ _byte_rate LeaderReplication	Taxa de bytes replicados por segundo para partições em que esse agente é o líder.
kafka_server_group_coordinator_metri cs_group_completed_rebalance_count	Número total de reequilíbrios concluídos de grupos de consumidores.
kafka_server_group_coordinator_metrics_offset _commit_count	Número total de operações de confirmação de deslocamento.
kafka_server_group_coordinator_metrics_offset _commit_rate	Taxa de operações de confirmação de deslocamento por segundo.

Métrica	Descrição e Objetivo
kafka_server_socket_server_metrics_connection_count	Número atual de conexões ativas.
kafka_server_socket_server_metrics_connection_creation_rate	Taxa de criação de novas conexões por segundo.
kafka_server_socket_server_metrics_connection_close_rate	Taxa de fechamentos de conexão por segundo.
kafka_server_socket_server_metrics_failed_authentication_total	Número total de tentativas de autenticação malsucedidas.
kafka_server_socket_server_metrics_incoming_byte_rate	Taxa de entrada de bytes por segundo.
kafka_server_socket_server_metrics_outgoing_byte_rate	Taxa de bytes de saída por segundo.
kafka_server_socket_server_metrics_request_rate	Taxa de solicitações por segundo.
kafka_server_socket_server_metrics_response_rate	Taxa de respostas por segundo.
kafka_server_socket_server_metrics_network_io_rate	Taxa de I/O operações de rede por segundo.
kafka_server_socket_server_metrics_io_ratio	Fração do tempo gasto em I/O operações.
kafka_server_controller_channel_metrics_connection_count	Número atual de conexões ativas para canais do controlador.
kafka_server_controller_channel_metrics_incoming_byte_rate	Taxa de bytes de entrada por segundo para canais do controlador.
kafka_server_controller_channel_metrics_outgoing_byte_rate	Taxa de bytes enviados por segundo para os canais do controlador.

Métrica	Descrição e Objetivo
kafka_server_controller_channel_metrics_request_rate	Taxa de solicitações por segundo para os canais do controlador.
kafka_server_replica_fetcher_metrics_connection_count	Número atual de conexões ativas para o buscador de réplicas.
kafka_server_replica_fetcher_metrics_incoming_byte_rate	Taxa de bytes de entrada por segundo para o buscador de réplicas.
kafka_server_replica_fetcher_metrics_request_rate	Taxa de solicitações por segundo para o buscador de réplicas.
kafka_server_replica_fetcher_metrics_failed_authentication_total	Número total de tentativas de autenticação malsucedidas para o coletor de réplicas.
kafka_server_ _Contagem ZooKeeperClientMetrics	Contagem total das operações ZooKeeper do cliente.
kafka_server_ _Mean ZooKeeperClientMetrics	Latência média das operações do ZooKeeper cliente.
kafka_server_ _Valor KafkaServer	Estado ou valor atual do servidor Kafka (normalmente indica que o servidor está em execução).
node_cpu_seconds_total	Total de segundos CPUs gastos em cada modo (usuário, sistema, inatividade etc.), dividido por CPU e modo.
node_disk_read_bytes_total	Número total de bytes lidos com sucesso dos discos, dividido por dispositivo.
node_disk_reads_completed_total	Número total de bytes lidos com sucesso dos discos, dividido por dispositivo.
node_disk_writes_completed_total	Número total de gravações concluídas com sucesso para discos, dividido por dispositivo.

Métrica	Descrição e Objetivo
node_disk_written_bytes_total	Número total de bytes gravados com sucesso nos discos, dividido por dispositivo.
node_filesystem_avail_bytes	Espaço disponível do sistema de arquivos em bytes para usuários que não são usuários-raiz, dividido por dispositivo e ponto de montagem.
node_filesystem_size_bytes	Tamanho total do sistema de arquivos em bytes, dividido por dispositivo e ponto de montagem.
node_filesystem_free_bytes	Espaço livre do sistema de arquivos, em bytes, dividido por dispositivo e ponto de montagem.
node_filesystem_files	Número total de nós de arquivo (inodes) no sistema de arquivos, dividido por dispositivo e ponto de montagem.
node_filesystem_files_free	Número de nós de arquivo livres (inodes) no sistema de arquivos, dividido por dispositivo e ponto de montagem.
node_filesystem_readonly	Indica se o sistema de arquivos está montado somente para leitura (1 = somente leitura, 0 = leitura-gravação).
node_filesystem_device_error	Indica se ocorreu um erro ao obter estatísticas do sistema de arquivos (1 = erro, 0 = sucesso).

Limitações

A integração atual do Amazon MSK com o Amazon Managed Service for Prometheus tem as seguintes limitações:

- Compatível somente com clusters provisionados do Amazon MSK (não disponível para o Amazon MSK Serverless)

- Não é compatível com clusters Amazon MSK com acesso público habilitado em combinação com o modo de KRaft metadados
- Não é compatível com os agentes do Amazon MSK Express
- Atualmente, oferece suporte a um mapeamento 1:1 entre os clusters do Amazon MSK e os coletores/espaços de trabalho do Amazon Managed Service for Prometheus

O que são métricas compatíveis com o Prometheus?

Para extrair métricas do Prometheus de suas aplicações e infraestrutura para uso no Amazon Managed Service for Prometheus, é necessário instrumentar e expor métricas compatíveis com o Prometheus a partir de endpoints do `/metrics` compatíveis com o Prometheus. É possível implementar suas próprias métricas, mas não é necessário. O Kubernetes (incluindo o Amazon EKS) e muitas outras bibliotecas e serviços implementam essas métricas diretamente.

Quando as métricas no Amazon EKS são exportadas para um endpoint compatível com o Prometheus, é possível fazer com que elas sejam extraídas automaticamente pelo coletor do Amazon Managed Service for Prometheus.

Para saber mais, consulte os seguintes tópicos:

- Para obter mais informações sobre bibliotecas e serviços existentes que exportam métricas como as do Prometheus, consulte Exporters and integrations na documentação do Prometheus.
- Para obter mais informações sobre como exportar métricas compatíveis com o Prometheus a partir do seu próprio código, consulte Writing exporters na documentação do Prometheus.
- Para obter mais informações sobre como configurar um coletor do Amazon Managed Service for Prometheus para extrair métricas dos clusters do Amazon EKS automaticamente, consulte Configurar coletores gerenciados para o Amazon EKS.

Monitorar coletores com logs fornecidos

Os coletores do Amazon Managed Service for Prometheus oferecem logs fornecidos para ajudar você a monitorar e solucionar problemas no processo de coleta de métricas. Esses registros são enviados automaticamente para o Amazon CloudWatch Logs e fornecem visibilidade das operações de descoberta de serviços, coleta de métricas e exportação de dados. O coletor fornece logs para três componentes principais do pipeline de coleta de métricas:

Tópicos

- Logs de descoberta de serviços
- Logs do coletor
- Logs do exportador
- Entender e usar logs fornecidos por coletores

Logs de descoberta de serviços

Os logs de descoberta de serviços fornecem informações sobre o processo de descoberta de destino, incluindo:

- Problemas de autenticação ou permissão ao acessar os recursos da API Kubernetes.
- Erros de configuração nas definições de descoberta de serviços.

Os exemplos a seguir demonstram erros comuns de autenticação e permissão na descoberta de serviços:

Cluster Amazon EKS inexistente

Quando o cluster do Amazon EKS especificado não existe, você recebe o seguinte erro:

```
{
  "component": "SERVICE_DISCOVERY",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "log": "Failed to watch Service - Verify your scraper source exists."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Permissões inválidas para serviços

Quando o coletor não tem as permissões de controle de acesso baseado em perfil (RBAC) adequadas para monitorar os serviços, você recebe este erro:

```
{
  "component": "SERVICE_DISCOVERY",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
```

```
    "log": "Failed to watch Service - Verify your scraper source permissions are
 valid."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Permissões inválidas para endpoints

Quando o coletor não tem as permissões de controle de acesso baseado em perfil (RBAC) adequadas para monitorar os endpoints, você recebe este erro:

```
{
  "component": "SERVICE_DISCOVERY",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "log": "Failed to watch Endpoints - Verify your scraper source permissions are
 valid."
  },
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Logs do coletor

Os logs do coletor fornecem informações sobre o processo de extração métrica, incluindo:

- Falhas de extração devido à indisponibilidade dos endpoints.
- Problemas de conexão ao tentar extrair destinos.
- Tempos limite durante as operações de extração.
- Erros de status HTTP retornados por destinos de extração.

Os exemplos a seguir demonstram erros comuns do coletor que você encontra durante o processo de coleta de métricas:

Endpoint de métricas ausente

Quando o endpoint `/metrics` não está disponível na instância de destino, você recebe este erro:

```
{
    "component": "COLLECTOR",
    "message": {
```

```
        "log": "Failed to scrape Prometheus endpoint - verify /metrics endpoint is
 available",
        "job": "pod_exporter",
        "targetLabels": "{__name__=\"up\", instance=\10.24.34.0\", job=
\"pod_exporter\"}"
    },
    "timestamp": "1752787969551",
    "scraperId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Conexão recusada

Quando o coletor não consegue estabelecer uma conexão com o endpoint de destino, você recebe este erro:

```
{
  "scrapeConfigId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "timestamp": "2025-04-30T17:25:41.946Z",
  "message": {
    "message": "Scrape failed",
    "scrape_pool": "pod_exporter",
    "target": "http://10.24.34.0:80/metrics",
    "error": "Get \"http://10.24.34.0:80/metrics\": dial tcp 10.24.34.0:80: connect:
 connection refused"
  },
  "component": "COLLECTOR"
}
```

Logs do exportador

Os logs do exportador fornecem informações sobre o processo de envio de métricas coletadas para seu espaço de trabalho do Amazon Managed Service for Prometheus, incluindo:

- Número de métricas e pontos de dados processados.
- Falhas na exportação devido a problemas no espaço de trabalho.
- Erros de permissão ao tentar gravar métricas.
- Falhas de dependência no pipeline de exportação.

O exemplo a seguir demonstra um erro comum do exportador durante o processo de exportação de métricas:

Espaço de trabalho não encontrado

Quando o espaço de trabalho de destino para exportação de métricas não é encontrado, você recebe este erro:

```
{
    "component": "EXPORTER",
    "message": {
        "log": "Failed to export to the target workspace - Verify your scraper
 destination.",
        "samplesDropped": 5
    },
    "timestamp": "1752787969664",
    "scraperId": "s-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Entender e usar logs fornecidos por coletores

Estrutura de logs

Todos os logs fornecidos por coletores seguem uma estrutura consistente com estes campos:

scrapeConfigId

O identificador exclusivo da configuração de extração que gerou o log.

timestamp

A hora em que a entrada do log foi gerada.

mensagem

O conteúdo da mensagem de log, que pode incluir campos estruturados adicionais.

componente

O componente que gerou o log (SERVICE_DISCOVERY, COLLECTOR ou EXPORTER)

Usar logs fornecidos para solução de problemas

Os logs fornecidos por coletores ajudam a solucionar problemas comuns com coleta de métricas:

1. Problemas de descoberta de serviços

- Verifique os logs do SERVICE_DISCOVERY em busca de erros de autenticação ou permissão.
- Verifique se o coletor tem as permissões necessárias para acessar os recursos do Kubernetes.

2. Problemas de extração métrica
   - Verifique nos logs do COLLECTOR se há falhas de extração.
   - Verifique se os endpoints de destino estão acessíveis e retornam métricas.
   - Certifique-se de que as regras de firewall permitam que o coletor se conecte aos endpoints de destino.
3. Problemas de exportação métrica
   - Verifique se há falhas na exportação nos logs do EXPORTER.
   - Verifique se o espaço de trabalho existe e se está configurado corretamente.
   - Certifique-se de que o coletor tenha as permissões necessárias para gravar no espaço de trabalho.

Acessar logs fornecidos por coletores

Os registros vendidos pelo Collector são enviados automaticamente para a Amazon CloudWatch Logs. Como acessar esses logs:

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. No painel de navegação, escolha Grupos de logs.
3. Encontre e selecione o grupo de logs para seu coletor:`/aws/prometheus/workspace_id/collector/collector_id`.
4. Navegue ou pesquise os eventos de logs para encontrar informações relevantes.

Você também pode usar o CloudWatch Logs Insights para consultar e analisar seus registros do coletor. Por exemplo, para encontrar todos os erros de descoberta de serviços:

```
fields @timestamp, message.message
| filter component = "SERVICE_DISCOVERY" and message.message like /Failed/
| sort @timestamp desc
```

Práticas recomendadas do monitoramento de coletores

Como monitorar com eficácia os coletores do Amazon Managed Service for Prometheus:

1. Configure CloudWatch alarmes para problemas críticos do coletor, como falhas persistentes de raspagem ou erros de exportação. Para obter mais informações, consulte Alarmes no Guia do CloudWatch usuário da Amazon.
2. Crie CloudWatch painéis para visualizar as métricas de desempenho do coletor junto com os dados de log vendidos. Para obter mais informações, consulte Painéis no Guia do CloudWatch usuário da Amazon.
3. Analise regularmente os logs de descoberta de serviços para garantir que os destinos sejam descobertos corretamente.
4. Monitore o número de destinos eliminados para identificar possíveis problemas de configuração.
5. Acompanhe as falhas de exportação para garantir que as métricas sejam enviadas com sucesso ao seu espaço de trabalho.

Coletores gerenciados pelo cliente

Esta seção contém informações sobre a ingestão de dados por meio da configuração de seus próprios coletores que enviam métricas para o Amazon Managed Service for Prometheus usando a gravação remota do Prometheus.

Quando você usa seus próprios coletores para enviar métricas para o Amazon Managed Service for Prometheus, você é responsável por proteger as métricas e garantir que o processo de ingestão atenda às suas necessidades de disponibilidade.

A maioria dos coletores gerenciados pelo cliente usa uma das seguintes ferramentas:

- AWS Distro for OpenTelemetry (ADOT) — ADOT é uma distribuição de código aberto totalmente suportada, segura e pronta para produção OpenTelemetry que fornece aos agentes a coleta de métricas. É possível usar o ADOT para coletar métricas e enviá-las ao espaço de trabalho do Amazon Managed Service for Prometheus. Para obter mais informações sobre o ADOT Collector, consulte AWS Distro for. OpenTelemetry
- Prometheus agent: você pode configurar sua própria instância do servidor Prometheus de código aberto, executado como agente, para coletar métricas e encaminhá-las para o espaço de trabalho do Amazon Managed Service for Prometheus.

Os tópicos a seguir descrevem o uso dessas duas ferramentas e incluem informações gerais sobre como configurar seus próprios coletores.

Tópicos

Proteger a ingestão de suas métricas

O Amazon Managed Service for Prometheus oferece maneiras de ajudar proteger a ingestão de suas métricas.

Usando AWS PrivateLink com o Amazon Managed Service para Prometheus

O tráfego de rede da ingestão das métricas no Amazon Managed Service for Prometheus pode ser feito por meio de um endpoint público da Internet ou por meio de um endpoint VPC. AWS PrivateLink O uso AWS PrivateLink garante que o tráfego de rede de você VPCs esteja protegido na AWS rede sem passar pela Internet pública. Para criar um AWS PrivateLink VPC endpoint para o Amazon Managed Service for Prometheus, consulte. Como utilizar o Amazon Managed Service for Prometheus com endpoints da VPC de interface

Autenticação e autorização

AWS O Identity and Access Management (IAM) é um serviço web que ajuda você a controlar com segurança o acesso aos recursos. AWS Com o IAM, você controla quem pode se autenticar (fazer login) e quem tem autorização (permissões) para acessar os recursos. O Amazon Managed Service for Prometheus se integra ao IAM para ajudar manter seus dados protegidos. Ao configurar o Amazon Managed Service for Prometheus, você precisará criar alguns perfis do IAM que permitam a ingestão de métricas dos servidores Prometheus e que permitam que os servidores Grafana consultem as métricas armazenadas nos espaços de trabalho do Amazon Managed Service for Prometheus. Para obter mais informações sobre o IAM, consulte O que é o IAM?

Outro recurso AWS de segurança que pode ajudar você a configurar o Amazon Managed Service para Prometheus é AWS o processo AWS de assinatura Signature Version 4 (SigV4). A versão 4 do Signature é o processo para adicionar informações de autenticação às AWS solicitações enviadas por HTTP. Por motivos de segurança, a maioria das solicitações AWS deve ser assinada com uma chave de acesso, que consiste em uma ID da chave de acesso e uma chave de acesso secreta.

Essas duas chaves são comumente conhecidas como suas credenciais de segurança. Para obter mais informações sobre o SigV4, consulte Processo de assinatura do Signature Version 4.

Usando o AWS Distro OpenTelemetry como coletor

Esta seção descreve como configurar o AWS Distro for OpenTelemetry (ADOT) Collector para extrair de um aplicativo instrumentado pelo Prometheus e enviar as métricas para o Amazon Managed Service for Prometheus. Para obter mais informações sobre o ADOT Collector, consulte AWS Distro for. OpenTelemetry

Os tópicos a seguir descrevem três maneiras diferentes de configurar o ADOT como um coletor para suas métricas, com base no fato de suas métricas serem provenientes do Amazon EKS, do Amazon ECS ou de uma instância do Amazon EC2.

Tópicos

- Configure a ingestão de métricas usando o AWS Distro para OpenTelemetry em um cluster do Amazon Elastic Kubernetes Service
- Configure a ingestão de métricas do Amazon ECS usando o AWS Distro for Open Telemetry
- Configure a ingestão de métricas de uma instância do Amazon EC2 usando a gravação remota

Configure a ingestão de métricas usando o AWS Distro para OpenTelemetry em um cluster do Amazon Elastic Kubernetes Service

Você pode usar o coletor AWS Distro for OpenTelemetry (ADOT) para extrair métricas de um aplicativo instrumentado pelo Prometheus e enviar as métricas para o Amazon Managed Service for Prometheus.

> **Note**
>
> Para obter mais informações sobre o coletor ADOT, consulte AWS Distro for. OpenTelemetry Para obter mais informações sobre os aplicativos instrumentados pelo Prometheus, consulte O que são métricas compatíveis com o Prometheus?.

A coleta de métricas do Prometheus com o ADOT envolve três OpenTelemetry componentes: o Prometheus Receiver, o Prometheus Remote Write Exporter e a Extensão de Autenticação Sigv4.

Você pode configurar o Prometheus Receiver usando sua configuração existente do Prometheus para realizar a descoberta de serviços e a coleta de métricas. O Prometheus Receiver coleta métricas no formato de exposição do Prometheus. Todos os aplicativos ou endpoints que você deseja coletar devem ser configurados com a biblioteca de clientes do Prometheus. O Prometheus Receiver suporta o conjunto completo de configurações de coleta e rerrotulagem do Prometheus descritas em Configuração na documentação do Prometheus. Você pode colar essas configurações diretamente nas suas configurações do ADOT Collector.

O Prometheus Remote Write Exporter usa o endpoint do `remote_write` para enviar as métricas coletadas para o espaço de trabalho do seu portal de gerenciamento. As solicitações HTTP para exportar dados serão assinadas com o AWS SigV4, o AWS protocolo para autenticação segura, com a Extensão de Autenticação Sigv4. Para obter mais informações, consulte Processo de assinatura do Signature Version 4.

O coletor descobre automaticamente os endpoints de métricas do Prometheus no Amazon EKS e usa a configuração encontrada em.<kubernetes_sd_config>.

A demonstração a seguir é um exemplo dessa configuração em um cluster executando o Amazon Elastic Kubernetes Service ou o Kubernetes autogerenciado. Para executar essas etapas, você deve ter AWS credenciais de qualquer uma das opções possíveis na cadeia de AWS credenciais padrão. Para obter mais informações, consulte Como configurar o AWS SDK for Go. Esta demonstração usa uma aplicação de amostra usada para testes de integração do processo. A aplicação de amostra expõe métricas no endpoint do `/metrics`, assim como a biblioteca de clientes do Prometheus.

Pré-requisitos

Antes de começar as etapas de configuração de ingestão a seguir, você deve configurar o perfil do IAM para a conta de serviço e a política de confiança.

Para configurar o perfil do IAM para a conta de serviço e a política de confiança

1. Crie o perfil do IAM para a conta de serviço seguindo as etapas em Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS.

   O ADOT Collector usará esse perfil ao coletar e exportar métricas.
2. Em seguida, edite a política de confiança. Abra o console do IAM em https://console.aws.amazon.com/iam/.
3. No painel de navegação esquerdo, escolha Funções e encontre as amp-iamproxy-ingest-roleque você criou na etapa 1.

4. Escolha a guia Relações de confiança e Editar relação de confiança.
5. No JSON da política de relação de confiança, substitua `aws-amp` por `adot-col` e, em seguida, escolha Atualizar política de confiança. A política de confiança resultante deverá ser algo semelhante a:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::111122223333:oidc-provider/oidc.eks.us-
east-1.amazonaws.com/id/EXAMPLED539D4633E53DE1B71EXAMPLE"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:sub": "system:serviceaccount:adot-
col:amp-iamproxy-ingest-service-account",
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:aud": "sts.amazonaws.com"
        }
      }
    }
  ]
}
```

6. Escolha a guia Permissões e certifique-se de que a política de permissões a seguir esteja anexada ao perfil.

   JSON

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

```
                "aps:RemoteWrite",
                "aps:GetSeries",
                "aps:GetLabels",
                "aps:GetMetricMetadata"
            ],
            "Resource": "*"
        }
    ]
}
```

Habilitar a coleta de métricas do Prometheus

> **Note**
>
> Quando você cria um namespace no Amazon EKS, o `alertmanager` e o exportador de nós são desabilitados por padrão.

Para habilitar a coleta do Prometheus em um cluster do Amazon EKS ou do Kubernetes

1. Bifurque e clone o aplicativo de amostra do repositório em. aws-otel-community

   Depois, execute os seguintes comandos.

   ```
   cd ./sample-apps/prometheus-sample-app
   docker build . -t prometheus-sample-app:latest
   ```

2. Envie essa imagem para um registro, como Amazon ECR ou DockerHub.
3. Implante o aplicativo de amostra no cluster copiando essa configuração do Kubernetes e aplicando-a. Altere a imagem para a imagem que você acabou de inserir substituindo `{{PUBLIC_SAMPLE_APP_IMAGE}}` no arquivo `prometheus-sample-app.yaml`.

   ```
   curl https://raw.githubusercontent.com/aws-observability/aws-otel-collector/
   main/examples/eks/aws-prometheus/prometheus-sample-app.yaml -o prometheus-sample-
   app.yaml
   kubectl apply -f prometheus-sample-app.yaml
   ```

4. Execute o comando a seguir para verificar se o aplicativo de amostra foi iniciado. Na saída do comando, você verá `prometheus-sample-app` na coluna `NAME`.

```
kubectl get all -n aoc-prometheus-pipeline-demo
```

5. Inicie uma instância padrão do ADOT Collector. Para fazer isso, primeiro insira o comando a seguir para extrair a configuração do Kubernetes para o ADOT Collector.

```
curl https://raw.githubusercontent.com/aws-observability/aws-otel-collector/main/
examples/eks/aws-prometheus/prometheus-daemonset.yaml -o prometheus-daemonset.yaml
```

   Em seguida, edite o arquivo de modelo, substituindo o endpoint remote_write do seu espaço de trabalho do Amazon Managed Service for Prometheus por `YOUR_ENDPOINT` e sua região por `YOUR_REGION`. Use o endpoint remote_write que é exibido no console do Amazon Managed Service for Prometheus ao examinar os detalhes do seu espaço de trabalho.

   Você também precisará alterar o ID da sua conta `YOUR_ACCOUNT_ID` na seção de conta de serviço da configuração do Kubernetes. AWS

   Neste exemplo, a configuração do ADOT Collector usa uma anotação (`scrape=true`) para informar quais endpoints de destino devem ser coletados. Isso permite que o ADOT Collector diferencie o endpoint do aplicativo de amostra dos endpoints do sistema kube em seu cluster. Você pode remover isso das configurações de renomeação se quiser coletar um aplicativo de amostra diferente.

6. Insira o comando a seguir para implantar o coletor ADOT.

```
kubectl apply -f prometheus-daemonset.yaml
```

7. Execute o comando a seguir para verificar se o coletor ADOT foi iniciado. Procure `adot-col` na coluna `NAMESPACE`.

```
kubectl get pods -n adot-col
```

8. Verifique se o pipeline funciona usando o exportador de log. Nosso modelo de exemplo já está integrado ao exportador de log. Insira os comandos a seguir:

```
kubectl get pods -A
kubectl logs -n adot-col name_of_your_adot_collector_pod
```

   Algumas das métricas coletadas do aplicativo de exemplo serão semelhantes às do exemplo a seguir.

```
Resource labels:
     -> service.name: STRING(kubernetes-service-endpoints)
     -> host.name: STRING(192.168.16.238)
     -> port: STRING(8080)
     -> scheme: STRING(http)
InstrumentationLibraryMetrics #0
Metric #0
Descriptor:
     -> Name: test_gauge0
     -> Description: This is my gauge
     -> Unit:
     -> DataType: DoubleGauge
DoubleDataPoints #0
StartTime: 0
Timestamp: 1606511460471000000
Value: 0.000000
```

9. Para testar se o Amazon Managed Service for Prometheus recebeu as métricas, use o `awscurl`. Essa ferramenta permite que você envie solicitações HTTP por meio da linha de comando com autenticação AWS Sigv4, portanto, você deve ter AWS credenciais configuradas localmente com as permissões corretas para fazer consultas no Amazon Managed Service for Prometheus. Para obter instruções sobre a instalação, consulte awscurl. `awscurl`

   No comando a seguir, substitua `AMP_REGION` e `AMP_ENDPOINT` pelas informações do seu espaço de trabalho do Amazon Managed Service for Prometheus.

   ```
   awscurl --service="aps" --region="AMP_REGION" "https://AMP_ENDPOINT/api/v1/query?
   query=adot_test_gauge0"
   {"status":"success","data":{"resultType":"vector","result":[{"metric":
   {"__name__":"adot_test_gauge0"},"value":[1606512592.493,"16.87214000011479"]}]}}
   ```

   Se você receber uma métrica como resposta, isso significa que a configuração do pipeline foi bem-sucedida e a métrica foi propagada com sucesso da aplicação de amostra para o Amazon Managed Service for Prometheus.

Limpeza

Para limpar essa demonstração, digite os comandos a seguir.

```
kubectl delete namespace aoc-prometheus-pipeline-demo
```

```
kubectl delete namespace adot-col
```

Configuração avançada

O Prometheus Receiver suporta o conjunto completo de configurações de coleta e rerrotulagem do Prometheus descritas em Configuração na documentação do Prometheus. Você pode colar essas configurações diretamente nas suas configurações do ADOT Collector.

A configuração do Prometheus Receiver inclui sua descoberta de serviços, configurações de coleta e configurações de rerrotulagem. A configuração do receptor se parece com as seguintes.

```
receivers:
  prometheus:
    config:
      [Your Prometheus configuration]
```

Veja a seguir um exemplo de configuração.

```
receivers:
  prometheus:
    config:
      global:
        scrape_interval: 1m
        scrape_timeout: 10s

      scrape_configs:
      - job_name: kubernetes-service-endpoints
        sample_limit: 10000
        kubernetes_sd_configs:
        - role: endpoints
        tls_config:
          ca_file: /var/run/secrets/kubernetes.io/serviceaccount/ca.crt
          insecure_skip_verify: true
        bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
```

Se você tiver uma configuração existente do Prometheus, deverá substituir os caracteres $ por $ $ para evitar que os valores sejam substituídos por variáveis de ambiente. *Isso é especialmente importante para o valor de substituição das relabel_configurations. Por exemplo, se você começar com a seguinte relabel_configuration:

```
relabel_configs:
```

```
- source_labels:
 [__meta_kubernetes_ingress_scheme,__address__,__meta_kubernetes_ingress_path]
  regex: (.+);(.+);(.+)
  replacement: ${1}://${2}${3}
  target_label: __param_target
```

Isso seria o seguinte:

```
relabel_configs:
- source_labels:
 [__meta_kubernetes_ingress_scheme,__address__,__meta_kubernetes_ingress_path]
  regex: (.+);(.+);(.+)
  replacement: $${1}://${2}${3}
  target_label: __param_target
```

Exportador de gravação remota do Prometheus e extensão de autenticação do Sigv4

A configuração do Prometheus Remote Write Exporter e do Sigv4 Authentication Extension é mais simples do que a do receptor do Prometheus. Neste estágio do pipeline, as métricas já foram ingeridas e estamos prontos para exportar esses dados para o Amazon Managed Service for Prometheus. O requisito mínimo para uma configuração bem-sucedida para se comunicar com o Amazon Managed Service for Prometheus é visto no exemplo a seguir.

```
extensions:
  sigv4auth:
    service: "aps"
    region: "user-region"
exporters:
  prometheusremotewrite:
    endpoint: "https://aws-managed-prometheus-endpoint/api/v1/remote_write"
    auth:
      authenticator: "sigv4auth"
```

Essa configuração envia uma solicitação HTTPS assinada pelo AWS SigV4 usando AWS credenciais da cadeia de credenciais padrão AWS . Para obter mais informações, consulte Configurar a AWS SDK para Go. O serviço deve ser especificado como `aps`.

Independentemente do método de implantação, o coletor ADOT deve ter acesso a uma das opções listadas na cadeia de AWS credenciais padrão. A extensão de autenticação Sigv4 depende do AWS SDK para Go e a usa para obter credenciais e autenticar. Você deve garantir que essas credenciais tenham permissões de gravação remota para o Amazon Managed Service for Prometheus.

Configure a ingestão de métricas do Amazon ECS usando o AWS Distro for Open Telemetry

Esta seção explica como coletar métricas do Amazon Elastic Container Service (Amazon ECS) e inseri-las no Amazon Managed Service for AWS Prometheus usando o Distro for Open Telemetry (ADOT). Também descreve como visualizar suas métricas no Amazon Managed Grafana.

Pré-requisitos

> **Important**
>
> Antes de começar, é preciso ter um ambiente Amazon ECS em um cluster do AWS Fargate com configurações padrão, um espaço de trabalho do Amazon Managed Service for Prometheus e um espaço de trabalho do Amazon Managed Grafana. Presumimos que você esteja familiarizado com as workloads de contêineres, o Amazon Managed Service for Prometheus e o Amazon Managed Grafana.

Para obter mais informações, consulte os seguintes links:

- Para obter informações sobre como criar um ambiente Amazon ECS em um cluster Fargate com configurações padrão, consulte Criação de um cluster no Guia do desenvolvedor do Amazon ECS.
- Para obter informações sobre como criar um espaço de trabalho do Amazon Managed Service for Prometheus, consulte Criação de um espaço de trabalho no Guia do usuário do Amazon Managed Service for Prometheus.
- Para obter informações sobre como criar um espaço de trabalho do Amazon Managed Grafana, consulte Criação de um espaço de trabalho no Guia do usuário do Amazon Managed Grafana.

Etapa 1: definir uma imagem personalizada de contêiner do coletor do ADOT

Use o arquivo de configuração a seguir como modelo para definir sua própria imagem de contêiner do coletor ADOT. *my-region*Substitua *my-remote-URL* e por seus `endpoint` `region` valores. Salve a configuração em um arquivo chamado adot-config.yaml.

> **Note**
>
> Essa configuração usa a extensão `sigv4auth` para autenticar chamadas para o Amazon Managed Service for Prometheus. Para obter mais informações sobre a configuração`sigv4auth`, consulte Autenticador - Sigv4 ativado. GitHub

```
receivers:
  prometheus:
    config:
      global:
        scrape_interval: 15s
        scrape_timeout: 10s
      scrape_configs:
      - job_name: "prometheus"
        static_configs:
        - targets: [ 0.0.0.0:9090 ]
  awsecscontainermetrics:
    collection_interval: 10s
processors:
  filter:
    metrics:
      include:
        match_type: strict
        metric_names:
          - ecs.task.memory.utilized
          - ecs.task.memory.reserved
          - ecs.task.cpu.utilized
          - ecs.task.cpu.reserved
          - ecs.task.network.rate.rx
          - ecs.task.network.rate.tx
          - ecs.task.storage.read_bytes
          - ecs.task.storage.write_bytes
exporters:
  prometheusremotewrite:
    endpoint: my-remote-URL
    auth:
      authenticator: sigv4auth
  logging:
    loglevel: info
extensions:
  health_check:
```

```
  pprof:
    endpoint: :1888
  zpages:
    endpoint: :55679
  sigv4auth:
    region: my-region
    service: aps
service:
  extensions: [pprof, zpages, health_check, sigv4auth]
  pipelines:
    metrics:
      receivers: [prometheus]
      exporters: [logging, prometheusremotewrite]
    metrics/ecs:
      receivers: [awsecscontainermetrics]
      processors: [filter]
      exporters: [logging, prometheusremotewrite]
```

Etapa 2: enviar sua imagem do contêiner do coletor do ADOT para um repositório do Amazon ECR

Use um Dockerfile para criar e enviar sua imagem de contêiner para um repositório do Amazon Elastic Container Registry (ECR).

1. Crie o Dockerfile para copiar e adicionar sua imagem de contêiner à imagem do OTEL Docker.

```
FROM public.ecr.aws/aws-observability/aws-otel-collector:latest
COPY adot-config.yaml /etc/ecs/otel-config.yaml
CMD ["--config=/etc/ecs/otel-config.yaml"]
```

2. Crie um repositório do Amazon ECR.

```
# create repo:
COLLECTOR_REPOSITORY=$(aws ecr create-repository --repository aws-otel-collector \
                              --query repository.repositoryUri --output text)
```

3. Crie sua imagem de contêiner.

```
# build ADOT collector image:
docker build -t $COLLECTOR_REPOSITORY:ecs .
```

> **Note**
>
> Isso pressupõe que você esteja criando seu contêiner no mesmo ambiente em que ele será executado. Caso contrário, talvez seja necessário usar o parâmetro `--platform` ao criar a imagem.

4. Faça login no repositório do Amazon ECR. *my-region*Substitua pelo seu `region` valor.

```
# sign in to repo:
aws ecr get-login-password --region my-region | \
        docker login --username AWS --password-stdin $COLLECTOR_REPOSITORY
```

5. Envie a imagem do seu contêiner.

```
# push ADOT collector image:
docker push $COLLECTOR_REPOSITORY:ecs
```

Etapa 3: criar uma definição de tarefa do Amazon ECS para extrair o Amazon Managed Service for Prometheus

Crie uma definição de tarefa do Amazon ECS para coletar o Amazon Managed Service for Prometheus. Sua definição de tarefa deve incluir um contêiner chamado `adot-collector` e um contêiner chamado`prometheus`. O `prometheus` gera métricas e o `adot-collector` coleta `prometheus`.

> **Note**
>
> O Amazon Managed Service for Prometheus é executado como um serviço, coletando métricas dos contêineres. Nesse caso, os contêineres executam o Prometheus localmente, no modo Atendente, que envia as métricas locais para o Amazon Managed Service for Prometheus.

Exemplo: Definição de tarefa

Veja a seguir um exemplo da possível aparência da definição de tarefa. Você pode usar esse exemplo como modelo para criar sua própria definição de tarefa. Substitua o valor `image` de `adot-`

`collector` pelo URL do seu repositório e pela tag da imagem (`$COLLECTOR_REPOSITORY:ecs`). Substitua os valores `region` de `adot-collector` e `prometheus` por seus valores `region`.

```
{
  "family": "adot-prom",
  "networkMode": "awsvpc",
  "containerDefinitions": [
    {
      "name": "adot-collector",
      "image": "account_id.dkr.ecr.region.amazonaws.com/image-tag",
      "essential": true,
      "logConfiguration": {
        "logDriver": "awslogs",
        "options": {
          "awslogs-group": "/ecs/ecs-adot-collector",
          "awslogs-region": "my-region",
          "awslogs-stream-prefix": "ecs",
          "awslogs-create-group": "True"
        }
      }
    },
    {
      "name": "prometheus",
      "image": "prom/prometheus:main",
      "logConfiguration": {
        "logDriver": "awslogs",
        "options": {
          "awslogs-group": "/ecs/ecs-prom",
          "awslogs-region": "my-region",
          "awslogs-stream-prefix": "ecs",
          "awslogs-create-group": "True"
        }
      }
    }
  ],
  "requiresCompatibilities": [
    "FARGATE"
  ],
  "cpu": "1024"
}
```

Etapa 4: conceder à sua tarefa permissões para acessar o Amazon Managed Service for Prometheus

Para enviar as métricas coletadas para o Amazon Managed Service for Prometheus, sua tarefa do Amazon ECS deve ter as permissões corretas para chamar as operações de API para você. AWS Você deve criar um perfil do IAM para as suas tarefas e anexar a política do `AmazonPrometheusRemoteWriteAccess` a ele. Para obter mais informações sobre como criar esse perfil e anexar a política, consulte Criação de um perfil e política do IAM para as suas tarefas.

Depois de anexar `AmazonPrometheusRemoteWriteAccess` ao seu perfil do IAM e usar esse perfil para suas tarefas, o Amazon ECS pode enviar suas métricas coletadas para o Amazon Managed Service for Prometheus.

Etapa 5: visualizar suas métricas no Amazon Managed Grafana

> **Important**
>
> Antes de começar, você deve executar uma tarefa do Fargate na definição de tarefa do Amazon ECS. Caso contrário, o Amazon Managed Service for Prometheus não poderá consumir suas métricas.

1. No painel de navegação do seu espaço de trabalho Amazon Managed Grafana, escolha Fontes de dados abaixo do ícone. AWS
2. Na guia Fontes de dados, em Serviço, selecione Amazon Managed Service for Prometheus e escolha a Região padrão.
3. Escolha Adicionar fonte de dados.
4. Use os prefixos `ecs` e `prometheus` para consultar e visualizar suas métricas.

Configure a ingestão de métricas de uma instância do Amazon EC2 usando a gravação remota

Esta seção explica como executar um servidor Prometheus com gravação remota em uma instância do Amazon Elastic Compute Cloud (Amazon EC2). Ela explica como coletar métricas de um aplicativo de demonstração escrito em Go e enviá-las para um espaço de trabalho do Amazon Managed Service for Prometheus.

Pré-requisitos

> **Important**
>
> Antes de começar, você deve ter instalado o Prometheus v2.26 ou posterior. Presumimos que você esteja familiarizado com o Prometheus, o Amazon EC2 e o Amazon Managed Service for Prometheus. Para obter informações sobre como instalar o Prometheus, consulte os Conceitos básicos no site do Prometheus.

Se você não estiver familiarizado com o Amazon EC2 ou com o Amazon Managed Service for Prometheus, recomendamos que comece lendo as seguintes seções:

- O que é o Amazon Elastic Compute Cloud?
- O que é o Amazon Managed Service for Prometheus?

Criar um perfil do IAM para o Amazon EC2

Para transmitir métricas, primeiro você deve criar uma função do IAM com a política AWS gerenciada AmazonPrometheusRemoteWriteAccess. Em seguida, você pode iniciar uma instância com o perfil e transmitir métricas para o seu espaço de trabalho do Amazon Managed Service for Prometheus.

1. Abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação, escolha Roles (Funções) e Create role (Criar função).
3. Para o tipo de entidade confiável, selecione AWS serviço. Para o caso de uso, escolha EC2. Escolha Próximo: Permissões.
4. Na barra de pesquisa, insira AmazonPrometheusRemoteWriteAccess. Em Nome da política AmazonPrometheusRemoteWriteAccess, selecione e escolha Anexar política. Selecione Next: Tags (Próximo: tags).
5. (Opcional) Crie tags do IAM para seu perfil do IAM. Escolha Próximo: revisar.
6. Insira um nome para o seu perfil. Selecione Criar política.

Iniciar uma instância do Amazon EC2

Para criar uma instância do Amazon EC2, siga as instruções em Executar uma instância no Guia do usuário do Amazon Elastic Compute Cloud para instâncias do Linux.

Execute o aplicativo de demonstração

Depois de criar seu perfil do IAM e iniciar uma instância do EC2 com o perfil, você poderá executar um aplicativo de demonstração para vê-lo em funcionamento.

Para executar um aplicativo de demonstração e métricas de teste

1. Use o modelo a seguir para criar um arquivo Go chamado `main.go`.

```
package main

import (
    "github.com/prometheus/client_golang/prometheus/promhttp"
    "net/http"
)

func main() {
    http.Handle("/metrics", promhttp.Handler())

    http.ListenAndServe(":8000", nil)
}
```

2. Execute os seguintes comandos para instalar as dependências corretas.

```
sudo yum update -y
sudo yum install -y golang
go get github.com/prometheus/client_golang/prometheus/promhttp
```

3. Execute o aplicativo de demonstração.

```
go run main.go
```

   O aplicativo de demonstração deve ser executado na porta 8000 e mostrar todas as métricas expostas do Prometheus. A seguir, veja um exemplo dessas métricas.

```
curl -s http://localhost:8000/metrics
...
process_max_fds 4096# HELP process_open_fds Number of open file descriptors.# TYPE
 process_open_fds gauge
process_open_fds 10# HELP process_resident_memory_bytes Resident memory size in
 bytes.# TYPE process_resident_memory_bytes gauge
```

```
process_resident_memory_bytes 1.0657792e+07# HELP process_start_time_seconds Start
 time of the process since unix epoch in seconds.# TYPE process_start_time_seconds
 gauge
process_start_time_seconds 1.61131955899e+09# HELP process_virtual_memory_bytes
 Virtual memory size in bytes.# TYPE process_virtual_memory_bytes gauge
process_virtual_memory_bytes 7.77281536e+08# HELP process_virtual_memory_max_bytes
 Maximum amount of virtual memory available in bytes.# TYPE
 process_virtual_memory_max_bytes gauge
process_virtual_memory_max_bytes -1# HELP
 promhttp_metric_handler_requests_in_flight Current number of scrapes being
 served.# TYPE promhttp_metric_handler_requests_in_flight gauge
promhttp_metric_handler_requests_in_flight 1# HELP
 promhttp_metric_handler_requests_total Total number of scrapes by HTTP status
 code.# TYPE promhttp_metric_handler_requests_total counter
promhttp_metric_handler_requests_total{code="200"} 1
promhttp_metric_handler_requests_total{code="500"} 0
promhttp_metric_handler_requests_total{code="503"} 0
```

Criar um espaço de trabalho do Amazon Managed Service for Prometheus

Para criar um espaço de trabalho do Amazon Managed Service for Prometheus, siga as instruções em Create a espaço de trabalho.

Executar um servidor Prometheus

1. Use o seguinte exemplo de arquivo YAML como modelo para criar um novo arquivo chamado `prometheus.yaml`. Para`url`, *`my-region`* substitua pelo valor da sua região e *`my-workspace-id`* pelo ID do espaço de trabalho que o Amazon Managed Service for Prometheus gerou para você. Para`region`, *`my-region`* substitua pelo valor da sua região.

   Exemplo: arquivo YAML

   ```
   global:
     scrape_interval: 15s
     external_labels:
       monitor: 'prometheus'

   scrape_configs:
     - job_name: 'prometheus'
       static_configs:
         - targets: ['localhost:8000']
   ```

```
remote_write:
  -
    url: https://aps-workspaces.my-region.amazonaws.com/workspaces/my-workspace-id/
api/v1/remote_write
    queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
    sigv4:
         region: my-region
```

2. Execute o servidor Prometheus para enviar as métricas do aplicativo de demonstração para seu espaço de trabalho do Amazon Managed Service for Prometheus.

```
prometheus --config.file=prometheus.yaml
```

O servidor Prometheus agora deverá enviar as métricas do aplicativo de demonstração para seu espaço de trabalho do Amazon Managed Service for Prometheus.

Usar uma instância do Prometheus como coletor

Você pode usar uma instância do Prometheus em execução no modo atendente (conhecido como agente do Prometheus) para extrair métricas e enviá-las ao seu espaço de trabalho do Amazon Managed Service for Prometheus.

Os tópicos a seguir descrevem maneiras diferentes de configurar uma instância do Prometheus em execução no modo Agente como um coletor para as métricas.

> **Warning**
>
> Ao criar um atendente do Prometheus, você é responsável pela respectiva configuração e manutenção. Habilite os recursos de segurança para evitar expor os endpoints de extração do Prometheus à Internet pública.

Se você configurou várias instâncias do Prometheus que monitoram o mesmo conjunto de métricas e as enviou para um único espaço de trabalho do Amazon Managed Service for Prometheus para obter alta disponibilidade, você precisará configurar a desduplicação. Se não seguir as etapas para configurar a desduplicação, você será cobrado por todas as amostras de dados enviadas ao

Amazon Managed Service for Prometheus, incluindo amostras duplicadas. Para ver instruções sobre como configurar a desduplicação, consulte Eliminar a duplicação de métricas de alta disponibilidade enviadas para o Amazon Managed Service for Prometheus.

Tópicos

- Configurar a ingestão de um novo servidor Prometheus usando o Helm
- Configurar a ingestão de um servidor Prometheus existente no Kubernetes no EC2
- Configurar a ingestão de um servidor Prometheus existente no Kubernetes no Fargate

Configurar a ingestão de um novo servidor Prometheus usando o Helm

As instruções nesta seção permitem que você comece a usar o Amazon Managed Service for Prometheus rapidamente. Você configura um novo servidor Prometheus em um cluster do Amazon EKS, e o novo servidor usa uma configuração padrão para enviar métricas para o Amazon Managed Service for Prometheus. Este método tem os seguintes pré-requisitos:

- Você deve ter um cluster do Amazon EKS do qual o novo servidor do Prometheus coletará métricas.
- Seu cluster do Amazon EKS deve ter um driver do Amazon EBS CSI instalado (necessário ao Helm).
- Você deve usar a CLI do Helm 3.0 ou posterior.
- Você deve usar um computador Linux ou macOS para executar as etapas nas seções a seguir.

Etapa 1: Adicionar novos repositórios de charts do Helm

Insira os comandos a seguir para adicionar novos repositórios de charts do Helm. Para obter mais informações sobre esses comandos, consulte o Repositório do Helm.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo add kube-state-metrics https://kubernetes.github.io/kube-state-metrics
helm repo update
```

Etapa 2: Criar um namespace do Prometheus

Digite o comando a seguir para criar um namespace do Prometheus para o servidor Prometheus e outros componentes de monitoramento. *prometheus-namespace*Substitua pelo nome que você deseja para esse namespace.

```
kubectl create namespace prometheus-namespace
```

Etapa 3: Configurar perfis do IAM para as contas de serviço

Para o método de integração que estamos documentando, é necessário usar perfis do IAM para as contas de serviço no cluster do Amazon EKS em que o servidor do Prometheus está em execução.

Com os perfis do IAM para contas de serviço, é possível associar um perfil do IAM a uma conta de serviço do Kubernetes. Essa conta de serviço pode fornecer permissões da AWS para os contêineres em qualquer pod que use essa conta de serviço. Para obter mais informações, consulte Perfis do IAM para contas de serviço.

Se você ainda não configurou esses perfis, siga as instruções em Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS para configurar os perfis. As instruções nessa seção exigem o uso do `eksctl`. Para obter mais informações, consulte Conceitos básicos do Amazon Elastic Kubernetes Service – `eksctl`.

> **Note**
>
> Quando você não está usando o EKS ou AWS está usando apenas a chave de acesso e a chave secreta para acessar o Amazon Managed Service para Prometheus, você não pode usar `EKS-IAM-ROLE` o SigV4 baseado.

Etapa 4: Configurar o novo servidor e começar a ingerir métricas

Para instalar o novo servidor Prometheus que envia métricas para seu espaço de trabalho do Amazon Managed Service for Prometheus, siga estas etapas.

Instalar o novo servidor Prometheus que envia métricas para seu espaço de trabalho do Amazon Managed Service for Prometheus

1. Use um editor de textos para criar um arquivo chamado `my_prometheus_values_yaml` com o conteúdo a seguir.

   - *`IAM_PROXY_PROMETHEUS_ROLE_ARN`*Substitua pelo ARN do amp-iamproxy-ingest-roleque você criou em. Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS

- *WORKSPACE_ID*Substitua pelo ID do seu espaço de trabalho do Amazon Managed Service for Prometheus.
- *REGION*Substitua pela região do seu espaço de trabalho do Amazon Managed Service for Prometheus.

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
## For the rest of prometheus helm chart values see: https://github.com/prometheus-
community/helm-charts/blob/main/charts/prometheus/values.yaml
##
serviceAccounts:
  server:
    name: amp-iamproxy-ingest-service-account
    annotations:
      eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
server:
  remoteWrite:
    - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
      sigv4:
        region: ${REGION}
      queue_config:
        max_samples_per_send: 1000
        max_shards: 200
        capacity: 2500
```

2. Insira o comando a seguir para criar o servidor Prometheus.

- Substitua *prometheus-chart-name* pelo nome da versão do Prometheus.
- *prometheus-namespace*Substitua pelo nome do seu namespace Prometheus.

```
helm install prometheus-chart-name prometheus-community/prometheus -n prometheus-
namespace \
-f my_prometheus_values_yaml
```

> **Note**
>
> É possível personalizar o comando `helm install` de várias maneiras. Para obter mais informações, consulte Helm install na documentação do Helm.

Configurar a ingestão de um servidor Prometheus existente no Kubernetes no EC2

O Amazon Managed Service for Prometheus oferece suporte à ingestão de métricas de servidores Prometheus em clusters em execução no Amazon EKS e em clusters Kubernetes autogerenciados em execução no Amazon EC2. As instruções detalhadas nesta seção são para um servidor Prometheus em um cluster Amazon EKS. As etapas para um cluster Kubernetes autogerenciado no Amazon EC2 são as mesmas, exceto que você mesmo precisará configurar o provedor OIDC e os perfis do IAM para contas de serviço no cluster Kubernetes.

As instruções nesta seção usam o Helm como gerenciador de pacotes do Kubernetes.

Tópicos

- Etapa 1: Configurar perfis do IAM para as contas de serviço
- Etapa 2: Fazer upgrade do servidor Prometheus existente usando o Helm

Etapa 1: Configurar perfis do IAM para as contas de serviço

Para o método de integração que estamos documentando, é necessário usar perfis do IAM para as contas de serviço no cluster do Amazon EKS em que o servidor do Prometheus está em execução. Esses perfis também são chamadas de perfis de serviço.

Com os perfis de serviço, é possível associar um perfil do IAM a uma conta de serviço do Kubernetes. Essa conta de serviço pode então fornecer AWS permissões para os contêineres em qualquer pod que use essa conta de serviço. Para obter mais informações, consulte Perfis do IAM para contas de serviço.

Se você ainda não configurou esses perfis, siga as instruções em Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS para configurar os perfis.

Etapa 2: Fazer upgrade do servidor Prometheus existente usando o Helm

As instruções nesta seção incluem a configuração de gravação remota e sigv4 para autenticar e autorizar o servidor Prometheus a gravar remotamente no espaço de trabalho do Amazon Managed Service for Prometheus.

Uso do Prometheus versão 2.26.0 ou posterior

Siga estas etapas se você estiver usando um chart do Helm com imagem do servidor Prometheus da versão 2.26.0 ou posterior.

Para configurar a gravação remota de um servidor Prometheus usando o chart do Helm

1. Crie uma nova seção de gravação remota em seu arquivo de configuração do Helm:

   - `${IAM_PROXY_PROMETHEUS_ROLE_ARN}`Substitua pelo ARN do amp-iamproxy-ingest-roleque você criou em. Etapa 1: Configurar perfis do IAM para as contas de serviço O ARN do perfil deve ter o formato de `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role`.
   - Substitua `${WORKSPACE_ID}` pela ID do seu espaço de trabalho do Amazon Managed Service for Prometheus.
   - Substitua `${REGION}` pela região do espaço de trabalho do Amazon Managed Service for Prometheus (como `us-west-2`).

```
## The following is a set of default values for prometheus server helm chart which
 enable remoteWrite to AMP
    ## For the rest of prometheus helm chart values see: https://github.com/
prometheus-community/helm-charts/blob/main/charts/prometheus/values.yaml
    ##
    serviceAccounts:
      server:
        name: amp-iamproxy-ingest-service-account
        annotations:
          eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}
    server:
      remoteWrite:
        - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
          sigv4:
            region: ${REGION}
```

```
          queue_config:
            max_samples_per_send: 1000
            max_shards: 200
            capacity: 2500
```

2. Atualize sua configuração existente do servidor Prometheus usando o Helm:

   - Substitua `prometheus-chart-name` pelo nome da versão do Prometheus.
   - Substitua `prometheus-namespace` pelo namespace Kubernetes em que seu servidor Prometheus está instalado.
   - Substitua `my_prometheus_values_yaml` pelo caminho para o arquivo de configuração do Helm.
   - Substitua `current_helm_chart_version` pela versão atual do chart do Helm do servidor Prometheus. Você pode encontrar a versão atual do gráfico usando o comando helm list.

```
helm upgrade prometheus-chart-name prometheus-community/prometheus \
       -n prometheus-namespace \
       -f my_prometheus_values_yaml \
       --version current_helm_chart_version
```

Usar versões anteriores do Prometheus

Siga estas etapas se você estiver usando uma versão do Prometheus anterior à 2.26.0. Essas etapas usam uma abordagem secundária, porque as versões anteriores do Prometheus não oferecem suporte nativo ao processo de AWS assinatura Signature Version 4 (SigV4).AWS

Essas instruções pressupõem que você está usando o Helm para implantar o Prometheus.

Para configurar a gravação remota de um servidor Prometheus

1. No seu servidor Prometheus, crie uma nova configuração de gravação remota. Primeiro, crie um novo arquivo de atualização. Chamaremos o arquivo de `amp_ingest_override_values.yaml`.

   Adicione os valores a seguir ao arquivo YAML.

```
serviceAccounts:
        server:
            name: "amp-iamproxy-ingest-service-account"
```

```
            annotations:
                eks.amazonaws.com/role-arn:
 "${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}"
    server:
        sidecarContainers:
            - name: aws-sigv4-proxy-sidecar
              image: public.ecr.aws/aws-observability/aws-sigv4-proxy:1.0
              args:
              - --name
              - aps
              - --region
              - ${REGION}
              - --host
              - aps-workspaces.${REGION}.amazonaws.com
              - --port
              - :8005
              ports:
              - name: aws-sigv4-proxy
                containerPort: 8005
        statefulSet:
            enabled: "true"
        remoteWrite:
            - url: http://localhost:8005/workspaces/${WORKSPACE_ID}/api/v1/
remote_write
```

Substitua `${REGION}` pela Região do espaço de trabalho do Amazon Managed Service for Prometheus.

`${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}`Substitua pelo ARN do amp-iamproxy-ingest-roleque você criou em. Etapa 1: Configurar perfis do IAM para as contas de serviço O ARN do perfil deve ter o formato de `arn:aws:iam::`*`your account ID`*`:role/amp-iamproxy-ingest-role`.

Substitua `${WORKSPACE_ID}` pelo ID do seu espaço de trabalho.

2. Faça o upgrade do seu chart do Helm do Prometheus. Primeiro, encontre o nome do chart do Helm digitando o comando a seguir. Na saída desse comando, procure um gráfico com um nome que inclua `prometheus`.

```
helm ls --all-namespaces
```

Depois, insira o comando a seguir.

```
helm upgrade --install prometheus-helm-chart-name prometheus-community/prometheus -
n prometheus-namespace -f ./amp_ingest_override_values.yaml
```

*prometheus-helm-chart-name*Substitua pelo nome do gráfico do leme do Prometheus retornado no comando anterior. Substitua *prometheus-namespace* pelo nome do seu namespace.

Download de charts do Helm

Se você ainda não tiver baixado os charts do Helm localmente, você pode usar o comando a seguir para baixá-los.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
    helm pull prometheus-community/prometheus --untar
```

Configurar a ingestão de um servidor Prometheus existente no Kubernetes no Fargate

O Amazon Managed Service for Prometheus oferece suporte à ingestão de métricas de servidores Prometheus em clusters Kubernetes autogerenciados em execução no Fargate. Para ingerir métricas dos servidores Prometheus em clusters Amazon EKS executados no Fargate, substitua as configurações padrão em um arquivo de configuração chamado amp_ingest_override_values.yaml da seguinte forma:

```
prometheus-node-exporter:
        enabled: false

    alertmanager:
        enabled: false

    serviceAccounts:
      server:
        name: amp-iamproxy-ingest-service-account
        annotations:
          eks.amazonaws.com/role-arn: ${IAM_PROXY_PROMETHEUS_ROLE_ARN}

    server:
      persistentVolume:
        enabled: false
      remoteWrite:
```

```
        - url: https://aps-workspaces.${REGION}.amazonaws.com/workspaces/
${WORKSPACE_ID}/api/v1/remote_write
          sigv4:
            region: ${REGION}
          queue_config:
            max_samples_per_send: 1000
            max_shards: 200
            capacity: 2500
```

Instalar o Prometheus usando as sobreposições com o seguinte comando:

```
helm install prometheus-for-amp prometheus-community/prometheus \
                  -n prometheus \
                  -f amp_ingest_override_values.yaml
```

Observe que, na configuração do chart do Helm, desativamos o exportador de nós e o gerenciador de alertas, além de executar a implantação do servidor Prometheus.

Você pode verificar a instalação com o exemplo de consulta de teste a seguir.

```
$ awscurl --region region --service aps "https://aps-
workspaces.region_id.amazonaws.com/workspaces/workspace_id/api/v1/query?
query=prometheus_api_remote_read_queries"
    {"status":"success","data":{"resultType":"vector","result":[{"metric":
{"__name__":"prometheus_api_remote_read_queries","instance":"localhost:9090","job":"prometheus"
[1648461236.419,"0"]}]}}21
```

Configurar o Amazon Managed Service for Prometheus para dados de alta disponibilidade

Quando você envia dados para o Amazon Managed Service for Prometheus, eles são automaticamente replicados em todas as zonas de disponibilidade da AWS na região e são servidos a você a partir de um cluster de hosts que fornecem escalabilidade, disponibilidade e segurança. Talvez você queira adicionar outros dispositivos de proteção contra falhas de alta disponibilidade, dependendo da configuração específica. Há duas maneiras comuns de adicionar seguranças de alta disponibilidade à configuração:

- Se você tiver vários contêineres ou instâncias com os mesmos dados, poderá enviar esses dados para o Amazon Managed Service for Prometheus e fazer com que as duplicatas dos dados sejam

automaticamente eliminadas. Isso ajuda a garantir que seus dados sejam enviados para o espaço de trabalho do Amazon Managed Service for Prometheus.

Para obter mais informações sobre a eliminação de duplicatas de dados de alta disponibilidade, consulte Eliminar a duplicação de métricas de alta disponibilidade enviadas para o Amazon Managed Service for Prometheus.

- Se você quiser garantir o acesso aos dados, mesmo quando a região da AWS não estiver disponível, poderá enviar as métricas para um segundo espaço de trabalho, em outra região.

  Para obter mais informações sobre o envio de dados de métricas para vários espaços de trabalho, consulte Use espaços de trabalho entre Regiões para adicionar alta disponibilidade no Amazon Managed Service for Prometheus.

Tópicos

Eliminar a duplicação de métricas de alta disponibilidade enviadas para o Amazon Managed Service for Prometheus

Você pode enviar dados de vários atendentes do Prometheus (instâncias do Prometheus em execução no modo Atendente) para o seu espaço de trabalho do Amazon Managed Service for Prometheus. Se algumas dessas instâncias estiverem registrando e enviando as mesmas métricas,

seus dados terão uma disponibilidade maior (mesmo que um dos atendentes pare de enviar dados, o espaço de trabalho do Amazon Managed Service for Prometheus ainda receberá os dados de outra instância). No entanto, você quer que seu espaço de trabalho do Amazon Managed Service for Prometheus elimine automaticamente a duplicação das métricas para que você não veja as métricas várias vezes e não seja cobrado pela ingestão e armazenamento de dados várias vezes.

Para que o Amazon Managed Service for Prometheus elimine automaticamente a duplicação de dados de vários atendentes do Prometheus, você atribui ao conjunto de atendentes que estão enviando os dados duplicados um único nome de cluster e a cada uma das instâncias um nome de réplica. O nome do cluster identifica as instâncias como tendo dados compartilhados, e o nome da réplica permite que o Amazon Managed Service for Prometheus identifique a origem de cada métrica. As métricas finais armazenadas incluem o rótulo do cluster, mas não a réplica, de modo que as métricas parecem estar vindo de uma única fonte.

> **Note**
>
> Determinadas versões do Kubernetes (1.28 e 1.29) podem emitir sua própria métrica com um rótulo `cluster`. Isso pode causar problemas com a desduplicação do Amazon Managed Service for Prometheus. Para obter mais informações, consulte High availability FAQ.

Os tópicos a seguir mostram como enviar dados e incluir os rótulos `cluster` e `__replica__` de modo que o Amazon Managed Service for Prometheus desduplique os dados automaticamente.

> **Important**
>
> Se você não configurar a eliminação de duplicatas, você será cobrado por todas as amostras de dados enviadas ao Amazon Managed Service for Prometheus. Essas amostras de dados incluem amostras duplicadas.

Enviar dados de alta disponibilidade para o Amazon Managed Service for Prometheus com o Prometheus

Para definir uma configuração de alta disponibilidade com o Prometheus, é necessário aplicar rótulos externos em todas as instâncias de um grupo de alta disponibilidade, para que o Amazon Managed Service for Prometheus possa identificá-las. Use o rótulo `cluster` para identificar um agente de instância do Prometheus como parte de um grupo de alta disponibilidade. Use o rótulo

`__replica__` para identificar cada réplica no grupo separadamente. Você precisa aplicar os rótulos `__replica__` e `cluster` para que a eliminação de duplicatas funcione.

> **Note**
>
> O rótulo `__replica__` é formatado com dois símbolos de sublinhado antes e depois da palavra `replica`.

Exemplo: trechos de código

Nos trechos de código a seguir, o rótulo `cluster` identifica o atendente de instância `prom-team1` do Prometheus, e o rótulo `_replica_` identifica as réplicas `replica1` e `replica2`.

```
cluster: prom-team1
__replica__: replica1
```

```
cluster: prom-team1
__replica__: replica2
```

Quando o Amazon Managed Service for Prometheus armazena amostras de dados de réplicas de alta disponibilidade com esses rótulos, ele retira o rótulo `replica` quando as amostras são aceitas. Isso significa que você só terá um mapeamento de série 1:1 para sua série atual, em vez de uma série por réplica. O rótulo `cluster` é mantido.

> **Note**
>
> Determinadas versões do Kubernetes (1.28 e 1.29) podem emitir sua própria métrica com um rótulo `cluster`. Isso pode causar problemas com a desduplicação do Amazon Managed Service for Prometheus. Para obter mais informações, consulte High availability FAQ.

Configurar dados de alta disponibilidade para o Amazon Managed Service for Prometheus usando o chart do Helm do Prometheus Operator

Para definir uma configuração de alta disponibilidade com o Prometheus Operator no Helm, você deve aplicar rótulos externos em todas as instâncias de um grupo de alta disponibilidade, para que

o Amazon Managed Service for Prometheus possa identificá-las. Você também deve definir os atributos `replicaExternalLabelName` e `externalLabels` o chart do Helm no Prometheus Operator.

Exemplo: cabeçalho YAML

No cabeçalho YAML a seguir, `cluster` é adicionado a `externalLabel` para identificar um atendente de instância do Prometheus como parte de um grupo de alta disponibilidade, e `replicaExternalLabels` identifica cada réplica no grupo.

```
replicaExternalLabelName: __replica__
externalLabels:
cluster: prom-dev
```

> **Note**
>
> Determinadas versões do Kubernetes (1.28 e 1.29) podem emitir sua própria métrica com um rótulo `cluster`. Isso pode causar problemas com a desduplicação do Amazon Managed Service for Prometheus. Para obter mais informações, consulte High availability FAQ.

Envie dados de alta disponibilidade para o Amazon Managed Service for AWS Prometheus com o Distro for OpenTelemetry

AWS Distro for OpenTelemetry (ADOT) é uma distribuição segura e pronta para produção do projeto. OpenTelemetry O ADOT fornece fontes APIs, bibliotecas e agentes, para que você possa coletar rastreamentos e métricas distribuídos para o monitoramento de aplicativos. Para obter informações sobre ADOT, consulte Sobre o AWS Distro for Open Telemetry.

Para configurar o ADOT com uma configuração de alta disponibilidade, você deve configurar uma imagem de contêiner do coletor ADOT e aplicar os rótulos externos ao exportador de `cluster` gravação `__replica__` remoto Prometheus AWS . Esse exportador envia suas métricas coletadas para o espaço de trabalho do Amazon Managed Service for Prometheus por meio do endpoint `remote_write`. Ao definir esses rótulos no exportador de gravação remota, você evita que métricas duplicadas sejam mantidas enquanto réplicas redundantes são executadas. Para obter mais informações sobre o exportador de gravação remota AWS Prometheus, consulte Introdução ao exportador de gravação remota Prometheus para o Amazon Managed Service for Prometheus.

> **Note**
>
> Determinadas versões do Kubernetes (1.28 e 1.29) podem emitir sua própria métrica com um rótulo `cluster`. Isso pode causar problemas com a desduplicação do Amazon Managed Service for Prometheus. Para obter mais informações, consulte High availability FAQ.

Enviar dados de alta disponibilidade para o Amazon Managed Service for Prometheus com o chart do Helm da comunidade do Prometheus

Para definir uma configuração de alta disponibilidade com o chart do Helm da comunidade do Prometheus, é necessário aplicar rótulos externos em todas as instâncias de um grupo de alta disponibilidade, de modo que o Amazon Managed Service for Prometheus possa identificá-las. Aqui está um exemplo de como você pode adicionar o `external_labels` a uma única instância do Prometheus do chart do Helm da comunidade do Prometheus.

```
server:
global:
  external_labels:
      cluster: monitoring-cluster
      __replica__: replica-1
```

> **Note**
>
> Se você quiser várias réplicas, precisará implantar o gráfico várias vezes com valores de réplica diferentes, pois o chart do Helm da comunidade do Prometheus não permite que você defina dinamicamente o valor da réplica ao aumentar o número de réplicas diretamente do grupo controlador. Se você preferir que o rótulo `replica` seja configurado automaticamente, use o chart do Helm prometheus-operator.

> **Note**
>
> Determinadas versões do Kubernetes (1.28 e 1.29) podem emitir sua própria métrica com um rótulo `cluster`. Isso pode causar problemas com a desduplicação do Amazon Managed Service for Prometheus. Para obter mais informações, consulte High availability FAQ.

Respostas a perguntas comuns sobre a configuração de alta disponibilidade no Amazon Managed Service for Prometheus

Devo incluir o valor __replica__ em outro rótulo para rastrear os pontos de amostra?

Em uma configuração de alta disponibilidade, o Amazon Managed Service for Prometheus garante que as amostras de dados não sejam duplicadas ao eleger um líder no cluster de instâncias do Prometheus. Se a réplica líder parar de enviar amostras de dados por 30 segundos, o Amazon Managed Service for Prometheus automaticamente transforma outra instância do Prometheus em uma réplica líder e ingere dados do novo líder, incluindo quaisquer dados perdidos. Portanto, a resposta é não, isso não é recomendado. Fazer isso pode causar problemas como:

- Consultar um `count` no PromQL pode retornar um valor maior do que o esperado durante o período de eleição de um novo líder.
- O número de `active series` aumenta durante o período de eleição de um novo líder e atinge o `active series limits`. Para obter mais informações, consulte Cotas do AMP.

O Kubernetes parece ter seu próprio rótulo de cluster e não está desduplicando minhas métricas. Como corrijo isso?

Uma nova métrica `apiserver_storage_size_bytes` foi introduzida no Kubernetes 1.28, com um rótulo `cluster`. Isso pode causar problemas com a desduplicação no Amazon Managed Service for Prometheus, que depende do rótulo `cluster`. No Kubernetes 1.3, o rótulo é renomeado para `storage-cluster_id` (ele também é renomeado em patches posteriores da 1.28 e 1.29). Se seu cluster estiver emitindo essa métrica com o rótulo `cluster`, o Amazon Managed Service for Prometheus não poderá desduplicar a série temporal associada. Recomendamos que você atualize seu cluster do Kubernetes para a versão corrigida mais recente para evitar esse problema. Outra possibilidade é renomear o rótulo `cluster` em sua métrica `apiserver_storage_size_bytes` antes de inseri-lo no Amazon Managed Service for Prometheus.

> **Note**
>
> Para obter mais detalhes sobre a mudança no Kubernetes, consulte Renomear o cluster Label para storage_cluster_id para a métrica apiserver_storage_size_bytes no projeto Kubernetes. GitHub

Use espaços de trabalho entre Regiões para adicionar alta disponibilidade no Amazon Managed Service for Prometheus

Para adicionar disponibilidade entre regiões aos seus dados, você pode enviar métricas para vários espaços de trabalho em AWS todas as regiões. O Prometheus oferece suporte tanto para vários gravadores quanto para gravação entre regiões.

O exemplo a seguir mostra como configurar um servidor Prometheus em execução no modo Agente para enviar métricas para dois espaços de trabalho em regiões diferentes com o Helm.

```
extensions:
      sigv4auth:
        service: "aps"

    receivers:
      prometheus:
        config:
          scrape_configs:
            - job_name: 'kubernetes-kubelet'
              scheme: https
              tls_config:
                ca_file: /var/run/secrets/kubernetes.io/serviceaccount/ca.crt
                insecure_skip_verify: true
              bearer_token_file: /var/run/secrets/kubernetes.io/serviceaccount/token
              kubernetes_sd_configs:
              - role: node
              relabel_configs:
              - action: labelmap
                regex: __meta_kubernetes_node_label_(.+)
              - target_label: __address__
                replacement: kubernetes.default.svc.cluster.local:443
              - source_labels: [__meta_kubernetes_node_name]
                regex: (.+)
                target_label: __metrics_path__
                replacement: /api/v1/nodes/$${1}/proxy/metrics

    exporters:
      prometheusremotewrite/one:
        endpoint: "https://aps-workspaces.workspace_1_region.amazonaws.com/workspaces/
ws-workspace_1_id/api/v1/remote_write"
        auth:
          authenticator: sigv4auth
      prometheusremotewrite/two:
```

```
        endpoint: "https://aps-workspaces.workspace_2_region.amazonaws.com/workspaces/
ws-workspace_2_id/api/v1/remote_write"
        auth:
          authenticator: sigv4auth

    service:
      extensions: [sigv4auth]
      pipelines:
        metrics/one:
          receivers: [prometheus]
          exporters: [prometheusremotewrite/one]
        metrics/two:
          receivers: [prometheus]
          exporters: [prometheusremotewrite/two]
```

Consultar as métricas do Prometheus

Agora que as métricas estão sendo ingeridas no espaço de trabalho, você pode consultá-las.

Para criar painéis com representações visuais de suas métricas, você pode usar um serviço como o Amazon Managed Grafana. O Amazon Managed Grafana (ou uma instância autônoma do Grafana) pode criar uma interface gráfica que mostra suas métricas em uma ampla variedade de estilos de apresentação. Para obter mais informações sobre o Amazon Managed Grafana, consulte o Guia do usuário do Amazon Managed Grafana.

Você também pode criar consultas pontuais, explorar seus dados ou escrever seus próprios aplicativos que usam suas métricas via consultas diretas. As consultas diretas usam a API do Amazon Managed Service for Prometheus e a linguagem de consulta padrão do Prometheus, PromQL, para obter dados do seu espaço de trabalho do Prometheus. Para obter mais informações sobre o PromQL e sua sintaxe, veja Consultando Prometheus na documentação do Prometheus.

Tópicos

Folha de dicas do PromQL

Use esta folha de dicas do PromQL (Prometheus Query Language) como referência rápida ao consultar métricas em seu espaço de trabalho do Amazon Managed Service for Prometheus. Com o PromQL, você pode selecionar e agregar dados de séries temporais em tempo real com a linguagem de consulta funcional dele.

Para obter mais detalhes sobre o PromQL, consulte a folha de dicas do PromQL no site. PromLabs

Seletores básicos

Selecione séries temporais por nome de métrica e correspondências de rótulos:

```
# Select all time series with the metric name http_requests_total
http_requests_total

# Select time series with specific label values
http_requests_total{job="prometheus", method="GET"}

# Use label matchers
http_requests_total{status_code!="200"}          # Not equal
http_requests_total{status_code=~"2.."}          # Regex match
http_requests_total{status_code!~"4.."}          # Negative regex match
```

Seletores de vetores de alcance

Selecione uma variedade de amostras ao longo do tempo:

```
# Select 5 minutes of data
http_requests_total[5m]

# Time units: s (seconds), m (minutes), h (hours), d (days), w (weeks), y (years)
cpu_usage[1h]
memory_usage[30s]
```

Operadores de agregação

Agregue dados em várias séries temporais:

```
# Sum all values
sum(http_requests_total)

# Sum by specific labels
sum by (job) (http_requests_total)
sum without (instance) (http_requests_total)

# Other aggregation operators
avg(cpu_usage)                    # Average
min(response_time)                # Minimum
max(response_time)                # Maximum
count(up)                         # Count of series
stddev(cpu_usage)                 # Standard deviation
```

Funções comuns

Aplique funções para transformar seus dados:

```
# Rate of increase per second (for counters)
rate(http_requests_total[5m])

# Increase over time range
increase(http_requests_total[1h])

# Derivative (for gauges)
deriv(cpu_temperature[5m])

# Mathematical functions
abs(cpu_usage - 50)               # Absolute value
round(cpu_usage, 0.1)             # Round to nearest 0.1
sqrt(memory_usage)                # Square root

# Time functions
time()                            # Current Unix timestamp
hour()                            # Hour of day (0-23)
```

```
day_of_week()                    # Day of week (0-6, Sunday=0)
```

Operadores binários

Execute operações aritméticas e lógicas:

```
# Arithmetic operators
cpu_usage + 10
memory_total - memory_available
disk_usage / disk_total * 100

# Comparison operators (return 0 or 1)
cpu_usage > 80
memory_usage < 1000
response_time >= 0.5

# Logical operators
(cpu_usage > 80) and (memory_usage > 1000)
(status_code == 200) or (status_code == 201)
```

Exemplos de consultas práticas

Consultas de monitoramento comuns que você pode usar no seu espaço de trabalho do Amazon Managed Service for Prometheus:

```
# CPU usage percentage
100 - (avg by (instance) (rate(node_cpu_seconds_total{mode="idle"}[5m])) * 100)

# Memory usage percentage
(1 - (node_memory_MemAvailable_bytes / node_memory_MemTotal_bytes)) * 100

# Request rate per second
sum(rate(http_requests_total[5m])) by (job)

# Error rate percentage
sum(rate(http_requests_total{status_code=~"5.."}[5m])) /
sum(rate(http_requests_total[5m])) * 100
```

```
# 95th percentile response time
histogram_quantile(0.95, sum(rate(http_request_duration_seconds_bucket[5m])) by (le))

# Top 5 instances by CPU usage
topk(5, avg by (instance) (cpu_usage))
```

Proteger suas consultas de métricas

O Amazon Managed Service for Prometheus oferece maneiras de ajudar você a proteger a consulta de suas métricas.

Utilizar AWS PrivateLink com o Amazon Managed Service para Prometheus

O tráfego de rede para consultar métricas no Amazon Managed Service for Prometheus pode ser feito por meio de um endpoint público da Internet ou por meio de um endpoint VPC. AWS PrivateLink Quando você usa AWS PrivateLink, o tráfego de rede de suas VPCs é protegido na AWS rede sem passar pela Internet pública. Para criar um AWS PrivateLink VPC endpoint para o Amazon Managed Service for Prometheus, consulte. Como utilizar o Amazon Managed Service for Prometheus com endpoints da VPC de interface

Autenticação e autorização

AWS Identity and Access Management é um serviço web que ajuda você a controlar com segurança o acesso aos AWS recursos. Com o IAM, você controla quem pode se autenticar (fazer login) e quem tem autorização (permissões) para acessar os recursos. O Amazon Managed Service for Prometheus se integra ao IAM para ajudar manter seus dados protegidos. Ao configurar o Amazon Managed Service for Prometheus, você precisará criar alguns perfis do IAM que permitam que os servidores Grafana consultem métricas armazenadas nos espaços de trabalho do Amazon Managed Service for Prometheus. Para obter mais informações sobre o IAM, consulte O que é o IAM?

Outro recurso AWS de segurança que pode ajudar você a configurar o Amazon Managed Service para Prometheus é AWS o processo AWS de assinatura Signature Version 4 (SigV4). A versão 4 do Signature é o processo para adicionar informações de autenticação às AWS solicitações enviadas por HTTP. Por motivos de segurança, a maioria das solicitações AWS deve ser assinada com uma chave de acesso, que consiste em uma ID da chave de acesso e uma chave de acesso secreta.

Essas duas chaves são comumente conhecidas como suas credenciais de segurança. Para obter mais informações sobre o SigV4, consulte Processo de assinatura do Signature Version 4.

Configurar o Amazon Managed Grafana para uso com o Amazon Managed Service for Prometheus

O Amazon Managed Grafana é um serviço totalmente gerenciado para o Grafana de código aberto que simplifica a conexão com ISVs de código aberto de terceiros AWS e serviços para visualizar e analisar suas fontes de dados em grande escala.

O Amazon Managed Service for Prometheus oferece suporte ao uso do Amazon Managed Grafana para consultar métricas em um espaço de trabalho. No console do Amazon Managed Grafana, você pode adicionar um espaço de trabalho do Amazon Managed Service for Prometheus como fonte de dados descobrindo suas contas existentes do Amazon Managed Service for Prometheus. O Amazon Managed Grafana gerencia a configuração das credenciais de autenticação necessárias para acessar o Amazon Managed Service for Prometheus. Para obter instruções detalhadas sobre como criar uma conexão com o Amazon Managed Service for Prometheus a partir do Amazon Managed Grafana, consulte as instruções no Guia do usuário do Amazon Managed Grafana.

Você também pode visualizar seus alertas do Amazon Managed Service for Prometheus no Amazon Managed Grafana. Para obter instruções sobre como configurar a integração com alertas, consulte Integrar alertas com o Amazon Managed Grafana ou o Grafana de código aberto.

Conexão com o Amazon Managed Grafana em uma VPC privada

O Amazon Managed Service for Prometheus fornece um endpoint de serviço ao qual o Amazon Managed Grafana pode se conectar ao consultar métricas e alertas.

Você pode configurar o Amazon Managed Grafana para usar uma VPC privada (para obter detalhes sobre como configurar uma VPC privada no Grafana, consulte Conexão com a Amazon VPC no Guia do usuário do Amazon Managed Grafana). Dependendo das configurações, essa VPC pode não ter acesso ao endpoint de serviço do Amazon Managed Service for Prometheus.

Para adicionar o Amazon Managed Service for Prometheus como fonte de dados a um espaço de trabalho do Amazon Managed Grafana configurado para usar uma VPC privada específica, primeiro é preciso conectar o Amazon Managed Service for Prometheus à mesma VPC criando um endpoint da VPC. Para obter mais informações sobre como criar um endpoint da VPC, consulte Criar um endpoint da VPC de interface para o Amazon Managed Service for Prometheus.

Configurar o Grafana de código aberto ou o Grafana Enterprise para uso com o Amazon Managed Service for Prometheus

Você pode usar uma instância do Grafana para consultar suas métricas no Amazon Managed Service for Prometheus. Este tópico mostra como consultar métricas do Amazon Managed Service for Prometheus usando uma instância autônoma do Grafana.

Pré-requisitos

Instância do Grafana: você deve ter uma instância do Grafana capaz de se autenticar com o Amazon Managed Service for Prometheus.

O Amazon Managed Service for Prometheus oferece suporte ao uso do Grafana versão 7.3.5 e posterior para consultar métricas em um espaço de trabalho. As versões 7.3.5 e posteriores incluem suporte para autenticação AWS Signature Version 4 (SigV4).

Para verificar sua versão do Grafana, digite o seguinte comando, *grafana_install_directory* substituindo pelo caminho para a instalação do Grafana:

```
grafana_install_directory/bin/grafana-server -v
```

Se você ainda não tem um Grafana independente ou precisa de uma versão mais recente, instale uma nova instância. Para obter instruções sobre como configurar um Grafana independente, consulte Instalar o Grafana na documentação do Grafana. Para obter informações sobre os conceitos básicos do Grafana, consulte Conceitos básicos do Grafana na documentação do Grafana.

Conta da AWS: você deve ter uma Conta da AWS com as permissões corretas para acessar suas métricas do Amazon Managed Service for Prometheus.

Para configurar o Grafana para funcionar com o Amazon Managed Service for Prometheus, você deve estar conectado a uma conta que tenha a AmazonPrometheusQueryAccesspolítica ou as permissões,, e. `aps:QueryMetrics` `aps:GetMetricMetadata` `aps:GetSeries` `aps:GetLabels` Para obter mais informações, consulte Permissões e políticas no IAM.

A próxima seção descreve com mais detalhes como configurar a autenticação do Grafana.

Etapa 1: Configurar AWS SigV4

O Amazon Managed Service for Prometheus trabalha AWS Identity and Access Management com (IAM) para proteger todas as chamadas para as APIs do Prometheus com credenciais do IAM.

Por padrão, a fonte de dados do Prometheus no Grafana presume que o Prometheus não requer autenticação. Para permitir que o Grafana aproveite os recursos de autenticação e autorização do Amazon Managed Service for Prometheus, você precisará habilitar o suporte à autenticação SigV4 na fonte de dados do Grafana. Siga as etapas desta página ao usar um servidor de código aberto autogerenciado do Grafana ou um servidor corporativo do Grafana. Se você estiver usando o Amazon Managed Grafana, a autenticação SIGv4 será totalmente automatizada. Para obter mais informações sobre o Amazon Managed Grafana, consulte What is Amazon Managed Grafana?

Para habilitar o SigV4 no Grafana, inicie o Grafana com as variáveis de ambiente `AWS_SDK_LOAD_CONFIG` e `GF_AUTH_SIGV4_AUTH_ENABLED` definidas como `true`. A variável de ambiente `GF_AUTH_SIGV4_AUTH_ENABLED` substitui a configuração padrão do Grafana para habilitar o suporte ao SigV4. Para obter mais informações, consulte Configuração na documentação do Grafana.

Linux

Para habilitar o SigV4 em um servidor Grafana autônomo no Linux, digite os seguintes comandos.

```
export AWS_SDK_LOAD_CONFIG=true
```

```
export GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
./bin/grafana-server
```

Windows

Para habilitar o SigV4 em um Grafana autônomo no Windows usando o prompt de comando do Windows, digite os comandos a seguir.

```
set AWS_SDK_LOAD_CONFIG=true
```

```
set GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

```
.\bin\grafana-server.exe
```

Etapa 2: adicionar a fonte de dados do Prometheus no Grafana

As etapas a seguir explicam como configurar a fonte de dados do Prometheus no Grafana para consultar suas métricas do Amazon Managed Service for Prometheus.

Para adicionar a fonte de dados do Prometheus no servidor Grafana

1. Abra o console do Grafana.
2. Em Configurações, escolha Fontes de dados.
3. Escolha Adicionar fonte de dados.
4. Escolha Prometheus.
5. Para o URL HTTP, especifique o Endpoint - URL de consulta exibido na página de detalhes do espaço de trabalho no console do Amazon Managed Service for Prometheus.
6. No URL HTTP que você acabou de especificar, remova a string `/api/v1/query` anexada ao URL, pois a fonte de dados do Prometheus a anexará automaticamente.

   O URL correto deve ser semelhante https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-1234a5b6-78cd-901e-2fgh-3i45j6k178l9a.
7. Em Auth, selecione o botão de alternância do SigV4 Auth para ativá-lo.
8. Você pode configurar a autorização do SigV4 especificando suas credenciais de longo prazo diretamente no Grafana ou usando uma cadeia de fornecedores padrão. Especificar suas credenciais de longo prazo diretamente ajuda você a começar mais rápido, e as etapas a seguir fornecem essas instruções primeiro. Quando você estiver mais familiarizado com o uso do Grafana com o Amazon Managed Service for Prometheus, recomendamos que você use uma cadeia de fornecedores padrão, pois ela oferece maior flexibilidade e segurança. Para obter mais informações sobre a configuração da cadeia de fornecedores padrão, consulte Especificar credenciais.
   - Para usar suas credenciais de longo prazo diretamente, faça o seguinte:
     a. Em Detalhes do SigV4 Auth, em Provedor de autenticação, escolha Acesso e chave secreta.
     b. Em ID da chave de acesso, informe o ID da chave de acesso do AWS .
     c. Em Chave de acesso secreta, informe sua chave de acesso secreta do AWS .

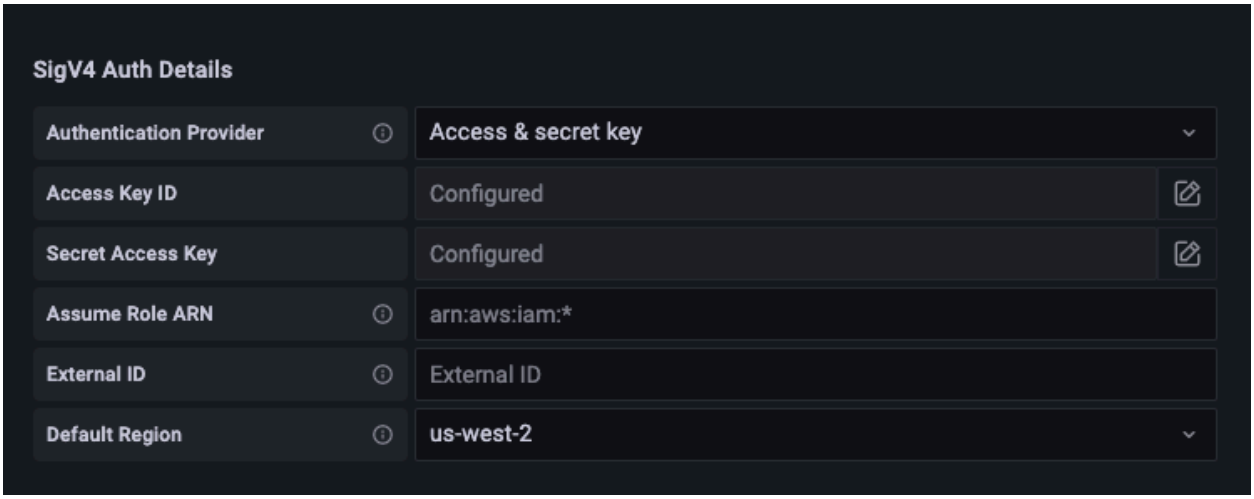
d. Deixe os campos Presumir ARN do perfil e ID externo em branco.

e. Em Região padrão, escolha a Região do seu espaço de trabalho do Amazon Managed Service for Prometheus. Essa região deve corresponder à região contida no URL que você listou na etapa 5.

f. Escolha Salvar e testar.

Você deverá ver a seguinte mensagem: A fonte de dados está funcionando

A captura de tela a seguir mostra a configuração de detalhes de autenticação da chave de acesso e da chave secreta do SigV4.



- Para usar uma cadeia de fornecedores padrão em vez disso (recomendada para um ambiente de produção), faça o seguinte:

a. Em Detalhes do SigV4 Auth, em Provedor de autenticação, escolha Padrão de SDK do AWS .

b. Deixe os campos Presumir ARN do perfil e ID externo em branco.

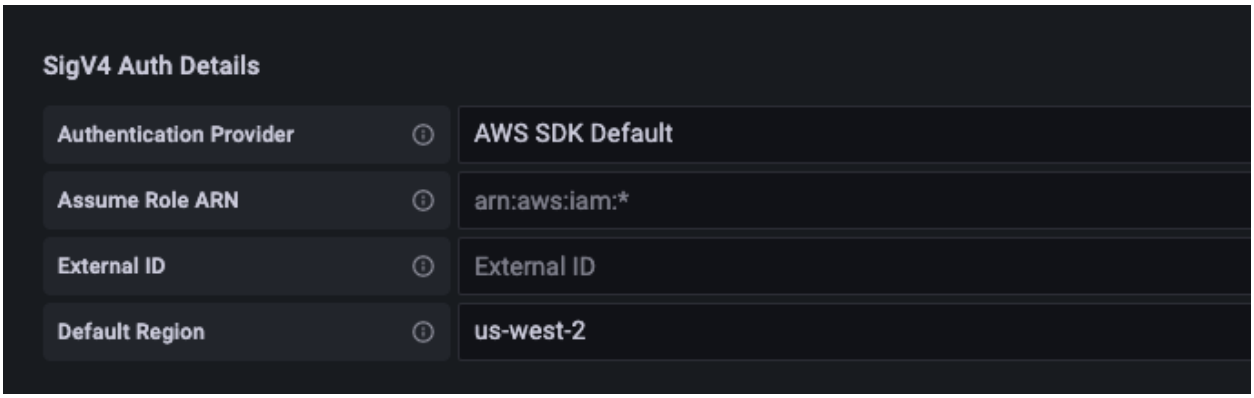
c. Em Região padrão, escolha a Região do seu espaço de trabalho do Amazon Managed Service for Prometheus. Essa região deve corresponder à região contida no URL que você listou na etapa 5.

d. Escolha Salvar e testar.

Você deverá ver a seguinte mensagem: A fonte de dados está funcionando

Se você não vir essa mensagem, confira a próxima seção com dicas para resolver problemas de conexão.

A captura de tela a seguir mostra a configuração dos detalhes de autenticação do SigV4 padrão do SDK.



9. Teste uma consulta PromQL contra a nova fonte de dados:

   a. Escolha Explorar.

   b. Execute um exemplo de consulta PromQL, como:

```
prometheus_tsdb_head_series
```

Etapa 3: (opcional) Solução de problemas se o Save & Test não funcionar

No procedimento anterior, se você encontrar um erro ao escolher Salvar e testar, verifique o seguinte.

HTTP Error Not Found

Verifique se o ID do espaço de trabalho no URL está correto.

HTTP Error Forbidden

Esse erro significa que as credenciais não são válidas. Verifique o seguinte:

- Verifique se a região especificada em Região padrão está correta.
- Verifique se há erros de digitação em sua credencial.
- Certifique-se de que a credencial que você está usando tenha a AmazonPrometheusQueryAccesspolítica. Para obter mais informações, consulte Permissões e políticas no IAM.

- Certifique-se de que a credencial que você está usando tenha acesso a esse espaço de trabalho do Amazon Managed Service for Prometheus.

HTTP Error Bad Gateway

Veja o log do servidor Grafana para solucionar esse erro. Para obter mais informações, consulte Solução de problemas na documentação do Grafana.

Se você ver**`Error http: proxy error: NoCredentialProviders: no valid providers in chain`**, a cadeia de provedores de credenciais padrão não conseguiu encontrar uma AWS credencial válida para usar. Certifique-se de ter configurado suas credenciais conforme documentado em Especificação de credenciais. Se você quiser usar uma configuração compartilhada, verifique se o ambiente `AWS_SDK_LOAD_CONFIG` está definido como `true`.

Consulta usando Grafana em execução em um cluster do Amazon EKS

O Amazon Managed Service for Prometheus oferece suporte ao uso do Grafana versão 7.3.5 e posteriores para consultar métricas em seu espaço de trabalho. As versões 7.3.5 e posteriores incluem suporte para autenticação AWS Signature Version 4 (SigV4).

Para configurar o Grafana para funcionar com o Amazon Managed Service for Prometheus, você deve estar conectado a uma conta que tenha a AmazonPrometheusQueryAccesspolítica ou as permissões,, e. `aps:QueryMetrics` `aps:GetMetricMetadata` `aps:GetSeries` `aps:GetLabels` Para obter mais informações, consulte Permissões e políticas no IAM.

Configurar AWS SigV4

A Grafana adicionou um novo recurso para oferecer suporte à autenticação AWS Signature Version 4 (SigV4). Para obter mais informações, consulte Processo de assinatura do Signature Version 4. Este atributo não está habilitado nos servidores Grafana por padrão. As instruções a seguir para habilitar esse atributo pressupõem que você esteja usando o Helm para implantar o Grafana em um cluster Kubernetes.

Para habilitar o SigV4 em seu servidor Grafana 7.3.5 ou posterior

1. Crie um novo arquivo de atualização para substituir sua configuração do Grafana e chame-o de `amp_query_override_values.yaml`.

2. Insira o conteúdo a seguir no arquivo e salve o arquivo. *account-id*Substitua pelo ID da AWS conta em que o servidor Grafana está sendo executado.

```
serviceAccount:
    name: "amp-iamproxy-query-service-account"
    annotations:
        eks.amazonaws.com/role-arn: "arn:aws:iam::account-id:role/amp-iamproxy-
query-role"
grafana.ini:
  auth:
    sigv4_auth_enabled: true
```

   Nesse conteúdo do arquivo YAML, `amp-iamproxy-query-role` é o nome do perfil que você criará na próxima seção, Configure perfis do IAM para as contas de serviço. Você pode substituir esse perfil pelo seu próprio nome de perfil, caso já tenha criado um perfil para consultar seu espaço de trabalho.

Você usará esse arquivo posteriormente, em Atualizar o servidor Grafana usando o Helm.

Configure perfis do IAM para as contas de serviço

Se você estiver usando um servidor Grafana em um cluster Amazon EKS, recomendamos que use perfis do IAM para contas de serviço, também conhecidas como perfis de serviço, para seu controle de acesso. Quando você faz isso para associar uma função do IAM a uma conta de serviço do Kubernetes, a conta de serviço pode então fornecer AWS permissões aos contêineres em qualquer pod que use essa conta de serviço. Para obter mais informações, consulte Perfis do IAM para contas de serviço.

Se você ainda não configurou esses perfis de serviço para consulta, siga as instruções em Configure perfis do IAM para contas de serviço para consulta de métricas para configurar os perfis.

Em seguida, você precisa adicionar a conta de serviço do Grafana nas condições da relação de confiança.

Para adicionar a conta de serviço do Grafana nas condições da relação de confiança

1. Em uma janela do terminal, determine o namespace e o nome da conta de serviço do seu servidor Grafana. Por exemplo, é possível usar o comando a seguir.

```
kubectl get serviceaccounts -n grafana_namespace
```

2. No console do Amazon EKS, abra o perfil do IAM para contas de serviço que está associado ao cluster EKS.
3. Selecione Editar relação de confiança.
4. Atualize a Condição para incluir o namespace do Grafana e o nome da conta de serviço do Grafana que você encontrou na saída do comando na etapa 1. Veja um exemplo do a seguir:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::111122223333:oidc-provider/oidc.eks.us-
east-1.amazonaws.com/id/EXAMPLED539D4633E53DE1B71EXAMPLE"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:sub": [
            "system:serviceaccount:aws-amp:amp-iamproxy-query-service-
account",
            "system:serviceaccount:grafana-namespace:grafana-service-account-
name"
          ],
          "oidc.eks.us-east-1.amazonaws.com/
id/EXAMPLED539D4633E53DE1B71EXAMPLE:aud": "sts.amazonaws.com"
        }
      }
    }
  ]
}
```

5. Selecione Atualizar política de confiança.

Atualizar o servidor Grafana usando o Helm

Esta etapa atualiza o servidor Grafana para usar as entradas que você adicionou ao arquivo `amp_query_override_values.yaml` na seção anterior.

Execute os seguintes comandos. Para obter mais informações sobre charts do Helm para o Grafana, consulte Charts do Helm da Comunidade Kubernetes do Grafana.

```
helm repo add grafana https://grafana.github.io/helm-charts
```

```
helm upgrade --install grafana grafana/grafana -n grafana_namespace -f ./
amp_query_override_values.yaml
```

Adicionar a fonte de dados do Prometheus no Grafana

As etapas a seguir explicam como configurar a fonte de dados do Prometheus no Grafana para consultar suas métricas do Amazon Managed Service for Prometheus.

Para adicionar a fonte de dados do Prometheus no servidor Grafana

1. Abra o console do Grafana.
2. Em Configurações, escolha Fontes de dados.
3. Escolha Adicionar fonte de dados.
4. Escolha Prometheus.
5. Para o URL HTTP, especifique o Endpoint - URL de consulta exibido na página de detalhes do espaço de trabalho no console do Amazon Managed Service for Prometheus.
6. No URL HTTP que você acabou de especificar, remova a string `/api/v1/query` anexada ao URL, pois a fonte de dados do Prometheus a anexará automaticamente.
7. Em Auth, selecione o botão de alternância do SigV4 Auth para ativá-lo.

   Deixe os campos Presumir ARN do perfil e ID externo em branco. Em seguida, em Região padrão, selecione a região onde está seu espaço de trabalho do Amazon Managed Service for Prometheus.
8. Escolha Salvar e testar.

   Você deverá ver a seguinte mensagem: A fonte de dados está funcionando
9. Teste uma consulta PromQL contra a nova fonte de dados:

a. Escolha Explorar.

b. Execute um exemplo de consulta PromQL, como:

```
prometheus_tsdb_head_series
```

Consulta usando Prometheus-compatible APIs

Embora usar uma ferramenta como o Amazon Managed Grafana seja a maneira mais fácil de visualizar e consultar suas métricas, o Amazon Managed Service for Prometheus também oferece suporte a várias Prometheus-compatible APIs que você pode usar para consultar suas métricas. Para obter mais informações sobre todas as Prometheus-compatible APIs disponíveis, consulteCompatível com Prometheus APIs.

As Prometheus-compatible APIs usam a linguagem de consulta Prometheus, PromQL, para especificar os dados que você deseja retornar. Para obter detalhes sobre PromQL e sua sintaxe, veja Como consultar o Prometheus na documentação do Prometheus.

Quando você usa essas APIs para consultar suas métricas, as solicitações devem ser assinadas com o processo de assinatura do AWS Signature Version 4. Você pode configurar o AWS Signature Version 4 para simplificar o processo de assinatura. Para obter mais informações, consulte aws-sigv4-proxy.

A assinatura por meio do proxy AWS SigV4 pode ser realizada usando. `awscurl` O tópico a seguir Usando awscurl para consultar Prometheus-compatible APIs explica como configurar o `awscurl` SigV4. AWS

Tópicos

- Use awscurl para consultar com APIs Prometheus-compatible

Use awscurl para consultar com APIs Prometheus-compatible

As solicitações de API para o Amazon Managed Service for Prometheus devem ser assinadas com o SigV4. Você pode usar o awscurl para simplificar o processo de consulta.

Para instalar o `awscurl`, você precisa ter o Python 3 e o gerenciador de pacotes pip instalados.

Em uma instância baseada no Linux, o comando a seguir instala o`awscurl`.

```
$ pip3 install awscurl
```

Em um computador macOS, o comando a seguir instala o `awscurl`.

```
$ brew install awscurl
```

Veja a seguir um exemplo de consulta do `awscurl`: Substitua *`Region`* as *`QUERY`* entradas *`Workspace-id`* e por valores apropriados para seu caso de uso:

```
# Define the Prometheus query endpoint URL. This can be found in the Amazon Managed
 Service for Prometheus console page
# under the respective workspace.

$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
workspaces/Workspace-id/api/v1/query

# credentials are infered from the default profile
$ awscurl -X POST --region Region \
                  --service aps "${AMP_QUERY_ENDPOINT}" -d 'query=QUERY'  --header
 'Content-Type: application/x-www-form-urlencoded'
```

> **Note**
>
> Sua string de consulta deve estar codificada em URL.

Para uma consulta como `query=up`, você pode obter resultados como:

```
{
  "status": "success",
  "data": {
    "resultType": "vector",
    "result": [
      {
        "metric": {
          "__name__": "up",
          "instance": "localhost:9090",
          "job": "prometheus",
          "monitor": "monitor"
        },
        "value": [
```

```
          1652452637.636,
          "1"
        ]
      },
    ]
  }
}
```

Para que o `awscurl` assine as solicitações fornecidas, você precisará passar as credenciais válidas de uma das seguintes formas:

- Forneça o ID da chave de acesso e a chave secreta para o perfil do IAM. Você pode encontrar a chave de acesso e a chave secreta para a função no https://console.aws.amazon.com/iam/.

  Por exemplo:

  ```
  $ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
  workspaces/Workspace_id/api/v1/query

  $ awscurl -X POST --region <Region> \
                    --access_key <ACCESS_KEY> \
                    --secret_key <SECRET_KEY> \
                    --service aps "$AMP_QUERY_ENDPOINT?query=<QUERY>"
  ```

- Faça referência aos arquivos de configuração armazenados no `.aws/credentials` e no arquivo `/aws/config`. Você também pode optar por especificar o nome do perfil a ser utilizado. Se não for especificado, o arquivo `default` será utilizado. Por exemplo:

  ```
  $ export AMP_QUERY_ENDPOINT=https://aps-workspaces.<Region>.amazonaws.com/workspaces/
  <Workspace_ID>/api/v1/query
  $ awscurl -X POST --region <Region> \
                    --profile <PROFILE_NAME>
                    --service aps "$AMP_QUERY_ENDPOINT?query=<QUERY>"
  ```

- Use o perfil de instância associado a uma instância do EC2.

Como executar solicitações de consulta usando o contêiner awscurl

Quando a instalação de uma versão diferente do Python e das dependências associadas não for viável, um contêiner pode ser usado para empacotar a aplicação `awscurl` e suas dependências.

O exemplo a seguir usa um runtime Docker para implantar o `awscurl`, mas qualquer runtime e imagem compatíveis com OCI funcionarão.

```
$ docker pull okigan/awscurl
$ export AMP_QUERY_ENDPOINT=https://aps-workspaces.Region.amazonaws.com/
workspaces/Workspace_id/api/v1/query
$ docker run --rm -it okigan/awscurl --access_key $AWS_ACCESS_KEY_ID  --secret_key
 $AWS_SECRET_ACCESS_KEY \ --region Region --service aps "$AMP_QUERY_ENDPOINT?
query=QUERY"
```

Obtenha estatísticas sobre o uso de cada consulta

O preço da consulta é baseado no número total de amostras de consulta processadas em um mês a partir das consultas executadas. Você pode obter estatísticas sobre cada consulta que você faz para acompanhar suas amostras processadas. A resposta da consulta para uma `query` ou uma API `queryRange` pode incluir os dados estatísticos sobre amostras de consulta processadas ao incluir o parâmetro de consulta `stats=all` na solicitação. Um objeto `samples` é criado no objeto `stats` e os dados de `stats` são retornados na resposta.

O objeto `samples` contém os seguintes atributos:

Atributo	Description
`totalQueryableSamples`	Número total de amostras de consulta processad as. Essas são as informações a serem usadas para cobrança.
`totalQueryableSamp lesPerStep`	O número de amostras de consulta processadas por cada etapa. Isso é estruturado como uma matriz de matrizes com a data e hora na época e o número de amostras carregadas na etapa específica.

Estes são alguns exemplos de solicitações e respostas que incluem as informações do `stats` na resposta:

Exemplo de `query`:

GET

```
endpoint/api/v1/query?query=up&time=1652382537&stats=all
```

Resposta


```
{
    "status": "success",
    "data": {
        "resultType": "vector",
        "result": [
            {
                "metric": {
                    "__name__": "up",
                    "instance": "localhost:9090",
                    "job": "prometheus"
                },
                "value": [
                    1652382537,
                    "1"
                ]
            }
        ],
        "stats": {
            "timings": {
                "evalTotalTime": 0.00453349,
                "resultSortTime": 0,
                "queryPreparationTime": 0.000019363,
                "innerEvalTime": 0.004508405,
                "execQueueTime": 0.000008786,
                "execTotalTime": 0.004554219
            },
            "samples": {
                "totalQueryableSamples": 1,
                "totalQueryableSamplesPerStep": [
                    [
                        1652382537,
                        1
                    ]
                ]
            }
        }
    }
}
```

Exemplo de `queryRange`:

GET

```
endpoint/api/v1/query_range?query=sum+%28rate+%28go_gc_duration_seconds_count%5B1m%5D
%29%29&start=1652382537&end=1652384705&step=1000&stats=all
```

Resposta

```
{
    "status": "success",
    "data": {
        "resultType": "matrix",
        "result": [
            {
                "metric": {},
                "values": [
                    [
                        1652383000,
                        "0"
                    ],
                    [
                        1652384000,
                        "0"
                    ]
                ]
            }
        ],
        "stats": {
            "samples": {
                "totalQueryableSamples": 8,
                "totalQueryableSamplesPerStep": [
                    [
                        1652382000,
                        0
                    ],
                    [
                        1652383000,
                        4
                    ],
                    [
                        1652384000,
                        4
```

```
                    ]
                ]
            }
        }
    }
}
```

Detecção de anomalias

O Amazon Managed Service for Prometheus fornece recursos de detecção de anomalias que usam algoritmos de machine learning para identificar automaticamente padrões incomuns em seus dados métricos. Esse recurso ajuda você a detectar proativamente possíveis problemas, reduzir a fadiga de alertas e melhorar a eficácia do monitoramento, concentrando-se em comportamentos verdadeiramente anômalos em vez de limites estáticos.

A detecção de anomalias no Amazon Managed Service for Prometheus usa o algoritmo Random Cut Forest (RCF), que analisa seus dados de séries temporais para estabelecer padrões normais de comportamento e identificar desvios desses padrões. O algoritmo se adapta às tendências sazonais, lida com os dados perdidos com tranquilidade e fornece pontuações de confiança para anomalias detectadas.

Como funciona a detecção de anomalias

A detecção de anomalias do Amazon Managed Service for Prometheus usa machine learning para identificar padrões incomuns em dados de métricas sem configuração manual de limites. O sistema aprende padrões normais de comportamento e variações sazonais, reduzindo falsos positivos e permitindo a detecção precoce de problemas. Ele se adapta continuamente às mudanças dos aplicativos, tornando-o adequado para ambientes de nuvem dinâmicos.

A detecção de anomalias monitora as métricas de desempenho do aplicativo, como tempos de resposta e taxas de erro, rastreia a integridade da infraestrutura por meio do uso da CPU e da memória, detecta comportamentos incomuns do usuário, identifica as necessidades de planejamento de capacidade pela análise de tráfego e monitora as métricas de negócios em busca de mudanças inesperadas. Funciona melhor com padrões previsíveis, variações sazonais ou tendências de crescimento gradual.

O algoritmo Random Cut Forest (RCF) é usado para analisar dados de séries temporais. O RCF cria árvores de decisão que particionam o espaço de dados e identificam pontos isolados longe da distribuição normal. O algoritmo aprende com os dados recebidos para criar um modelo dinâmico de comportamento normal para cada métrica.

Quando ativado, ele analisa dados históricos para estabelecer padrões básicos e tendências sazonais e, em seguida, gera previsões para valores esperados e identifica desvios. O algoritmo produz quatro saídas principais:

- upper_band: o limite superior dos valores normais esperados
- lower_band: o limite inferior dos valores normais esperados
- score: uma pontuação de anomalia numérica que indica o quão incomum é o ponto de dados
- value: o valor métrico real observado

Conceitos básicos da detecção de anomalias

Para começar a usar a detecção de anomalias com suas métricas do Prometheus, você precisa de dados históricos suficientes para que o algoritmo aprenda os padrões normais. Recomendamos ter pelo menos 14 dias de dados métricos consistentes antes de ativar a detecção de anomalias para obter os melhores resultados.

Você pode visualizar como a detecção de anomalias funcionará com suas métricas usando a API `PreviewAnomalyDetector`. Use `PreviewAnomalyDetector` para testar o algoritmo em relação aos seus dados históricos e avaliar a eficácia dele antes de implementá-lo no monitoramento da produção. Para obter mais informações, consulte PreviewAnomalyDetector API.

Ao implementar a detecção de anomalias, considere estas práticas recomendadas:

- Comece com métricas estáveis: comece com métricas com padrões consistentes e evite inicialmente dados altamente voláteis ou esparsos.
- Use dados agregados: aplique a detecção de anomalias a métricas agregadas (como médias ou somas) em vez de dados brutos de alta cardinalidade para obter melhor desempenho e precisão.
- Ajuste a sensibilidade: ajuste os parâmetros do algoritmo com base em seu caso de uso específico e na tolerância a falsos positivos versus anomalias perdidas.
- Monitore o desempenho do algoritmo: revise regularmente as anomalias detectadas para garantir que o algoritmo continue fornecendo informações valiosas à medida que seu sistema evolui.

PreviewAnomalyDetector API

Use a operação `PreviewAnomalyDetector` para criar um endpoint que demonstre como seus dados métricos serão analisados pelo algoritmo de detecção de anomalias durante o período especificado. Esse endpoint ajuda você a avaliar e validar o desempenho do detector antes da implementação.

Verbos HTTP válidos:

`GET`, `POST`

Tipos de carga útil compatíveis

Parâmetros codificados em URL

`application/x-www-form-urlencoded` para `POST`

Parâmetros compatíveis

`query=<string>` Uma string de consulta da expressão Prometheus.

`start=<rfc3339 | unix_timestamp>` Inicie o timestamp se você estiver usando `query_range` para consultar por um intervalo de tempo.

`end=<rfc3339 | unix_timestamp>` Carimbo de data/hora de término se você estiver usando `query_range` para consultar por um intervalo de tempo.

`step=<duration | float>` Largura da etapa de resolução da consulta em `duration` formato ou em `float` alguns segundos. Use somente se você estiver usando `query_range` para consultar por um intervalo de tempo e for necessário para essas consultas.

Formatação do parâmetro da solicitação

Envolva sua expressão original do PromQL com a pseudofunção RandomCutForest (RCF) no parâmetro de consulta. Para obter mais informações, consulte a RandomCutForestConfigurationReferência de API do Amazon Managed Service for Prometheus.

A função RCF usa este formato:

```
RCF(<query>
[,shingle size
[,sample size
[,ignore near expected from above
[,ignore near expected from below
[,ignore near expected from above ratio
[,ignore near expected from below ratio]]]]])
```

Todos os parâmetros, exceto a consulta, são opcionais e usam valores padrão quando omitidos. A sintaxe é:

```
RCF(<query>)
```

A consulta deve ser integrada com uma função de agregação. Para usar parâmetros opcionais específicos e omitir outros, deixe posições vazias na função:

```
RCF(<query>,,,,,1.0,1.0)
```

Esse exemplo define somente os parâmetros de proporção que ignoram os picos e quedas de detecção de anomalias com base na proporção entre os valores esperados e observados.

Solicitação e reposta da API

As chamadas bem-sucedidas retornam o mesmo formato da QueryMetrics API. Além da série temporal original, a API retorna estas novas séries temporais quando amostras suficientes estão disponíveis:

- `anomaly_detector_preview:lower_band`: banda inferior para o valor esperado do resultado da expressão PromQL
- `anomaly_detector_preview:score`: pontuação de anomalia entre 0 e 1, onde 1 indica alta confiança de uma anomalia naquele ponto de dados
- `anomaly_detector_preview:upper_band`: banda superior para o valor esperado do resultado da expressão PromQL

Exemplo de solicitação

```
POST /workspaces/workspace-id/anomalydetectors/preview
Content-Type: application/x-www-form-urlencoded

query=RCF%28avg%28vector%28time%28%29%29%29%2C%208%2C
%20256%29&start=1735689600&end=1735695000&step=1m
```

Exemplo de resposta

```
200 OK
...

{
```

```
  "status": "success",
  "data": {
    "result": [
      {
        "metric": {},
        "values": [
          [
            1735689600,
            "1735689600"
          ],
          [
            1735689660,
            "1735689660"
          ],
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "upper_band"
        },
        "values": [
          [
            1735693500,
            "1.7356943E9"
          ],
          [
            1735693560,
            "1.7356945E9"
          ]
          ],
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "lower_band"
        },
        "values": [
          [
            1735693500,
            "1.7356928E9"
          ],
          [
```

```
            1735693560,
            "1.7356929E9"
          ],
          .........
        ]
      },
      {
        "metric": {
          "anomaly_detector_preview": "score"
        },
        "values": [
          [
            1735693500,
            "0.0"
          ],
          [
            1735695000,
            "0.0"
          ],
          .........
        ]
      }
    ],
    "resultType": "matrix"
  }
}
```

Como usar regras para modificar ou monitorar métricas à medida são recebidas

Você pode configurar regras para agir de acordo com as métricas à medida que são recebidas pelo Amazon Managed Service for Prometheus. Essas regras podem monitorar as métricas ou até mesmo criar novas métricas computadas com base nas métricas recebidas.

O Amazon Managed Service for Prometheus oferece suporte a dois tipos de regras que ele avalia em intervalos regulares:

- As regras de gravação permitem que você pré-compute expressões frequentemente necessárias ou computacionalmente caras e salve seus resultados como um novo conjunto de séries temporais. Consultar o resultado pré-computado geralmente é muito mais rápido do que executar a expressão original sempre que necessário.
- As regras de alerta permitem que você defina condições de alerta com base no PromQL e em um limite. Quando a regra aciona o limite, uma notificação é enviada ao gerenciador de alertas, que pode ser configurado para gerenciar as regras ou encaminhá-las para notificação downstream a receptores como o Amazon Simple Notification Service.

Para usar regras no Amazon Managed Service for Prometheus, você cria um ou mais arquivos de regras YAML que definem as regras. Um arquivo de regras do Amazon Managed Service for Prometheus tem o mesmo formato de um arquivo de regras no Prometheus autônomo. Para obter mais informações, consulte Definição de regras de gravação e Regras de alerta na documentação do Prometheus.

Você pode ter vários arquivos de regras em um espaço de trabalho. Cada arquivo de regras separado está contido em um namespace separado. Ter vários arquivos de regras permite importar arquivos de regras existentes do Prometheus para um espaço de trabalho sem precisar alterá-los ou combiná-los. Namespaces de grupos de regras diferentes também podem ter tags diferentes.

Sequenciamento de regras

Em um arquivo de regras, as regras estão contidas em grupos de regras. As regras dentro de um único grupo de regras em um arquivo de regras são sempre avaliadas em ordem de cima para baixo. Portanto, nas regras de gravação, o resultado de uma regra de gravação pode ser usado no cálculo de uma regra de gravação posterior ou em uma regra de alerta no mesmo grupo de regras. No

entanto, como você não pode especificar a ordem na qual executar arquivos de regras separados, não é possível usar os resultados de uma regra de gravação para calcular uma regra em um grupo de regras diferente ou em um arquivo de regras diferente.

Tópicos

Noções básicas das permissões do IAM necessárias para usar regras

É necessário conceder aos usuários as permissões de usar as regras no Amazon Managed Service for Prometheus. Crie uma política AWS Identity and Access Management (IAM) com as seguintes permissões e atribua a política aos seus usuários, grupos ou funções.

> **Note**
>
> Para ter mais informações sobre o IAM, consulte Gerenciamento de identidade e acesso para Amazon Managed Service for Prometheus.

Política para dar acesso às regras de uso

A política a seguir dá acesso às regras de uso para todos os recursos da sua conta.

JSON

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

```
            "Action": [
                "aps:CreateRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:DescribeRuleGroupsNamespace",
                "aps:PutRuleGroupsNamespace",
                "aps:DeleteRuleGroupsNamespace"
            ],
            "Resource": "*"
        }
    ]
}
```

Política para dar acesso a apenas um namespace

Você também pode criar uma política que dê acesso somente a políticas específicas. O exemplo de política a seguir dá acesso somente ao `RuleGroupNamespace` especificado. Para usar essa política, substitua *<account><region>*,*<workspace-id>*, e *<namespace-name>* por valores apropriados para sua conta.

Criar um arquivo de regras

Para usar regras no Amazon Managed Service for Prometheus, você cria um arquivo de regras que define as regras. Um arquivo de regras do Amazon Managed Service for Prometheus é um arquivo de texto YAML que tem o mesmo formato de um arquivo de regras no Prometheus independente. Para obter mais informações, consulte Definição de regras de gravação e Regras de alerta na documentação do Prometheus.

Este é um exemplo básico de um arquivo de regras:

```
groups:
  - name: cpu_metrics
    interval: 60s
    rules:
     - record: avg_cpu_usage
       expr: avg(rate(node_cpu_seconds_total[5m])) by (instance)
     - alert: HighAverageCPU
       expr: avg_cpu_usage > 0.8
       for: 10m
       keep_firing_for: 20m
       labels:
```

```
        severity: critical
      annotations:
        summary: "Average CPU usage across cluster is too high"
```

Esse exemplo cria um grupo de regras `cpu_metrics` avaliado a cada 60 segundos. Esse grupo de regras cria uma nova métrica usando uma regra de gravação chamada `avg_cpu_usage` e, depois, usa essa regra em um alerta. Veja a seguir a descrição de algumas propriedades usadas. Para obter mais informações sobre regras de alerta e outras propriedades que você pode incluir, consulte Regras de alerta na documentação do Prometheus.

- `record: avg_cpu_usage`: essa regra de gravação cria uma nova métrica chamada `avg_cpu_usage`.
- O intervalo de avaliação padrão dos grupos de regras é de 60 segundos se a propriedade `interval` não for especificada.
- `expr: avg(rate(node_cpu_seconds_total[5m])) by (instance)`: essa expressão da regra de gravação calcula a taxa média de uso da CPU nos últimos 5 minutos para cada nó, agrupando pelo rótulo `instance`.
- `alert: HighAverageCPU`: essa regra de alerta cria um novo alerta chamado `HighAverageCPU`
- `expr: avg_cpu_usage > 0.8` : essa expressão instrui o alerta a procurar amostras em que o uso médio da CPU esteja acima de 80%.
- `for: 10m`: o alerta só será acionado se o uso médio da CPU exceder 80% por pelo menos 10 minutos.

  Nesse caso, a métrica é calculada como uma média dos últimos 5 minutos. Portanto, o alerta só será acionado se houver pelo menos duas amostras consecutivas de 5 minutos (10 minutos no total) em que o uso médio da CPU esteja acima de 80%.
- `keep_firing_for: 20m`: esse alerta continuará acionando até que as amostras estejam abaixo do limite por pelo menos 20 minutos. Isso pode ser útil para evitar que o alerta suba e desça repetidamente em sucessão.

> **Note**
>
> Você pode criar um arquivo de definição de regras localmente e, em seguida, carregá-lo no Amazon Managed Service for Prometheus ou você pode criar, editar e carregar a definição diretamente no console do Amazon Managed Service for Prometheus. De qualquer forma, as mesmas regras de formatação são aplicadas. Para saber mais sobre como carregar e editar

seu arquivo, consulte Carregar um arquivo de configuração de regras no Amazon Managed Service for Prometheus.

Carregar um arquivo de configuração de regras no Amazon Managed Service for Prometheus

Depois de saber quais regras você quer em seu arquivo de configuração de regras, você pode criá-las e editá-las no console ou carregar um arquivo com o console ou o AWS CLI.

> **Note**
>
> Se você estiver executando um cluster do Amazon EKS, também poderá carregar um novo arquivo de configuração usando Controladores da AWS para Kubernetes.

Para usar o console do Amazon Managed Service for Prometheus a fim de editar ou carregar sua configuração de regras e criar o namespace

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os espaços de trabalho.
3. Escolha a ID do espaço de trabalho e, em seguida, escolha a guia Gerenciamento de regras.
4. Escolha Adicionar um namespace.
5. Escolha Escolher arquivo e selecione o arquivo de definição de regras.

   Outra possibilidade é criar e editar um arquivo de definição de regras diretamente no console do Amazon Managed Service for Prometheus selecionando Definir configuração. Isso criará uma amostra de arquivo de definição padrão para você editar antes do carregamento.
6. (Opcional) Para adicionar tags ao namespace, selecione Adicionar nova tag.

   Em seguida, em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.

   Para adicionar outra tag, escolha Adicionar nova tag.

7. Escolha Continuar. O Amazon Managed Service for Prometheus cria um novo namespace com o mesmo nome do arquivo de regras que você selecionou.

Para usar o AWS CLI para carregar uma configuração do gerenciador de alertas em um espaço de trabalho em um novo namespace

1. O Base64 codifica o conteúdo do seu arquivo do gerenciador de alertas. Em um sistema Linux, use o seguinte comando:

   ```
   base64 input-file output-file
   ```

   No macOS, use o seguinte comando:

   ```
   openssl base64 input-file output-file
   ```

2. Digite um dos comandos a seguir para criar o namespace e fazer upload do arquivo.

   Na AWS CLI versão 2, digite:

   ```
   aws amp create-rule-groups-namespace --data file://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

   Na AWS CLI versão 1, digite:

   ```
   aws amp create-rule-groups-namespace --data fileb://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

3. São necessários alguns segundos para que a configuração do Alert Manager entre em vigor. Para verificar o status, insira o comando a seguir:

   ```
   aws amp describe-rule-groups-namespace --workspace-id workspace_id --
   name namespace-name --region region
   ```

   Se `status` for `ACTIVE`, seu arquivo de regras entrou em vigor.

Editar ou substituir um arquivo de configuração de regras

Se você quiser alterar as regras em um arquivo de regras que você já carregou no Amazon Managed Service for Prometheus, carregue um novo arquivo de regras para substituir a configuração existente ou edite a configuração atual diretamente no console. Opcionalmente, você pode baixar o arquivo atual, editá-lo em um editor de texto e, em seguida, fazer upload de uma nova versão.

Para usar o console do Amazon Managed Service for Prometheus para editar sua configuração de regras

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os espaços de trabalho.
3. Escolha a ID do espaço de trabalho e, em seguida, escolha a guia Gerenciamento de regras.
4. Selecione o nome do arquivo de configuração de regras que você deseja editar.
5. (Opcional) Se você quiser baixar o arquivo de configuração de regras atual, selecione Baixar ou Copiar.
6. Selecione Modificar para editar a configuração diretamente no console. Selecione Salvar ao concluir.

   Outra possibilidade é selecionar Substituir configuração para carregar um novo arquivo de configuração. Se esse for caso, selecione o novo arquivo de definição de regras e Continuar para carregá-lo.

Para usar o AWS CLI para editar um arquivo de configuração de regras

1. O Base64 codifica o conteúdo do seu arquivo de regras. Em um sistema Linux, use o seguinte comando:

   ```
   base64 input-file output-file
   ```

   No macOS, use o seguinte comando:

   ```
   openssl base64 input-file output-file
   ```

2. Digite um dos comandos a seguir para fazer upload do novo arquivo.

Na AWS CLI versão 2, digite:

```
aws amp put-rule-groups-namespace --data file://path_to_base_64_output_file --
name namespace-name  --workspace-id my-workspace-id --region region
```

Na AWS CLI versão 1, digite:

```
aws amp put-rule-groups-namespace --data fileb://path_to_base_64_output_file --
name namespace-name  --workspace-id my-workspace-id --region region
```

3. São necessários alguns segundos para que seu arquivo de regras entre em vigor. Para verificar o status, insira o comando a seguir:

```
aws amp describe-rule-groups-namespace --workspace-id workspace_id --
name namespace-name --region region
```

Se `status` for `ACTIVE`, seu arquivo de regras entrou em vigor. Até lá, a versão anterior desse arquivo de regras ainda estará ativa.

Solucionar problemas em avaliações de regras

Este guia fornece procedimentos step-by-step de solução de problemas comuns com avaliações de regras no Amazon Managed Service for Prometheus (AMP). Siga estes procedimentos para diagnosticar e resolver problemas com as regras de alerta e gravação.

Tópicos

Validar o status de disparo do alerta

Ao solucionar problemas de avaliação de regras, primeiro verifique se o alerta foi acionado consultando a série temporal sintética `ALERTS`. As séries temporais `ALERTS` incluem os seguintes rótulos:

- alertname: o nome do alerta.
- alertstate pending ou firing.
  - pending: o alerta está aguardando a duração especificada na cláusula `for`.
  - firing: o alerta atendeu às condições da duração especificada. Rótulos adicionais são definidos em sua regra de alerta.

> **Note**
>
> Quando um alerta tem o rótulo firing ou pending, o valor da amostra é 1. Quando o alerta está inativo, nenhuma amostra é produzida.

Solucionar falta de notificações de alertas

Se os alertas estiverem sendo disparados, mas as notificações não chegarem, verifique as seguintes configurações do Alertmanager:

1. Verifique a configuração do Alertmanager: verifique se as rotas, os receptores e as configurações estão configuradas corretamente. Revise as configurações de bloqueio de rotas, incluindo tempos de espera, intervalos de tempo e rótulos necessários, que podem afetar o disparo de alertas. Compare as regras de alerta com as rotas e receptores correspondentes para confirmar a correspondência adequada. Para rotas com `time_interval`, verifique se os carimbos de data/hora estão dentro dos intervalos especificados.
2. Verifique as permissões do receptor de alertas: ao usar um tópico do Amazon SNS, verifique se o AMP tem as permissões necessárias para publicar notificações. Para obter mais informações, consulte Conceder ao Amazon Managed Service for Prometheus permissão para enviar mensagens ao seu tópico do Amazon SNS.
3. Valide a compatibilidade da carga útil do receptor: confirme se seu receptor de alerta aceita o formato de carga útil do Alertmanager. Para ver os requisitos do Amazon SNS, consulte Noções básicas das regras de validação de mensagens do Amazon SNS.

4. Analise os logs do Alertmanager: o AMP oferece logs fornecidos do Alertmanager para ajudar a depurar problemas de notificação. Para obter mais informações, consulte Monitore eventos do Amazon Managed Service para Prometheus com registros CloudWatch.

Para obter mais informações sobre o Alertmanager, consulte Como gerenciar e encaminhar alertas no Amazon Managed Service for Prometheus com o gerenciador de alertas.

Verificar o status de integridade da regra

Regras malformadas podem causar falhas na avaliação. Use os métodos a seguir para identificar por que uma regra não foi avaliada:

Example

Use a ListRules API

A API ListRules fornece informações sobre a integridade das regras. Verifique os campos `lastError` e `health` para diagnosticar problemas.

Exemplo de resposta:

```
{
  "status": "success",
  "data": {
    "groups": [
      {
        "name": "my_rule_group",
        "file": "my_namespace",
        "rules": [
          {
            "state": "firing",
            "name": "broken_alerting_rule",
            "query": "...",
            "duration": 0,
            "keepFiringFor": 0,
            "labels": {},
            "annotations": {},
            "alerts": [],
            "health": "err",
            "lastError": "vector contains metrics with the same labelset after applying
 alert labels",
```

```
            "type": "alerting",
            "lastEvaluation": "1970-01-01T00:00:00.00000000Z",
            "evaluationTime": 0.08
          }
        ]
      }
    ]
  }
}
```

Example

Use logs fornecidos

A ListRules API exibe apenas as informações mais recentes. Para obter um histórico mais detalhado, habilite os logs fornecidos em seu espaço de trabalho para acessar:

- Carimbos de data/hora de falhas de avaliação
- Mensagens de erro detalhadas
- Dados históricos de avaliação

Exemplo de mensagem de log fornecido:

```
{
  "workspaceId": "ws-a2c55905-e0b4-4065-a310-d83ce597a391",
  "message": {
    "log": "Evaluating rule failed, name=broken_alerting_rule, group=my_rule_group,
 namespace=my_namespace, err=vector contains metrics with the same labelset after
 applying alert labels",
    "level": "ERROR",
    "name": "broken_alerting_rule",
    "group": "my_rule_group",
    "namespace": "my_namespace"
  },
  "component": "ruler"
}
```

Para obter mais exemplos de logs do Ruler ou do Alertmanager, consulte Solução de problemas do Ruler e Como gerenciar e encaminhar alertas no Amazon Managed Service for Prometheus com o gerenciador de alertas.

Usar deslocamento em consultas para lidar com atrasos na ingestão

Por padrão, as expressões são avaliadas sem deslocamento (consulta instantânea), usando valores no momento da avaliação. Se a ingestão de métricas for atrasada, as regras de gravação podem não representar os mesmos valores de quando você avalia manualmente a expressão depois que todas as métricas são ingeridas.

> **Tip**
>
> Usar o modificador de deslocamento pode reduzir os problemas causados por atrasos na ingestão. Para obter mais informações, consulte Offset modifier na documentação do Prometheus.

Exemplo: lidar com métricas atrasadas

Se a sua regra for avaliada às 12:00, mas a amostra mais recente da métrica for das 11:45 devido ao atraso na ingestão, a regra não encontrará amostras no carimbo de data/hora das 12:00. Para mitigar isso, adicione um deslocamento, como: **`my_metric_name offset 15m`** .

Exemplo: lidar com métricas de várias fontes

Quando as métricas são provenientes de fontes diferentes, como dois servidores, elas podem ser ingeridas em momentos diferentes. Para mitigar isso, forme uma expressão, como: **`metric_from_server_A / metric_from_server_B`** .

Se a regra for avaliada entre os tempos de ingestão do servidor A e do servidor B, você obterá resultados inesperados. Usar um deslocamento pode ajudar a alinhar os tempos de avaliação.

Problemas e soluções comuns de

Lacunas nos dados da regra de gravação

Se você notar lacunas nos dados da regra de gravação em comparação com a avaliação manual (ao executar diretamente a expressão PromQL original da regra de gravação pela API ou IU de consulta), isso pode ser devido a um dos seguintes motivos:

1. Tempos de avaliação longos: um grupo de regras não pode ter várias avaliações simultâneas. Se o tempo de avaliação exceder o intervalo configurado, as avaliações subsequentes poderão ser

perdidas. Várias avaliações perdidas consecutivas excedendo o intervalo configurado podem fazer com que a regra de gravação fique obsoleta. Para obter mais informações, consulte Staleness na documentação do Prometheus. Você pode monitorar a duração da avaliação usando a CloudWatch métrica `RuleGroupLastEvaluationDuration` para identificar grupos de regras que estão demorando muito para serem avaliados.

2. Monitoramento de avaliações perdidas — O AMP fornece a `RuleGroupIterationsMissed` CloudWatch métrica para rastrear quando as avaliações são ignoradas. A ListRules API exibe a hora da avaliação e a hora da última avaliação de cada regra/grupo, o que pode ajudar a identificar padrões de avaliações perdidas. Para obter mais informações, consulte ListRules.

Recomendação: divida as regras em grupos separados

Para reduzir a duração da avaliação, divida as regras em grupos de regras separados. As regras em um grupo são executadas sequencialmente, enquanto os grupos de regras podem ser executados paralelamente. Mantenha regras relacionadas que dependam umas das outras no mesmo grupo. Geralmente, grupos de regras menores garantem avaliações mais consistentes e menos lacunas.

Práticas recomendadas para avaliações de regras

1. Otimize o tamanho do grupo de regras: mantenha os grupos de regras pequenos para garantir avaliações consistentes. Agrupe as regras relacionadas, mas evite grupos de regras grandes.
2. Defina intervalos de avaliação adequados: equilíbrio entre alertas oportunos e carga do sistema. Analise os padrões de estabilidade de suas métricas monitoradas para entender as faixas normais de flutuação.
3. Use modificadores de deslocamento para métricas atrasadas: adicione deslocamentos para compensar os atrasos na ingestão. Ajuste a duração do deslocamento com base nos padrões de ingestão observados.
4. Monitore o desempenho da avaliação: acompanhe a métrica `RuleGroupIterationsMissed`. Analise os tempos de avaliação na ListRules API.
5. Valide expressões de regras: certifique-se de que as expressões correspondam exatamente entre as definições de regras e as consultas manuais. Teste expressões com intervalos de tempo diferentes para entender o comportamento.
6. Revise a integridade das regras regularmente: verifique se há erros nas avaliações das regras. Monitore os logs fornecidos em busca de problemas recorrentes.

Ao seguir essas etapas de solução de problemas e as práticas recomendadas, você pode identificar e resolver problemas comuns com avaliações de regras no Amazon Managed Service for Prometheus.

Solução de problemas do Ruler

Utilizando Monitore eventos do Amazon Managed Service para Prometheus com registros CloudWatch, você pode solucionar problemas relacionados ao gerenciador de alertas e ao Ruler. Esta seção contém tópicos de solução de problemas relacionados ao ruler.

Quando o log contém o seguinte erro de falha do ruler

```
{
    "workspaceId": "ws-12345c67-89c0-4d12-345b-f14db70f7a99",
    "message": {
        "log": "Evaluating rule failed, name=failure,
 group=canary_long_running_vl_namespace, namespace=canary_long_running_vl_namespace,
 err=found duplicate series for the match group {dimension1=\\\"1\\\"} on the right
 hand-side of the operation: [{__name__=\\\"fake_metric2\\\", dimension1=\\\"1\\
\", dimension2=\\\"b\\\"}, {__name__=\\\"fake_metric2\\\", dimension1=\\\"1\\\",
 dimension2=\\\"a\\\"}];many-to-many matching not allowed: matching labels must be
 unique on one side",
        "level": "ERROR",
        "name": "failure",
        "group": "canary_long_running_vl_namespace",
        "namespace": "canary_long_running_vl_namespace"
    },
    "component": "ruler"
}
```

Isso significa que ocorreu algum erro ao executar a regra.

Medida a ser tomada

Use a mensagem de erro para solucionar problemas de execução de regra.

Como gerenciar e encaminhar alertas no Amazon Managed Service for Prometheus com o gerenciador de alertas

Quando as regras de alerta executáveis pelo Amazon Managed Service for Prometheus são acionadas, o gerenciador de alertas controla os alertas enviados. Ele elimina duplicatas, agrupa e encaminha os alertas para os destinatários finais. O Amazon Managed Service for Prometheus oferece suporte somente ao Amazon Simple Notification Service como receptor e pode rotear mensagens para tópicos do Amazon SNS na mesma conta. Você também pode usar o gerenciador de alertas para silenciar e inibir os alertas.

O gerenciador de alertas fornece funcionalidade semelhante ao Alertmanager no Prometheus.

Use o arquivo de configuração do gerenciador de alertas nos seguintes casos:

- Agrupamento — O agrupamento coleta alertas similares em uma única notificação. Isso é especialmente útil durante interrupções maiores, quando muitos sistemas falham ao mesmo tempo e centenas de alertas podem ser acionados simultaneamente. Por exemplo, suponha que uma falha na rede cause uma falha em muitos de seus nós simultaneamente. Se esses tipos de alertas estiverem agrupados, o gerenciador de alertas enviará uma única notificação.

  O agrupamento de alertas e o período das notificações agrupadas são configurados por uma árvore de roteamento no arquivo de configuração do gerenciador de alertas. Para obter mais informações, consulte <route>.
- Inibição — A inibição suprime as notificações de determinados alertas quando outros alertas já estiverem acionados. Por exemplo, se tiver um alerta acionado sobre um cluster inacessível, você pode configurar o gerenciador de alertas para silenciar todos os outros alertas relacionados a esse cluster. Isso evita notificações de centenas ou milhares de alertas de acionamento não relacionados ao problema real. Para obter mais informações sobre como escrever regras de inibição, consulte <inhibit_rule>.
- Silencia — Silencia alertas sem som por um período específico, por exemplo, durante uma janela de manutenção. Os alertas recebidos são verificados para conferir se têm todas as correspondências de igualdade ou expressão regular de um silêncio ativo. Se forem correspondentes, nenhuma notificação será enviada de tal alerta.

  Para criar um silêncio, você usa a API `PutAlertManagerSilences`. Para obter mais informações, consulte PutAlertManagerSilences.

Modelagem de Prometheus

O Prometheus autônomo é compatível com modelos por meio de arquivos de modelo separados. Os modelos podem usar condicionais e formatar dados, entre outras coisas.

No Amazon Managed Service for Prometheus, você coloca seus modelos no mesmo arquivo de configuração do gerenciador de alertas em que está sua configuração do gerenciador de alertas.

Tópicos

- Noções básicas das permissões do IAM necessárias para trabalhar com o gerenciador de alertas
- Criar uma configuração do gerenciador de alertas no Amazon Managed Service for Prometheus para gerenciar e encaminhar alertas
- Encaminhar alertas para um receptor de alertas no Amazon Managed Service for Prometheus
- Carregar seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus
- Integrar alertas com o Amazon Managed Grafana ou o Grafana de código aberto
- Solucione problemas do gerenciador de alertas com CloudWatch o Logs

Noções básicas das permissões do IAM necessárias para trabalhar com o gerenciador de alertas

Você precisa conceder aos usuários as permissões para usar o gerenciador de alertas no Amazon Managed Service for Prometheus. Crie uma política do AWS Identity and Access Management (IAM) com as seguintes permissões e atribua a política aos seus usuários, grupos ou perfis.

Criar uma configuração do gerenciador de alertas no Amazon Managed Service for Prometheus para gerenciar e encaminhar alertas

Para usar o gerenciador de alertas e a modelagem no Amazon Managed Service for Prometheus, você cria um arquivo YAML de configuração do gerenciador de alertas. Um arquivo do gerenciador de alertas do Amazon Managed Service for Prometheus tem duas seções principais:

- `template_files:` contém os modelos utilizados para mensagens enviadas pelos destinatários. Para obter mais informações, consulte Referência de modelo e Exemplos de modelos na documentação do Prometheus.
- `alertmanager_config:` contém a configuração do gerenciador de alertas. Utiliza a mesma estrutura de um arquivo de configuração do gerenciador de alertas no Prometheus autônomo. Para obter mais informações, consulte Configuração na documentação do Alertmanager.

> **Note**
>
> A configuração `repeat_interval` descrita na documentação do Prometheus acima tem uma limitação adicional no Amazon Managed Service for Prometheus. O valor máximo permitido é de cinco dias. Se você definir um período maior que cinco dias, será tratado como cinco dias e as notificações serão enviadas novamente após o término do período de cinco dias.

> **Note**
>
> Você também pode editar o arquivo de configuração diretamente pelo console do Amazon Managed Service for Prometheus, mas ele ainda deve seguir o formato especificado aqui. Para obter mais informações sobre como carregar ou editar um arquivo de configuração, consulte Carregar seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus.

No Amazon Managed Service for Prometheus, seu arquivo de configuração do gerenciador de alertas deve ter todo o seu conteúdo de configuração do gerenciador de alertas dentro de uma chave `alertmanager_config` na raiz do arquivo YAML.

Veja a seguir um exemplo básico de arquivo de configuração do gerenciador de alertas:

```
alertmanager_config: |
  route:
    receiver: 'default'
  receivers:
    - name: 'default'
      sns_configs:
      - topic_arn: arn:aws:sns:us-east-2:123456789012:My-Topic
        sigv4:
```

```
          region: us-east-2
        attributes:
          key: key1
          value: value1
```

No momento, o único receptor suportado é o Amazon Simple Notification Service (Amazon SNS). Se você tiver outros tipos de receptores listados na configuração, a mesma será rejeitada.

Aqui está outro exemplo de arquivo de configuração do gerenciador de alertas que utiliza o bloco `template_files` e o bloco `alertmanager_config`.

```
template_files:
  default_template: |
    {{ define "sns.default.subject" }}[{{ .Status | toUpper }}{{ if eq .Status
 "firing" }}:{{ .Alerts.Firing | len }}{{ end }}]{{ end }}
    {{ define "__alertmanager" }}AlertManager{{ end }}
    {{ define "__alertmanagerURL" }}{{ .ExternalURL }}/#/alerts?receiver={{ .Receiver |
 urlquery }}{{ end }}
alertmanager_config: |
  global:
  templates:
    - 'default_template'
  route:
    receiver: default
  receivers:
    - name: 'default'
      sns_configs:
      - topic_arn: arn:aws:sns:us-east-2:accountid:My-Topic
        sigv4:
          region: us-east-2
        attributes:
          key: severity
          value: SEV2
```

Bloco de modelos padrão do Amazon SNS

A configuração padrão do Amazon SNS usa o modelo a seguir, a menos que você o substitua expressamente.

```
{{ define "sns.default.message" }}{{ .CommonAnnotations.SortedPairs.Values | join "
 " }}
  {{ if gt (len .Alerts.Firing) 0 -}}
```

```
  Alerts Firing:
    {{ template "__text_alert_list" .Alerts.Firing }}
  {{- end }}
  {{ if gt (len .Alerts.Resolved) 0 -}}
  Alerts Resolved:
    {{ template "__text_alert_list" .Alerts.Resolved }}
  {{- end }}
{{- end }}
```

Encaminhar alertas para um receptor de alertas no Amazon Managed Service for Prometheus

Quando um alerta é gerado por uma regra de alerta, ele é enviado ao gerenciador de alertas. O gerenciador de alertas executa funções como eliminação de duplicatas de alertas, inibição de alertas durante a manutenção ou agrupamento deles conforme necessário. Em seguida, ele encaminha o alerta como mensagem para um receptor de alertas. Você pode configurar um receptor de alertas para que possa notificar os operadores, ter respostas automatizadas ou responder aos alertas de outras formas.

Você pode configurar o Amazon Simple Notification Service (Amazon SNS) PagerDuty e como receptor de alertas no Amazon Managed Service for Prometheus. Os tópicos a seguir descrevem como criar e configurar seu receptor de alertas.

Tópicos

- Usar o Amazon SNS como receptor de alertas
- Use PagerDuty como receptor de alertas

Usar o Amazon SNS como receptor de alertas

Você pode usar um tópico existente do Amazon SNS como receptor de alertas para o Amazon Managed Service for Prometheus ou criar um novo. Recomendamos que você use um tópico do tipo Padrão para poder encaminhar alertas do tópico para o e-mail, SMS ou HTTP.

Para criar um novo tópico do Amazon SNS para utilizar como receptor do gerenciador de alertas, siga as etapas da Etapa 1: Criar um tópico. Certifique-se de escolher Padrão para o tipo de tópico.

Se você quiser receber e-mails sempre que uma mensagem for enviada para esse tópico do Amazon SNS, siga as etapas da Etapa 2: Crie uma assinatura para o tópico.

Seja usando um tópico existente ou novo do Amazon SNS, você precisará do nome do recurso da Amazon (ARN) do seu tópico do Amazon SNS para concluir as tarefas a seguir.

Tópicos

- Conceder ao Amazon Managed Service for Prometheus permissão para enviar mensagens ao seu tópico do Amazon SNS
- Configurar o gerenciador de alertas para enviar mensagens ao tópico do Amazon SNS
- Configurar o gerenciador de alertas para que envie mensagens ao seu tópico do Amazon SNS como JSON
- Configurar o Amazon SNS para que envie mensagens de alertas a outros destinos
- Noções básicas das regras de validação de mensagens do Amazon SNS

Conceder ao Amazon Managed Service for Prometheus permissão para enviar mensagens ao seu tópico do Amazon SNS

Você deve conceder permissão ao Amazon Managed Service for Prometheus para enviar mensagens ao seu tópico do Amazon SNS. A instrução de política a seguir concederá essa permissão. Ela contém uma instrução `Condition` para ajudar a evitar um problema de segurança conhecido como problema do substituto confuso. A declaração `Condition` restringe o acesso ao tópico do Amazon SNS para permitir somente operações provenientes dessa conta específica e do espaço de trabalho do Amazon Managed Service for Prometheus. Para obter mais informações sobre o problema confused deputy, veja Prevenção contra o ataque do “substituto confuso” em todos os serviços.

Para dar permissão ao Amazon Managed Service for Prometheus para enviar mensagens para seu tópico do Amazon SNS

1. Abra o console do Amazon SNS em https://console.aws.amazon.com/sns/ v3/home.
2. No painel de navegação, escolha Tópicos.
3. Escolha o nome do tópico que você está usando com o Amazon Managed Service for Prometheus.
4. Escolha Editar.
5. Escolha Política de acesso e adicione a seguinte declaração de política à política existente.

```
{
    "Sid": "Allow_Publish_Alarms",
```

```
    "Effect": "Allow",
    "Principal": {
        "Service": "aps.amazonaws.com"
    },
    "Action": [
        "sns:Publish",
        "sns:GetTopicAttributes"
    ],
    "Condition": {
        "ArnEquals": {
            "aws:SourceArn": "workspace_ARN"
        },
        "StringEquals": {
            "AWS:SourceAccount": "account_id"
        }
    },
    "Resource": "arn:aws:sns:region:account_id:topic_name"
}
```

[Opcional] Se o tópico do Amazon SNS estiver habilitado para a criptografia do lado do serviço (SSE), você precisa permitir que o Amazon Managed Service for Prometheus envie mensagens para esse tópico criptografado adicionando as `kms:Decrypt` permissões `kms:GenerateDataKey*` e à política de chaves da AWS KMS chave usada para criptografar o tópico.

Por exemplo, você poderia adicionar isto à política:

```
{
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": "aps.amazonaws.com"
    },
    "Action": [
      "kms:GenerateDataKey*",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }]
}
```

Para obter mais informações, consulte AWS Permissões KMS para Tópico SNS.

6. Escolha Salvar alterações.

> **Note**
>
> Por padrão, o Amazon SNS cria a política de acesso com a condição em `AWS:SourceOwner`. Para mais informações, consulte a política de acesso do SNS.

> **Note**
>
> O IAM segue a primeira regra mais restritiva da política. Em seu tópico do SNS, se houver um bloco de política mais restritivo do que o bloco documentado na política do Amazon SNS, não será concedida a permissão na política do tópico. Para avaliar a sua política e saber quais as concessões, consulte a Lógica de avaliação da política.

Configuração de tópicos do SNS para regiões opcionais

Você pode usar `aps.amazonaws.com` para configurar um tópico do Amazon SNS da mesma forma Região da AWS que seu espaço de trabalho do Amazon Managed Service for Prometheus. Para usar um tópico do SNS de uma non-opt-in região (como us-east-1) com uma região opcional (como af-south-1), você precisa usar o formato principal do serviço regional. No princípio do serviço regional, *`us-east-1`* substitua pela non-opt-in região que você deseja usar:**`aps.`*`us-east-1`*`.amazonaws.com`**.

A tabela a seguir lista as regiões opcionais e as entidades principais de serviço regional correspondentes:

Regiões opcionais e entidades principais de serviço regional

Nome da região	Região	Entidade principal do serviço regional
África (Cidade do Cabo)	af-south-1	af-south-1.aps.amazonaws.com
Ásia-Pacífico (Hong Kong)	ap-east-1	ap-east-1.aps.amazonaws.com

Nome da região	Região	Entidade principal do serviço regional
Ásia-Pacífico (Tailândia)	ap-southeast-7	ap-southeast-7.aps.amazonaws.com
Europa (Milão)	eu-south-1	eu-south-1.aps.amazonaws.com
Europa (Zurique)	eu-central-2	eu-central-2.aps.amazonaws.com
Oriente Médio (Emirados Árabes Unidos)	me-central-1	me-central-1.aps.amazonaws.com
Ásia-Pacífico (Malásia)	ap-southeast-5	ap-southeast-5.aps.amazonaws.com

Para obter informações sobre como habilitar uma região opcional, consulte Gerenciamento de Regiões da AWS no Guia do usuário do IAM em Referência geral da Amazon Web Services.

Ao configurar seu tópico do Amazon SNS para essas regiões opcionais, certifique-se de usar a entidade principal de serviço regional correta para permitir a entrega de alertas entre regiões.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

“Confused deputy” é um problema de segurança no qual uma entidade sem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executá-la. 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 o uso das chaves de contexto de condição global `aws:SourceArn` e `aws:SourceAccount` nas políticas de recursos para restringir as permissões do recurso que o Amazon Managed Service for Prometheus Amazon concede ao Amazon SNS. Se você utilizar

ambas as chaves de contexto de condição global, o valor `aws:SourceAccount` e a conta no valor `aws:SourceArn` deverão utilizar o mesmo ID de conta quando utilizados na mesma instrução de política.

O valor de `aws:SourceArn` deve ser o ARN do espaço de trabalho do Amazon Managed Service for Prometheus.

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:servicename::123456789012:*`

A política mostrada em Conceder ao Amazon Managed Service for Prometheus permissão para enviar mensagens ao seu tópico do Amazon SNS como usar as chaves de contexto de condição globais `aws:SourceArn` e `aws:SourceAccount` no Amazon Managed Service for Prometheus para evitar o problema confused deputy.

Configurar o gerenciador de alertas para enviar mensagens ao tópico do Amazon SNS

Depois de ter um tópico (novo ou existente) do Amazon SNS do tipo padrão, você poderá adicioná-lo à sua configuração do gerenciador de alertas como receptor de alertas. O gerenciador de alertas pode encaminhar seus alertas para um receptor de alertas configurado. Para fazer isso, você precisa saber o nome do recurso da Amazon (ARN) do seu tópico do Amazon SNS.

Para obter mais informações sobre a configuração do receptor Amazon SNS, consulte <sns_configs> na documentação de configuração do Prometheus.

Propriedades não suportadas

O Amazon Managed Service for Prometheus oferece suporte ao Amazon SNS como receptor de alertas. No entanto, devido às restrições de serviço, nem todas as propriedades do receptor do Amazon SNS são suportadas. As seguintes propriedades não são permitidas em um arquivo de configuração do gerenciador de alertas do Amazon Managed Service for Prometheus:

- `api_url:` – O Amazon Managed Service for Prometheus define `api_url` para você, portanto, essa propriedade não é permitida.
- `Http_config` – Essa propriedade permite que você defina proxies externos. No momento, o Amazon Managed Service for Prometheus não é compatível com esse atributo.

Além disso, é necessário que as configurações do SigV4 tenham uma propriedade de Região. Sem a propriedade Região, o Amazon Managed Service for Prometheus não tem informações suficientes para fazer a solicitação de autorização.

Como configurar o gerenciador de alertas com seu tópico do Amazon SNS como receptor

1. Se você estiver usando um arquivo de configuração do gerenciador de alertas existente, abra-o em um editor de texto.
2. Se houver receptores presentes que não sejam o Amazon SNS no bloco `receivers`, remova-os. Você pode configurar vários tópicos do Amazon SNS para serem receptores colocando-os em blocos `sns_config` separados dentro do bloco `receivers`.
3. Adicione o seguinte bloco YAML na seção `receivers`.

```
- name: name_of_receiver
  sns_configs:
    - sigv4:
        region: Região da AWS
      topic_arn: ARN_of_SNS_topic
      subject: yoursubject
      attributes:
        key: yourkey
        value: yourvalue
```

Se não for especificado `subject`, por padrão, será gerado um assunto com o modelo padrão com o nome do rótulo e os valores, o que pode resultar em um valor muito longo para o SNS. Para alterar o modelo aplicado ao assunto, consulte Configurar o gerenciador de alertas para que envie mensagens ao seu tópico do Amazon SNS como JSON neste guia.

Agora você deve fazer upload do seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus. Para obter mais informações, consulte Carregar seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus.

Configurar o gerenciador de alertas para que envie mensagens ao seu tópico do Amazon SNS como JSON

Por padrão, o gerenciador de alertas do Amazon Managed Service for Prometheus gera mensagens em um formato de lista com texto sem formatação. Isso pode tornar mais difícil a análise por outros serviços. Em vez disso, você pode configurar o gerenciador de alertas para que envie alertas no

formato JSON. O JSON pode simplificar o processamento das mensagens a jusante do Amazon SNS em ou em AWS Lambda endpoints de recebimento de webhooks. Em vez de usar o modelo padrão, você pode definir um modelo personalizado para gerar o conteúdo da mensagem em JSON, facilitando a análise em funções posteriores.

Para enviar mensagens do gerenciador de alertas para o Amazon SNS no formato JSON, atualize a configuração do gerenciador de alertas para conter o seguinte código na seu seção raiz `template_files`:

```
default_template: |
   {{ define "sns.default.message" }}{{ "{" }}"receiver": "{{ .Receiver }}","status":
 "{{ .Status }}","alerts": [{{ range $alertIndex, $alerts := .Alerts }}{{ if
 $alertIndex }}, {{ end }}{{ "{" }}"status": "{{ $alerts.Status }}"{{ if
 gt (len $alerts.Labels.SortedPairs) 0 -}},"labels": {{ "{" }}{{ range
 $index, $label := $alerts.Labels.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $label.Name }}": "{{ $label.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ if gt (len $alerts.Annotations.SortedPairs )
 0 -}},"annotations": {{ "{" }}{{ range $index, $annotations :=
 $alerts.Annotations.SortedPairs }}{{ if $index }}, {{ end }}"{{ $annotations.Name }}":
 "{{ $annotations.Value }}"{{ end }}{{ "}" }}{{- end }},"startsAt":
 "{{ $alerts.StartsAt }}","endsAt": "{{ $alerts.EndsAt }}","generatorURL":
 "{{ $alerts.GeneratorURL }}","fingerprint": "{{ $alerts.Fingerprint }}"{{ "}" }}
{{ end }}]{{ if gt (len .GroupLabels) 0 -}},"groupLabels": {{ "{" }}{{ range
 $index, $groupLabels := .GroupLabels.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $groupLabels.Name }}": "{{ $groupLabels.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ if gt (len .CommonLabels) 0 -}},"commonLabels": {{ "{" }}
{{ range $index, $commonLabels := .CommonLabels.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $commonLabels.Name }}": "{{ $commonLabels.Value }}"{{ end }}{{ "}" }}{{-
 end }}{{ if gt (len .CommonAnnotations) 0 -}},"commonAnnotations": {{ "{" }}{{ range
 $index, $commonAnnotations := .CommonAnnotations.SortedPairs }}{{ if $index }},
 {{ end }}"{{ $commonAnnotations.Name }}": "{{ $commonAnnotations.Value }}"{{ end }}
{{ "}" }}{{- end }}{{ "}" }}{{ end }}
   {{ define "sns.default.subject" }}[{{ .Status | toUpper }}{{ if eq .Status
 "firing" }}:{{ .Alerts.Firing | len }}{{ end }}]{{ end }}
```

> **Note**
>
> Esse modelo cria JSON a partir de dados alfanuméricos. Se seus dados tiverem caracteres especiais, codifique-os antes de usar esse modelo.

Para garantir que esse modelo seja usado nas notificações enviadas, faça referência a ele em seu bloco `alertmanager_config` da seguinte forma:

```
alertmanager_config: |
  global:
  templates:
    - 'default_template'
```

> **Note**
>
> Esse modelo é para o corpo inteiro da mensagem como o da mensagem JSON. Esse modelo substitui o corpo inteiro da mensagem. Você não pode substituir o corpo da mensagem se quiser usar esse modelo específico. Todas as substituições feitas manualmente terão precedência sobre o modelo.

Para obter mais informações sobre:

- O arquivo de configuração do gerenciador de alertas, consulte Criar uma configuração do gerenciador de alertas no Amazon Managed Service for Prometheus para gerenciar e encaminhar alertas.
- Como fazer o upload do seu arquivo de configuração, consulte Carregar seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus.

Configurar o Amazon SNS para que envie mensagens de alertas a outros destinos

O Amazon Managed Service for Prometheus só pode enviar mensagens de alertas ao Amazon Simple Notification Service (Amazon SNS). Para enviar essas mensagens para outros destinos, como e-mail, webhook, Slack ou OpsGenie, você deve configurar o Amazon SNS para encaminhar as mensagens para esses endpoints.

As próximas seções descrevem como configurar o Amazon SNS para que encaminhe alertas para outros destinos.

Tópicos

- E-mail
- Webhook

- Slack
- OpsGenie

E-mail

Para configurar um tópico do Amazon SNS para enviar mensagens para e-mail, crie uma assinatura. No console do Amazon SNS, escolha a guia Assinaturas para abrir a página da lista de Assinaturas. Escolha Criar assinatura e selecione E-mail. O Amazon SNS envia um e-mail de confirmação ao endereço de e-mail listado. Depois de aceitar a confirmação, você poderá receber notificações do Amazon SNS, como e-mails do tópico em que você se inscreveu. Para obter mais informações, consulte Assinatura de um tópico do Amazon SNS.

Webhook

Para configurar um tópico do Amazon SNS para enviar mensagens para um endpoint de webhook, crie uma assinatura. No console do Amazon SNS, escolha a guia Assinaturas para abrir a página da lista de Assinaturas. Escolha Criar assinatura e selecione HTTP/HTTPS. Depois de criar a assinatura, você deve seguir as etapas de confirmação para ativá-la. Quando estiver ativo, seu endpoint HTTP deve receber as notificações do Amazon SNS. Para obter mais informações, consulte Assinatura de um tópico do Amazon SNS. Para obter mais informações, consulte Como uso webhooks para publicar mensagens do Amazon SNS no Amazon Chime, Slack ou Microsoft Teams?

Slack

Para configurar um tópico do Amazon SNS para enviar mensagens para o Slack, você tem duas opções. Você pode fazer a integração com a email-to-channel integração do Slack, que permite que o Slack aceite mensagens de e-mail e as encaminhe para um canal do Slack, ou você pode usar uma função Lambda para reescrever a notificação do Amazon SNS para o Slack. Para obter mais informações sobre o encaminhamento de e-mails para os canais do Slack, consulte Confirmação da assinatura do AWS SNS Topic para o Slack Webhook. Para obter mais informações sobre a criação de uma função do Lambda para converter mensagens do Amazon SNS em Slack, consulte Como integrar o Amazon Managed Service for Prometheus com o Slack.

OpsGenie

Para obter informações sobre como configurar um tópico do Amazon SNS para enviar mensagens OpsGenie, consulte Integrar o Opsgenie com o Amazon SNS de entrada.

Noções básicas das regras de validação de mensagens do Amazon SNS

O Amazon Simple Notification Service (Amazon SNS) exige que as mensagens atendam a determinados padrões. As mensagens que não atendem a esses padrões serão modificadas quando forem recebidas. Se necessário, as mensagens de alertas serão validadas, truncadas ou modificadas pelo receptor do Amazon SNS com base nas seguintes regras:

- A mensagem contém caracteres não utf.
  - A mensagem será substituída por Error - not a valid UTF-8 encoded string.
  - Um atributo de mensagem será adicionado com a chave truncated e o valor true.
  - Um atributo de mensagem será adicionado com a chave de item modified e o valor de Message: Error - not a valid UTF-8 encoded string.
- A mensagem está vazia.
  - A mensagem será substituída por Error - Message should not be empty.
  - Um atributo de mensagem será adicionado com a chave de item modified e o valor de Message: Error - Message should not be empty.
- A mensagem foi truncada.
  - A mensagem terá o conteúdo truncado.
  - Um atributo de mensagem será adicionado com a chave truncated e o valor true.
  - Um atributo de mensagem será adicionado com a chave de “modificado” e o valor de Mensagem: Erro - A mensagem foi truncada de ***X*** KB, porque excede o limite de tamanho de 256 KB.
- O assunto contém caracteres de controle ou não ASCII.
  - Se o assunto tiver caracteres de controle ou caracteres não ASCII, o SNS substituirá o assunto por Error - contains control- or non-ASCII characters.
  - Para assuntos de e-mail do SNS, remova os caracteres de controle, como novas linhas: \n.
- O assunto não é ASCII.
  - O assunto será substituído por Error - contains non printable ASCII characters.
  - Um atributo de mensagem será adicionado com a chave de item modified e o valor de Subject: Error - contains non-printable ASCII characters.
- O assunto foi truncado.
  - O assunto terá o conteúdo truncado.

  - Um atributo de mensagem será adicionado com a chave de modificado e o valor de Assunto: Erro - O assunto foi truncado de ***X*** caracteres, pois excede o limite de tamanho de 100 caracteres.
- O atributo da mensagem tem chave/valor inválido.
  - O atributo de mensagem inválido será removido.
  - Um atributo de mensagem será adicionado com a chave de modificado e o valor de MessageAttribute: Erro - se os atributos ***X*** da mensagem foram removidos por causa de MessageAttributeKey ou MessageAttributeValue inválido.
- O atributo da mensagem foi truncado.
  - Os atributos extras da mensagem serão removidos.
  - Um atributo de mensagem será adicionado com a chave de modificada e o valor de MessageAttribute: Erro - se os atributos ***X*** da mensagem tiverem sido removidos, pois excede o limite de tamanho de 256 KB.

Use PagerDuty como receptor de alertas

Você pode configurar o Amazon Managed Service for Prometheus para enviar alertas diretamente para. PagerDuty Essa integração exige que você armazene sua chave de PagerDuty integração AWS Secrets Manager e conceda permissão ao Amazon Managed Service for Prometheus para ler o segredo.

PagerDuty a integração permite fluxos de trabalho automatizados de resposta a incidentes e garante que alertas críticos cheguem aos membros certos da equipe no momento certo. Ao usar PagerDuty como receptor de alertas, você pode aproveitar as políticas de escalonamento, PagerDuty o agendamento de plantões e os recursos de gerenciamento de incidentes para garantir que os alertas sejam reconhecidos e resolvidos rapidamente. Essa integração é especialmente útil em ambientes de produção em que a resposta rápida aos problemas do sistema é essencial para manter a disponibilidade do serviço e atender aos requisitos de SLA. Para obter mais informações, consulte a Base de PagerDuty Conhecimento no PagerDuty site.

PagerDuty opções de configuração

Opção	Description	Obrigatório
`routing_key`	A chave PagerDuty de roteamento para uma integração em um serviço. Você deve especificá-la como um ARN do Secrets Manager	Sim
`service_key`	A chave PagerDuty de serviço para uma integração em um serviço. Você deve especific á-la como um ARN do Secrets Manager	Sim (para a API de eventos v1)
`client`	A identificação do cliente do notificador	Não
`client_url`	Um backlink para o remetente da notificação	Não
`description`	Descrição do incidente	Não
`details`	Um conjunto de key/value pares arbitrários que fornecem mais detalhes sobre o incidente	Não
`severity`	Gravidade do incidente	Não
`class`	A classe, ou tipo, do evento	Não
`component`	Componente da máquina de origem responsável pelo evento	Não
`group`	Agrupamento lógico de componentes	Não

Opção	Description	Obrigatório
`source`	A localização exclusiva do sistema afetado	Não

> **Note**
>
> As opções `url`, `service_key_file`, `routing_key_file` e `http_config` não são compatíveis.

Os tópicos a seguir descrevem como configurar PagerDuty como um receptor de alertas no Amazon Managed Service para Prometheus.

Tópicos

- Configuração AWS Secrets Manager e permissões
- Configure o gerenciador de alertas para enviar alertas para PagerDuty

Configuração AWS Secrets Manager e permissões

Antes de enviar alertas para PagerDuty, você deve armazenar com segurança sua chave de PagerDuty integração e configurar as permissões necessárias. Esse processo envolve criar um segredo AWS Secrets Manager, criptografá-lo com uma chave gerenciada pelo cliente AWS Key Management Service (AWS KMS) e conceder ao Amazon Managed Service for Prometheus as permissões necessárias para acessar o segredo e sua chave de criptografia. Os procedimentos a seguir orientam você em cada etapa desse processo de configuração.

Para criar um segredo no Secrets Manager para PagerDuty

Para usar PagerDuty como receptor de alertas, você deve armazenar sua chave de PagerDuty integração no Secrets Manager. Siga estas etapas:

1. Abra o console do Secrets Manager.
2. Selecione Armazenar um novo segredo.
3. Em Tipo de segredo, escolha Outro tipo de segredo.

4. Para pares chave/valor, insira sua chave de PagerDuty integração como valor secreto. Essa é a chave de roteamento ou a chave de serviço da sua PagerDuty integração.
5. Escolha Próximo.
6. Insira um nome e uma descrição para o seu segredo e selecione Avançar.
7. Defina as configurações de rotação, se desejar, e escolha Avançar.
8. Revise as configurações e clique em Armazenar.
9. Após criar o segredo, anote seu ARN. Você precisará dele ao configurar o gerenciador de alertas.

Para criptografar seu segredo com uma chave gerenciada pelo cliente AWS KMS

Você deve conceder permissão ao Amazon Managed Service for Prometheus para acessar seu segredo e a chave de criptografia dele:

1. Política de recursos secretos: abra seu segredo no console do Secrets Manager.
   a. Escolha Permissões de recursos.
   b. Escolha Editar permissões.
   c. Adicione a seguinte declaração de política: Na declaração, substitua o *highlighted values* por seus valores específicos.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "aps.amazonaws.com"
  },
  "Action": "secretsmanager:GetSecretValue",
  "Resource": "*",
  "Condition": {
    "ArnEquals": {
      "aws:SourceArn": "arn:aws:aps:aws-
region:123456789012:workspace/WORKSPACE_ID"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012"
    }
  }
}
```

   d. Escolha Salvar.

2. Política de chaves do KMS: abra sua AWS KMS chave no AWS KMS console.

   a. Escolha Política de chave.

   b. Escolha Editar.

   c. Adicione a seguinte declaração de política: Na declaração, substitua o *highlighted values* por seus valores específicos.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "aps.amazonaws.com"
  },
  "Action": "kms:Decrypt",
  "Resource": "*",
  "Condition": {
    "ArnEquals": {
      "aws:SourceArn": "arn:aws:aps:aws-
region:123456789012:workspace/WORKSPACE_ID"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012"
    }
  }
}
```

   d. Escolha Salvar.

Próximos passos: continue para o próximo tópico, Configure o gerenciador de alertas para enviar alertas para PagerDuty.

Configure o gerenciador de alertas para enviar alertas para PagerDuty

Para configurar o gerenciador de alertas para o qual enviar alertas PagerDuty, você precisa atualizar sua definição do gerenciador de alertas. Você pode fazer isso usando o Console de gerenciamento da AWS, AWS CLI, ou AWS SDKs.

Example configuração do gerenciador de alertas

A seguir, é apresentado um exemplo de configuração do gerenciador de alertas que envia alertas para PagerDuty o. No exemplo, substitua o *highlighted values* por seus valores específicos.

```
alertmanager_config: |
  route:
    receiver: 'pagerduty-receiver'
    group_by: ['alertname']
    group_wait: 30s
    group_interval: 5m
    repeat_interval: 1h
  receivers:
    - name: 'pagerduty-receiver'
      pagerduty_configs:
      - routing_key:
          aws_secrets_manager:
            secret_arn: 'arn:aws:secretsmanager:aws-
region:123456789012:secret:YOUR_SECRET_NAME'
            secret_key: 'YOUR_SECRET_KEY'
            refresh_interval: 5m
        description: '{{ .CommonLabels.alertname }}'
        severity: 'critical'
        details:
          firing: '{{ .Alerts.Firing | len }}'
          status: '{{ .Status }}'
          instance: '{{ .CommonLabels.instance }}'
```

Example AWS CLI

A seguir, é apresentado um AWS CLI comando usado para atualizar sua definição do gerenciador de alertas. No exemplo, substitua o *highlighted values* por seus valores específicos.

```
aws amp put-alert-manager-definition \
  --workspace-id WORKSPACE_ID \
  --data file://alertmanager-config.yaml
```

Solução de problemas de PagerDuty integração

Se os alertas não estiverem sendo enviados para PagerDuty, verifique os seguintes itens:

- Verifique se seu segredo existe e contém a chave de PagerDuty integração correta.
- Confirme se o seu segredo está criptografado com uma chave do KMS gerenciada pelo cliente.

- Certifique-se de que as políticas de recursos tanto para a chave secreta quanto para a chave KMS concedam as permissões necessárias ao Amazon Managed Service for Prometheus.
- Verifique se o ARN na configuração do gerenciador de alertas faz referência correta ao seu segredo.
- Verifique se sua chave de PagerDuty integração é válida e está ativa em sua PagerDuty conta.

O Amazon Managed Service para Prometheus oferece suporte ao CloudWatch Amazon Logs e às seguintes métricas para CloudWatch ajudar na solução de problemas. Para obter mais informações, consulte Monitore eventos do Amazon Managed Service para Prometheus com registros CloudWatch e Use CloudWatch métricas para monitorar os recursos do Amazon Managed Service for Prometheus.

- `SecretFetchFailure`
- `AlertManagerNotificationsThrottledByIntegration`
- `AlertManagerNotificationsFailedByIntegration`

Carregar seu arquivo de configuração do gerenciador de alertas no Amazon Managed Service for Prometheus

Depois de identificar o que você quer no seu arquivo de configuração do gerenciador de regras, você poderá criar e editar isso no console ou carregar um arquivo existente com o AWS CLI ou o console do Amazon Managed Service for Prometheus.

> **Note**
>
> Se você estiver executando um cluster do Amazon EKS, também poderá carregar um novo arquivo de configuração do gerenciador de alertas usando Controladores da AWS para Kubernetes.

Para usar o console do Amazon Managed Service for Prometheus para editar ou substituir sua configuração do gerenciador de alertas

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/

2. No canto superior esquerdo da página, selecione o ícone do menu e escolha Todos os espaços de trabalho.
3. Selecione o ID do espaço de trabalho e, em seguida, selecione a guia Gerenciador de alertas.
4. Se o espaço de trabalho ainda não tiver uma definição de gerenciador de alertas, selecione Adicionar definição.

   > **Note**
   >
   > Se o espaço de trabalho tiver uma definição do gerenciador de alertas que você deseja substituir, selecione Modificar.

5. Selecione Escolher arquivo, selecione o arquivo de definição do gerenciador de alertas e Continuar.

   > **Note**
   >
   > Outra possibilidade é criar um novo arquivo e editá-lo diretamente pelo console, selecionando a opção Criar definição. Isso criará uma amostra de configuração padrão para você editar antes do carregamento.

Para usar o AWS CLI para carregar uma configuração do gerenciador de alertas em um espaço de trabalho pela primeira vez

1. O Base64 codifica o conteúdo do seu arquivo do gerenciador de alertas. Em um sistema Linux, use o seguinte comando:

   ```
   base64 input-file output-file
   ```

   No macOS, use o seguinte comando:

   ```
   openssl base64 input-file output-file
   ```

2. Para fazer o upload, insira um dos seguintes comandos.

   Na AWS CLI versão 2, digite:

```
aws amp create-alert-manager-definition --data file://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

Na AWS CLI versão 1, digite:

```
aws amp create-alert-manager-definition --data fileb://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

3. São necessários alguns segundos para que a configuração do Alert Manager entre em vigor. Para verificar o status, insira o comando a seguir:

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

Se o `status` estiver `ACTIVE`, a sua nova definição do gerenciador de alertas está em vigor.

Para usar o AWS CLI para substituir a configuração do gerenciador de alertas de um espaço de trabalho por uma nova

1. O Base64 codifica o conteúdo do seu arquivo do gerenciador de alertas. Em um sistema Linux, use o seguinte comando:

```
base64 input-file output-file
```

No macOS, use o seguinte comando:

```
openssl base64 input-file output-file
```

2. Para fazer o upload, insira um dos seguintes comandos.

Na AWS CLI versão 2, digite:

```
aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
workspace-id my-workspace-id --region region
```

Na AWS CLI versão 1, digite:

```
aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
workspace-id my-workspace-id --region region
```

3. São necessários alguns segundos para que a sua nova configuração do gerenciador de alertas fique ativa. Para verificar o status, insira o comando a seguir:

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

Se o `status` estiver `ACTIVE`, a sua nova definição do gerenciador de alertas está em vigor. Antes disso, a sua configuração anterior do gerenciador de alertas ainda está ativa.

Integrar alertas com o Amazon Managed Grafana ou o Grafana de código aberto

As regras de alerta que você criou no Alertmanager dentro do Amazon Managed Service for Prometheus podem ser encaminhadas e visualizadas no Amazon Managed Grafana e no Grafana, unificando suas regras de alerta e alertas em um único ambiente. No Amazon Managed Grafana, você pode visualizar suas regras de alerta e os alertas que são gerados.

Pré-requisitos

Antes de começar a integrar o Amazon Managed Service for Prometheus ao Amazon Managed Grafana, você deve preencher os seguintes pré-requisitos:

- Você deve ter credenciais existentes Conta da AWS e do IAM para criar programaticamente as funções do Amazon Managed Service for Prometheus e do IAM.

  Para obter mais informações sobre a criação de credenciais de um Conta da AWS e do IAM, consulteConfigurar AWS.
- Você deve ter um espaço de trabalho do Amazon Managed Service for Prometheus e estar ingerindo dados nele. Para configurar um novo espaço de trabalho, consulte Criar um espaço de trabalho do Amazon Managed Service for Prometheus. Você também deve estar familiarizado com os conceitos do Prometheus, como o Alertmanager e o Ruler. Para obter informações sobre esses tópicos, consulte a documentação do Prometheus.

- Você tem uma configuração do Alertmanager e um arquivo de regras já configurados no Amazon Managed Service for Prometheus. Para obter mais informações sobre Alertmanager no Amazon Managed Service for Prometheus, consulte Como gerenciar e encaminhar alertas no Amazon Managed Service for Prometheus com o gerenciador de alertas. Para saber mais sobre regras de , consulte Como usar regras para modificar ou monitorar métricas à medida são recebidas.
- Você deve ter o Amazon Managed Grafana configurado ou a versão de código aberto do Grafana em execução.
  - Se você estiver usando o Amazon Managed Grafana, deverá usar alertas do Grafana. Para obter mais informações, consulte Migração de alertas de painéis legados para alertas do Grafana.
  - Se você estiver usando a versão de código aberto do Grafana, deverá executar a versão 9.1 ou superior.

    > **Note**
    >
    > Você pode usar versões anteriores do Grafana, mas deve habilitar o atributo de alerta unificado (alerta do Grafana) e talvez seja necessário configurar um proxy sigv4 para fazer chamadas do Grafana para o Amazon Managed Service for Prometheus. Para obter mais informações, consulte Configurar o Grafana de código aberto ou o Grafana Enterprise para uso com o Amazon Managed Service for Prometheus.
- O Amazon Managed Grafana deve ter as seguintes permissões para seus recursos do Prometheus. Você deve adicioná-los às políticas gerenciadas pelo serviço ou pelo cliente descritas em https://docs.aws.amazon.com/grafana/latest/userguide/AMG-manage-permissions.html.
  - `aps:ListRules`
  - `aps:ListAlertManagerSilences`
  - `aps:ListAlertManagerAlerts`
  - `aps:GetAlertManagerStatus`
  - `aps:ListAlertManagerAlertGroups`
  - `aps:PutAlertManagerSilences`
  - `aps:DeleteAlertManagerSilence`

Configuração do Amazon Managed Grafana

Se você já configurou regras e alertas em sua instância do Amazon Managed Service for Prometheus, a configuração para usar o Amazon Managed Grafana como um painel para esses alertas é feita inteiramente dentro do Amazon Managed Grafana.

Para configurar o Amazon Managed Grafana como seu painel de alertas

1. Abra o console do Grafana em seu espaço de trabalho.
2. Em Configurações, escolha Fontes de dados.
3. Crie ou abra sua fonte de dados do Prometheus. Se você ainda não configurou uma fonte de dados do Prometheus, consulte Etapa 2: adicionar a fonte de dados do Prometheus no Grafana para obter mais informações.
4. Na fonte de dados do Prometheus, selecione Gerenciar alertas por meio da interface do usuário do Alertmanager.
5. Volte para a interface de fontes de dados.
6. Crie uma fonte de dados do Alertmanager.
7. Na página de configuração da fonte de dados do Alertmanager, adicione as seguintes configurações:

   - Defina a Implementação como `Prometheus`.
   - Para a configuração do URL, use o URL do seu espaço de trabalho do Prometheus, remova tudo após o ID do espaço de trabalho e anexe o `/alertmanager` ao final. No exemplo a seguir, *`variables`* substitua as por suas próprias informações (específicas da conta):

     ```
     https://aps-workspaces.US East (N. Virginia).amazonaws.com/workspaces/ws-example-1234-5678-abcd-xyz00000001/alertmanager.
     ```

   - Em Auth, ative SigV4Auth. Isso diz ao Grafana para usar a autenticação da AWS para as solicitações.
   - Em Detalhes de SigV4Auth, em Região padrão, forneça a região da sua instância do Prometheus, por exemplo, `us-east-1`.
   - Defina a opção Padrão como `true`.
8. Escolha Save and test.
9. Seus alertas do Amazon Managed Service for Prometheus agora devem ser configurados para funcionar com sua instância do Grafana. Verifique se você pode ver Regras de alerta, Grupos

de alerta (incluindo alertas ativos) e Silêncios da sua instância do Amazon Managed Service for Prometheus na página de Alertas do Grafana.

Solucione problemas do gerenciador de alertas com CloudWatch o Logs

Utilizando Monitore eventos do Amazon Managed Service para Prometheus com registros CloudWatch, você pode solucionar problemas relacionados ao gerenciador de alertas e ao Ruler. Esta seção contém tópicos de solução de problemas relacionados ao gerenciador de alertas.

Tópicos

Aviso de alertas ativos

Quando o log contém o seguinte aviso

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "too many alerts, limit: 1000",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Isso significa que a cota de alertas ativos do gerenciador de alertas foi excedida.

Medida a ser tomada

Solicite um aumento da cota. Faça login Console de gerenciamento da AWS e abra o console Service Quotas em. https://console.aws.amazon.com/servicequotas/

Aviso de tamanho do grupo de agregação de alertas

Quando o log contém o seguinte aviso

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "Too many aggregation groups, cannot create new group for alert,
 groups=1000, limit=1000, alert=sample-alert",
        "level": "WARN"
    },
    "component": "alertmanager"
}
```

Isso significa que a cota de tamanho do grupo de agregação de alertas do gerenciador de alertas foi excedida.

Medida a ser tomada

Reduza o tamanho do grupo de agregação de alertas usando o parâmetro `group_by`. Para obter mais informações, consulte Configurações relacionadas à rota na documentação do Prometheus.

Também é possível solicitar um aumento da cota. Faça login Console de gerenciamento da AWS e abra o console Service Quotas em. https://console.aws.amazon.com/servicequotas/

Aviso de alerta muito grande

Quando o log contém o seguinte aviso

```
{
    "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
    "message": {
        "log": "alerts too big, total size limit: 20000000 bytes",
        "level": "WARN"
```

```
    },
    "component": "alertmanager"
}
```

Isso significa que a cota de alertas por espaço de trabalho do Alert Manager foi excedida.

Medida a ser tomada

Remova anotações e rótulos desnecessários para reduzir o tamanho do alerta.

Aviso de conteúdo vazio

Quando o log contém o seguinte aviso

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Message has been modified because the content was empty."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Isso significa que o modelo do gerenciador de alertas resolveu o alerta de saída em uma mensagem vazia.

Medida a ser tomada

Valide o seu modelo do gerenciador de alertas e garanta que você tenha um modelo válido para todos os caminhos do receptor.

Aviso de **`key/value`** inválido

Quando o log contém o seguinte aviso

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "MessageAttributes has been removed because of invalid key/value,
 numberOfRemovedAttributes=1"
       "level": "WARN"
   },
```

```
   "component": "alertmanager"
}
```

Isso significa que alguns dos atributos da mensagem foram removidos por keys/values serem inválidos.

Medida a ser tomada

Reavalie os modelos que você está usando para preencher os atributos da mensagem e certifique-se de que eles estão resultando em um atributo de mensagem do SNS válido. Para obter mais informações sobre como validar uma mensagem em um tópico do Amazon SNS, consulte o tópico Validar SNS

Aviso de limite de mensagens

Quando o log contém o seguinte aviso

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Message has been truncated because it exceeds size limit,
 originSize=266K, truncatedSize=12K"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Isso significa que parte do tamanho da mensagem é muito grande.

Medida a ser tomada

Veja o modelo de mensagem do receptor de alerta e reformule-o para caber dentro do limite de tamanho.

Nenhum erro da política baseada no recurso

Quando o log contém o seguinte erro

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
```

```
       "log": "Notify for alerts failed, AMP is not authorized to perform: SNS:Publish
 on resource: arn:aws:sns:us-west-2:12345:testSnsReceiver because no resource-based
 policy allows the SNS:Publish action"
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Isso significa que o Amazon Managed Service for Prometheus não tem as permissões para enviar o alerta para o tópico do SNS especificado.

Medida a ser tomada

Verifique se a política de acesso no tópico do Amazon SNS concede ao Amazon Managed Service for Prometheus a capacidade de enviar mensagens do SNS para o tópico. Crie uma política de acesso do SNS para que o serviço `aps.amazonaws.com` (Amazon Managed Service for Prometheus) acesse seu tópico do Amazon SNS. Para obter mais informações sobre políticas de acesso do SNS, consulte Como usar a linguagem de políticas de acesso e Casos de exemplo para o controle de acesso do Amazon SNS no Guia do desenvolvedor do Amazon Simple Notification Service.

Aviso não ASCII

Quando o log contém o seguinte aviso

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Subject has been modified because it contains control or non-ASCII
 characters."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

Isso significa que o assunto tem caracteres não ASCII.

Medida a ser tomada

Remova as referências no campo de assunto do seu modelo dos rótulos que possam conter caracteres não ASCII.

Não autorizado a chamar o KMS

Quando o log contém o seguinte erro do AWS KMS

```
{
   "workspaceId": "ws-abcd1234-ef56-78ab-cd90-1234abcd0000",
   "message": {
       "log": "Notify for alerts failed, AMP is not authorized to call KMS",
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Medida a ser tomada

Valide se a política de chave da chave usada para criptografar o tópico do Amazon SNS permite que a entidade principal do serviço do Amazon Managed Service for Prometheus `aps.amazonaws.com` execute as seguintes ações: `kms:GenerateDataKey*` e `kms:Decrypt`. Para obter mais informações, consulte o tópico Permissões do AWS KMS para SNS.

Erro de modelo

Quando o log contém o seguinte erro

```
              {
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "message": {
       "log": "Notify for alerts failed. There is an error in a receiver that is using
 templates in the AlertManager definition. Make sure that the syntax is correct and
 only template functions and variables that exist are used in the receiver 'default',
 sns_configs position #2, section 'attributes'"
       "level": "ERROR"
   },
   "component": "alertmanager"
}
```

Isso significa que há um erro em um modelo sendo usado na AlertManager definição. A entrada de erro contém instruções sobre qual receptor, a posição no sns_configs e a propriedade que contém erros.

Medida a ser tomada

Valide sua definição do Alert Manager. Verifique se a sintaxe está correta e se você faz referência às variáveis e funções do modelo que existem. Para obter mais informações, consulte Notification Template Reference na documentação de código aberto do Prometheus.

Registro e monitoramento de espaços de trabalho do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus usa a CloudWatch Amazon para fornecer dados sobre sua operação. Você pode usar CloudWatch métricas para aprender sobre o uso de recursos e solicitações para seus espaços de trabalho do Amazon Managed Service for Prometheus. Você pode ativar o suporte a CloudWatch registros para obter registros de eventos que acontecem em seus espaços de trabalho.

Os tópicos a seguir descrevem o uso CloudWatch com mais detalhes.

Use CloudWatch métricas para monitorar os recursos do Amazon Managed Service for Prometheus

O Amazon Managed Service para Prometheus vende métricas de uso para. CloudWatch Essas métricas fornecem visibilidade sobre a utilização do seu espaço de trabalho. As métricas vendidas podem ser encontradas nos `AWS/Prometheus` namespaces `AWS/Usage` e em. CloudWatch Essas métricas estão disponíveis CloudWatch gratuitamente. Para obter mais informações sobre métricas de uso, consulte Métricas de uso da CloudWatch .

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount*	CreateAlertManagerAlertsTPS	`AWS/Usage`	Número máximo de operações da API `CreateAlertManagerAlerts` por segundo, por espaço de trabalho.
ResourceCount*	DeleteAlertManagerSilencesTPS	`AWS/Usage`	Número máximo de operações da API `DeleteAlertManagerSilences` por segundo, por espaço de trabalho.

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount*	GetAlertManagerSilenceTPS	AWS/Usage	Número máximo de operações da API GetAlertManagerSilence por segundo, por espaço de trabalho.
ResourceCount*	GetAlertManagerStatusTPS	AWS/Usage	Número máximo de operações da API GetAlertManagerStatus por segundo, por espaço de trabalho.
ResourceCount*	GetLabelsTPS	AWS/Usage	Número máximo de operações da API GetLabels por segundo, por espaço de trabalho.
ResourceCount*	GetMetricMetadataTPS	AWS/Usage	Número máximo de operações da API GetMetricMetadata por segundo, por espaço de trabalho.
ResourceCount*	GetSeriesTPS	AWS/Usage	Número máximo de operações da API GetSeries por segundo, por espaço de trabalho.
ResourceCount	InhibitionRulesInAlertManagerDefinition	AWS/Usage	Número máximo de regras de inibição no arquivo de definição do gerenciador de alertas.

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount*	ListAlertManagerAlertGroupInfosTPS	AWS/Usage	Número máximo de operações da API ListAlertManagerAlertGroupInfos por segundo, por espaço de trabalho.
ResourceCount*	ListAlertManagerAlertGroupsTPS	AWS/Usage	Número máximo de operações da API ListAlertManagerAlertGroups por segundo, por espaço de trabalho.
ResourceCount*	ListAlertManagerAlertsTPS	AWS/Usage	Número máximo de operações da API ListAlertManagerAlerts por segundo, por espaço de trabalho.
ResourceCount*	ListAlertManagerReceiversTPS	AWS/Usage	Número máximo de operações da API ListAlertManagerReceivers por segundo, por espaço de trabalho.
ResourceCount*	ListAlertManagerSilencesTPS	AWS/Usage	Número máximo de operações da API ListAlertManagerSilences por segundo, por espaço de trabalho.

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount*	ListAlertsTPS	`AWS/Usage`	Número máximo de operações da API `ListAlerts` por segundo, por espaço de trabalho.
ResourceCount*	ListRulesTPS	`AWS/Usage`	Número máximo de operações da API `ListRules` por segundo, por espaço de trabalho.
ResourceCount*	PutAlertManagerSilencesTPS	`AWS/Usage`	Número máximo de operações da API `PutAlertManagerSilences` por segundo, por espaço de trabalho.
ResourceCount	HAReplicaGroupCount	`AWS/Usage`	Número de grupos de réplicas de alta disponibilidade
ResourceCount*	QueryMetricsTPS	`AWS/Usage`	Operações de solicitação por segundo
ResourceCount*	RemoteWriteTPS	`AWS/Usage`	Operações de gravação remota por segundo.
ResourceCount	ActiveAlerts	`AWS/Usage`	Número de alertas ativos por espaço de trabalho Unidades: contagem Estatísticas válidas: média, mínima, máxima

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount	ActiveSeries	AWS/Usage	Número de séries ativas por espaço de trabalho Unidades: contagem Estatísticas válidas: média, mínima, máxima
ResourceCount	AlertAggregationGroupSize	AWS/Usage	Tamanho máximo do grupo de agregação de alertas no arquivo de definição do gerenciador de alertas. Cada combinação de valores de rótulo de `group_by` cria um grupo de agregação.
ResourceCount	AlertManagerDefinitionSizeBytes	AWS/Usage	Tamanho máximo de um arquivo de definição do gerenciador de alertas, em bytes.
ResourceCount	AllSilences	AWS/Usage	Número máximo de silêncios, incluindo silêncios expirados, ativos e pendentes, por espaço de trabalho.
ResourceCount	IngestionRate	AWS/Usage	Taxa de ingestão da amostra Unidades: contagem por segundo Estatísticas válidas: média, mínima, máxima

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount	RuleEvaluationInterval	AWS/Usage	O intervalo mínimo de avaliação de regras
ResourceCount	RuleGroupNamespaceDefinitionSizeBytes	AWS/Usage	O tamanho máximo de um arquivo de definição de namespace de grupo de regras, em bytes.
ResourceCount	TemplatesInAlertManagerDefinition	AWS/Usage	O número máximo de modelos no arquivo de definição do gerenciador de alertas.
ResourceCount	WorkspaceCount	AWS/Usage	O número máximo de espaços de trabalho por Região, por conta.
ResourceCount	SizeOfAlerts	AWS/Usage	Tamanho total de todos os alertas no espaço de trabalho, em bytes Unidade: bytes Estatísticas válidas: média, mínima, máxima

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount	SuppressedAlerts	AWS/Usage	Número de alertas em estado suprimido por espaço de trabalho. Um alerta pode ser suprimido por um silêncio ou uma inibição. Unidades: contagem Estatísticas válidas: média, mínima, máxima
ResourceCount	UnprocessedAlerts	AWS/Usage	Número de alertas em estado não processado por espaço de trabalho. Um alerta fica em estado não processado depois de recebido AlertManager, mas aguarda a próxima avaliação do grupo de agregação. Unidades: contagem Estatísticas válidas: média, mínima, máxima
ResourceCount	AllAlerts	AWS/Usage	Número de alertas em qualquer estado por espaço de trabalho Unidades: contagem Estatísticas válidas: média, mínima, máxima

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
ResourceCount	AllRules	AWS/Usage	Número de regras em qualquer estado por espaço de trabalho Unidades: contagem Estatísticas válidas: média, mínima, máxima
ActiveSeriesPerLabelSet	-	AWS/Prometheus	O uso atual da série ativa para cada conjunto de rótulos definido pelo usuário Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
ActiveSeriesLimitPerLabelSet	-	AWS/Prometheus	O valor atual do limite de séries ativas para cada conjunto de rótulos definido pelo usuário. Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
AlertManagerAlerts Received	-	AWS/Prometheus	Total de alertas concluídos recebidos pelo gerenciador de alertas Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
AlertManagerNotificationsFailed	-	AWS/Prometheus	Número de entregas de alertas com falha Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
AlertManagerNotificationsThrottled	-	AWS/Prometheus	Número de alertas com controle de utilização Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
AnomalyDetectors	WorkspaceId	AWS/Prometheus	Número total de detectores de anomalias para um determinado espaço de trabalho Unidades: contagem Estatísticas válidas: média, mínima, máxima
AnomalyDetectorEvaluations	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Número total de avaliações de detectores de anomalias Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
AnomalyDetectorEvaluationFailures	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Número de falhas no detector de anomalias no intervalo Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
AnomalyDetectorLastEvaluationDuration	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Duração da última avaliação de um detector de anomalias Unidades: segundos Estatísticas válidas: média, mínimo, máximo, soma
AnomalyDetectorMissedEvaluations	WorkspaceId, AnomalyDetectorId	AWS/Prometheus	Número de avaliações perdidas de detectores de anomalias no intervalo Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
Discarded Samples**	-	AWS/Prometheus	Número de amostras descartadas por motivo Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
Discarded Series**	-	AWS/Prometheus	Número de séries que contêm uma amostra descartada por motivo Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
Discarded SamplesPerLabelSet	-	AWS/Prometheus	A contagem de amostras descartadas para cada conjunto de etiquetas definido pelo usuário Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
Discarded SeriesPerLabelSet	-	AWS/Prometheus	A contagem de séries que contêm uma amostra descartada para cada conjunto de rótulos definido pelo usuário Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
Ingestion RatePerLabelSet	-	`AWS/Prometheus`	A taxa de ingestão para cada conjunto de rótulos definido pelo usuário Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
QuerySamp lesProcessed	-	`AWS/Prometheus`	Número de amostras de consulta processadas. Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
RuleEvaluations	-	`AWS/Prometheus`	Número total de avaliações de regras Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
RuleEvalu ationFailures	-	`AWS/Prometheus`	Número de falhas na avaliação de regras no intervalo Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma

CloudWatch nome da métrica	Nome do recurso	CloudWatch namespace	Description
RuleGroup IterationsMissed	-	`AWS/Prometheus`	Número de iterações de grupos de regras perdidas no intervalo. Unidades: contagem Estatísticas válidas: média, mínimo, máximo, soma
RuleGroup LastEvaluationDuration	-	`AWS/Prometheus`	Duração da última avaliação de um grupo de regras. Unidades: segundos Estatísticas válidas: média, mínimo, máximo, soma

* As métricas do TPS são geradas a cada minuto e são uma média por segundo durante esse minuto. Períodos curtos de intermitência não serão capturados nas métricas do TPS.

** Alguns dos motivos que fazem com que as amostras sejam descartadas são os seguintes. Nem todos os motivos abaixo aparecem na DiscardedSeries métrica.

Motivo	Significado
greater_than_max_sample_age	Descarte de amostras com mais de uma hora.
new-value-for-timestamp	As amostras duplicadas são enviadas com o mesmo carimbo de data/hora da amostra anterior, mas com valores diferentes.
per_labelset_series_limit	O usuário atingiu o limite total de séries ativas por conjunto de rótulos.
per_metric_series_limit	O usuário atingiu o limite ativo da série por métrica.

Motivo	Significado
per_user_series_limit	O usuário atingiu o limite total de séries ativas.
rate_limited	Taxa de ingestão limitada.
sample-out-of-order	As amostras são enviadas fora de ordem e não podem ser processadas.
label_value_too_long	O valor do rótulo é maior do que o limite permitido de caracteres.
max_label_names_per_series	O usuário atingiu o limite de nomes dos rótulos por métrica.
missing_metric_name	O nome da métrica não foi fornecido.
metric_name_invalid	Nome da métrica inválido fornecido.
label_invalid	Rótulo inválido fornecido.
duplicate_label_names	Nomes de rótulos duplicados fornecidos.

> **Note**
>
> Uma métrica inexistente ou ausente é o mesmo que o valor dessa métrica ser 0.

> **Note**
>
> `RuleGroupIterationsMissed`, `RuleEvaluations`, `RuleEvaluationFailures` e `RuleGroupLastEvaluationDuration` têm a dimensão `RuleGroup` da seguinte estrutura:
> *`RuleGroupNamespace`*;*`RuleGroup`*

Definindo um CloudWatch alarme nas métricas vendidas do Prometheus

Você pode monitorar o uso dos recursos do Prometheus usando alarmes. CloudWatch

Para definir um alarme para o número de ActiveSeries em Prometheus

1. Escolha a guia Métricas representadas graficamente e role para baixo até o ActiveSeriesrótulo.

   Na visualização de Métricas gráficas, somente as métricas que estão sendo ingeridas no momento aparecerão.
2. Escolha o ícone de notificação na coluna Ações.
3. Em Especificar métrica e condições, insira a condição limite no campo Valor das condições e escolha Avançar.
4. Em Configurar ações, selecione um tópico existente do SNS ou crie um novo tópico do SNS para o qual enviar a notificação.
5. Em Adicionar nome e descrição, adicione o nome do alarme e uma descrição opcional.
6. Selecione Criar alarme.

Monitore eventos do Amazon Managed Service para Prometheus com registros CloudWatch

O Amazon Managed Service for Prometheus registra eventos de erro e aviso do Alert Manager e do Ruler em grupos de registros no Amazon Logs. CloudWatch Para obter mais informações sobre o Alert Manager e o Rulers, consulte o tópico Alert Manager neste guia. Você pode publicar os dados de registros do espaço de trabalho em fluxos de registros no CloudWatch Logs. Você pode configurar os logs que deseja monitorar no console do Amazon Managed Service for Prometheus ou usando o AWS CLI. Você pode visualizar ou consultar esses registros no CloudWatch console. Para obter mais informações sobre como visualizar fluxos de CloudWatch registros no console, consulte Como trabalhar com grupos de registros e fluxos de registros CloudWatch no guia do CloudWatch usuário.

O nível CloudWatch gratuito permite que até 5 GB de registros sejam publicados no CloudWatch Logs. Os registros que excederem o limite de nível gratuito serão cobrados com base no plano de CloudWatch preços.

Tópicos

- Configurando registros CloudWatch

Configurando registros CloudWatch

O Amazon Managed Service for Prometheus registra eventos de erro e aviso do Alert Manager e do Ruler em grupos de registros no Amazon Logs. CloudWatch

Você pode definir a configuração de registro de CloudWatch registros no console do Amazon Managed Service for Prometheus ou no `create-logging-configuration` chamando AWS CLI a solicitação de API.

Pré-requisitos

Antes de ligar`create-logging-configuration`, anexe a política a seguir ou permissões equivalentes ao ID ou à função que você usará para configurar CloudWatch os registros.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups",
                "aps:CreateLoggingConfiguration",
                "aps:UpdateLoggingConfiguration",
                "aps:DescribeLoggingConfiguration",
                "aps:DeleteLoggingConfiguration"
            ],
            "Resource": "*"
        }
    ]
}
```

Para configurar CloudWatch registros

Você pode configurar o registro no Amazon Managed Service para Prometheus usando o console ou AWS o. AWS CLI

Console

Para configurar o registro em log no console Amazon Managed Service for Prometheus

1. Navegue até a guia Logs no painel de detalhes do seu espaço de trabalho.
2. Escolha Gerenciar logs no canto superior direito do painel Logs.
3. Escolha tudo na lista suspensa Nível de log.
4. Escolha o grupo de logs no qual você deseja publicar seus logs na lista suspensa Grupo de logs.

   Você também pode criar um novo grupo de registros no CloudWatch console.
5. Escolha Salvar alterações.

AWS CLI

Você pode definir a configuração de registro usando AWS CLI o.

Para configurar o registro usando o AWS CLI

- Usando o AWS CLI, execute o comando a seguir.

```
aws amp create-logging-configuration --workspace-id my_workspace_ID
                                --log-group-arn my-log-group-arn
```

Limitações

- Nem todos os eventos foram registrados em log

  O Amazon Managed Service for Prometheus registra logs de eventos somente quando estão no nível `warning` ou `error`.
- Limite de tamanho da política

  CloudWatch As políticas de recursos de registros estão limitadas a 5120 caracteres. Quando CloudWatch os registros detectam que uma política se aproxima desse limite de tamanho, eles ativam automaticamente grupos de registros que começam com`/aws/vendedlogs/`.

Quando você cria uma regra de alerta com o registro ativado, o Amazon Managed Service for Prometheus deve atualizar CloudWatch sua política de recursos de registros com o grupo de registros que você especificar. Para evitar atingir o limite de tamanho da política de recursos de CloudWatch registros, prefixe os nomes dos grupos de CloudWatch registros de registros com `/aws/vendedlogs/`. Quando você cria um grupo de log no console do Amazon Managed Service for Prometheus, os nomes dos grupos de logs são prefixados com `/aws/vendedlogs/`. Para obter mais informações, consulte Habilitar o registro de determinados AWS serviços no Guia do usuário de CloudWatch registros.

Gerenciar o custo da consulta no Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus oferece a capacidade de limitar o custo da consulta fornecendo limites para a quantidade de amostras de consulta processadas (QSP) que podem ser usadas por uma única consulta. Você pode configurar dois tipos de limites para QSP, aviso e erro, para gerenciar e controlar os custos de consulta de modo eficaz.

Quando as consultas atingem o limite de aviso, uma mensagem de aviso aparece na resposta da consulta da API. Para consultas exibidas no Amazon Managed Grafana, o aviso estará visível na IU do Amazon Managed Grafana, ajudando os usuários a identificar consultas caras. As consultas que atingirem o limite de erro não serão cobradas e serão rejeitadas com um erro.

Além da limitação de consultas, o Amazon Managed Service for Prometheus oferece a capacidade de registrar dados de desempenho de consultas no Logs. CloudWatch Esse recurso permite que você analise as consultas em detalhes para otimizar suas consultas do Amazon Managed Service for Prometheus e gerenciar os custos com mais eficiência. O registro em log de consultas captura informações sobre consultas que excedem os limites especificados de amostras de consulta processadas (QSP). Esses dados são então publicados no CloudWatch Logs, permitindo que você investigue e analise o desempenho da consulta. As consultas registradas incluem consultas de API e consultas de regras. Por padrão, o registro de consultas está desativado para minimizar o uso desnecessário de CloudWatch registros. Você pode ativar esse recurso quando necessário para análise de consultas.

Tópicos

- Configurar o registro em log de consultas
- Configurar limites de controle de utilização de consultas

- Conteúdo do log
- Limitações

Configurar o registro em log de consultas

Você pode configurar o registro de consultas no console do Amazon Managed Service for Prometheus ou na AWS CLI chamando a solicitação de API. `create-query-logging-configuration` Esse corpo da API contém uma lista de destinos, mas, por enquanto, só oferecemos suporte a CloudWatch registros como destino e os destinos devem conter exatamente um elemento com CloudWatch configurações.

Pré-requisitos

Verifique se o `logGroup` já foi criado. O ID ou o perfil usado para configurar deve ter a política ou as permissões equivalentes a seguir.

JSON

```
{
"Version":"2012-10-17",
    "Statement": [
        {
"Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups",
                "aps:CreateQueryLoggingConfiguration",
                "aps:UpdateQueryLoggingConfiguration",
                "aps:DescribeQueryLoggingConfiguration",
                "aps:DeleteQueryLoggingConfiguration"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Configurar CloudWatch registros

Você pode configurar CloudWatch os registros fazendo login no Amazon Managed Service for Prometheus usando o ou o. Console de gerenciamento da AWS AWS CLI

Como configurar o registro em log de consultas usando o console do Amazon Managed Service for Prometheus

1. Navegue até a guia Logs no painel de detalhes do seu espaço de trabalho.
2. Em Insights de consultas, escolha Criar.
3. Selecione o menu suspenso Grupo de logs e escolha o grupo de logs para publicar seus logs.

   Você também pode criar um novo grupo de registros no CloudWatch console.
4. Insira o Limite (QSP).
5. Escolha Salvar.

Para configurar o registro de consultas AWS CLI usando o comando

```
aws amp create-query-logging-configuration \
--workspace-id my_workspace_ID \
--destinations '[{"cloudWatchLogs":{"logGroupArn":"$my-log-group-arn"},"filters":
{"qspThreshold":$qspThreshold}}]'
```

Para obter informações sobre como atualizar, excluir e descrever operações, consulte Referência da API do Amazon Managed Service for Prometheus.

Configurar limites de controle de utilização de consultas

Para configurar os limites de QSP, você deve fornecer os parâmetros de consulta na QueryMetrics API.

- max_samples_processed_warning_threshold: define o limite de avisos para amostras de consulta processadas
- max_samples_processed_error_threshold: define o limite de erros para as amostras de consulta processadas.

Usuários do Amazon Managed Grafana podem usar a configuração da fonte de dados grafana para aplicar limites a todas as consultas da fonte de dados:

1. Navegue até a configuração da fonte de dados do Amazon Managed Service for Prometheus no Amazon Managed Grafana.
2. Em Parâmetros de consulta personalizados, adicione os cabeçalhos de limite.
3. Escolha Salvar.

Conteúdo do log

Para consultas originadas de regras, você verá as seguintes informações sobre a consulta nos CloudWatch registros:

```
{
    workspaceId: "workspace_id",
    message: {
        query: "avg(rate(go_goroutines[1m])) > 1",
        name: "alert_rule",
        kind: "alerting",
        group: "test-alert",
        namespace: "test",
        samples: "59321",
    },
    component: "ruler"
}
```

Para consultas originadas de chamadas de API, você verá as seguintes informações sobre a consulta nos CloudWatch registros:

```
{
    workspaceId: "ws-5e7658c2-7ccf-4c30-9de9-2ab26fa30639",
    message: {
        query: "sum by (instance) (go_memstats_alloc_bytes{job=\"node\"})",
        queryType: "range",
        start: "1683308700000",
        end: "1683913500000",
        step: "300000",
        samples: "11496",
        userAgent: "AWSPrometheusDPJavaClient/2.0.436.0 ",
        dashboardUid: "11234",
```

```
        panelId: "12"
    },
    component: "query-frontend"
}
```

Limitações

Limites de tamanho da política — as políticas de recursos de CloudWatch registros estão limitadas a 5120 caracteres. Quando o CloudWatch Logs detecta que a política está se aproximando do limite de tamanho, ele ativa automaticamente grupos de registros que começam com. `/aws/vendedlogs/` Quando você ativa o registro de consultas, o Amazon Managed Service for Prometheus deve atualizar CloudWatch sua política de recursos de registros com o grupo de registros que você especificar. Para evitar atingir o limite de tamanho da política de recursos de CloudWatch registros, prefixe os nomes dos grupos de CloudWatch registros de registros com`/aws/vendedlogs/`.

Entender e otimizar custos no Amazon Managed Service for Prometheus

As seguintes perguntas frequentes e suas respostas podem ser úteis para entender e otimizar os custos associados ao Amazon Managed Service for Prometheus.

O que contribui para meus custos?

Para a maioria dos clientes, a ingestão de métricas contribui para a maioria dos custos. Clientes com alto uso de consultas também verão alguns custos com base nas amostras de consultas processadas, com o armazenamento de métricas sendo um pequeno fator dos custos gerais. Para obter mais informações sobre os preços de cada um destes, consulte Preços na página do produto Amazon Managed Service for Prometheus.

Qual é a melhor maneira de reduzir meus custos? Como faço para reduzir os custos de ingestão?

Para a maioria dos clientes, as taxas de ingestão (não o armazenamento das métricas) constituem a maior parte dos custos. Você pode reduzir as taxas de ingestão reduzindo a frequência de coleta (aumentando o intervalo de coleta) ou reduzindo o número de séries ativas ingeridas.

Você pode aumentar o intervalo de coleta (raspagem) do seu agente de coleta: tanto o servidor Prometheus (executado no modo Agente) quanto o coletor AWS Distro for OpenTelemetry (ADOT) suportam a configuração. `scrape_interval` Por exemplo, aumentar o intervalo de coleta de 30 segundos para 60 segundos reduzirá seu uso de ingestão para a metade.

Você também pode filtrar as métricas enviadas ao Amazon Managed Service for Prometheus usando o `<relabel_config>`. Para obter mais informações sobre a renomeação na configuração do agente Prometheus, consulte https://prometheus. io/docs/prometheus/latest/configuration/ configuration/#relabel_config na documentação do Prometheus.

Qual é a melhor maneira de reduzir meus custos de consulta?

Os custos de consulta são baseados no número de amostras processadas. Você pode reduzir a frequência das consultas para reduzir seus custos de consulta.

Para obter mais visibilidade das consultas que estão contribuindo mais para seus custos de consulta, consulteGerenciar o custo da consulta no Amazon Managed Service for Prometheus.

Se eu diminuir o período de retenção das minhas métricas, isso ajudará a reduzir o total da minha fatura?

Você pode reduzir seu período de retenção, mas é improvável que isso reduza substancialmente seus custos.

Para obter informações sobre como configurar o período de retenção de um espaço de trabalho, consulte Configurar seu espaço de trabalho.

Como posso manter meus custos de consulta de alerta baixos?

Os alertas criam consultas com base em seus dados, o que aumenta seus custos de consulta. Estas são algumas estratégias que você pode usar para otimizar suas consultas de alerta e manter seus custos mais baixos.

- Use alertas do Amazon Managed Service for Prometheus: os sistemas de alerta fora do Amazon Managed Service for Prometheus podem exigir consultas adicionais para adicionar resiliência ou alta disponibilidade, já que o serviço externo consulta as métricas de várias zonas ou regiões de disponibilidade. Isso inclui alertas no Grafana para alta disponibilidade. Isso pode multiplicar seus custos por três vezes ou mais. Os alertas no Amazon Managed Service for Prometheus são otimizados e fornecem alta disponibilidade e resiliência com o menor número de consultas.

  Recomendamos usar os alertas nativos no Amazon Managed Service for Prometheus no lugar de sistemas de alerta externos.
- Otimize seu intervalo de alertas: uma maneira rápida de otimizar suas consultas de alerta é aumentar o intervalo de atualização automática. Se você tem um alerta que consulta a cada minuto, mas só é necessário a cada cinco minutos, aumentar o intervalo de atualização automática pode economizar cinco vezes os custos de consulta desse alerta.
- Use uma retrospectiva ideal: uma janela de retrospectiva maior em sua consulta aumenta os custos da consulta, já que ela extrai mais dados. Certifique-se de que a janela de retrospectiva em sua consulta PromQL tenha um tamanho razoável para os dados que precisam de alerta. Por exemplo, na regra a seguir, a expressão inclui uma janela de retrospectiva de dez minutos:

```
    - alert: metric:alerting_rule
```

```
      expr: avg(rate(container_cpu_usage_seconds_total[10m])) > 0
      for: 2m
```

Alterar o `expr` para `avg(rate(container_cpu_usage_seconds_total[5m])) > 0` pode ajudar a reduzir seus custos de consulta.

Em geral, analise suas regras de alertas e verifique se está usando alertas com base nas melhores métricas do seu serviço. É fácil criar alertas sobrepostos nas mesmas métricas ou em vários alertas que fornecem as mesmas informações, especialmente quando você adiciona alertas ao longo do tempo. Se você achar frequente a visualização de grupos de alertas ocorrendo ao mesmo tempo, será possível otimizar seus alertas e não incluir todos eles.

Essas sugestões podem ajudar você a reduzir custos. Em última análise, você deve equilibrar os custos com a criação do conjunto certo de alertas para entender o estado do seu sistema.

Para obter mais informações sobre alertas no Amazon Managed Service for Prometheus, consulte Como gerenciar e encaminhar alertas no Amazon Managed Service for Prometheus com o gerenciador de alertas.

Posso verificar minha fatura a qualquer momento?

O AWS Cost and Usage Report rastreia seu AWS uso e fornece cobranças estimadas associadas à sua conta dentro de um período de cobrança. Para obter mais informações, consulte O que são relatórios de AWS custo e uso? no Guia do usuário de relatórios de AWS custo e uso

Quais métricas posso usar para monitorar meus custos?

As amostras métricas que você ingere são o principal fator de custo do Amazon Managed Service for Prometheus. O número de amostras ingeridas determina diretamente suas cobranças mensais, tornando essencial monitorar e entender seus padrões de ingestão.

AWS Cost Exploreré a fonte confiável para monitorar o Amazon Managed Service quanto aos custos do Prometheus. Você pode monitorar o histórico e day-by-day as tendências de custo do Cost Explorer no Amazon Managed Service for Prometheus em várias dimensões, incluindo amostras ingeridas.AWS A detecção de anomalias de custos também pode fornecer a capacidade de monitorar mudanças inesperadas em seus padrões de gastos.

O uso de `IngestionRate` métricas fornece um método auxiliar para monitorar tendências na ingestão que estão diretamente correlacionadas ao custo. As vantagens de usar `IngestionRate` como métrica adicional incluem:

- Rastreamento no nível do espaço de trabalho — monitore a ingestão por espaço de trabalho e não apenas no nível da conta.
- Visibilidade granular — acompanhe os padrões de ingestão de hora em hora ou até mesmo para obter minute-by-minute insights em tempo real.
- Monitoramento proativo — defina CloudWatch alarmes para detectar picos de uso antes que eles apareçam no faturamento.

> **Note**
>
> `IngestionRate`pode ser usado para estimar custos e tendências ou atribuir o custo por espaço de trabalho, mas não é 100% preciso. Como `IngestionRate` relata uma taxa média de amostragem em intervalos de 1 minuto, multiplicar essa taxa pelo tempo fornece uma aproximação em vez de uma contagem exata das amostras ingeridas. Além disso, a política CloudWatch de retenção de dados da Amazon afeta a granularidade disponível para consultas históricas, com dados com mais de 63 dias limitados a intervalos de 1 hora.

Para obter mais informações sobre o monitoramento das métricas CloudWatch do Amazon Managed Service for Prometheus em, consulte. Use CloudWatch métricas para monitorar os recursos do Amazon Managed Service for Prometheus

Como faço para ver meus custos em AWS Cost Explorer?

Como fonte fidedigna dos custos do Amazon Managed Service for Prometheus AWS Cost Explorer , fornece o uso real faturado e as cobranças do Amazon Managed Service para amostras do Prometheus ingeridas, incluindo dados históricos de faturamento por mês e região. Use o Cost Explorer para seus valores finais faturados e tendências de day-by-day custo.

Para ver seus custos do Amazon Managed Service para Prometheus:

Acesso AWS Cost Explorer

1. Faça login no AWS Management Console.

2. Navegue até o painel Billing and Cost Management.
3. Selecione Cost Explorer no menu de navegação à esquerda.
4. Escolha Launch Cost Explorer (se for a primeira vez que você o usa).

Configurar o relatório

1. Defina seu intervalo de tempo para o período de cobrança desejado (por exemplo, março de 2025 a fevereiro de 2026).
2. Em Filtros, selecione:
   - Serviço: escolha “Amazon Managed Service for Prometheus”.
   - Tipo de uso: Filtro para "MetricSampleCount" para isolar as cargas de ingestão de amostras.

Agrupe e visualize dados

1. Em Agrupar por, selecione Região para ver os dados de custo e uso por região.
2. Escolha sua visualização preferida (gráfico de barras, gráfico de linhas ou tabela).
3. Escolha Aplicar para gerar o relatório.

Exportar dados (opcional)

1. Escolha Baixar CSV no canto superior direito para exportar os dados.
2. O arquivo CSV conterá: período de cobrança, região, tipo de uso, valor faturado e quantidade de uso (número de amostras faturadas).

> **Note**
>
> Os dados do Cost Explorer normalmente têm um atraso de 24 horas. Para o período de cobrança mais atual, os dados podem não estar disponíveis até o dia seguinte.

Como faço para calcular o número de amostras ingeridas em um mês?

Você pode calcular o número aproximado de amostras ingeridas usando as `IngestionRate` métricas CloudWatch da Amazon com o. AWS Command Line Interface Isso é útil para revisar as faturas mensais e entender os padrões de uso nos espaços de trabalho.

Para recuperar dados de ingestão:

```
aws cloudwatch get-metric-data \
    --region your-region \
    --start-time start-timestamp \
    --end-time end-timestamp \
    --metric-data-queries '[
        {
            "Id": "e1",
            "Expression": "SUM(METRICS())",
            "Period": 3600
        },
        {
            "Id": "ws1",
            "MetricStat": {
                "Metric": {
                    "Namespace": "AWS/Usage",
                    "MetricName": "ResourceCount",
                    "Dimensions": [
                        {"Name": "Service",    "Value": "Prometheus"},
                        {"Name": "Resource",   "Value": "IngestionRate"},
                        {"Name": "Type",       "Value": "Resource"},
                        {"Name": "Class",      "Value": "None"},
                        {"Name": "ResourceId", "Value": "YOUR_AMP_WORKSPACE_ID"}
                    ]
                },
                "Period": 3600,
                "Stat": "Average"
            }
        }
    ]'
```

O comando retorna `IngestionRate` valores médios horários, medidos em amostras por segundo. Para calcular o número aproximado de amostras ingeridas em um mês, multiplique cada ponto de

dados por hora por 3600 (segundos por hora) para obter as amostras ingeridas naquela hora e, em seguida, some todos os totais horários do mês:

```
Monthly samples ≈ Σ (hourly IngestionRate average × 3600)
```

Por exemplo, se uma única hora retornar uma média `IngestionRate` de 500 amostras por segundo, essa hora contribuiu com aproximadamente 500 × 3600 = 1.800.000 amostras. Repita isso para cada hora do mês e some os resultados para obter sua contagem aproximada de ingestão mensal.

Principais parâmetros:

- `Period`: 3600 (1 hora em segundos)
- `StartTime`: Início do mês (por exemplo,`2026-02-01T00:00:00Z`)
- `EndTime`: Fim do mês (por exemplo,`2026-03-01T00:00:00Z`)
- `Stat`: Média

Para encontrar seu espaço de trabalho IDs:

```
aws amp list-workspaces --region your-region
```

Use o ID do espaço de trabalho para filtrar as métricas e mostrar dados somente para o espaço de trabalho especificado, em vez de agregá-los a todos os recursos do Prometheus na região.

Qual granularidade de dados está disponível para análise histórica de custos?

A política CloudWatch de retenção de dados da Amazon afeta a granularidade disponível para consultas históricas:

- Dados com menos de 15 dias: consulta em intervalos de 1 minuto (`Period`: 60)
- Dados de 15 a 63 dias: consulta em intervalos de 5 minutos (`Period`: 300)
- Dados com mais de 63 dias: limitados a intervalos de 1 hora (`Period`: 3600)

Para análises históricas além de 63 dias, reduza CloudWatch automaticamente a amostra dos dados para um período mínimo de 1 hora. Ao revisar o faturamento de meses anteriores a 63 dias, você

deve usar dados agregados por hora. O cálculo da amostra mensal usa esses pontos de dados médios por hora, somando cada valor multiplicado por 3600 em todo o mês.

Essa granularidade reduzida contribui ainda mais para o motivo pelo qual `IngestionRate` fornece estimativas em vez de contagens exatas para dados mais antigos. Sempre consulte o Cost Explorer para ver seus valores faturados autorizados.

Para obter mais detalhes sobre retenção de CloudWatch métricas, consulte Retenção de métricas no Guia CloudWatch do usuário da Amazon.

Quais são as melhores práticas para monitorar os custos do Amazon Managed Service for Prometheus?

Para gerenciar e otimizar com eficácia seu Amazon Managed Service para gastos com o Prometheus, considere implementar as seguintes práticas de monitoramento:

- Monitore o Cost Explorer regularmente para acompanhar as tendências reais de gastos e identificar anomalias de custo em várias dimensões, incluindo amostras ingeridas.
- Ative a detecção de anomalias de AWS custo para receber alertas sobre aumentos inesperados de custos em seus gastos com o Amazon Managed Service for Prometheus.
- Configure CloudWatch alarmes `IngestionRate` para monitoramento em nível de espaço de trabalho e detecção precoce de picos de ingestão.
- Exporte dados do Cost Explorer regularmente para análises e relatórios de custos de longo prazo.

Por que minha fatura é maior no início do mês do que no final do mês?

O Amazon Managed Service for Prometheus tem um modelo de preços em camadas para a ingestão, o que resulta em custos mais altos em seu uso inicial. À medida que seu uso atinge camadas mais altas de ingestão, com custos mais baixos, seus custos são menores. Para obter mais informações sobre os preços, incluindo camadas de ingestão, consulte Preços na página do produto Amazon Managed Service for Prometheus.

> **Note**
>
> - Os níveis são para uso dentro de uma região, não entre regiões. O uso dentro de uma região deve atingir o próximo nível para que use a taxa mais baixa.
> - Em uma organização em AWS Organizations, o uso do nível é contabilizado por conta do pagador, não por conta (a conta do pagador é sempre a conta de gerenciamento da organização). Quando o total de métricas ingeridas (dentro de uma região) para todas as contas em uma organização atinge o próximo nível, todas as contas são cobradas com a taxa mais baixa.

Excluí todos os meus espaços de trabalho do Amazon Managed Service for Prometheus, mas parece que ainda há cobranças. O que pode estar acontecendo?

Uma possibilidade nesse caso é que você ainda tenha raspadores AWS gerenciados configurados para enviar métricas aos seus espaços de trabalho excluídos. Siga as instruções em Encontrar e excluir extratores.

Integração a outros serviços da AWS

O Amazon Managed Service for Prometheus se integra a outros serviços AWS. Esta seção descreve a integração com o Amazon Elastic Kubernetes Service (Amazon EKS), o monitoramento de custos (com o Kubecost) e como ingerir métricas do CloudWatch usando o Amazon Data Firehose. Também descreve como configurar e gerenciar o Amazon Managed Service para Prometheus com módulos do Terraform do AWS Observability Accelerator ou usando controladores da AWS para Kubernetes.

Tópicos

Integração com o monitoramento de custos do Amazon EKS

O Amazon Managed Service for Prometheus se integra ao monitoramento de custos do Amazon Elastic Kubernetes Service (Amazon EKS) (com o Kubecost) para realizar cálculos de alocação de custos e fornecer informações sobre como otimizar seus clusters do Kubernetes. Usando o Amazon Managed Service for Prometheus com Kubecost, você pode escalar de forma confiável seu monitoramento de custos para suportar clusters maiores.

A integração com o Kubecost oferece visibilidade granular dos custos do seu cluster do Amazon EKS. Você pode agregar custos pela maioria dos contextos do Kubernetes, desde o nível do contêiner até o nível do cluster e até mesmo no nível de vários clusters. Você pode gerar relatórios em contêineres ou clusters para rastrear custos para fins de devolução ou estorno.

A seguir, são apresentadas instruções para integração com o Kubecost em um cenário de um ou vários clusters:

- Integração com um único cluster: para saber como integrar o monitoramento de custos do Amazon EKS a um único cluster, consulte a postagem no blog da AWS, Integrando o Kubecost com o Amazon Managed Service for Prometheus.
- Integração com vários clusters: para saber como integrar o monitoramento de custos do Amazon EKS a vários clusters, consulte a postagem no blog da AWS, Monitoramento de custos de vários clusters para o Amazon EKS usando o Kubecost e o Amazon Managed Service for Prometheus.

> **Note**
>
> Para obter mais informações sobre o uso do Kubecost, consulte Monitoramento de custos no Guia do usuário do Amazon EKS.

Configure o Amazon Managed Service para AWS Prometheus com o Observability Accelerator

AWS fornece ferramentas de observabilidade, incluindo monitoramento, registro, alertas e painéis, para seus projetos do Amazon Elastic Kubernetes Service (Amazon EKS). Isso inclui o Amazon Managed Service para Prometheus, o Amazon Managed Grafana, o AWS OpenTelemetryDistro for e outras ferramentas. Para ajudá-lo a usar essas ferramentas em conjunto, AWS fornece módulos do Terraform que configuram a observabilidade com esses serviços, chamados de AWS Observability Accelerator.

AWS O Observability Accelerator fornece dois perfis de coletor para o Amazon Managed Service for Prometheus:

- Métricas gerenciadas (sem agente) — usa o coletor Amazon Managed Service for Prometheus, um raspador totalmente gerenciado e sem agentes que é executado fora do seu cluster. Não há cápsulas de coletor para gerenciar. Somente métricas.
- Autogerenciado — implanta um OpenTelemetry coletor via Helm em seu cluster. Suporta métricas, rastreamentos (AWS X-Ray) e registros (Amazon CloudWatch).

Esta seção aborda as duas opções, começando com a abordagem sem agente recomendada.

Os modelos e instruções detalhadas do Terraform podem ser encontrados na página do AWS Observability Accelerator for Terraform. GitHub

Pré-requisitos

Para usar o AWS Observability Accelerator, você deve ter um cluster Amazon EKS existente e os seguintes pré-requisitos:

- AWS CLI— usado para chamar a AWS funcionalidade a partir da linha de comando.
- kubectl: usado para controlar seu cluster do EKS a partir da linha de comando.

- Terraform (>= 1.5.0) — usado para automatizar a criação dos recursos para essa solução. Você deve configurar o AWS provedor com uma função do IAM que tenha acesso para criar e gerenciar o Amazon Managed Service para Prometheus, Amazon Managed Grafana e IAM em sua conta. AWS Para obter mais informações sobre como configurar o provedor da AWS para o Terraform, consulte o provedor da AWS na documentação do Terraform.

Usando o exemplo de métricas gerenciadas (sem agente)

Este exemplo usa o coletor Amazon Managed Service for Prometheus para extrair métricas do Prometheus do seu cluster Amazon EKS sem implantar nenhum pod de coletor. O coletor exige pelo menos duas sub-redes em duas zonas de disponibilidade distintas. Para obter mais detalhes, consulte o eks-amp-managedexemplo em GitHub.

Para usar o módulo Terraform de monitoramento de infraestrutura sem agente

1. Na pasta em que você deseja criar seu projeto, clone o repositório usando o comando a seguir.

```
git clone https://github.com/aws-observability/terraform-aws-observability-
accelerator.git
```

2. Inicialize o Terraform com os comandos a seguir.

```
cd examples/eks-amp-managed

terraform init
```

3. Crie um arquivo `terraform.tfvars`, como no exemplo a seguir. Use a AWS região, o ID do cluster e os detalhes da rede VPC para seu cluster Amazon EKS. O coletor exige pelo menos duas sub-redes em duas zonas de disponibilidade distintas.

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"

# (mandatory) Subnets for the managed scraper (>= 2 AZs)
scraper_subnet_ids = ["subnet-aaa", "subnet-bbb"]

# (mandatory) Security group allowing scraper access to the EKS API
```

```
scraper_security_group_ids = ["sg-xxx"]
```

4. Crie um espaço de trabalho do Amazon Managed Grafana, se você ainda não tiver um que queira usar. Para obter informações sobre como criar um espaço de trabalho, consulte Crie seu primeiro espaço de trabalho no Guia do usuário do Amazon Managed Grafana.
5. Crie duas variáveis para que o Terraform use seu espaço de trabalho do Grafana executando os seguintes comandos na linha de comando. Você precisará substituí-lo pelo ID do *grafana-workspace-id* seu espaço de trabalho da Grafana.

```
export TF_VAR_managed_grafana_workspace_id=grafana-workspace-id
export TF_VAR_grafana_api_key=`aws grafana create-workspace-api-key --key-name
 "observability-accelerator-$(date +%s)" --key-role ADMIN --seconds-to-live 1200 --
workspace-id $TF_VAR_managed_grafana_workspace_id --query key --output text`
```

6. [Opcional] Para usar um espaço de trabalho existente do Amazon Managed Service para Prometheus, adicione o ID ao arquivo, como no exemplo `terraform.tfvars` a seguir, substituindo-o pelo ID do espaço de trabalho do *prometheus-workspace-id* Prometheus. Se você não especificar um espaço de trabalho existente, um espaço de trabalho do Prometheus será criado para você.

```
# (optional) Leave it empty for a new workspace to be created
managed_prometheus_workspace_id = "prometheus-workspace-id"
```

7. Implante a solução com o seguinte comando.

```
terraform apply -var-file=terraform.tfvars
```

Isso criará recursos em sua AWS conta, incluindo o seguinte:

- Um novo espaço de trabalho do Amazon Managed Service for Prometheus (a menos que você tenha optado por usar um espaço de trabalho existente).
- Um coletor do Amazon Managed Service for Prometheus (raspador sem agente) configurado para extrair métricas do Prometheus do seu cluster Amazon EKS.
- Regras de gravação e alerta do Prometheus em seu espaço de trabalho do Amazon Managed Service for Prometheus.
- kube-state-metrics e node-exporter implantado em seu cluster Amazon EKS para métricas de infraestrutura.

- Nova fonte de dados e painéis do Amazon Managed Grafana em seu espaço de trabalho atual. Os painéis serão listados em Monitoramento EKS.

Alternativa: coletor autogerenciado OpenTelemetry

Se você precisar de rastreamentos, registros ou controle total sobre o pipeline de coleta, use o perfil autogerenciado. Isso implanta um OpenTelemetry Collector via Helm em seu cluster Amazon EKS, configurado para coletar métricas do Prometheus e gravar remotamente no Amazon Managed Service for Prometheus. Ele também suporta traços (AWS X-Ray) e registros (Amazon CloudWatch). Para obter mais detalhes, consulte o eks-amp-otelexemplo em GitHub.

Para usar o módulo autogerenciado do Terraform

1. Clone o repositório e inicialize o Terraform.

```
git clone https://github.com/aws-observability/terraform-aws-observability-
accelerator.git
cd examples/eks-amp-otel
terraform init
```

2. Crie um arquivo `terraform.tfvars`, como no exemplo a seguir.

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"
```

3. Configure seu espaço de trabalho e chave de API do Amazon Managed Grafana usando as mesmas etapas do exemplo de métricas gerenciadas (etapas 4 a 6 acima).
4. Implante a solução com o seguinte comando.

```
terraform apply -var-file=terraform.tfvars
```

Isso criará os seguintes recursos em sua AWS conta (diferentemente da abordagem sem agente, o coletor é executado dentro do seu cluster):

- Um espaço de trabalho do Amazon Managed Service para Prometheus (se não for fornecido).

- Um espaço de trabalho Amazon Managed Grafana com fonte de dados e painéis.
- Um OpenTelemetry coletor implantado via Helm em seu cluster Amazon EKS, configurado para coletar métricas do Prometheus e gravar remotamente no Amazon Managed Service for Prometheus.
- Uma função do IAM para contas de serviço (IRSA) para o OpenTelemetry Collector.
- Rastreia o pipeline até o AWS X-Ray (ativado por padrão).
- Registra o pipeline na Amazon CloudWatch (ativado por padrão).

Visualizando painéis

Para visualizar seus novos painéis, abra o painel específico em seu espaço de trabalho do Amazon Managed Grafana. Os painéis de infraestrutura são provisionados automaticamente pelo Terraform. Para obter mais informações sobre o uso do Amazon Managed Grafana, consulte Trabalhar em seu espaço de trabalho do Grafana, no Guia do usuário do Amazon Managed Grafana.

Gerencie o Amazon Managed Service para Prometheus AWS com controladores para Kubernetes

O Amazon Managed Service for Prometheus é integrado aos AWS Controllers for Kubernetes (ACK), com suporte para gerenciar seus recursos de espaço de trabalho, Alert Manager e Ruler no Amazon EKS. Você pode usar AWS controladores para definições de recursos personalizadas (CRDs) do Kubernetes e objetos nativos do Kubernetes sem precisar definir nenhum recurso fora do seu cluster.

Esta seção descreve como configurar AWS controladores para Kubernetes e Amazon Managed Service para Prometheus em um cluster Amazon EKS existente.

Você também pode ler as postagens do blog que apresentam AWS os controladores para Kubernetes e o controlador ACK para o Amazon Managed Service for Prometheus.

Pré-requisitos

Antes de começar a integrar AWS os Controllers for Kubernetes e o Amazon Managed Service for Prometheus com seu cluster Amazon EKS, você deve ter os seguintes pré-requisitos.

- Você deve ter uma conta existente Conta da AWS e ter permissões para criar programaticamente as funções do Amazon Managed Service para Prometheus e IAM.
- Você deve ter um cluster do Amazon EKS existente com o OpenID Connect (OIDC) habilitado.

Se o OIDC não estiver habilitado, você pode usar o comando a seguir para habilitá-lo. Lembre-se de substituir o *YOUR_CLUSTER_NAME* e *AWS_REGION* pelos valores corretos para sua conta.

```
eksctl utils associate-iam-oidc-provider \
    --cluster ${YOUR_CLUSTER_NAME} --region ${AWS_REGION} \
    --approve
```

Para obter mais informações sobre o uso do OIDC com o Amazon EKS, consulte Autenticação do provedor de identidade do OIDC e Criação de um provedor OIDC do IAM no Guia do usuário do Amazon EKS.

- Você deve ter o driver da CSI do Amazon EBS instalado no seu cluster do Amazon EKS.
- É necessário ter a AWS CLI instalada. O AWS CLI é usado para chamar a AWS funcionalidade a partir da linha de comando.
- O Helm, o gerenciador de pacotes do Kubernetes, deve estar instalado.
- As métricas do ambiente de gerenciamento com o Prometheus devem ser configuradas em seu cluster do Amazon EKS.
- Você deve ter um tópico do Amazon Simple Notification Service (Amazon SNS) para o qual você deseja enviar alertas do seu novo espaço de trabalho. Verifique se você concedeu permissão ao Amazon Managed Service for Prometheus para enviar mensagens para o tópico.

Quando seu cluster do Amazon EKS estiver configurado adequadamente, você poderá ver as métricas formatadas para o Prometheus chamando `kubectl get --raw /metrics`. Agora você está pronto para instalar um controlador de serviço AWS Controllers for Kubernetes e usá-lo para implantar recursos do Amazon Managed Service for Prometheus.

Implantação de um espaço de trabalho com AWS controladores para Kubernetes

Para implantar um novo espaço de trabalho do Amazon Managed Service para Prometheus, você instalará AWS um controlador Controllers for Kubernetes e o usará para criar o espaço de trabalho.

Para implantar um novo espaço de trabalho AWS do Amazon Managed Service para Prometheus com controladores para Kubernetes

1. Use os comandos a seguir para usar o Helm para instalar o controlador de serviço do Amazon Managed Service for Prometheus. Para obter mais informações, consulte Instalar um controlador

ACK na documentação de AWS Controllers for Kubernetes em. GitHub Use o correto *region* para o seu sistema, como`us-east-1`.

```
export SERVICE=prometheusservice
export RELEASE_VERSION=`curl -sL https://api.github.com/repos/aws-controllers-k8s/
$SERVICE-controller/releases/latest | jq -r '.tag_name | ltrimstr("v")'`
export ACK_SYSTEM_NAMESPACE=ack-system
export AWS_REGION=region

aws ecr-public get-login-password --region us-east-1 | helm registry login --
username AWS --password-stdin public.ecr.aws
helm install --create-namespace -n $ACK_SYSTEM_NAMESPACE ack-$SERVICE-controller \
  oci://public.ecr.aws/aws-controllers-k8s/$SERVICE-chart --version=
$RELEASE_VERSION --set=aws.region=$AWS_REGION
```

Após alguns instantes, você verá um resultado semelhante ao seguinte, indicando êxito.

```
You are now able to create Amazon Managed Service for Prometheus (AMP) resources!
The controller is running in "cluster" mode.
The controller is configured to manage AWS resources in region: "us-east-1"
```

Opcionalmente, você pode verificar se o AWS controlador Controllers for Kubernetes foi instalado com êxito com o comando a seguir.

```
helm list --namespace $ACK_SYSTEM_NAMESPACE -o yaml
```

Isso retornará informações sobre o controlador `ack-prometheusservice-controller`, incluindo o `status: deployed`.

2. Crie um arquivo denominado `workspace.yaml` com o seguinte texto. Ele será usado como configuração para o espaço de trabalho que você está criando.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: Workspace
metadata:
  name: my-amp-workspace
spec:
  alias: my-amp-workspace
  tags:
    ClusterName: EKS-demo
```

3. Execute o comando a seguir para criar seu espaço de trabalho (esse comando depende das variáveis do sistema que você configurou na etapa 1).

```
kubectl apply -f workspace.yaml -n $ACK_SYSTEM_NAMESPACE
```

Em alguns instantes, você poderá ver um novo espaço de trabalho, chamado `my-amp-workspace` em sua conta.

Executando o comando a seguir para visualizar os detalhes e o status do seu espaço de trabalho, incluindo o ID do espaço de trabalho. Como alternativa, você pode visualizar o novo espaço de trabalho no console do Amazon Managed Service for Prometheus.

```
kubectl describe workspace my-amp-workspace -n $ACK_SYSTEM_NAMESPACE
```

> **Note**
>
> Você também pode usar um espaço de trabalho existente em vez de criar um novo.

4. Crie dois novos arquivos yaml como configuração para os grupos de regras e AlertManager que você criará em seguida usando a configuração a seguir.

Salve essa configuração como `rulegroup.yaml`. *`WORKSPACE-ID`*Substitua pela ID do espaço de trabalho da etapa anterior.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: RuleGroupsNamespace
metadata:
  name: default-rule
spec:
  workspaceID: WORKSPACE-ID
  name: default-rule
  configuration: |
    groups:
    - name: example
      rules:
      - alert: HostHighCpuLoad
        expr: 100 - (avg(rate(node_cpu_seconds_total{mode="idle"}[2m])) * 100) > 60
        for: 5m
        labels:
          severity: warning
```

```
          event_type: scale_up
        annotations:
          summary: Host high CPU load (instance {{ $labels.instance }})
          description: "CPU load is > 60%\n  VALUE = {{ $value }}\n  LABELS =
 {{ $labels }}"
      - alert: HostLowCpuLoad
        expr: 100 - (avg(rate(node_cpu_seconds_total{mode="idle"}[2m])) * 100) < 30
        for: 5m
        labels:
          severity: warning
          event_type: scale_down
        annotations:
          summary: Host low CPU load (instance {{ $labels.instance }})
          description: "CPU load is < 30%\n  VALUE = {{ $value }}\n  LABELS =
 {{ $labels }}"
```

Salve a configuração a seguir como `alertmanager.yaml`. *WORKSPACE-ID*Substitua pela ID do espaço de trabalho da etapa anterior. *TOPIC-ARN*Substitua pelo ARN do tópico do Amazon SNS para o qual enviar notificações *REGION* e pelo que você está usando Região da AWS . Lembre-se de que o Amazon Managed Service for Prometheus deve ter permissões para o tópico do Amazon SNS.

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: AlertManagerDefinition
metadata:
  name: alert-manager
spec:
  workspaceID: WORKSPACE-ID
  configuration: |
    alertmanager_config: |
      route:
         receiver: default_receiver
      receivers:
        - name: default_receiver
          sns_configs:
          - topic_arn: TOPIC-ARN
            sigv4:
              region: REGION
            message: |
              alert_type: {{ .CommonLabels.alertname }}
              event_type: {{ .CommonLabels.event_type }}
```

> **Note**
>
> Para saber mais sobre os formatos desses arquivos de configuração, consulte RuleGroupsNamespaceData e AlertManagerDefinitionData.

5. Execute os comandos a seguir para criar seu grupo de regras e a configuração do gerenciador de alertas (esse comando depende das variáveis do sistema que você configurou na etapa 1).

```
kubectl apply -f rulegroup.yaml -n $ACK_SYSTEM_NAMESPACE
kubectl apply -f alertmanager.yaml -n $ACK_SYSTEM_NAMESPACE
```

As mudanças estarão disponíveis em instantes.

> **Note**
>
> Para atualizar um recurso, em vez de criá-lo, basta atualizar o arquivo yaml e executar o comando `kubectl apply` novamente.
> Para excluir um recurso, execute o comando a seguir. *`ResourceType`*Substitua pelo tipo de recurso que você deseja excluir `WorkspaceAlertManagerDefinition`, ou`RuleGroupNamespace`. *`ResourceName`*Substitua pelo nome do recurso a ser excluído.
>
> ```
> kubectl delete ResourceType ResourceName -n $ACK_SYSTEM_NAMESPACE
> ```

Isso conclui a implantação do novo espaço de trabalho. A próxima seção descreve como configurar seu cluster para enviar métricas para esse espaço de trabalho.

Configuração do cluster do Amazon EKS para gravar no espaço de trabalho do Amazon Managed Service for Prometheus

Esta seção descreve como usar o Helm para configurar o Prometheus em execução no seu cluster do Amazon EKS para gravar de forma remota métricas no espaço de trabalho do Amazon Managed Service for Prometheus que você criou na seção anterior.

Para esse procedimento, você precisará do nome do perfil do IAM que você criou para usar na ingestão de métricas. Se ainda não o tiver feito isso, consulte Configurar perfis de serviço para a

ingestão de métricas de clusters do Amazon EKS para obter mais informações e instruções. Se você seguir essas instruções, o perfil do IAM será chamado `amp-iamproxy-ingest-role`.

Para configurar o cluster do Amazon EKS para gravação remota

1. Use o comando a seguir para obter o `prometheusEndpoint` para o espaço de trabalho. *`WORKSPACE-ID`*Substitua pela ID do espaço de trabalho da seção anterior.

   ```
   aws amp describe-workspace --workspace-id WORKSPACE-ID
   ```

   O prometheusEndpoint estará nos resultados de retorno e será formatado assim:

   ```
   https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-a1b2c3d4-a123-b456-
   c789-ac1234567890/
   ```

   Salve esse URL para uso nas próximas etapas.

2. Crie um arquivo com o texto a seguir e chame-o de `prometheus-config.yaml`. *`account`*Substitua pelo ID da sua conta, *`workspaceURL/`* pelo URL que você acabou de encontrar e *`region`* pelo apropriado Região da AWS para o seu sistema.

   ```
   serviceAccounts:
           server:
               name: "amp-iamproxy-ingest-service-account"
               annotations:
                   eks.amazonaws.com/role-arn: "arn:aws:iam::account:role/amp-
   iamproxy-ingest-role"
   server:
       remoteWrite:
           - url: workspaceURL/api/v1/remote_write
             sigv4:
               region: region
             queue_config:
               max_samples_per_send: 1000
               max_shards: 200
               capacity: 2500
   ```

3. Encontre os nomes do gráfico e do namespace do Prometheus, bem como a versão do gráfico, com o seguinte comando Helm.

   ```
   helm ls --all-namespaces
   ```

Com base nas etapas até aqui, o gráfico e o namespace do Prometheus devem ser nomeados `prometheus`, e a versão do gráfico pode ser `15.2.0`

4. Execute o comando a seguir*PrometheusChartName*, usando o*PrometheusNamespace*,, e *PrometheusChartVersion* encontrado na etapa anterior.

```
helm upgrade PrometheusChartName prometheus-community/prometheus -
n PrometheusNamespace -f prometheus-config.yaml --version PrometheusChartVersion
```

Depois de alguns minutos, você verá uma mensagem informando que a atualização ocorreu com êxito.

5. Opcionalmente, valide se as métricas estão sendo enviadas com êxito consultando o endpoint do Amazon Managed Service for Prometheus via `awscurl`. *Region*Substitua pelo Região da AWS que você está usando e *workspaceURL/* pelo URL encontrado na etapa 1.

```
awscurl --service="aps" --region="Region" "workspaceURL/api/v1/query?
query=node_cpu_seconds_total"
```

Agora você criou um espaço de trabalho do Amazon Managed Service for Prometheus e se conectou a ele a partir do seu cluster do Amazon EKS usando arquivos YAML como configuração. Esses arquivos, chamados de definições de recursos personalizados (CRDs), residem em seu cluster Amazon EKS. Você pode usar o controlador AWS Controllers for Kubernetes para gerenciar todos os seus recursos do Amazon Managed Service for Prometheus diretamente do cluster.

Integração de métricas do CloudWatch com o Amazon Managed Service for Prometheus

Ter todas as suas métricas em um só lugar pode ajudar. O Amazon Managed Service for Prometheus não ingere métricas do Amazon CloudWatch automaticamente. No entanto, você pode usar o Amazon Data Firehose e o AWS Lambda para enviar métricas do CloudWatch ao Amazon Managed Service for Prometheus.

Esta seção descreve como instrumentar um fluxo de métricas do Amazon CloudWatch e usar o Amazon Data Firehose e o AWS Lambda para ingerir métricas no Amazon Managed Service for Prometheus.

Você definirá uma pilha usando o kit de desenvolvimento em nuvem (CDK) da AWS para criar um fluxo de entrega do Firehose, um bucket do Amazon S3 e Lambda para demonstrar um cenário completo.

Infraestrutura

A primeira coisa que você deve fazer é configurar a infraestrutura dessa fórmula.

Os fluxos de métricas do CloudWatch permitem o encaminhamento dos dados métricos de transmissão para um endpoint HTTP ou um bucket do Amazon S3.

A configuração da infraestrutura consistirá em 4 etapas:

- Configurar pré-requisitos
- Criação de um espaço de trabalho do Amazon Managed Service for Prometheus
- Instalar as dependências
- Implantar a pilha

Pré-requisitos

- A AWS CLI estar instalada e configurada em seu ambiente.
- O AWS CDK Typescript estar instalado em seu ambiente.
- O Node.js e o Go estarem instalados em seu ambiente.
- O repositório do GitHub do exportador de métricas do CloudWatch de observabilidade da AWS (`CWMetricsStreamExporter`) ter sido clonado em sua máquina local.

Para criar um espaço de trabalho do Amazon Managed Service for Prometheus

1. O aplicativo de demonstração dessa fórmula será executado no Amazon Managed Service for Prometheus. Crie seu espaço de trabalho do Amazon Managed Service for Prometheus por meio do seguinte comando:

   ```
   aws amp create-workspace --alias prometheus-demo-recipe
   ```

2. Verifique se o seu espaço de trabalho foi criado com o seguinte comando:

   ```
   aws amp list-workspaces
   ```

Para obter mais informações sobre o Amazon Managed Service for Prometheus, consulte o Guia do usuário do Amazon Managed Service for Prometheus.

Para instalar dependências

1. Instale as dependências

   Na raiz do repositório `aws-o11y-recipes`, altere seu diretório para `CWMetricStreamExporter` usando o comando:

   ```
   cd sandbox/CWMetricStreamExporter
   ```

   A partir de agora, esse será considerado a raiz do repositório.

2. Altere o diretório para `/cdk` por meio do comando a seguir:

   ```
   cd cdk
   ```

3. Instale as dependências do CDK por meio do seguinte comando:

   ```
   npm install
   ```

4. Altere o diretório de volta para a raiz do repositório e, em seguida, altere o diretório para `/lambda` usando o seguinte comando:

   ```
   cd lambda
   ```

5. Uma vez na pasta `/lambda`, instale as dependências do Go usando:

   ```
   go get
   ```

   Agora todas as dependências estão instaladas.

Para implantar a pilha

1. Na raiz do repositório, abra `config.yaml` e modifique o URL do espaço de trabalho do Amazon Managed Service for Prometheus substituindo o `{workspace}` pelo ID do espaço de trabalho

recém-criado e pela região em que está seu espaço de trabalho do Amazon Managed Service for Prometheus.

Por exemplo, modifique o seguinte com:

```
AMP:
    remote_write_url: "https://aps-workspaces.us-east-2.amazonaws.com/workspaces/
{workspaceId}/api/v1/remote_write"
    region: us-east-2
```

Altere os nomes do fluxo de entrega do Firehose e do bucket do Amazon S3 como preferir.

2. Para criar o AWS CDK e o código Lambda, execute o seguinte comando na raiz do repositório:

```
npm run build
```

Essa etapa de criação garante que o binário do Go Lambda seja criado e implanta o CDK no CloudFormation.

3. Para concluir a implantação, revise e aceite as alterações do IAM exigidas pela pilha.
4. (Opcional) Você pode verificar se a pilha foi criada executando o seguinte comando.

```
aws cloudformation list-stacks
```

Uma pilha chamada `CDK Stack` estará na lista.

Criação de um fluxo do Amazon CloudWatch

Agora que você tem uma função do Lambda para lidar com as métricas, você pode criar o fluxo de métricas a partir do Amazon CloudWatch.

Para criar um fluxo de métricas do CloudWatch

1. Vá até o console do CloudWatch, em https://console.aws.amazon.com/cloudwatch/home#metric-streams:streamsList, e selecione Criar fluxo de métricas.
2. Selecione as métricas necessárias, sejam todas as métricas ou somente aquelas dentro dos namespaces selecionados.
3. Em `Configuration`, escolha Selecionar um Firehose existente pertencente à sua conta.

4. Você usará o Firehose criado anteriormente pelo CDK. No menu suspenso Selecionar seu fluxo do Kinesis Data Firehose, selecione o fluxo criado anteriormente. Ele terá um nome como `CdkStack-KinesisFirehoseStream123456AB-sample1234`.
5. Altere o formato de saída para JSON.
6. Dê ao fluxo de métricas um nome que signifique alguma coisa para você.
7. Escolha Create metric stream (Criar filtro de métrica).
8. (Opcional) Para verificar a invocação da função do Lambda, vá até o console do Lambda e escolha a função `KinesisMessageHandler`. Selecione a guia Monitorar e a subguia Registros e, em Invocações recentes, deve haver entradas da função do Lambda sendo acionadas.

> **Note**
>
> Pode levar até 5 minutos até que as invocações comecem a ser exibidas na guia Monitorar.

Suas métricas agora estão sendo transmitidas do Amazon CloudWatch para o Amazon Managed Service for Prometheus.

Limpeza

Você pode precisar limpar os recursos usados neste exemplo. O procedimento a seguir explica como. Isso interromperá o fluxo de métricas que você criou.

Como limpar recursos

1. Comece excluindo a pilha do CloudFormation com os seguintes comandos:

```
cd cdk
cdk destroy
```

2. Remova o espaço de trabalho do Amazon Managed Service for Prometheus:

```
aws amp delete-workspace --workspace-id \
    `aws amp list-workspaces --alias prometheus-sample-app --query
 'workspaces[0].workspaceId' --output text`
```

3. Por fim, remova o fluxo de métricas do Amazon CloudWatch usando o console do Amazon CloudWatch.

Segurança no Amazon Managed Service for Prometheus

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados 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 isso 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 terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de AWS de . Para saber mais sobre os programas de conformidade que se aplicam ao Amazon Managed Service for Prometheus, AWS consulte Serviços no escopo do programa de conformidade Serviços no escopo AWS de conformidade.
- 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 a entender como aplicar o modelo de responsabilidade compartilhada ao usar o Amazon Managed Service for Prometheus. Os tópicos a seguir mostram como configurar o Amazon Managed Service for Prometheus para atender aos seus objetivos de segurança e compatibilidade. Você também aprende a usar outros AWS serviços que ajudam você a monitorar e proteger seus recursos do Amazon Managed Service for Prometheus.

Tópicos

- Configure perfis do IAM para as contas de serviço
- Como utilizar o Amazon Managed Service for Prometheus com endpoints da VPC de interface

Proteção de dados no Amazon Managed Service for Prometheus

O modelo de responsabilidade AWS compartilhada O modelo de se aplica à proteção de dados no Amazon Managed Service for Prometheus. 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 obter mais informações sobre privacidade de dados, consulte Perguntas frequentes sobre privacidade de dados . Para obter informações sobre proteção de dados na Europa, consulte o Centro de Regulamento Geral sobre a Proteção de Dados (RGPD).

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 Amazon Managed Service for Prometheus ou Serviços da AWS outros usando o console, a API AWS CLI ou os SDKs. AWS 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.

Tópicos

Dados coletados pelo Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus coleta e armazena métricas operacionais que você configura para serem enviadas dos servidores Prometheus em execução na sua conta para o Amazon Managed Service for Prometheus. Esses dados incluem o seguinte:

- Valores da métrica
- Rótulos métricos (ou pares arbitrários de valores-chave) que ajudam a identificar e classificar dados
- Carimbos de data e hora para amostras de dados

IDs de locatário exclusivos isolam dados de diferentes clientes. Esses IDs limitam quais dados do cliente podem ser acessados. Os clientes não podem alterar as IDs dos locatários.

O Amazon Managed Service for Prometheus criptografa os dados que ele armazena AWS Key Management Service com chaves ().AWS KMS O Amazon Managed Service for Prometheus gerencia essas chaves.

> **Note**
>
> O Amazon Managed Service for Prometheus permite a criação de chaves gerenciadas pelo cliente para criptografar seus dados. Para obter mais informações sobre as chaves que o

Amazon Managed Service for Prometheus usa por padrão e sobre como usar as próprias chaves gerenciadas pelo cliente, consulte Criptografia em repouso.

Os dados em trânsito são criptografados automaticamente com HTTPS. O Amazon Managed Service for Prometheus protege conexões entre zonas de disponibilidade em AWS uma região usando HTTPS internamente.

Criptografia em repouso

Por padrão, o Amazon Managed Service for Prometheus fornece automaticamente a criptografia em repouso e faz isso AWS usando chaves de criptografia próprias.

- AWS chaves próprias — O Amazon Managed Service for Prometheus usa essas chaves para criptografar automaticamente os dados enviados para o seu espaço de trabalho. Você não pode visualizar, gerenciar ou usar chaves AWS próprias nem auditar seu uso. No entanto, não é necessário tomar nenhuma medida nem alterar qualquer programa para proteger as chaves que criptografam seus dados. Para obter mais informações, consulte chaves de propriedade da AWS no Guia do desenvolvedor do AWS Key Management Service .

A criptografia de dados em repouso ajuda a reduzir a sobrecarga operacional e a complexidade da proteção de dados confidenciais do cliente, como informações de identificação pessoal. Isso permite que você crie aplicações seguras que atendam aos rigorosos requisitos regulatórios e de conformidade de criptografia.

Como alternativa, é possível usar uma chave gerenciada pelo cliente ao criar o espaço de trabalho:

- Chaves gerenciadas pelo cliente: o Amazon Managed Service for Prometheus é compatível com o uso de uma chave simétrica gerenciada pelo cliente que você cria, detém e gerencia para criptografar os dados no espaço de trabalho. Como você tem controle total dessa criptografia, é possível realizar tarefas como:
  - Estabelecer e manter as políticas de chave
  - Estabelecer e manter subsídios e IAM policies
  - Habilitar e desabilitar políticas de chaves
  - Alternar os materiais de criptografia de chave
  - Adicionar etiquetas
  - Criar réplicas de chaves

- Chaves de agendamento para exclusão

Para obter mais informações, consulte chaves gerenciadas pelo cliente no Guia do desenvolvedor do AWS Key Management Service .

Escolha se deseja usar as chaves gerenciadas pelo cliente ou as chaves AWS próprias com cuidado. Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem ser convertidos para usar chaves AWS próprias posteriormente (e vice-versa).

> **Note**
>
> O Amazon Managed Service for Prometheus ativa automaticamente a criptografia em repouso AWS usando chaves próprias para proteger seus dados sem nenhum custo. No entanto, AWS KMS cobranças são cobradas pelo uso de uma chave gerenciada pelo cliente. Para saber mais sobre preços, consulte Preços do AWS Key Management Service.

Para obter mais informações sobre AWS KMS, consulte O que é AWS Key Management Service?

> **Note**
>
> Os espaços de trabalho criados com chaves gerenciadas pelo cliente não podem usar coletores gerenciados pela AWS para ingestão.

Como o Amazon Managed Service for Prometheus usa subsídios em AWS KMS

O Amazon Managed Service for Prometheus exige três concessões para usar a chave gerenciada pelo cliente.

Quando você cria um espaço de trabalho do Amazon Managed Service para Prometheus criptografado com uma chave gerenciada pelo cliente, o Amazon Managed Service for Prometheus cria as três concessões em seu nome enviando solicitações para. CreateGrant AWS KMS As concessões AWS KMS são usadas para dar ao Amazon Managed Service for Prometheus acesso à chave KMS em sua conta, mesmo quando não são chamadas diretamente em seu nome (por exemplo, ao armazenar dados de métricas que foram extraídos de um cluster do Amazon EKS).

O Amazon Managed Service for Prometheus exige as concessões para usar a chave gerenciada pelo cliente para as seguintes operações internas:

- Envie DescribeKeysolicitações AWS KMS para verificar se a chave KMS simétrica gerenciada pelo cliente fornecida ao criar um espaço de trabalho é válida.
- Envie GenerateDataKeysolicitações AWS KMS para gerar chaves de dados criptografadas pela chave gerenciada pelo cliente.
- Envie solicitações de descriptografia para AWS KMS descriptografar as chaves de dados criptografadas para que elas possam ser usadas para criptografar seus dados.

O Amazon Managed Service for Prometheus cria três concessões para a chave que permitem que AWS KMS o Amazon Managed Service for Prometheus use a chave em seu nome. É possível remover o acesso à chave alterando a política de chaves, desabilitando a chave ou revogando a concessão. É necessário entender as consequências dessas ações antes de executá-las. Isso pode causar perda de dados no espaço de trabalho.

Se você remover o acesso a qualquer uma das concessões de alguma forma, o Amazon Managed Service for Prometheus não poderá acessar nenhum dos dados criptografados pela chave gerenciada pelo cliente, nem armazenar novos dados enviados para o espaço de trabalho, o que afetará as operações que dependem desses dados. Novos dados enviados para o espaço de trabalho não estarão acessíveis e poderão ser perdidos permanentemente.

⚠ Warning

- Se você desabilitar a chave ou remover o acesso do Amazon Managed Service for Prometheus na política de chaves, os dados do espaço de trabalho não estarão mais acessíveis. Novos dados enviados para o espaço de trabalho não estarão acessíveis e poderão ser perdidos permanentemente.

  É possível acessar os dados do espaço de trabalho e começar a receber novos dados novamente restaurando o acesso à chave do Amazon Managed Service for Prometheus.
- Se você revogar uma concessão, ela não poderá ser recriada e os dados no espaço de trabalho serão perdidos permanentemente.

Etapa 1: criar uma chave gerenciada pelo cliente

Você pode criar uma chave simétrica gerenciada pelo cliente usando o Console de gerenciamento da AWS, ou as AWS KMS APIs. A chave não precisa estar na mesma conta do espaço de trabalho

do Amazon Managed Service for Prometheus, desde que você forneça o acesso correto por meio da política, conforme descrito abaixo.

Para criar uma chave simétrica gerenciada pelo cliente

Siga as etapas de Criar uma chave simétrica gerenciada pelo cliente 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, você pode especificar uma política de chaves. Para obter mais informações, consulte Gerenciamento do acesso às chaves gerenciadas pelo cliente no Guia do desenvolvedor do AWS Key Management Service .

Para usar a chave gerenciada pelo cliente com os espaços de trabalho do Amazon Managed Service for Prometheus, as seguintes operações de API deverão ser permitidas na política de chave:

- `kms:CreateGrant`: adiciona uma concessão a uma chave gerenciada pelo cliente. Concede acesso de controle a uma chave do KMS especificada, que permite o acesso às operações de concessão exigidas pelo Amazon Managed Service for Prometheus. Para obter mais informações, consulte Uso de concessões no Guia do desenvolvedor do AWS Key Management Service .

  Com isso, o Amazon Managed Service for Prometheus pode:

  - Ligar para `GenerateDataKey` para gerar uma chave de dados criptografada e armazená-la, porque a chave de dados não é usada imediatamente para criptografar.
  - Ligar para `Decrypt` para usar a chave de dados criptografada armazenada para acessar os dados criptografados.
- `kms:DescribeKey`: fornece os detalhes da chave gerenciada pelo cliente para permitir que o Amazon Managed Service for Prometheus valide a chave.

Veja a seguir exemplos de declarações de política que você pode adicionar ao Amazon Managed Service for Prometheus:

```
 "Statement" : [
   {
     "Sid" : "Allow access to Amazon Managed Service for Prometheus principal within
your account",
```

```
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "*"
      },
      "Action" : [
        "kms:DescribeKey",
        "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "kms:ViaService" : "aps.region.amazonaws.com",
          "kms:CallerAccount" : "111122223333"
        }
    },
    {
      "Sid": "Allow access for key administrators - not required for Amazon Managed
Service for Prometheus",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
       },
      "Action" : [
        "kms:*"
       ],
      "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
    },
    <other statements needed for other non-Amazon Managed Service for Prometheus
scenarios>
 ]
```

- Para obter mais informações sobre especificar permissões em uma política, consulte o Guia do desenvolvedor do AWS Key Management Service .
- Para obter mais informações sobre solução de problemas de acesso à chave, consulte o Guia do Desenvolvedor do AWS Key Management Service .

Etapa 2: especificar chave gerenciada pelo cliente para o Amazon Managed Service for Prometheus

Ao criar um espaço de trabalho, você pode especificar a chave gerenciada pelo cliente inserindo um ARN da chave do KMS, que o Amazon Managed Service for Prometheus usa para criptografar os dados armazenados pelo espaço de trabalho.

Etapa 3: acessar dados de outros serviços, como o Amazon Managed Grafana

Esta etapa é opcional; só é necessária se você precisar acessar seus dados do Amazon Managed Service for Prometheus de outro serviço.

Seus dados criptografados não podem ser acessados por outros serviços, a menos que eles também tenham acesso para usar a AWS KMS chave. Por exemplo, se você quiser usar o Amazon Managed Grafana para criar um painel ou um alerta sobre seus dados, você deve conceder ao Amazon Managed Grafana acesso à chave.

Para conceder ao Amazon Managed Grafana acesso à chave gerenciada pelo cliente

1. Na sua lista de espaços de trabalho do Amazon Managed Grafana, selecione o nome do espaço de trabalho que você deseja que tenha acesso ao Amazon Managed Service for Prometheus. Isso mostra informações resumidas sobre seu espaço de trabalho do Amazon Managed Grafana.
2. Anote o nome do perfil do IAM usado pelo seu espaço de trabalho. O nome deve estar no formato `AmazonGrafanaServiceRole-<unique-id>`. O console mostra o ARN completo do perfil. Especifique esse nome no console do AWS KMS em uma etapa posterior.
3. Na sua lista de chaves do AWS KMS gerenciadas pelo cliente, escolha aquela que você usou durante a criação do seu espaço de trabalho do Amazon Managed Service for Prometheus. Isso abre a página de detalhes da configuração da chave.
4. Ao lado de Usuários de chaves, selecione o botão Adicionar.
5. Na lista de nomes, escolha o perfil do IAM do Amazon Managed Grafana que você anotou anteriormente. Para facilitar a localização, é possível pesquisar pelo nome também.
6. Selecione Adicionar para adicionar o perfil do IAM à lista de usuários de chaves.

Seu espaço de trabalho do Amazon Managed Grafana agora pode acessar os dados no seu espaço de trabalho do Amazon Managed Service for Prometheus. Você pode adicionar outros usuários ou perfis aos usuários de chaves para permitir que outros serviços acessem seu espaço de trabalho.

Contexto de criptografia do Amazon Managed Service for Prometheus

Um contexto de criptografia é um conjunto opcional de pares 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 do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus usa o mesmo contexto de criptografia em AWS KMS todas as operações criptográficas, onde a chave `aws:amp:arn` está e o valor é o Amazon Resource Name (ARN) do espaço de trabalho.

Example

```
"encryptionContext": {
    "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-
abcd-56ef-7890abcd12ef"
}
```

Uso do contexto de criptografia para monitoramento

Ao usar uma chave simétrica gerenciada pelo cliente para criptografar os dados do espaço de trabalho, você também pode utilizar o contexto de criptografia em registros de auditoria e logs para identificar como a chave gerenciada pelo cliente está sendo utilizada. O contexto de criptografia também aparece nos registros gerados pelo AWS CloudTrail ou Amazon CloudWatch Logs.

Uso do contexto de criptografia para controlar o acesso à chave gerenciada pelo cliente

Você pode usar o contexto de criptografia nas políticas de chaves e políticas do IAM como `conditions` e controlar o acesso à sua chave simétrica gerenciada pelo cliente. Você também pode usar restrições no contexto de criptografia em uma concessão.

O Amazon Managed Service for Prometheus utiliza uma restrição de contexto de criptografia em concessões para controlar o acesso à chave gerenciada pelo cliente na conta ou região. A restrição de concessão exige que as operações permitidas pela concessão usem o contexto de criptografia especificado.

Example

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 CreateGrant",
     "Effect": "Allow",
     "Principal": {
         "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": "kms:CreateGrant",
     "Resource": "*",
     "Condition": {
         "StringEquals": {
             "kms:EncryptionContext:aws:aps:arn": "arn:aws:aps:us-
west-2:111122223333:workspace/ws-sample-1234-abcd-56ef-7890abcd12ef"
          }
     }
}
```

Monitorar as chaves de criptografia do Amazon Managed Service for Prometheus

Ao usar uma chave gerenciada pelo AWS KMS cliente com seus espaços de trabalho do Amazon Managed Service for Prometheus, você pode usar AWS CloudTrailo Amazon Logs para rastrear solicitações enviadas pelo CloudWatch Amazon Managed Service for Prometheus. AWS KMS

Os exemplos a seguir são AWS CloudTrail eventos para`CreateGrant`, `GenerateDataKeyDecrypt`, e `DescribeKey` para monitorar operações do KMS chamadas pelo Amazon Managed Service para que o Prometheus acesse dados criptografados pela chave gerenciada pelo cliente:

CreateGrant

Quando você usa uma chave gerenciada pelo AWS KMS cliente para criptografar seu espaço de trabalho, o Amazon Managed Service for Prometheus envia três `CreateGrant` solicitações em seu nome para acessar a chave KMS que você especificou. As concessões que o Amazon Managed Service for Prometheus cria são específicas do recurso associado à chave gerenciada pelo cliente do AWS KMS .

O evento de exemplo a seguir registra uma operação `CreateGrant`:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "TESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE-KEY-ID1",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "TESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "retiringPrincipal": "aps.region.amazonaws.com",
        "operations": [
```

```
            "GenerateDataKey",
            "Decrypt",
            "DescribeKey"
        ],
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "granteePrincipal": "aps.region.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKey

Quando você ativa uma chave gerenciada pelo AWS KMS cliente para seu espaço de trabalho, o Amazon Managed Service for Prometheus cria uma chave exclusiva. Ele envia uma `GenerateDataKey` solicitação AWS KMS que especifica a chave gerenciada pelo AWS KMS cliente para o recurso.

O evento de exemplo a seguir registra a operação `GenerateDataKey`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "aps.amazonaws.com"
```

```
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "encryptionContext": {
            "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-
sample-1234-abcd-56ef-7890abcd12ef"
        },
        "keySpec": "AES_256",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "57f5dbee-16da-413e-979f-2c4c6663475e"
}
```

Decrypt

Quando uma consulta é gerada em um espaço de trabalho criptografado, o Amazon Managed Service for Prometheus chama a operação `Decrypt` para usar a chave de dados criptografada armazenada para acessar os dados criptografados.

O evento de exemplo a seguir registra a operação `Decrypt`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:10:51Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "encryptionContext": {
            "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-
sample-1234-abcd-56ef-7890abcd12ef"
        },
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "dc129381-1d94-49bd-b522-f56a3482d088"
}
```

DescribeKey

O Amazon Managed Service for Prometheus usa a operação `DescribeKey` para verificar se a chave gerenciada pelo cliente do AWS KMS associada ao espaço de trabalho existe na conta e na região.

O evento de exemplo a seguir registra a operação `DescribeKey`:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "TESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE-KEY-ID1",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "TESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "aps.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "keyId": "00dd0db0-0000-0000-ac00-b0c000SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
```

```
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

Saiba mais

Os recursos a seguir fornecem mais informações sobre a criptografia de dados em pausa.

- Para obter mais informações sobre conceitos básicos do AWS Key Management Service, consulte o Guia do desenvolvedor do AWS Key Management Service .
- Para obter mais informações sobre as melhores práticas de segurança para AWS Key Management Service, consulte o Guia do AWS Key Management Service desenvolvedor.

Gerenciamento de identidade e acesso para Amazon Managed Service for Prometheus

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 (ter permissões) para utilizar os recursos do Amazon Managed Service for Prometheus. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- Gerenciar o acesso usando políticas
- Como o Amazon Managed Service for Prometheus funciona com o IAM
- Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus
- Resolução de problemas de identidade e acesso no Amazon Managed Service for Prometheus

Público

A forma como você usa AWS Identity and Access Management (IAM) difere com base na sua função:

- Usuário do serviço: solicite permissões ao seu administrador se você não conseguir acessar os atributos (consulte Resolução de problemas de identidade e acesso no Amazon Managed Service for Prometheus).
- Administrador do serviço: determine o acesso do usuário e envie solicitações de permissão (consulte Como o Amazon Managed Service for Prometheus funciona com o IAM)
- Administrador do IAM: escreva políticas para gerenciar o acesso (consulte Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus)

Autenticação com identidades

A autenticação é a forma como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado como usuário do IAM ou assumindo uma função do IAM. Usuário raiz da conta da AWS

Você pode fazer login como uma identidade federada usando credenciais de uma fonte de identidade como Centro de Identidade do AWS IAM (IAM Identity Center), autenticação de login único ou credenciais. Google/Facebook Para ter mais informações sobre como fazer login, consulte Como fazer login em sua Conta da AWS no Guia do usuário do Início de Sessão da AWS .

Para acesso programático, AWS fornece um SDK e uma CLI para assinar solicitações criptograficamente. Para ter mais informações, consulte AWS Signature Version 4 para solicitações de API no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar um Conta da AWS, você começa com uma identidade de login chamada usuário Conta da AWS raiz que tem acesso completo a todos Serviços da AWS os recursos. É altamente

recomendável não usar o usuário-raiz em tarefas diárias. Consulte as tarefas que exigem credenciais de usuário-raiz em Tarefas que exigem credenciais de usuário-raiz no Guia do usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários humanos usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório corporativo, provedor de identidade da web ou Directory Service que acessa Serviços da AWS usando credenciais de uma fonte de identidade. As identidades federadas assumem funções que oferecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos Centro de Identidade do AWS IAM. Para saber mais, consulte O que é o IAM Identity Center? no Guia do usuário do Centro de Identidade do AWS IAM .

Usuários e grupos do IAM

Um usuário do IAM é uma identidade com permissões específicas para uma única pessoa ou aplicação. É recomendável usar credenciais temporárias, em vez de usuários do IAM com credenciais de longo prazo. Para obter mais informações, consulte Exigir que usuários humanos usem a federação com um provedor de identidade para acessar AWS usando credenciais temporárias no Guia do usuário do IAM.

Um grupo do IAM especifica um conjunto de usuários do IAM e facilita o gerenciamento de permissões para grandes conjuntos de usuários. Para ter mais informações, consulte Casos de uso de usuários do IAM no Guia do usuário do IAM.

Perfis do IAM

Uma perfil do IAM é uma identidade com permissões específicas que oferece credenciais temporárias. Você pode assumir uma função mudando de um usuário para uma função do IAM (console) ou chamando uma operação de AWS API AWS CLI ou. Para saber mais, consulte Métodos para assumir um perfil no Manual do usuário do IAM.

Os perfis do IAM são úteis para acesso de usuário federado, permissões de usuário do IAM temporárias, acesso entre contas, acesso entre serviços e aplicações em execução no Amazon EC2. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política define permissões quando associada a uma identidade ou recurso. AWS avalia essas políticas quando um diretor faz uma solicitação. A maioria das políticas é armazenada AWS como documentos JSON. Para ter mais informações sobre documentos de política JSON, consulte Visão geral das políticas JSON no Guia do usuário do IAM.

Por meio de políticas, os administradores especificam quem tem acesso a que, definindo qual entidade principal pode realizar ações em quais recursos e sob quais condições.

Por padrão, usuários e perfis não têm permissões. Um administrador do IAM cria políticas do IAM e as adiciona aos perfis, os quais os usuários podem então assumir. As políticas do IAM definem permissões, independentemente do método usado para realizar a operação.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissão JSON que você anexa a uma identidade (usuário, grupo ou perfil). Essas políticas controlam quais ações as identidades podem realizar, em quais recursos e sob quais 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.

As políticas baseadas em identidade podem ser políticas em linha (incorporadas diretamente em uma única identidade) ou políticas gerenciadas (políticas autônomas anexadas a várias identidades). Para saber como escolher entre uma política gerenciada e políticas em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. Entre os exemplos estão políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. É necessário especificar uma entidade principal em uma política baseada em recursos.

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais que podem definir o máximo de permissões concedidas por tipos de políticas mais comuns:

- Limites de permissões: definem o número máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. Para saber mais sobre limites de permissões, consulte Limites de permissões para identidades do IAM no Guia do usuário do IAM.
- Políticas de controle de serviço (SCPs) — Especifique as permissões máximas para uma organização ou unidade organizacional em AWS Organizations. Para saber mais, consulte Políticas de controle de serviço no Guia do usuário do AWS Organizations .
- Políticas de controle de recursos (RCPs) — Defina o máximo de permissões disponíveis para recursos em suas contas. Para obter mais informações, consulte Políticas de controle de recursos (RCPs) no Guia AWS Organizations do usuário.
- Políticas de sessão: políticas avançadas transmitidas como um parâmetro durante a criação de uma sessão temporária para um perfil ou um usuário federado. Para saber mais, consulte Políticas de sessão no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como o Amazon Managed Service for Prometheus funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao Amazon Managed Service for Prometheus, entenda que atributos do IAM estão disponíveis para uso com o Amazon Managed Service for Prometheus.

Atributos do IAM que você pode usar com o Amazon Managed Service for Prometheus

Atributo do IAM	Suporte ao Amazon Managed Service for Prometheus
Políticas baseadas em identidade	Sim

Atributo do IAM	Suporte ao Amazon Managed Service for Prometheus
Políticas baseadas em atributos	Sim
Ações de políticas	Sim
Recursos de políticas	Sim
Chaves de condição de políticas	Não
ACLs	Não
ABAC (tags em políticas)	Sim
Credenciais temporárias	Sim
Sessões de acesso direto (FAS)	Não
Perfis de serviço	Não
Perfis vinculados ao serviço	Sim

Para ter uma visão de alto nível de como o Amazon Managed Service for Prometheus e AWS outros serviços funcionam com a maioria dos recursos do IAM, AWS consulte os serviços que funcionam com o IAM no Guia do usuário do IAM.

Políticas baseadas em identidade do Amazon Managed Service for Prometheus

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 do Amazon Managed Service for Prometheus

Para visualizar exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus, consulte Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus.

Políticas baseadas em recursos do Amazon Managed Service for Prometheus

Compatível com políticas baseadas em recursos: sim

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. É necessário especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, é possível especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Ações de políticas para o Amazon Managed Service for Prometheus

Compatível com ações 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 `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista das ações do Amazon Managed Service for Prometheus, consulte Ações definidas pelo Amazon Managed Service for Prometheus na Referência de autorização de serviço.

As ações de política no Amazon Managed Service for Prometheus usam o seguinte prefixo antes da ação:

```
aps
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [
      "aps:action1",
      "aps:action2"
         ]
```

Para visualizar exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus, consulte Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus.

Recursos de políticas do Amazon Managed Service for Prometheus

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": "*"
```

Para ver uma lista dos tipos de recursos do Amazon Managed Service for Prometheus e ARNs seus, consulte Recursos definidos pelo Amazon Managed Service for Prometheus na Referência de Autorização de Serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte Ações definidas pelo Amazon Managed Service for Prometheus.

Para visualizar exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus, consulte Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus.

Chaves de condição de políticas para o Amazon Managed Service for Prometheus

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 de chaves de condição do Amazon Managed Service for Prometheus, consulte Chaves de condição para o Amazon Managed Service for Prometheus na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar a chave de condição, consulte Ações definidas pelo Amazon Managed Service for Prometheus.

Para visualizar exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus, consulte Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus.

Listas de controle de acesso (ACLs) no Amazon Managed Service para Prometheus

Suportes ACLs: Não

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

Controle de acesso por atributos (ABAC) com o Amazon Managed Service for Prometheus

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.

Uso de credenciais temporárias com o Amazon Managed Service for Prometheus

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.

Encaminhar sessões de acesso para o Amazon Managed Service for Prometheus

Compatível com sessões de acesso direto (FAS): não

As sessões de acesso direto (FAS) usam as permissões do principal chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) de fazer solicitações aos serviços posteriores. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.

Perfis de serviço para o Amazon Managed Service for Prometheus

Compatível com perfis de serviço: não

O perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para saber mais, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.

> **Warning**
>
> A alteração das permissões de um perfil de serviço pode interromper a funcionalidade do Amazon Managed Service for Prometheus. Edite perfis de serviço somente quando o Amazon Managed Service for Prometheus fornecer orientação para isso.

Perfis vinculados ao serviço para o Amazon Managed Service for Prometheus

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 ou gerenciar perfis vinculados ao serviço do Amazon Managed Service for Prometheus, consulte Usar perfis vinculados ao serviço para o Amazon Managed Service for Prometheus.

Exemplos de políticas baseadas em identidade do Amazon Managed Service for Prometheus

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do Amazon Managed Service for Prometheus. 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 Managed Service for Prometheus, incluindo o formato de cada um ARNs dos tipos de recursos, consulte Ações, recursos e chaves de condição do Amazon Managed Service for Prometheus na Referência de Autorização de Serviço.

Tópicos

- Práticas recomendadas de política
- Usar o console do Amazon Managed Service for Prometheus

- Permitir que os usuários visualizem suas próprias permissões

Práticas recomendadas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do Amazon Managed Service for Prometheus 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.

Usar o console do Amazon Managed Service for Prometheus

Para acessar o console do Amazon Managed Service for Prometheus, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os recursos do Amazon Managed Service for Prometheus em sua Conta da AWS. Caso crie uma política baseada em identidade mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) 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 a ações que correspondam à operação de API que estiverem tentando executar.

Para garantir que usuários e funções ainda possam usar o console do Amazon Managed Service for Prometheus, anexe também o Amazon Managed Service for `ConsoleAccess` `ReadOnly` AWS Prometheus ou a política gerenciada às entidades. Para obter informações, consulte Adicionar permissões a um usuário no Guia do usuário do IAM.

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
```

```
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Resolução de problemas de identidade e acesso no Amazon Managed Service for Prometheus

Use as informações a seguir para ajudar a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o Amazon Managed Service for Prometheus e o IAM.

Tópicos

- Não tenho autorização para executar uma ação no Amazon Managed Service for Prometheus
- Não estou autorizado a realizar iam: PassRole
- Quero permitir que pessoas fora da minha AWS conta acessem meus recursos do Amazon Managed Service for Prometheus

Não tenho autorização para executar uma ação no Amazon Managed Service for Prometheus

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM `mateojackson` tenta usar o console para visualizar detalhes sobre um atributo *`my-example-widget`* fictício, mas não tem as permissões `aps:`*`GetWidget`* fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 aps:GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso *`my-example-widget`* usando a ação `aps:`*`GetWidget`*.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar iam: PassRole

Caso receba uma mensagem de erro informando que você não tem autorização para executar a ação `iam:PassRole`, as políticas deverão ser atualizadas para permitir a transmissão de um perfil ao Amazon Managed Service for Prometheus.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O erro exemplificado a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no Amazon Managed Service for Prometheus. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha AWS conta acessem meus recursos do Amazon Managed Service for Prometheus

É possível criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o Amazon Managed Service for Prometheus é compatível com esses atributos, consulte Como o Amazon Managed Service for Prometheus funciona com o IAM.
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS a terceiros no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte Conceder acesso a usuários autenticados externamente (federação de identidades) no Guia do usuário do IAM.
- Para conhecer a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Permissões e políticas no IAM

O acesso às ações e dados do Amazon Managed Service for Prometheus requer credenciais. Essas credenciais devem ter permissões para realizar as ações e acessar os AWS recursos, como recuperar dados do Amazon Managed Service for Prometheus sobre seus recursos de nuvem. As seções a seguir fornecem detalhes sobre como você pode usar o AWS Identity and Access Management (IAM) e o Amazon Managed Service for Prometheus para ajudar a proteger seus recursos, controlando quem pode acessá-los. Para obter mais informações, consulte Políticas e permissões no IAM.

Permissões do Amazon Managed Service for Prometheus

Para ver a lista de ações, tipos de recurso e chaves de condição possíveis do Amazon Managed Service for Prometheus, consulte Ações, recursos e chaves de condição do Amazon Managed Service for Prometheus.

Políticas do IAM de exemplo

Esta seção fornece exemplos de outras políticas autogerenciadas que você pode criar.

A política do IAM a seguir concede acesso total ao Amazon Managed Service for Prometheus e também permite que um usuário descubra clusters do Amazon EKS e veja os detalhes sobre eles.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:*",
                "eks:DescribeCluster",
                "eks:ListClusters"
            ],
            "Resource": "*"
        }
    ]
}
```

Validação de conformidade para o Amazon Managed Service for Prometheus

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 Managed Service for Prometheus

A infraestrutura global da AWS é criada com base em regiões da AWS e zonas de disponibilidade. As regiões da AWS As regiões fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, conectadas com baixa latência, throughput elevado e redes altamente redundantes. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre regiões e zonas de disponibilidade da AWS, consulte Infraestrutura global da AWS.

Além da infraestrutura global da AWS, o Amazon Managed Service for Prometheus oferece vários atributos para oferecer suporte às suas necessidades de resiliência e backup de dados, incluindo suporte para dados de alta disponibilidade.

Segurança de infraestrutura no Amazon Managed Service for Prometheus

Como um serviço gerenciado, o Amazon Managed Service for Prometheus é protegido pela segurança de rede global da AWS. Para obter informações sobre serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte Segurança na Nuvem AWS. Para projetar seu ambiente da AWS usando as práticas recomendadas de segurança da infraestrutura, consulte Proteção de Infraestrutura em Pilar de Segurança: AWS Estrutura bem arquitetada.

Você usa as chamadas de API da AWS publicadas para acessar o Amazon Managed Service for Prometheus 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.

Usar perfis vinculados ao serviço para o Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus AWS Identity and Access Management usa funções vinculadas a serviços (IAM). Um perfil vinculado ao serviço é um tipo exclusivo de perfil do IAM vinculado diretamente ao Amazon Managed Service for Prometheus. Os perfis vinculados ao serviço são predefinidos pelo Amazon Managed Service for Prometheus e incluem todas as permissões que o serviço precisa para chamar outros serviços da AWS em seu nome.

Um perfil vinculado ao serviço facilita a configuração do Amazon Managed Service for Prometheus porque você não precisa adicionar as permissões necessárias manualmente. O Amazon Managed Service for Prometheus define as permissões dos perfis vinculados ao serviço e, exceto definido de outra forma, somente o Amazon Managed Service for Prometheus pode assumir os perfis. As permissões definidas incluem a política de confiança e a política de permissões, que não pode ser anexada a nenhuma outra entidade do IAM.

Usar perfis para extrair métricas do EKS

Ao coletar métricas automaticamente usando o coletor gerenciado do Amazon Managed Service for Prometheus, a função AWSService RoleForAmazonPrometheusScraper vinculada ao serviço é usada para facilitar a configuração do coletor gerenciado, porque você não precisa adicionar manualmente as permissões necessárias. O Amazon Managed Service for Prometheus define as permissões e somente ele pode assumir o perfil.

Para obter informações sobre outros serviços que são compatíveis com perfis vinculados ao serviço, consulte Serviços da AWS que funcionam com o IAM e procure por serviços que indiquem Sim na coluna Perfis vinculados ao serviço. Escolha um Sim com um link para visualizar a documentação do perfil vinculado a esse serviço.

Permissões de perfil vinculadas ao serviço para o Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus usa uma função vinculada ao serviço nomeada com o prefixo AWSServiceRoleForAmazonPrometheusScraperpara permitir que o Amazon Managed Service for Prometheus extraia automaticamente métricas em seus clusters do Amazon EKS.

A função AWSService RoleForAmazonPrometheusScraper vinculada ao serviço confia nos seguintes serviços para assumir a função:

- `scraper.aps.amazonaws.com`

A política de permissões de função nomeada AmazonPrometheusScraperServiceRolePolicy permite que o Amazon Managed Service for Prometheus conclua as seguintes ações nos recursos especificados:

- Ler e modificar a configuração de rede para se conectar à rede que contém o cluster do Amazon EKS.
- Ler métricas de clusters do Amazon EKS e gravar métricas nos espaços de trabalho do Amazon Managed Service for Prometheus.

É necessário configurar permissões para permitir que usuários, grupos ou perfis criem um perfil vinculado ao serviço. Para obter mais informações, consulte Service-linked role permissions (Permissões de nível vinculado a serviços) no Guia do usuário do IAM.

Criar um perfil vinculado ao serviço para o Amazon Managed Service for Prometheus

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você cria uma instância de coletor gerenciada usando o Amazon EKS ou o Amazon Managed Service for Prometheus na, na ou na AWS API, Console de gerenciamento da AWS AWS CLI o Amazon Managed Service for Prometheus cria a função vinculada ao serviço para você.

> **Important**
>
> Esse perfil vinculado ao serviço pode aparecer em sua conta se você concluiu uma ação em outro serviço que usa os atributos compatíveis com esse perfil. Para saber mais, consulte Uma nova função apareceu no meu Conta da AWS.

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 instância do coletor gerenciada usando o Amazon EKS ou o Amazon Managed Service for Prometheus, o Amazon Managed Service for Prometheus cria um perfil vinculado ao serviço para você novamente.

Editar um perfil vinculado ao serviço para o Amazon Managed Service for Prometheus

O Amazon Managed Service para Prometheus não permite que você edite AWSService RoleForAmazonPrometheusScraper a função vinculada ao 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 obter mais informações, consulte Editar uma função vinculada a serviço no Guia do usuário do IAM.

Excluir um perfil vinculado ao serviço para o Amazon Managed Service for Prometheus

Você não precisa excluir manualmente a AWSService RoleForAmazonPrometheusScraper função. Quando você exclui todas as instâncias gerenciadas do coletor associadas à função na Console de gerenciamento da AWS, na ou na AWS API AWS CLI, o Amazon Managed Service for Prometheus limpa os recursos e exclui a função vinculada ao serviço para você.

Regiões compatíveis com perfis vinculados ao serviço do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus é compatível com o uso de perfis vinculados ao serviço em todas as regiões em que o serviço está disponível. Para obter mais informações, consulte Regiões aceitas.

Registro de chamadas de API do Amazon Managed Service para Prometheus usando AWS CloudTrail

O Amazon Managed Service for Prometheus está integrado AWS CloudTrailcom, um serviço que fornece um registro das ações realizadas por um usuário, função ou um. AWS service (Serviço da AWS) CloudTrailcaptura todas as chamadas de API para o Amazon Managed Service for Prometheus como eventos. As chamadas capturadas incluem aquelas do console do Amazon Managed Service for Prometheus e as chamadas de código para as operações de API do Amazon Managed Service for Prometheus. Usando as informações coletadas por CloudTrail, você pode

determinar a solicitação que foi feita ao Amazon Managed Service for Prometheus, o endereço IP a partir do qual a solicitação foi feita, quando foi feita e detalhes adicionais.

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 raiz ou credenciais de usuário.
- Se a solicitação foi feita em nome de um usuário do Centro de Identidade do 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 service (Serviço da AWS).

CloudTrail está ativo Conta da AWS quando você cria a conta e você tem acesso automático ao histórico de CloudTrail eventos. O histórico de CloudTrail eventos fornece um registro visível, pesquisável, baixável e imutável dos últimos 90 dias de eventos de gerenciamento registrados em um. Região da AWS Para obter mais informações, consulte Trabalhando com o histórico de CloudTrail eventos no Guia AWS CloudTrail do usuário. Não há CloudTrail cobrança pela visualização do histórico de eventos.

Para um registro contínuo dos eventos dos Conta da AWS últimos 90 dias, crie uma trilha ou um armazenamento de dados de eventos do CloudTrail Lake.

CloudTrail trilhas

Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Todas as trilhas criadas usando o Console de gerenciamento da AWS são multirregionais. Só é possível criar uma trilha de região única ou de várias regiões usando a AWS CLI. É recomendável criar uma trilha multirregional porque você captura todas as atividades Regiões da AWS em sua conta. Ao criar uma trilha de região única, é possível visualizar somente os eventos registrados na Região da AWS da trilha. Para obter mais informações sobre trilhas, consulte Criar uma trilha para a Conta da AWS e Criar uma trilha para uma organização no Guia do usuário do AWS CloudTrail .

Você pode entregar uma cópia dos seus eventos de gerenciamento contínuos para o bucket do Amazon S3 sem nenhum custo CloudTrail criando uma trilha. No entanto, há cobranças de armazenamento do Amazon S3. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços. Para receber informações sobre a definição de preços do Amazon S3, consulte Definição de preços do Amazon S3.

CloudTrail Armazenamentos de dados de eventos em Lake

CloudTrail O Lake permite que você execute consultas baseadas em SQL em seus eventos. CloudTrail O Lake converte eventos existentes no formato JSON baseado em linhas para o formato Apache ORC. O ORC é um formato colunar de armazenamento otimizado para recuperação rápida de dados. Os eventos são agregados em armazenamentos de dados de eventos, que são coleções imutáveis de eventos baseados nos critérios selecionados com a aplicação de seletores de eventos avançados. Os seletores que aplicados a um armazenamento de dados de eventos controlam quais eventos persistem e estão disponíveis para consulta. Para obter mais informações sobre o CloudTrail Lake, consulte Trabalhando com o AWS CloudTrail Lake no Guia AWS CloudTrail do Usuário.

CloudTrail Os armazenamentos e consultas de dados de eventos em Lake incorrem em custos. Ao criar um armazenamento de dados de eventos, você escolhe a opção de preço que deseja usar para ele. A opção de preço determina o custo para a ingestão e para o armazenamento de eventos, e o período de retenção padrão e máximo para o armazenamento de dados de eventos. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços.

Eventos de gerenciamento do Amazon Managed Service for Prometheus em CloudTrail

Os eventos de gerenciamento fornecem informações sobre as operações de gerenciamento que são realizadas nos recursos do seu Conta da AWS. Também são conhecidas como operações de ambiente de gerenciamento. Por padrão, CloudTrail registra eventos de gerenciamento.

O Amazon Managed Service for Prometheus gera logs de todas as operações do ambiente de gerenciamento do Amazon Managed Service for Prometheus como eventos de gerenciamento. Para obter uma lista das operações do plano de controle do Amazon Managed Service for Prometheus nas quais o Amazon Managed Service for Prometheus se conecta CloudTrail, consulte a Referência da API do Amazon Managed Service for Prometheus.

Exemplos de eventos do Amazon Managed Service for Prometheus

Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a operação de API solicitada, a data e a hora da operação, 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, os eventos não aparecem em nenhuma ordem específica.

Exemplo: CreateWorkspace

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a CreateWorkspace ação.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-11-30T23:39:29Z"
            }
        }
    },
    "eventTime": "2020-11-30T23:43:21Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateWorkspace",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "aws-cli/1.11.167 Python/2.7.10 Darwin/16.7.0 botocore/1.7.25",
    "requestParameters": {
        "alias": "alias-example",
        "clientToken": "12345678-1234-abcd-1234-12345abcd1"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
```

```
        "arn": "arn:aws:aps:us-west-2:123456789012:workspace/ws-abc123456-
abcd-1234-5678-1234567890",
        "status": {
            "statusCode": "CREATING"
        },
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

Exemplo: CreateAlertManagerDefinition

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a CreateAlertManagerDefinition ação.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-09-23T20:20:14Z"
```

```
            }
        }
    },
    "eventTime": "2021-09-23T20:22:43Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateAlertManagerDefinition",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "Boto3/1.17.46 Python/3.6.14 Linux/4.14.238-182.422.amzn2.x86_64 exec-
env/AWS_ECS_FARGATE Botocore/1.20.46",
    "requestParameters": {
        "data":
 "YWxlcnRtYW5hZ2VyX2NvbmZpZzogfAogIGdsb2JhbDoKICAgIHNtdHBfc21hcnRob3N0OiAnbG9jYWxob3N0OjI1JwogI
        "clientToken": "12345678-1234-abcd-1234-12345abcd1",
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
        "status": {
            "statusCode": "CREATING"
        }
    },
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

Exemplo: CreateRuleGroupsNamespace

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a CreateRuleGroupsNamespace ação.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE123EXAMPLE123-1234567890616",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/admin",
```

```
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "creationDate": "2021-09-23T20:22:19Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2021-09-23T20:25:08Z",
    "eventSource": "aps.amazonaws.com",
    "eventName": "CreateRuleGroupsNamespace",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "34.212.33.165",
    "userAgent": "Boto3/1.17.63 Python/3.6.14 Linux/4.14.238-182.422.amzn2.x86_64 exec-
env/AWS_ECS_FARGATE Botocore/1.20.63",
    "requestParameters": {
        "data":
 "Z3JvdXBzOgogIC0gbmFtZTogdGVzdFJ1bGVHcm91cHNOYW1lc3BhY2UKICAgIHJ1bGVzOgogICAgLSBhbGVydDogdGVzd
        "clientToken": "12345678-1234-abcd-1234-12345abcd1",
        "name": "exampleRuleGroupsNamespace",
        "workspaceId": "ws-12345678-1234-abcd-1234-1234567890"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
trace-id,x-amzn-errormessage,x-amz-apigw-id,date",
        "name": "exampleRuleGroupsNamespace",
        "arn": "arn:aws:aps:us-west-2:492980759322:rulegroupsnamespace/ws-
ae46a85c-1609-4c22-90a3-2148642c3b6c/exampleRuleGroupsNamespace",
        "status": {
            "statusCode": "CREATING"
        },
        "tags": {}
    },
```

```
    "requestID": "890b8639-e51f-11e7-b038-EXAMPLE",
    "eventID": "874f89fa-70fc-4798-bc00-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "123456789012"
}
```

Para obter informações sobre o conteúdo do CloudTrail registro, consulte o conteúdo do CloudTrail registro no Guia AWS CloudTrail do usuário.

Configure perfis do IAM para as contas de serviço

Com os perfis do IAM para contas de serviço, é possível associar um perfil do IAM a uma conta de serviço do Kubernetes. Essa conta de serviço pode então fornecer AWS permissões para os contêineres em qualquer pod que use essa conta de serviço. Para obter mais informações, consulte Perfis do IAM para contas de serviço.

Os perfis do IAM para contas de serviço também são conhecidos como perfis de serviço.

No Amazon Managed Service for Prometheus, o uso de perfis de serviço pode ajudar a obter os perfis necessários para autorizar e autenticar entre o Amazon Managed Service for Prometheus, os servidores do Prometheus e os servidores do Grafana.

Pré-requisitos

Os procedimentos nesta página exigem que você tenha a interface de linha de comando AWS CLI e EKSCTL instalada.

Configurar perfis de serviço para a ingestão de métricas de clusters do Amazon EKS

Para configurar os perfis de serviço para permitir que o Amazon Managed Service for Prometheus consuma métricas dos servidores do Prometheus nos clusters do Amazon EKS, você deve estar conectado a uma conta com as seguintes permissões:

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`

- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Para configurar o perfil de serviço para ingestão no Amazon Managed Service for Prometheus

1. Crie um arquivo chamado `createIRSA-AMPIngest.sh` com o conteúdo a seguir. Substitua <my_amazon_eks_clustername> pelo nome do cluster e substitua <my_prometheus_namespace> pelo namespace do Prometheus.

```
#!/bin/bash -e
CLUSTER_NAME=<my_amazon_eks_clustername>
SERVICE_ACCOUNT_NAMESPACE=<my_prometheus_namespace>
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)
OIDC_PROVIDER=$(aws eks describe-cluster --name $CLUSTER_NAME --query
 "cluster.identity.oidc.issuer" --output text | sed -e "s/^https:\/\///")
SERVICE_ACCOUNT_AMP_INGEST_NAME=amp-iamproxy-ingest-service-account
SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE=amp-iamproxy-ingest-role
SERVICE_ACCOUNT_IAM_AMP_INGEST_POLICY=AMPIngestPolicy
#
# Set up a trust policy designed for a specific combination of K8s service account
 and namespace to sign in from a Kubernetes cluster which hosts the OIDC Idp.
#
cat <<EOF > TrustPolicy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::${AWS_ACCOUNT_ID}:oidc-provider/
${OIDC_PROVIDER}"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "${OIDC_PROVIDER}:sub": "system:serviceaccount:
${SERVICE_ACCOUNT_NAMESPACE}:${SERVICE_ACCOUNT_AMP_INGEST_NAME}"
        }
      }
    }
  ]
}
```

```
EOF
#
# Set up the permission policy that grants ingest (remote write) permissions for
 all AMP workspaces
#
cat <<EOF > PermissionPolicyIngest.json
{
  "Version": "2012-10-17",
   "Statement": [
       {"Effect": "Allow",
        "Action": [
           "aps:RemoteWrite",
           "aps:GetSeries",
           "aps:GetLabels",
           "aps:GetMetricMetadata"
        ],
        "Resource": "*"
      }
   ]
}
EOF

function getRoleArn() {
  OUTPUT=$(aws iam get-role --role-name $1 --query 'Role.Arn' --output text 2>&1)

  # Check for an expected exception
  if [[ $? -eq 0 ]]; then
    echo $OUTPUT
  elif [[ -n $(grep "NoSuchEntity" <<< $OUTPUT) ]]; then
    echo ""
  else
    >&2 echo $OUTPUT
    return 1
  fi
}

#
# Create the IAM Role for ingest with the above trust policy
#
SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN=$(getRoleArn
 $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE)
if [ "$SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN" = "" ];
then
  #
```

```
  # Create the IAM role for service account
  #
  SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN=$(aws iam create-role \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE \
  --assume-role-policy-document file://TrustPolicy.json \
  --query "Role.Arn" --output text)
  #
  # Create an IAM permission policy
  #
  SERVICE_ACCOUNT_IAM_AMP_INGEST_ARN=$(aws iam create-policy --policy-name
 $SERVICE_ACCOUNT_IAM_AMP_INGEST_POLICY \
  --policy-document file://PermissionPolicyIngest.json \
  --query 'Policy.Arn' --output text)
  #
  # Attach the required IAM policies to the IAM role created above
  #
  aws iam attach-role-policy \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE \
  --policy-arn $SERVICE_ACCOUNT_IAM_AMP_INGEST_ARN
else
    echo "$SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN IAM role for ingest already
 exists"
fi
echo $SERVICE_ACCOUNT_IAM_AMP_INGEST_ROLE_ARN
#
# EKS cluster hosts an OIDC provider with a public discovery endpoint.
# Associate this IdP with AWS IAM so that the latter can validate and accept the
 OIDC tokens issued by Kubernetes to service accounts.
# Doing this with eksctl is the easier and best approach.
#
eksctl utils associate-iam-oidc-provider --cluster $CLUSTER_NAME --approve
```

2. Use o seguinte comando para dar ao script os privilégios necessários.

```
chmod +x createIRSA-AMPIngest.sh
```

3. Executar o script.

Configure perfis do IAM para contas de serviço para consulta de métricas

Para configurar o perfil do IAM para a conta de serviço (perfil de serviço) para permitir a consulta de métricas de workspaces do Amazon Managed Service for Prometheus, você deve estar conectado a uma conta com as seguintes permissões:

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

Para configurar perfis de serviço para a consulta das métricas do Amazon Managed Service for Prometheus;

1. Crie um arquivo chamado `createIRSA-AMPQuery.sh` com o conteúdo a seguir. Substitua <my_amazon_eks_clustername> pelo nome do seu cluster e substitua <my_prometheus_namespace> pelo seu namespace do Prometheus.

```
#!/bin/bash -e
CLUSTER_NAME=<my_amazon_eks_clustername>
SERVICE_ACCOUNT_NAMESPACE=<my_prometheus_namespace>
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)
OIDC_PROVIDER=$(aws eks describe-cluster --name $CLUSTER_NAME --query
 "cluster.identity.oidc.issuer" --output text | sed -e "s/^https:\/\///")
SERVICE_ACCOUNT_AMP_QUERY_NAME=amp-iamproxy-query-service-account
SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE=amp-iamproxy-query-role
SERVICE_ACCOUNT_IAM_AMP_QUERY_POLICY=AMPQueryPolicy
#
# Setup a trust policy designed for a specific combination of K8s service account
 and namespace to sign in from a Kubernetes cluster which hosts the OIDC Idp.
#
cat <<EOF > TrustPolicy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
```

```
        "Federated": "arn:aws:iam::${AWS_ACCOUNT_ID}:oidc-provider/
${OIDC_PROVIDER}"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "${OIDC_PROVIDER}:sub": "system:serviceaccount:
${SERVICE_ACCOUNT_NAMESPACE}:${SERVICE_ACCOUNT_AMP_QUERY_NAME}"
        }
      }
    }
  ]
}
EOF
#
# Set up the permission policy that grants query permissions for all AMP workspaces
#
cat <<EOF > PermissionPolicyQuery.json
{
  "Version": "2012-10-17",
   "Statement": [
       {"Effect": "Allow",
        "Action": [
           "aps:QueryMetrics",
           "aps:GetSeries",
           "aps:GetLabels",
           "aps:GetMetricMetadata"
        ],
        "Resource": "*"
      }
   ]
}
EOF

function getRoleArn() {
  OUTPUT=$(aws iam get-role --role-name $1 --query 'Role.Arn' --output text 2>&1)

  # Check for an expected exception
  if [[ $? -eq 0 ]]; then
    echo $OUTPUT
  elif [[ -n $(grep "NoSuchEntity" <<< $OUTPUT) ]]; then
    echo ""
  else
    >&2 echo $OUTPUT
```

```
    return 1
  fi
}

#
# Create the IAM Role for query with the above trust policy
#
SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN=$(getRoleArn
 $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE)
if [ "$SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN" = "" ];
then
  #
  # Create the IAM role for service account
  #
  SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN=$(aws iam create-role \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE \
  --assume-role-policy-document file://TrustPolicy.json \
  --query "Role.Arn" --output text)
  #
  # Create an IAM permission policy
  #
  SERVICE_ACCOUNT_IAM_AMP_QUERY_ARN=$(aws iam create-policy --policy-name
 $SERVICE_ACCOUNT_IAM_AMP_QUERY_POLICY \
  --policy-document file://PermissionPolicyQuery.json \
  --query 'Policy.Arn' --output text)
  #
  # Attach the required IAM policies to the IAM role create above
  #
  aws iam attach-role-policy \
  --role-name $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE \
  --policy-arn $SERVICE_ACCOUNT_IAM_AMP_QUERY_ARN
else
    echo "$SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN IAM role for query already
 exists"
fi
echo $SERVICE_ACCOUNT_IAM_AMP_QUERY_ROLE_ARN
#
# EKS cluster hosts an OIDC provider with a public discovery endpoint.
# Associate this IdP with AWS IAM so that the latter can validate and accept the
 OIDC tokens issued by Kubernetes to service accounts.
# Doing this with eksctl is the easier and best approach.
#
eksctl utils associate-iam-oidc-provider --cluster $CLUSTER_NAME --approve
```

2. Use o seguinte comando para dar ao script os privilégios necessários.

```
chmod +x createIRSA-AMPQuery.sh
```

3. Executar o script.

Como utilizar o Amazon Managed Service for Prometheus com endpoints da VPC de interface

Se você usa a Amazon Virtual Private Cloud (Amazon VPC) para hospedar seus AWS recursos, você pode estabelecer conexões privadas entre sua VPC e o Amazon Managed Service for Prometheus. Você pode usar essas conexões para habilitar o Amazon Managed Service for Prometheus para se comunicar com os seus recursos no seu VPC sem passar pela Internet pública.

O Amazon VPC é um AWS serviço que você pode usar para lançar AWS recursos em uma rede virtual que você define. Com a VPC, você tem controle sobre as configurações de rede, como o intervalo de endereços IP, sub-redes, tabelas de rotas e gateways de rede. Para conectar a sua VPC ao Amazon Managed Service for Prometheus, você define um endpoint da VPC de interface para conectar a sua VPC aos serviços da AWS . O endpoint fornece uma conectividade confiável e escalável ao Amazon Managed Service for Prometheus sem precisar de um gateway da Internet, instância de conversão de endereços de rede (NAT) ou uma conexão VPN. Para obter mais informações, consulte O que é a Amazon VPC? no Guia do usuário da Amazon VPC.

Os endpoints VPC da Interface são alimentados por AWS PrivateLink uma AWS tecnologia que permite a comunicação privada entre AWS serviços usando uma interface de rede elástica com endereços IP privados. Para obter mais informações, consulte a postagem do blog New — AWS PrivateLink for AWS Services.

As informações a seguir são para os usuários da Amazon VPC. Para obter mais informações sobre como iniciar a Amazon VPC, consulte Conceitos básicos e no Guia do usuário da Amazon VPC.

Criar um endpoint da VPC de interface para o Amazon Managed Service for Prometheus

Crie um endpoint da VPC de interface para começar a usar o Amazon Managed Service for Prometheus. Escolha entre os seguintes endpoints do nome do serviço:

- `com.amazonaws.region.aps-workspaces`

Escolha esse nome de serviço para trabalhar com o Prometheus compatível APIs. Para obter mais informações, consulte Compatível com Prometheus APIs no Guia do usuário do Amazon Managed Service for Prometheus.

- `com.amazonaws.region.aps`

  Escolha este nome de serviço para realizar tarefas de gerenciamento do workspace. Para obter mais informações, consulte Amazon Managed Service for APIs Prometheus no Guia do usuário do Amazon Managed Service for Prometheus.

> **Note**
>
> Se você estiver usando remote_write em uma VPC sem acesso direto à Internet, também deverá criar uma interface para a VPC endpoint, para permitir que o sigv4 funcione por AWS Security Token Service meio do endpoint. Para obter informações sobre como criar um VPC endpoint para AWS STS, consulte Como usar endpoints AWS STS VPC de interface no Guia do usuário.AWS Identity and Access Management Você deve configurar AWS STS para usar endpoints regionalizados.

Para obter mais informações, incluindo step-by-step instruções para criar uma interface VPC endpoint, consulte Criação de um endpoint de interface no Guia do usuário da Amazon VPC.

> **Note**
>
> Você pode usar políticas de endpoint da VPC para controlar o acesso ao seu endpoint de VPC da interface Amazon Managed Service for Prometheus. Consulte a próxima seção para obter mais informações.

Se você criou um endpoint da VPC de interface para o Amazon Managed Service for Prometheus e já tiver o fluxo de dados para os workspaces localizados em sua VPC, as métricas fluirão por meio do endpoint da VPC de interface por padrão. O Amazon Managed Service for Prometheus usa endpoints públicos ou privados da interface (aqueles que estiverem em uso) para realizar essa tarefa.

Controle do acesso ao endpoint da VPC do seu Amazon Managed Service for Prometheus

Você pode usar políticas de endpoint da VPC para controlar o acesso ao seu endpoint de VPC da interface Amazon Managed Service for Prometheus. Uma política de endpoint da VPC é uma política de recursos do IAM que você anexa a um endpoint quando cria ou modifica o endpoint. Se você não associar uma política ao criar um endpoint, a Amazon VPC associará uma política padrão que permita o acesso total ao serviço. Uma política de endpoint não substitui as políticas do IAM nem as políticas fundamentadas na identidade e específicas do serviço. É uma política separada para controlar o acesso do endpoint ao serviço especificado.

Para obter mais informações, consulte Controlar o acesso a serviços com VPC endpoints no Guia do usuário da Amazon VPC.

Veja a seguir um exemplo de política de endpoint do Amazon Managed Service for Prometheus. Essa política permite aos usuários com função `PromUser` se conectarem ao Amazon Managed Service for Prometheus através da VPC para visualizar workspaces e grupos de regras, mas não permite, por exemplo, criar ou excluir workspaces.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonManagedPrometheusPermissions",
            "Effect": "Allow",
            "Action": [
                "aps:DescribeWorkspace",
                "aps:DescribeRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:ListWorkspaces"
            ],
            "Resource": "arn:aws:aps:*:*:/workspaces*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::111122223333:role/PromUser"
                ]
            }
        }
    ]
```

```
}
```

O exemplo a seguir mostra uma política que só permite a efetivação de solicitações provenientes de um endereço IP especificado na VPC estabelecida. Solicitações de outros endereços IP não são aceitas.

```
{
    "Statement": [
        {
            "Action": "aps:*",
            "Effect": "Allow",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:VpcSourceIp": "192.0.2.123"
                },
        "StringEquals": {
                    "aws:SourceVpc": "vpc-555555555555"
                }
            }
        }
    ]
}
```

Solucionar erros do Amazon Managed Service for Prometheus

Use as seções a seguir para solucionar problemas com o Amazon Managed Service for Prometheus.

Tópicos

Erros 429 ou de limite de excedido

Se você ver um erro 429 semelhante ao exemplo a seguir, suas solicitações excederam as cotas de ingestão do Amazon Managed Service for Prometheus.

```
ts=2020-10-29T15:34:41.845Z caller=dedupe.go:112 component=remote level=error
 remote_name=e13b0c
url=http://iamproxy-external.prometheus.uswest2-prod.eks:9090/workspaces/workspace_id/
api/v1/remote_write
msg="non-recoverable error" count=500 err="server returned HTTP status 429
Too Many Requests: ingestion rate limit (6666.666666666667) exceeded while adding 499
 samples and 0 metadata
```

Se você ver um erro 429 semelhante ao exemplo a seguir, suas solicitações excederam a cota do Amazon Managed Service for Prometheus para o número de métricas ativas em um espaço de trabalho.

```
ts=2020-11-05T12:40:33.375Z caller=dedupe.go:112 component=remote level=error
 remote_name=aps
url=http://iamproxy-external.prometheus.uswest2-prod.eks:9090/workspaces/workspace_id/
api/v1/remote_write
msg="non-recoverable error" count=500 err="server returned HTTP status 429 Too Many
 Requests: user=accountid_workspace_id:
```

```
per-user series limit (local limit: 0 global limit: 3000000 actual local limit: 500000)
 exceeded
```

Se você vir um erro 429 semelhante ao exemplo a seguir, suas solicitações excederam a cota do Amazon Managed Service for Prometheus para a taxa (transações por segundo) em que você pode enviar dados para seu espaço de trabalho usando a API `RemoteWrite` compatível com o Prometheus.

```
ts=2024-03-26T16:50:21.780708811Z caller=dedupe.go:112 component=remote level=error
 remote_name=ab123c
url=https://aps-workspaces.us-east-1.amazonaws.com/workspaces/workspace_id/api/v1/
remote_write
msg="non-recoverable error" count=1000 exemplarCount=0 err="server returned HTTP status
 429 Too Many Requests: {\"message\":\"Rate exceeded\"}"
```

Se você vir um erro 400 semelhante ao exemplo a seguir, suas solicitações excederam as cotas do Amazon Managed Service for Prometheus para séries temporais ativas. Para obter detalhes sobre como as cotas de séries temporais ativas são processadas, consulte Cotas padrão de séries ativas..

```
ts=2024-03-26T16:50:21.780708811Z caller=push.go:53 level=warn
url=https://aps-workspaces.us-east-1.amazonaws.com/workspaces/workspace_id/api/v1/
remote_write
msg="non-recoverable error" count=500 exemplarCount=0
err="server returned HTTP status 400 Bad Request: maxFailure (quorum) on a given error
 family, rpc error: code = Code(400)
desc = addr=10.1.41.23:9095 state=ACTIVE zone=us-east-1a, rpc error: code = Code(400)
desc = user=accountid_workspace_id: per-user series limit of 10000000 exceeded,
Capacity from 2,000,000 to 10,000,000 is automatically adjusted based on the last 30
 min of usage.
If throttled above 10,000,000 or in case of incoming surges, please contact
 administrator to raise it.
(local limit: 0 global limit: 10000000 actual local limit: 92879)"
```

Para obter mais informações sobre as Service Quotas do Amazon Managed Service for Prometheus e sobre como solicitar aumentos, consulte Service Quotas do Amazon Managed Service for Prometheus

Vejo amostras duplicadas

Se você estiver usando um grupo Prometheus de alta disponibilidade, precisará usar rótulos externos em suas instâncias do Prometheus para configurar a desduplicação. Para obter mais informações, consulte Eliminar a duplicação de métricas de alta disponibilidade enviadas para o Amazon Managed Service for Prometheus.

Outros problemas relacionados a dados duplicados são discutidos na próxima seção.

Vejo erros sobre carimbos de data/hora de amostra

O Amazon Managed Service for Prometheus ingere dados em ordem e espera que cada amostra tenha um registro de data e hora posterior à amostra anterior.

Se seus dados não chegarem em ordem, você poderá ver erros sobre `out-of-order samples`, `duplicate sample for timestamp` ou `samples with different value but same timestamp`. Esses problemas geralmente são causados pela configuração incorreta do cliente que está enviando dados para o Amazon Managed Service for Prometheus. Se você estiver usando um cliente do Prometheus em execução no modo atendente, verifique a configuração de regras com nome de série duplicado ou destinos duplicados. Se suas métricas fornecerem diretamente o carimbo de data/hora, verifique se elas não estão fora de ordem.

Para obter mais detalhes sobre como isso funciona ou maneiras de verificar sua configuração, consulte a postagem do blog Entendendo amostras duplicadas e erros de registro de Out-of-order data e hora no Prometheus da Prom Labs.

Vejo uma mensagem de erro relacionada a um limite

> **Note**
>
> O Amazon Managed Service para Prometheus CloudWatch fornece métricas de uso para monitorar o uso dos recursos do Prometheus. Usando o recurso de alarme de métricas de CloudWatch uso, você pode monitorar os recursos e o uso do Prometheus para evitar erros de limite.

Se você receber uma das mensagens de erro a seguir, poderá solicitar um aumento em uma das cotas do Amazon Managed Service for Prometheus para resolver o problema. Para obter mais informações, consulte Service Quotas do Amazon Managed Service for Prometheus.

- limite de série por usuário *<value>* excedido, entre em contato com o administrador para aumentá-lo
- limite de série por métrica *<value>* excedido, entre em contato com o administrador para aumentá-lo
- limite de taxa de ingestão (...) excedido
- a série tem muitos rótulos (...) series: '%s'
- o intervalo de tempo de consulta excede o limite (comprimento da consulta: xxx, limite: yyy)
- a consulta atingiu o limite máximo de partes ao buscar partes dos ingestores
- Limite excedido. Máximo de espaços de trabalho por conta.

A saída local do servidor Prometheus excede o limite.

O Amazon Managed Service for Prometheus tem Service Quotas para a quantidade de dados que um espaço de trabalho pode receber dos servidores Prometheus. Para encontrar a quantidade de dados que seu servidor Prometheus está enviando para o Amazon Managed Service for Prometheus, você pode executar as seguintes consultas em seu servidor Prometheus. Se você descobrir que sua produção do Prometheus está excedendo o limite do Amazon Managed Service for Prometheus, você pode solicitar um aumento de Service Quota correspondente. Para obter mais informações, consulte Service Quotas do Amazon Managed Service for Prometheus.

Consultas em seu servidor Prometheus autônomo local para encontrar os limites de saída.

Tipo de dados	Consulta a ser usada	
Séries ativas atuais	`prometheus_tsdb_head_series`	
Taxa de ingestão atual	`rate(prometheus_tsdb_head_samples_ap`	

Tipo de dados	Consulta a ser usada	
	`pended_total[5m])`	
Most-to-least lista de séries ativas por nome de métrica	`sort_desc(count by(__name__)({__name__!=""}))`	
Número de rótulos por série métrica	`group by(mylabelname)({__name__!=""})`	

Alguns dos meus dados não estão aparecendo

Os dados enviados ao Amazon Managed Service for Prometheus podem ser descartados por vários motivos. A tabela a seguir mostra os motivos pelos quais os dados podem ser descartados em vez de serem ingeridos.

Você pode rastrear a quantidade e os motivos pelos quais os dados são descartados usando a Amazon. CloudWatch Para obter mais informações, consulte Use CloudWatch métricas para monitorar os recursos do Amazon Managed Service for Prometheus.

Motivo	Significado
greater_than_max_sample_age	Descarte de linhas de log que são mais antigas do que a hora atual
new-value-for-timestamp	As amostras duplicadas são enviadas com o mesmo carimbo de data/hora da amostra anterior, mas com valores diferentes.

Motivo	Significado
per_metric_series_limit	O usuário atingiu o limite ativo da série por métrica
per_user_series_limit	O usuário atingiu o limite total de séries ativas
rate_limited	Taxa de ingestão limitada
sample-out-of-order	As amostras são enviadas fora de ordem e não podem ser processadas
label_value_too_long	O valor do rótulo é maior do que o limite permitido de caracteres
max_label_names_per_series	O usuário atingiu o limite de nomes dos rótulos por métrica
missing_metric_name	O nome da métrica não foi fornecido
metric_name_invalid	Nome da métrica inválido fornecido
label_invalid	Rótulo inválido fornecido
duplicate_label_names	Nomes de rótulos duplicados fornecidos

Como atribuir tags ao Amazon Managed Service for Prometheus

Uma tag é um rótulo de atributo personalizado que você atribui ou AWS atribui a um AWS recurso. Cada AWS tag tem duas partes:

- Uma chave de tag (por exemplo `CostCenter`, `Environment`, `Project` ou `Secret`). As chaves de tag diferenciam maiúsculas de minúsculas
- Um campo opcional conhecido como um valor de tag (por exemplo, `111122223333`, `Production` ou um nome de equipe). Omitir o valor da tag é o mesmo que usar uma string vazia. Assim como as chaves de tag, os valores de tag diferenciam maiúsculas de minúsculas.

Juntos, esses são conhecidos como pares de chave-valor. Você pode ter até 50 tags atribuídas a cada espaço de trabalho.

As tags ajudam você a 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 um espaço de trabalho do Amazon Managed Service for Prometheus que você atribui a um bucket do Amazon S3. Para ter mais informações sobre estratégias de marcação, consulte Marcar recursos da AWS.

No Amazon Managed Service for Prometheus, os namespaces de espaços de trabalho e grupos de regras podem ser marcados. Você pode usar o console AWS CLI, o APIs, ou SDKs para adicionar, gerenciar e remover tags desses recursos. Além de identificar, organizar e rastrear seus de espaços de trabalho e namespaces de grupos de regras com tags, você pode usar tags em políticas do IAM para ajudar a controlar quem pode visualizar e interagir com seus recursos do Amazon Managed Service for Prometheus.

Restrições de tags

As restrições básicas a seguir se aplicam a tags:

- Cada recurso pode ter um máximo de 50 tags.
- Em todos os recursos, cada chave de tag deve ser exclusiva e pode ter apenas um valor.
- O comprimento máximo da chave da tag é de 128 caracteres Unicode em UTF-8.

- O comprimento máximo do valor da tag é de 256 caracteres Unicode em UTF-8.
- Se seu esquema de marcação for usado em vários AWS serviços e recursos, lembre-se de que outros serviços podem ter restrições quanto aos caracteres permitidos. Os caracteres permitidos são letras, números, espaços representáveis em UTF-8, além dos seguintes caracteres: . : + = @ _ / - (hífen).
- As chaves e valores das tags diferenciam maiúsculas de minúsculas. Como prática recomendada, decida-se sobre uma estratégia para letras maiúsculas em tags e implemente-a de forma consistente em todos os tipos de recursos. Por exemplo, decida se deseja usar `Costcenter`, `costcenter` ou `CostCenter` e use a mesma convenção para todas as tags. Evite usar tags semelhantes com tratamento do tamanho de letra inconsistente.
- Não use `aws:`, `AWS:` ou qualquer combinação de letras maiúsculas e minúsculas como prefixo para chaves ou valores. Eles são reservados apenas para AWS uso. Você não pode editar nem excluir chaves nem valores de tags com esse prefixo. Tags com esse prefixo não contam para o seu tags-per-resource limite.

Tópicos

Atribuir tags ao espaço de trabalho do Amazon Managed Service for Prometheus

Tags são rótulos personalizados que podem ser atribuídos a um recurso. Elas contêm uma chave exclusiva e um valor opcional (em um par de chave/valor). As tags ajudam a identificar e organizar os recursos da AWS . No Amazon Managed Service for Prometheus, é possível atribuir tags a espaços de trabalho (e namespaces de grupos de regras). Você pode usar o console, a AWS CLI ou SDKs adicionar, gerenciar e remover tags desses recursos. APIs Além de identificar, organizar e rastrear seus espaços de trabalho com tags, você pode usar tags em políticas do IAM para ajudar a controlar quem pode visualizar e interagir com seus recursos do Amazon Managed Service for Prometheus.

Use os procedimentos desta seção para trabalhar com tags para espaços de trabalho do Amazon Managed Service for Prometheus.

Tópicos

- Visualização de tags de um espaço de trabalho
- Editar tags para um espaço de trabalho
- Remova uma tag de um espaço de trabalho

Adicionar uma tag a um espaço de trabalho

Adicionar tags a um espaço de trabalho do Amazon Managed Service for Prometheus pode ajudar a identificar e organizar seus recursos da AWS e gerenciar o acesso a eles. Primeiro, adicione uma ou mais tags (pares de chave/valor) a um projeto. Depois que tiver tags, você poderá criar políticas do IAM para gerenciar o acesso ao espaço de trabalho com base nessas tags. Você pode usar o console ou o AWS CLI para adicionar tags a um espaço de trabalho do Amazon Managed Service for Prometheus.

> **Important**
>
> Adicionar tags a um espaço de trabalho pode afetar o acesso a esse espaço de trabalho. Antes de adicionar uma tag a um grupo de relatórios, revise as políticas do IAM que possam usar tags para controlar o acesso a recursos, como grupo de relatórios.

Para obter mais informações sobre como adicionar tags a um espaço de trabalho do Amazon Managed Service for Prometheus ao criá-lo, consulte Criar um espaço de trabalho do Amazon Managed Service for Prometheus.

Tópicos

- Adicionar uma tag a um espaço de trabalho (console)
- Adicionar uma tag a um espaço de trabalho (AWS CLI)

Adicionar uma tag a um espaço de trabalho (console)

Você pode usar o console para adicionar uma ou mais tags a um espaço de trabalho do Amazon Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.

3. Escolha Todos os espaços de trabalho.
4. Escolha o ID de espaço de trabalho do espaço de trabalho que você quiser gerenciar.
5. Escolha a guia Tags.
6. Se nenhuma tag tiver sido adicionada ao espaço de trabalho do Amazon Managed Service for Prometheus, escolha Create tag. Caso contrário, escolha Gerenciar tags.
7. Em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.
8. (Opcional) Para adicionar outra tag, selecione Adicionar tag novamente.
9. Quando terminar de adicionar tags, escolha Salvar alterações.

Adicionar uma tag a um espaço de trabalho (AWS CLI)

Siga estas etapas para usar o AWS CLI para adicionar uma tag a um espaço de trabalho do Amazon Managed Service for Prometheus. Para adicionar uma tag a um pipeline ao criá-lo, consulte Criar um espaço de trabalho do Amazon Managed Service for Prometheus.

Nessas etapas, presumimos que você já tenha instalado uma versão recente do AWS CLI ou atualizado para a versão atual. Para obter mais informações, consulte Instalar a AWS Command Line Interface.

No terminal ou na linha de comando, execute o comando tag-resource, especificando o nome do recurso da Amazon (ARN) do espaço de trabalho no qual você deseja adicionar tags e a chave e o valor da tag que você deseja adicionar. Você pode adicionar mais de uma tag a um espaço de trabalho. Por exemplo, para marcar um espaço de trabalho do Amazon Managed Service para Prometheus chamado My-Workspace com duas tags, uma chave de tag com o valor de tag de e uma chave *Status* de tag com o valor *Secret* de tag de: *Team My-Team*

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspaces/IDstring
--tags Status=Secret,Team=My-Team
```

Se houver êxito, o comando não retornará nada.

Visualização de tags de um espaço de trabalho

As tags podem ajudar você a identificar e organizar seus AWS recursos e gerenciar o acesso a eles. Para obter mais informações sobre estratégias de marcação, consulte Recursos de marcação AWS.

Exibir tags para um espaço de trabalho do Amazon Managed Service for Prometheus (console)

Você pode usar o console para visualizar as tags associadas a um espaço de trabalho do Prometheus Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID de espaço de trabalho do espaço de trabalho que você quiser gerenciar.
5. Escolha a guia Tags.

Exibir tags para um espaço de trabalho do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para visualizar as AWS tags de um espaço de trabalho. Se não foram adicionadas tags, a lista retornará vazia.

No terminal ou na linha de comando, execute o comando list-tags-for-resource. Por exemplo, para visualizar uma lista de chaves de tag e valores de tag para um espaço de trabalho:

```
aws amp list-tags-for-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
```

Se houver êxito, o comando retornará informações semelhantes às seguintes:

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

Editar tags para um espaço de trabalho

É possível alterar o valor de uma tag associada a um projeto. Também é possível alterar o nome da chave, o que é equivalente a excluir a tag atual e adicionar outra com o novo nome e o mesmo valor da outra chave.

> **Important**
>
> A edição de tags de um espaço de trabalho do Amazon Managed Service for Prometheus pode afetar o acesso a esse espaço de trabalho. Antes de editar o nome (chave) ou o valor de uma tag de um repositório, revise as políticas do IAM que podem usar essa chave ou esse valor para uma tag a fim de controlar o acesso a recursos, como repositórios.

Editar tags para um espaço de trabalho do Amazon Managed Service for Prometheus (console)

Você pode usar o console para visualizar as tags associadas a um espaço de trabalho do Amazon Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID de espaço de trabalho do espaço de trabalho que você quiser gerenciar.
5. Escolha a guia Tags.
6. Se nenhuma tag tiver sido adicionada ao grupo de relatórios, selecione Adicionar tag. Caso contrário, escolha Gerenciar tags.
7. Em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.
8. (Opcional) Para adicionar outra tag, selecione Adicionar tag novamente.
9. Quando terminar de adicionar tags, escolha Salvar alterações.

Editar tags para um espaço de trabalho do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para atualizar uma tag para um espaço de trabalho. Você pode alterar o valor para uma chave existente ou adicionar outra chave.

No terminal ou na linha de comando, execute o comando tag-resource, especificando o nome do recurso da Amazon (ARN) do espaço de trabalho do Amazon Managed Service for Prometheus onde você deseja atualizar uma tag e especifique a chave e o valor da tag:

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tags Team=New-Team
```

Remova uma tag de um espaço de trabalho

É possível remover uma ou mais tags associadas a um projeto. A remoção de uma tag não exclui a tag de outros AWS recursos associados a essa tag.

> **Important**
>
> A remoção de tags de um espaço de trabalho do Amazon Managed Service for Prometheus pode afetar o acesso a esse espaço de trabalho. Antes de excluir uma tag de um espaço de trabalho, revise as políticas do IAM que podem usar a chave ou o valor para uma tag a fim de controlar o acesso a recursos, como repositórios.

Remover tags de um espaço de trabalho do Amazon Managed Service for Prometheus (console)

É possível usar o console para remover a associação entre uma tag e um espaço de trabalho.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID de espaço de trabalho do espaço de trabalho que você quiser gerenciar.
5. Escolha a guia Tags.
6. Selecione Gerenciar tags.

7. Encontre a tag que você deseja excluir e selecione Remover.

Remover tags de um espaço de trabalho do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para remover uma tag de um espaço de trabalho. Remover uma tag não a exclui, apenas remove a associação entre a tag e o espaço de trabalho.

> **Note**
>
> Se você excluir um espaço de trabalho do Amazon Managed Service for Prometheus, todas as associações de tags serão removidas do espaço de trabalho excluído. Você não precisa remover as tags antes de excluir um espaço de trabalho.

No terminal ou na linha de comando, execute o comando untag-resource, especificando o nome do recurso da Amazon (ARN) do espaço de trabalho no qual você deseja remover tags e a chave da tag que você deseja remover. Por exemplo, para remover uma tag em um espaço de trabalho chamado My-Workspace com a chave de tag: *Status*

```
aws amp untag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tag-keys Status
```

Se houver êxito, o comando não retornará nada. Para verificar as tags associadas ao espaço de trabalho, execute o comando list-tags-for-resource.

Marcação de namespaces de grupos de regras

Tags são rótulos personalizados que podem ser atribuídos a um recurso. Elas contêm uma chave exclusiva e um valor opcional (em um par de chave/valor). As tags ajudam a identificar e organizar os recursos da AWS . No Amazon Managed Service for Prometheus, é possível atribuir tags a namespaces de grupos de regras (e espaços de trabalho). Você pode usar o console, a AWS CLI ou SDKs adicionar, gerenciar e remover tags desses recursos. APIs Além de identificar, organizar e rastrear seus namespaces de grupos de regras com tags, você pode usar tags em políticas do IAM para ajudar a controlar quem pode visualizar e interagir com seus recursos do Amazon Managed Service for Prometheus.

Use os procedimentos desta seção para trabalhar com tags para namespaces de grupos de regras do Amazon Managed Service for Prometheus.

Tópicos

Adicionar uma tag a um namespace de grupos de regras

Adicionar tags a um namespace de grupos de regras do Amazon Managed Service for Prometheus pode ajudar você a identificar e organizar AWS seus recursos e gerenciar o acesso a eles. Primeiro, adicione uma ou mais tags (pares chave/valor) a um grupo de relatórios. Depois que tiver tags, você poderá criar políticas do IAM para gerenciar o acesso ao namespace com base nessas tags. Você pode usar o console ou o AWS CLI para adicionar tags a um namespace de grupos de regras do Amazon Managed Service for Prometheus.

> **Important**
>
> Adicionar tags a um namespace de grupos de regras pode afetar o acesso a esse namespace de grupos de regras. Antes de adicionar uma tag a um repositório, revise as políticas do IAM que possam usar tags para controlar o acesso a recursos, como projetos de compilação.

Para obter mais informações sobre como adicionar tags a um grupo de relatórios ao criá-lo, consulte Criar um arquivo de regras.

Tópicos

Adicionar uma tag a um namespace de grupos de regras (console)

Você pode usar o console para adicionar uma ou mais tags a um namespace de grupos de regras do Amazon Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID do workspace do workspace que você quiser gerenciar.
5. Escolha a guia Gerenciamento de regras.
6. Escolha o botão ao lado do nome do namespace de nomes e selecione Editar.
7. Selecione Criar tags, Adicionar nova tag.
8. Em Chave, insira um nome para a tag. É possível adicionar um valor opcional para a tag em Valor.
9. (Opcional) Para adicionar outra tag, selecione Adicionar tag novamente.
10. Quando terminar de adicionar tags, escolha Salvar alterações.

Adicionar uma tag a um namespace de grupos de regras (AWS CLI)

Siga estas etapas para usar o AWS CLI para adicionar uma tag a um namespace de grupos de regras do Amazon Managed Service for Prometheus. Para adicionar uma tag a um namespace de grupos de regras ao criá-la, consulte Carregar um arquivo de configuração de regras no Amazon Managed Service for Prometheus.

Nessas etapas, presumimos que você já tenha instalado uma versão recente do AWS CLI ou atualizado para a versão atual. Para obter mais informações, consulte Instalar a AWS Command Line Interface.

No terminal ou na linha de comando, execute o comando tag-resource, especificando o nome do recurso da Amazon (ARN) do pipeline no qual você deseja adicionar tags e a chave e o valor da tag que você deseja adicionar. Você pode adicionar mais de uma tag a um namespace de grupos de regras. Por exemplo, para marcar um namespace do Amazon Managed Service for Prometheus chamado My-Workspace com duas tags, uma chave de tag com o valor de tag de e uma chave `Status` de tag com o valor `Secret` de tag de: `Team My-Team`

```
aws amp tag-resource \
    --resource-arn arn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name \
    --tags Status=Secret,Team=My-Team
```

Se houver êxito, o comando não retornará nada.

Visualização de tags de um namespace de grupos de regras

As tags podem ajudar você a identificar e organizar seus AWS recursos e gerenciar o acesso a eles. Para obter mais informações sobre estratégias de marcação, consulte Recursos de marcação AWS.

Exibir tags para um namespace de grupos de regras do Amazon Managed Service for Prometheus (console)

Você pode usar o console para visualizar as tags associadas a um namespace de grupos de regras do Amazon Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID do workspace do workspace que você quiser gerenciar.
5. Escolha a guia Gerenciamento de regras.
6. Selecione o nome do namespace.

Exibir tags para um espaço de trabalho do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para visualizar as AWS tags de um namespace de grupos de regras. Se não foram adicionadas tags, a lista retornará vazia.

No terminal ou na linha de comando, execute o comando list-tags-for-resource. Por exemplo, para visualizar uma lista de chaves e valores de tag para um namespace de grupos de regras:

```
aws amp list-tags-for-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name
```

Se houver êxito, o comando retornará informações semelhantes às seguintes:

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

Editar tags para um namespace de grupos de regras

É possível alterar o valor de uma tag associada a um grupo de relatórios. Também é possível alterar o nome da chave, o que é equivalente a excluir a tag atual e adicionar outra com o novo nome e o mesmo valor da outra chave.

> **Important**
>
> A edição de tags para um namespace de grupos de regras pode afetar o acesso a ele. Antes de editar o nome (chave) ou o valor de uma tag de um grupo de relatórios, revise as políticas do IAM que podem usar essa chave ou esse valor para uma tag a fim de controlar o acesso a recursos, como grupo de relatórios.

Editar tags para um namespace de grupos de regras do Amazon Managed Service for Prometheus (console)

Você pode usar o console para editar as tags associadas a um namespace de grupos de regras do Amazon Managed Service for Prometheus.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/
2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID do workspace do workspace que você quiser gerenciar.
5. Escolha a guia Gerenciamento de regras.
6. Escolha o nome do namespace.
7. Escolha Gerenciar e Adicionar nova tag.

8. Para alterar o valor de uma tag existente, insira o novo valor para Value.
9. Para adicionar mais tags, selecione Adicionar nova tag.
10. Quando terminar de adicionar e editar tags, escolha Salvar alterações.

Editar tags para um namespace de grupos de regras do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para atualizar uma tag para um namespace de grupos de regras. Você pode alterar o valor para uma chave existente ou adicionar outra chave.

No terminal ou na linha de comando, execute o comando tag-resource, especificando o nome do recurso da Amazon (ARN) do repositório em que você deseja atualizar uma tag e especifique a chave e o valor da tag:

```
aws amp tag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tags Team=New-Team
```

Remova uma tag de um namespace de grupos de regras

É possível excluir uma ou mais tags associadas a um grupo de relatórios. A remoção de uma tag não exclui a tag de outros AWS recursos associados a essa tag.

> **Important**
>
> A remoção de tags de um recurso pode afetar o acesso a esse recurso. Antes de excluir uma tag de um recurso, revise as políticas do IAM que podem usar a chave ou o valor para uma tag a fim de controlar o acesso a recursos, como repositórios.

Remover tags de um namespace de grupos de regras do Amazon Managed Service for Prometheus (console)

É possível usar o console para remover a associação entre uma tag e um grupo de relatórios do namespace.

1. Abra o console do Amazon Managed Service for Prometheus em. https://console.aws.amazon.com/prometheus/

2. No painel de navegação, escolha o ícone de calendário.
3. Escolha Todos os espaços de trabalho.
4. Escolha o ID do workspace do workspace que você quiser gerenciar.
5. Escolha a guia Gerenciamento de regras.
6. Escolha o nome do namespace.
7. Selecione Gerenciar tags.
8. Ao lado da tag que você deseja excluir e selecione Remover.
9. Ao terminar, selecione Salvar alterações.

Remover uma tag de um namespace de grupos de regras do Amazon Managed Service for Prometheus (AWS CLI)

Siga estas etapas para usar o AWS CLI para remover uma tag de um namespace de grupos de regras. Remover uma tag não a exclui, apenas remove a associação entre a tag e o namespace do grupo de regras.

> **Note**
>
> Se você excluir um namespace de grupos de regras do Amazon Managed Service for Prometheus, todas as associações de tags serão removidas do namespace excluído. Você não precisa remover as tags antes de excluir um namespace.

No terminal ou na linha de comando, execute o comando untag-resource, especificando o nome do recurso da Amazon (ARN) do namespace de grupos de regras no qual você deseja remover tags e a chave da tag que você deseja remover. Por exemplo, para remover uma tag em um espaço de trabalho chamado My-Workspace com a chave de tag: *Status*

```
aws amp untag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tag-keys Status
```

Se houver êxito, o comando não retornará nada. Para verificar as tags associadas ao recurso, execute o comando list-tags-for-resource.

Service Quotas do Amazon Managed Service for Prometheus

As duas seções a seguir descrevem as cotas e os limites associados ao Amazon Managed Service for Prometheus.

Cotas de serviço

O Amazon Managed Service for Prometheus tem as cotas a seguir. O Amazon Managed Service for Prometheus vende métricas de uso para monitorar o CloudWatch uso dos recursos do Prometheus. Usando o recurso de alarme CloudWatch de métricas de uso da Amazon, você pode monitorar os recursos e o uso do Prometheus para evitar erros de limite.

À medida que seus projetos e espaços de trabalho crescem, as cotas mais comuns que você deve monitorar ou solicitar aumento são: séries ativas por espaço de trabalho e taxa de ingestão por espaço de trabalho.

Para todas as cotas ajustáveis, você pode solicitar um aumento de cota selecionando o link na coluna Ajustável ou solicitando um aumento de cota.

O limite da série ativa por espaço de trabalho é aplicado dinamicamente. Para obter mais informações, consulte Cotas padrão de séries ativas.. A cota da taxa de ingestão por espaço de trabalho determina a rapidez com que você pode ingerir dados no seu espaço de trabalho. Para obter mais informações, consulte Controle de utilização da ingestão.

> **Note**
>
> Salvo indicação em contrário, essas cotas são por espaço de trabalho. O valor máximo para séries ativas por espaço de trabalho é de um bilhão.

Nome	Padrão	Ajustá	Description
Métricas ativas com metadados por espaço de trabalho	Cada região com suporte: 20.000	Não	O número de métricas ativas exclusivas com metadados por espaço

Nome	Padrão	Ajustá	Description
			de trabalho. Observaçã o: se o limite for atingido, a amostra da métrica será registrada, mas os metadados acima do limite serão descartados.
Série ativa por espaço de trabalho	Cada região com suporte: 50.000.00 0	Sim	O número de séries ativas exclusivas por espaço de trabalho (até um máximo de 1 bilhão). Uma série está ativa se uma amostra tiver sido relatada nas últimas 2 horas. A capacidade de 2 M a 50 M é ajustada automaticamente com base nos últimos 30 minutos de uso.
Tamanho do grupo de agregação de alertas no arquivo de definição do gerenciador de alertas	Cada região com suporte: 1.000	Sim	Tamanho máximo do grupo de agregação de alertas no arquivo de definição do gerenciad or de alertas. Cada combinação de valores de rótulo de group_by cria um grupo de agregação.
Tamanho do arquivo de definição do gerenciador de alertas	Cada região com suporte: 1.000.000	Não	Tamanho máximo de um arquivo de definição do gerenciador de alertas, em bytes.

Nome	Padrão	Ajustá	Description
Tamanho da carga útil do alerta no gerenciador de alertas	Cada região com suporte: 20.000.000	Não	O tamanho máximo da carga útil de todos os alertas do Alert Manager por espaço de trabalho, em bytes. O tamanho do alerta depende dos rótulos e das anotações.
Alertas no gerenciador de alertas	Cada região com suporte: 1.000	Sim	O número máximo de alertas simultâneos do gerenciador de alertas por espaço de trabalho.
Clusters de rastreadores HA	Cada região com suporte: 500	Não	O número máximo de clusters que o rastreador HA rastreará para amostras ingeridas por espaço de trabalho.
Taxa de ingestão por espaço de trabalho	Cada região suportada: 1.666.666	Sim	Taxa métrica de ingestão de amostras por espaço de trabalho por segundo. O limite é ajustado automaticamente para ser 1/30 da série ativa por limite de espaço de trabalho, até 1.666.666.
Regras de inibição no arquivo de definição do gerenciador de alertas	Cada região com suporte: 100	Sim	Número máximo de regras de inibição no arquivo de definição do gerenciador de alertas.

Nome	Padrão	Ajustá	Description
Tamanho do label	Cada região compatível: 7	Não	O tamanho máximo combinado de todas as etiquetas e valores de etiquetas aceitos para uma série, em quilobytes.
LabelSet limites por espaço de trabalho	Cada região compatível: 100	Sim	O número máximo de limites do conjunto de etiquetas que podem ser criados por espaço de trabalho.
Rótulos por série métrica	Cada região com suporte: 150	Sim	Número de rótulos por série métrica.
Tamanho dos metadados	Cada região compatível: 1	Não	O comprimento máximo aceito para metadados de métricas, em quilobytes. Os metadados são referentes a nome da métrica, tipo, unidade e texto de ajuda.
Metadados por métrica	Cada região com suporte: 10	Não	O número de metadados por métrica. Observação: se o limite for atingido, a amostra da métrica será registrada, mas os metadados acima do limite serão descartados.
Nós na árvore de roteamento do gerenciador de alertas	Cada região com suporte: 100	Sim	O número máximo de nós na árvore de roteamento do gerenciador de alertas.

Nome	Padrão	Ajustá	Description
Número de operações de API por região em transações por segundo	Cada região com suporte: 10	Sim	Número máximo de operações de API por segundo por região para todas as APIs do Amazon Managed Service para Prometheus, incluindo APIs CRUD de espaço de trabalho, APIs de marcação, APIs CRUD de namespace de grupos de regras e APIs CRUD de definição de gerenciador de alertas.
Número GetLabels e operações GetSeries de GetMetricMetadata API por espaço de trabalho em transações por segundo	Cada região com suporte: 10	Não	O número máximo de operações GetSeries de GetMetricMetadata Prometheus-compatible API GetLabels e de API por segundo por espaço de trabalho.
Número de operações de QueryMetrics API por espaço de trabalho em transações por segundo	Cada região com suporte: 300	Não	O número máximo de operações de QueryMetrics Prometheus-compatible API por segundo por espaço de trabalho.
Número de operações de RemoteWrite API por espaço de trabalho em transações por segundo	Cada região com suporte: 3.000	Não	O número máximo de operações de RemoteWrite Prometheus-compatible API por segundo por espaço de trabalho.

Nome	Padrão	Ajustá	Description
Número de outras operações de Prometheus-compatible API por espaço de trabalho em transações por segundo	Cada região compatível: 100	Não	O número máximo de operações de API por segundo por espaço de trabalho para todas as outras Prometheus-compatible APIs ListAlerts, incluindo ListRules, etc.
Taxa de ingestão fora de ordem por espaço de trabalho	Cada região suportada: 83.333	Sim	Taxa de ingestão de amostras fora de ordem por espaço de trabalho por segundo. A menos que seja substituído, o limite é ajustado automaticamente para ser de 5% da taxa de ingestão por limite de espaço de trabalho.
Janela de tempo fora de serviço por espaço de trabalho	Cada região compatível: 600	Sim	A janela de tempo máxima para amostras fora de ordem por espaço de trabalho, em segundos.
Bytes de consulta para consultas instantâneas	Cada região compatível: 5	Não	O máximo de bytes que podem ser analisados por uma única consulta instantânea, em gigabytes.

Nome	Padrão	Ajustá	Description
Bytes de consulta para consultas de intervalo	Cada região compatível: 5	Não	O máximo de bytes que podem ser verificados por intervalo de 24 horas em uma única consulta de intervalo, em gigabytes.
Consultas de exemplo	Cada região com suporte: 50.000.000	Não	O número máximo de amostras que podem ser digitalizadas por intervalo de 24 horas em uma única consulta de intervalo ou em uma única consulta instantânea.
Série de consultas obtida	Cada região com suporte: 12.000.000	Não	O número máximo de séries que podem ser verificadas por intervalo de 24 horas em uma única consulta de intervalo ou em uma única consulta instantânea.
Intervalo de tempo de consulta em dias	Cada região suportada: 95	Não	O intervalo máximo de tempo de QueryMetrics, GetSeries, e GetLabels APIs.
Dimensão da solicitação	Cada região compatível: 1	Não	O tamanho máximo da solicitação para ingestão ou consulta, em megabytes.

Nome	Padrão	Ajustá	Description
Intervalo de avaliação da regra	Cada região compatível: 30	Sim	O intervalo mínimo de avaliação de regras de um grupo de regras por espaço de trabalho, em segundos.
Tamanho do arquivo de definição do namespace do grupo de regras	Cada região com suporte: 1.000.000	Não	O tamanho máximo de um arquivo de definição de namespace de grupo de regras, em bytes.
Regras por espaço de trabalho	Cada região com suporte: 2.000	Sim	O número máximo de regras por WorkSpace.
Silêncios por espaço de trabalho	Cada região com suporte: 1.000	Sim	Número máximo de silêncios, incluindo silêncios expirados, ativos e pendentes, por espaço de trabalho.
Modelos no arquivo de definição do gerenciador de alertas	Cada região com suporte: 100	Sim	O número máximo de modelos no arquivo de definição do gerenciador de alertas.
Espaços de trabalho por região por conta	Cada região com suporte: 25	Sim	O número máximo de tags por espaços de trabalho.

Cotas padrão de séries ativas.

Os espaços de trabalho do Amazon Managed Service for Prometheus se adaptam automaticamente ao seu uso de ingestão. À medida que seu uso aumenta, o serviço aumenta automaticamente sua capacidade de séries temporais até o limite padrão.

Seu espaço de trabalho do Amazon Managed Service for Prometheus é escalado automaticamente, com base no seu uso, de duas maneiras:

1. Quando seu uso médio de 30 minutos está abaixo de 5 milhões de séries, a capacidade dobra (por exemplo, um espaço de trabalho com 3,5 milhões de uso obtém 7 milhões de capacidade).
2. Quando o uso excede 5 milhões de séries, o espaço de trabalho adiciona um buffer de 10 milhões (por exemplo, um espaço de trabalho com 25 milhões de uso obtém 35 milhões de capacidade).

O Amazon Managed Service for Prometheus aloca automaticamente mais capacidade à medida que seu consumo aumenta, até sua cota. Isso ajuda a garantir que sua workload não sofra controle de utilização contínuo. No entanto, pode ocorrer controle de utilização se você dobrar ou exceder em 10 milhões o seu valor de referência anterior calculado nos últimos 30 minutos. Para evitar controle de utilização, o Amazon Managed Service for Prometheus recomenda aumentar a ingestão gradualmente ao ultrapassar seu baseline anterior.

> **Note**
>
> A capacidade mínima para séries temporais ativas é de 2 milhões; e não há controle de utilização quando você tem menos de 2 milhões de séries.
> Para ir além de sua cota padrão, solicite um aumento de cota.

Escalar acima da cota padrão

Quando você solicita um aumento de cota acima da cota padrão da série ativa, o Amazon Managed Service for Prometheus ajusta a capacidade do seu espaço de trabalho adequadamente. Se você não utilizar totalmente a capacidade aumentada, o serviço recuperará a parte não usada ao longo do tempo. Conforme seu uso aumenta, o espaço de trabalho aumentará a escala verticalmente novamente de modo automático.

No entanto, pode ocorrer controle de utilização se você mais do que dobrar ou exceder 50 milhões de séries temporais ativas dentro de seu pico anterior dentro de 2 horas. Por exemplo:

- Se sua cota for de 100 milhões e sua linha de base for de 30 milhões, você poderá aumentar a escala verticalmente até 60 milhões em 2 horas sem controle de utilização.
- Se sua cota for de 100 milhões e sua linha de base for de 50 milhões, você pode aumentar a escala verticalmente até os 100 milhões completos dentro de 2 horas sem controle de utilização.

Controle de utilização da ingestão

O Amazon Managed Service for Prometheus tem controle de utilização da ingestão em cada espaço de trabalho, com base nos seus limites atuais. Isso ajuda a manter o desempenho dos espaços de trabalho. Se você exceder o limite, você verá `DiscardedSamples` nas CloudWatch métricas (com o `rate_limited` motivo). Você pode usar CloudWatch para monitorar sua ingestão e criar um alarme para avisá-lo quando estiver perto de atingir os limites de limitação. Para obter mais informações, consulte Use CloudWatch métricas para monitorar os recursos do Amazon Managed Service for Prometheus.

O Amazon Managed Service for Prometheus usa o algoritmo do bucket de tokens para implementar o controle de utilização da ingestão. Com esse algoritmo, sua conta tem um bucket que contém um número específico de tokens. O número de tokens no bucket representa seu limite de ingestão em qualquer segundo.

Cada amostra de dados ingerida remove um token do bucket. Se o tamanho do seu bucket (Taxa de ingestão por espaço de trabalho) for 1.000.000, o seu espaço de trabalho poderá ingerir um milhão de amostras de dados em um segundo. Se ele exceder um milhão de amostras para ingestão, o controle de utilização será aplicado e nenhum outro registro será ingerido. As amostras de dados adicionais serão descartadas.

O bucket será recarregado automaticamente a uma taxa definida. Se o bucket ficar abaixo da capacidade máxima, um determinado número de tokens será adicionado novamente a ele a cada segundo até atingir sua capacidade máxima. Se o bucket estiver cheio quando os tokens de recarga forem adicionados, eles serão descartados. O bucket não pode conter mais do que seu número máximo de tokens. A taxa de recarga para amostras de ingestão é definida pelo limite da taxa de ingestão por espaço de trabalho. Se sua taxa de ingestão por espaço de trabalho estiver definida como 170.000, a taxa de recarga do bucket será de 170.000 tokens por segundo.

Se seu espaço de trabalho ingerir 1.000.000 de amostras de dados por segundo, seu bucket será imediatamente reduzido para zero token. O bucket será então recarregado com 170.000 tokens a cada segundo até atingir sua capacidade máxima de 1.000.000 tokens. Se não houver mais ingestão, o bucket anteriormente vazio retornará à sua capacidade máxima em 6 segundos.

> **Note**
>
> A ingestão ocorre via solicitações em lote. Se você tiver 100 tokens disponíveis e enviar uma solicitação com 101 amostras, a solicitação inteira será rejeitada. O Amazon Managed

Service for Prometheus não aceita solicitações parcialmente. Se você estiver gravando um coletor, poderá gerenciar novas tentativas (com lotes menores ou após algum tempo).

Para que o bucket possa ingerir mais amostras de dados, você não precisa esperar que o bucket esteja cheio. Você pode usar tokens à medida que eles são adicionados ao bucket. Se você usar imediatamente os tokens de recarga, o bucket não atingirá sua capacidade máxima. Por exemplo, se você esgotar o bucket, poderá continuar ingerindo 170.000 amostras de dados por segundo. O bucket poderá ser recarregado até a capacidade máxima somente se você ingerir menos de 170.000 amostras de dados por segundo.

Limites adicionais para dados ingeridos

O Amazon Managed Service for Prometheus também tem os seguintes requisitos adicionais para ingestão de dados no espaço de trabalho. Eles não são ajustáveis.

- Amostras métricas com mais de 1 hora não podem ser ingeridas.
- Cada amostra e metadado deve ter um nome de métrica.

Referência de API do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus oferece dois tipos de APIs:

1. APIs do Amazon Managed Service para Prometheus: essas APIs permitem que você crie e gerencie seus espaços de trabalho do Amazon Managed Service for Prometheus, incluindo operações para espaços de trabalho, extratores, definições do gerenciador de alertas, namespaces de grupos de regras e registros em log. Você usa os SDKs da AWS, disponíveis para várias linguagens de programação, para interagir com essas APIs.
2. APIs compatíveis com o Prometheus: o Amazon Managed Service for Prometheus aceita APIs HTTP compatíveis com o Prometheus. Essas APIs permitem criar aplicativos personalizados, automatizar fluxos de trabalho, integrar-se a outros serviços ou ferramentas e consultar e interagir com seus dados de monitoramento usando a linguagem de consulta do Prometheus (PromQL).

Esta seção lista as operações de API e as estruturas de dados suportadas pelo Amazon Managed Service for Prometheus.

Para obter informações sobre cotas para as séries, os rótulos e as solicitações de API, consulte Service Quotas do Amazon Managed Service for Prometheus no Guia do usuário do Amazon Managed Service for Prometheus.

Tópicos

APIs do Amazon Managed Service for Prometheus

O Amazon Managed Service for Prometheus oferece operações de API criando e mantendo seus espaços de trabalho do Amazon Managed Service for Prometheus. Isso inclui APIs para espaços de trabalho, extratores, definições do gerenciador de alertas, namespaces de grupos de regras e registros em log.

Para obter informações detalhadas sobre as APIs do Amazon Managed Service for Prometheus, consulte a Referência de API do Amazon Managed Service for Prometheus.

Como usar o Amazon Managed Service for Prometheus com um SDK da AWS

Os kits de desenvolvimento de software (software development kits, ou SDKs) AWS estão disponíveis em muitas linguagens de programação populares. Cada SDK fornece uma API, exemplos de código e documentação que facilitam a criação de aplicações AWS em seu idioma preferido pelos desenvolvedores. Para ver uma lista de SDKs e ferramentas por linguagem, consulte FTools to Build on AWS no Centro do Desenvolvedor da AWS.

> **Versões do SDK**
>
> Recomendamos que você use a versão mais recente do AWS SDK e quaisquer outros SDKs usados em seus projetos e que mantenha os SDKs atualizados. O AWS SDK fornece os atributos e funcionalidades mais recentes, além de atualizações de segurança.

Compatível com Prometheus APIs

O Amazon Managed Service para Prometheus é compatível com o seguinte Prometheus. APIs

Para obter mais informações sobre como usar o compatível com Prometheus APIs, consulte. Consulta usando Prometheus-compatible APIs

Tópicos

- CreateAlertManagerAlerts
- DeleteAlertManagerSilence
- GetAlertManagerStatus
- GetAlertManagerSilence
- GetLabels
- GetMetricMetadata
- GetSeries
- ListAlerts
- ListAlertManagerAlerts
- ListAlertManagerAlertGroups

- ListAlertManagerReceivers
- ListAlertManagerSilences
- ListRules
- PutAlertManagerSilences
- QueryMetrics
- RemoteWrite

CreateAlertManagerAlerts

A operação `CreateAlertManagerAlerts` cria um alerta no espaço de trabalho.

Verbos HTTP válidos:

`POST`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

Parâmetros da consulta de URL:

`alerts` Uma matriz de objetos, em que cada objeto representa um alerta. Veja a seguir um exemplo de um caminho de objeto alerta:

```
[
  {
    "startsAt": "2021-09-24T17:14:04.995Z",
    "endsAt": "2021-09-24T17:14:04.995Z",
    "annotations": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "labels": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "generatorURL": "string"
  }
```

```
]
```

Exemplo de solicitação

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts
 HTTP/1.1
Content-Length: 203,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0

[
  {
    "labels": {
      "alertname": "test-alert"
    },
    "annotations": {
      "summary": "this is a test alert used for demo purposes"
    },
    "generatorURL": "https://www.amazon.com/"
  }
]
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin
```

DeleteAlertManagerSilence

O `DeleteSilence` exclui um silêncio de alerta.

Verbos HTTP válidos:

`DELETE`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/silence/silenceID`

Parâmetros de consulta de URL: nenhum

Exemplo de solicitação

```
DELETE /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silence/
d29d9df3-9125-4441-912c-70b05f86f973 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin
```

GetAlertManagerStatus

O `GetAlertManagerStatus` recupera informações sobre o status do gerenciador de alertas.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/status`

Parâmetros de consulta de URL: nenhum

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/status
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 941
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "cluster": null,
    "config": {
        "original": "global:\n  resolve_timeout: 5m\n  http_config:\n
 follow_redirects: true\n  smtp_hello: localhost\n  smtp_require_tls: true\nroute:
\n  receiver: sns-0\n  group_by:\n  - label\n  continue: false\nreceivers:\n-
 name: sns-0\n  sns_configs:\n  - send_resolved: false\n    http_config:\n
      follow_redirects: true\n    sigv4: {}\n    topic_arn: arn:aws:sns:us-
west-2:123456789012:test\n    subject: '{{ template \"sns.default.subject\" . }}'\n
    message: '{{ template \"sns.default.message\" . }}'\n    workspace_arn:
 arn:aws:aps:us-west-2:123456789012:workspace/ws-58a6a446-5ec4-415b-9052-a449073bbd0a
\ntemplates: []\n"
    },
    "uptime": null,
    "versionInfo": null
}
```

GetAlertManagerSilence

O `GetAlertManagerSilence` recupera informações sobre um alerta silencioso.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/silence/silenceID`

Parâmetros de consulta de URL: nenhum

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silence/
d29d9df3-9125-4441-912c-70b05f86f973 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 310
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "id": "d29d9df3-9125-4441-912c-70b05f86f973",
    "status": {
        "state": "active"
    },
    "updatedAt": "2021-10-22T19:32:11.763Z",
    "comment": "hello-world",
    "createdBy": "test-person",
    "endsAt": "2023-07-24T01:05:36.000Z",
    "matchers": [
        {
            "isEqual": true,
            "isRegex": true,
            "name": "job",
            "value": "hello"
        }
```

```
    ],
    "startsAt": "2021-10-22T19:32:11.763Z"
}
```

GetLabels

A operação `GetLabels` recupera os rótulos associados a uma série temporal.

Verbos HTTP válidos:

`GET`, `POST`

Válido URIs:

`/workspaces/`*`workspaceId`*`/api/v1/labels`

`/workspaces/`*`workspaceId`*`/api/v1/label/`*`label-name`*`/values` Esse URI é compatível somente com solicitações GET.

Parâmetros da consulta de URL:

`match[]=<series_selector>` Argumento repetido do seletor de série que seleciona a série da qual ler os nomes dos rótulos. Opcional.

`start=<rfc3339 | unix_timestamp>` Carimbo de data/hora de início. Opcional.

`end=<rfc3339 | unix_timestamp>` Carimbo de data e hora de término. Opcional.

Solicitação de amostra para **`/workspaces/workspaceId/api/v1/labels`**

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/labels HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta para **`/workspaces/workspaceId/api/v1/labels`**

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 1435
Connection: keep-alive
```

```
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": [
        "__name__",
        "access_mode",
        "address",
        "alertname",
        "alertstate",
        "apiservice",
        "app",
        "app_kubernetes_io_instance",
        "app_kubernetes_io_managed_by",
        "app_kubernetes_io_name",
        "area",
        "beta_kubernetes_io_arch",
        "beta_kubernetes_io_instance_type",
        "beta_kubernetes_io_os",
        "boot_id",
        "branch",
        "broadcast",
        "buildDate",
        ...
    ]
}
```

Solicitação de amostra para **/workspaces/workspaceId/api/v1/label/label-name/values**

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/label/access_mode/values
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta para **/workspaces/workspaceId/api/v1/label/label-name/values**

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 74
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": [
        "ReadWriteOnce"
    ]
}
```

GetMetricMetadata

A operação `GetMetricMetadata` recupera metadados sobre métricas que estão sendo extraídas dos alvos no momento. Ele não fornece nenhuma informação sobre o alvo.

A seção de dados do resultado da consulta consiste em um objeto em que cada chave é um nome de métrica e cada valor é uma lista de objetos de metadados exclusivos, conforme exposto para esse nome de métrica em todos os destinos.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/`*`workspaceId`*`/api/v1/metadata`

Parâmetros da consulta de URL:

`limit=<number>` O número máximo de linhas a serem retornadas.

`metric=<string>` Um nome de métrica para filtrar metadados. Se você mantiver isso vazio, todos os metadados métricos serão recuperados.

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/metadata HTTP/1.1
```

```
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
Transfer-Encoding: chunked

{
    "status": "success",
    "data": {
        "aggregator_openapi_v2_regeneration_count": [
            {
                "type": "counter",
                "help": "[ALPHA] Counter of OpenAPI v2 spec regeneration count broken
 down by causing APIService name and reason.",
                "unit": ""
            }
        ],
        ...
    }
}
```

GetSeries

A operação `GetSeries` recupera a lista de séries temporais que correspondem a um determinado conjunto de rótulos.

Verbos HTTP válidos:

`GET`, `POST`

Válido URIs:

`/workspaces/workspaceId/api/v1/series`

Parâmetros da consulta de URL:

`match[]=<series_selector>` Argumento repetido do seletor de série que seleciona a série a ser retornada. Pelo menos um `match[]` deve ser fornecido.

`start=<rfc3339 | unix_timestamp>` Carimbo de data/hora de início. Opcional

`end=<rfc3339 | unix_timestamp>` Carimbo de data e hora de término. Opcional

Exemplo de solicitação

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/series --data-urlencode
 'match[]=node_cpu_seconds_total{app="prometheus"}' --data-urlencode 'start=1634936400'
 --data-urlencode 'end=1634939100' HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
content-encoding: gzip

{
    "status": "success",
    "data": [
        {
            "__name__": "node_cpu_seconds_total",
            "app": "prometheus",
            "app_kubernetes_io_managed_by": "Helm",
            "chart": "prometheus-11.12.1",
            "cluster": "cluster-1",
            "component": "node-exporter",
            "cpu": "0",
            "heritage": "Helm",
            "instance": "10.0.100.36:9100",
```

```
            "job": "kubernetes-service-endpoints",
            "kubernetes_name": "servicesstackprometheuscf14a6d7-node-exporter",
            "kubernetes_namespace": "default",
            "kubernetes_node": "ip-10-0-100-36.us-west-2.compute.internal",
            "mode": "idle",
            "release": "servicesstackprometheuscf14a6d7"
        },
        {
            "__name__": "node_cpu_seconds_total",
            "app": "prometheus",
            "app_kubernetes_io_managed_by": "Helm",
            "chart": "prometheus-11.12.1",
            "cluster": "cluster-1",
            "component": "node-exporter",
            "cpu": "0",
            "heritage": "Helm",
            "instance": "10.0.100.36:9100",
            "job": "kubernetes-service-endpoints",
            "kubernetes_name": "servicesstackprometheuscf14a6d7-node-exporter",
            "kubernetes_namespace": "default",
            "kubernetes_node": "ip-10-0-100-36.us-west-2.compute.internal",
            "mode": "iowait",
            "release": "servicesstackprometheuscf14a6d7"
        },
        ...
    ]
}
```

ListAlerts

A operação `ListAlerts` recupera os alertas atualmente ativos no espaço de trabalho.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/workspaceId/api/v1/alerts`

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/alerts HTTP/1.1
```

```
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 386
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
  "status": "success",
  "data": {
    "alerts": [
      {
        "labels": {
          "alertname": "test-1.alert",
          "severity": "none"
        },
        "annotations": {
          "message": "message"
        },
        "state": "firing",
        "activeAt": "2020-12-01T19:37:25.429565909Z",
        "value": "1e+00"
      }
    ]
  },
  "errorType": "",
  "error": ""
}
```

ListAlertManagerAlerts

Ele `ListAlertManagerAlerts` recupera informações sobre os alertas atualmente disparados no gerenciador de alertas no espaço de trabalho.

Verbos HTTP válidos:

```
GET
```

Válido URIs:

```
/workspaces/workspaceId/alertmanager/api/v2/alerts
```

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 354
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "annotations": {
            "summary": "this is a test alert used for demo purposes"
        },
        "endsAt": "2021-10-21T22:07:31.501Z",
        "fingerprint": "375eab7b59892505",
        "receivers": [
            {
                "name": "sns-0"
            }
        ],
        "startsAt": "2021-10-21T22:02:31.501Z",
        "status": {
            "inhibitedBy": [],
```

```
            "silencedBy": [],
            "state": "active"
        },
        "updatedAt": "2021-10-21T22:02:31.501Z",
        "labels": {
            "alertname": "test-alert"
        }
    }
]
```

ListAlertManagerAlertGroups

A operação `ListAlertManagerAlertGroups` recupera uma lista de grupos de alertas configurados no gerenciador de alertas no espaço de trabalho.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/`*`workspaceId`*`/alertmanager/api/v2/alerts/groups`

Parâmetros da consulta de URL:

Booleano `active`. Se verdadeiro, a lista retornada inclui alertas ativos. O padrão é true. Opcional

Booleano `silenced`. Se verdadeiro, a lista retornada inclui alertas silenciados. O padrão é true. Opcional

Booleano `inhibited`. Se verdadeiro, a lista retornada inclui alertas inibidos. O padrão é true. Opcional

`filter` Uma matriz de strings. Uma lista de correspondências para filtrar os alertas. Opcional

`receiver` String. Uma expressão regular que combina receptores pelos quais filtrar alertas. Opcional

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/alerts/
groups HTTP/1.1
Content-Length: 0,
```

```
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 443
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "alerts": [
            {
                "annotations": {
                    "summary": "this is a test alert used for demo purposes"
                },
                "endsAt": "2021-10-21T22:07:31.501Z",
                "fingerprint": "375eab7b59892505",
                "receivers": [
                    {
                        "name": "sns-0"
                    }
                ],
                "startsAt": "2021-10-21T22:02:31.501Z",
                "status": {
                    "inhibitedBy": [],
                    "silencedBy": [],
                    "state": "unprocessed"
                },
                "updatedAt": "2021-10-21T22:02:31.501Z",
                "generatorURL": "https://www.amazon.com/",
                "labels": {
                    "alertname": "test-alert"
                }
            }
        ],
        "labels": {},
```

```
        "receiver": {
            "name": "sns-0"
        }
    }
]
```

ListAlertManagerReceivers

A operação `ListAlertManagerReceivers` recupera informações sobre os receptores configurados no gerenciador de alertas.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/receivers`

Parâmetros de consulta de URL: nenhum

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/receivers
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 19
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
```

```
    {
        "name": "sns-0"
    }
]
```

ListAlertManagerSilences

A operação `ListAlertManagerSilences` recupera informações sobre os silêncios de alerta configurados no espaço de trabalho.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/silences`

Exemplo de solicitação

```
GET /workspaces/ws-58a6a446-5ec4-415b-9052-a449073bbd0a/alertmanager/api/v2/silences
 HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 312
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

[
    {
        "id": "d29d9df3-9125-4441-912c-70b05f86f973",
        "status": {
```

```
            "state": "active"
        },
        "updatedAt": "2021-10-22T19:32:11.763Z",
        "comment": "hello-world",
        "createdBy": "test-person",
        "endsAt": "2023-07-24T01:05:36.000Z",
        "matchers": [
            {
                "isEqual": true,
                "isRegex": true,
                "name": "job",
                "value": "hello"
            }
        ],
        "startsAt": "2021-10-22T19:32:11.763Z"
    }
]
```

ListRules

O `ListRules` recupera informações sobre as regras configuradas no espaço de trabalho.

Verbos HTTP válidos:

`GET`

Válido URIs:

`/workspaces/`*`workspaceId`*`/api/v1/rules`

Exemplo de solicitação

```
GET /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/rules HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
```

```
Content-Length: 423
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "status": "success",
    "data": {
        "groups": [
            {
                "name": "test-1.rules",
                "file": "test-rules",
                "rules": [
                    {
                        "name": "record:1",
                        "query": "sum(rate(node_cpu_seconds_total[10m:1m]))",
                        "labels": {},
                        "health": "ok",
                        "lastError": "",
                        "type": "recording",
                        "lastEvaluation": "2021-10-21T21:22:34.429565909Z",
                        "evaluationTime": 0.001005399
                    }
                ],
                "interval": 60,
                "lastEvaluation": "2021-10-21T21:22:34.429563992Z",
                "evaluationTime": 0.001010504
            }
        ]
    },
    "errorType": "",
    "error": ""
}
```

PutAlertManagerSilences

A operação `PutAlertManagerSilences` cria um novo silêncio de alerta ou atualiza um existente.

Verbos HTTP válidos:

`POST`

Válido URIs:

`/workspaces/workspaceId/alertmanager/api/v2/silences`

Parâmetros da consulta de URL:

`silence` Um objeto que representa o silêncio. Este é o formato:

```
{
  "id": "string",
  "matchers": [
    {
      "name": "string",
      "value": "string",
      "isRegex": Boolean,
      "isEqual": Boolean
    }
  ],
  "startsAt": "timestamp",
  "endsAt": "timestamp",
  "createdBy": "string",
  "comment": "string"
}
```

Exemplo de solicitação

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/alertmanager/api/v2/silences
 HTTP/1.1
Content-Length: 281,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0

{
   "matchers":[
      {
         "name":"job",
         "value":"up",
         "isRegex":false,
         "isEqual":true
      }
   ],
   "startsAt":"2020-07-23T01:05:36+00:00",
```

```
   "endsAt":"2023-07-24T01:05:36+00:00",
   "createdBy":"test-person",
   "comment":"test silence"
}
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 53
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
vary: Origin

{
    "silenceID": "512860da-74f3-43c9-8833-cec026542b32"
}
```

QueryMetrics

A operação `QueryMetrics` avalia uma consulta instantânea em um único momento ou em um intervalo de tempo.

Verbos HTTP válidos:

`GET`, `POST`

Válido URIs:

`/workspaces/`*`workspaceId`*`/api/v1/query` Esse URI avalia uma consulta instantânea em um único momento.

`/workspaces/`*`workspaceId`*`/api/v1/query_range` Esse URI avalia uma consulta instantânea em um intervalo de tempo.

Parâmetros da consulta de URL:

`query=<string>` Uma string de consulta da expressão Prometheus. Usado em ambos `query` e `query_range`.

`time=<rfc3339 | unix_timestamp>` (Opcional) Carimbo de data/hora de avaliação se você estiver usando o `query` para uma consulta instantânea em um único momento.

`timeout=<duration>` (Opcional) Tempo limite de avaliação. O padrão é e é limitado pelo valor do sinalizador `-query.timeout`. Usado em ambos `query` e `query_range`.

`start=<rfc3339 | unix_timestamp>` Inicie o timestamp se você estiver usando `query_range` para consultar por um intervalo de tempo.

`end=<rfc3339 | unix_timestamp>` Carimbo de data/hora de término se você estiver usando `query_range` para consultar por um intervalo de tempo.

`step=<duration | float>` Largura da etapa de resolução da consulta em `duration` formato ou em `float` alguns segundos. Use somente se você estiver usando `query_range` para consultar por um intervalo de tempo e for necessário para essas consultas.

`max_samples_processed_warning_threshold=<integer>` (Opcional) Define o limite de aviso para amostras de consulta processadas (QSP). Quando as consultas atingirem esse limite, uma mensagem de aviso será retornada na resposta da API.

`max_samples_processed_error_threshold=<integer>>` (Opcional) Define o limite de erro para amostras de consulta processadas (QSP). As consultas que excederem esse limite serão rejeitadas com um erro e não serão cobradas. Usado para evitar custos excessivos de consulta.

Duration (Duração)

A `duration` em uma API compatível com o Prometheus é um número, seguido imediatamente por uma das seguintes unidades:

- `ms` milissegundos
- `s` segundos
- `m` minutos
- `h` horas
- `d` dias, supondo que um dia sempre tenha 24h
- `w` semanas, supondo que uma semana sempre tenha 7 dias
- `y` anos, supondo que um ano sempre tenha 365 dias

Exemplo de solicitação

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/query?
query=sum(node_cpu_seconds_total) HTTP/1.1
Content-Length: 0,
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Grafana/8.1.0
```

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length: 132
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
Content-Type: application/json
Server: amazon
content-encoding: gzip

{
    "status": "success",
    "data": {
        "resultType": "vector",
        "result": [
            {
                "metric": {},
                "value": [
                    1634937046.322,
                    "252590622.81000024"
                ]
            }
        ]
    }
}
```

RemoteWrite

A operação `RemoteWrite` grava métricas de um servidor Prometheus em uma URL remota em um formato padronizado. Normalmente, você usará um cliente existente, como um servidor Prometheus, para chamar essa operação.

Verbos HTTP válidos:

`POST`

Válido URIs:

`/workspaces/workspaceId/api/v1/remote_write`

Parâmetros da consulta de URL:

Nenhum

`RemoteWrite` tem uma taxa de ingestão de 70.000 amostras por segundo e um tamanho de pico de ingestão de 1.000.000 de amostras.

Exemplo de solicitação

```
POST /workspaces/ws-b226cc2a-a446-46a9-933a-ac50479a5568/api/v1/remote_write --data-
binary "@real-dataset.sz" HTTP/1.1
Authorization: AUTHPARAMS
X-Amz-Date: 20201201T193725Z
User-Agent: Prometheus/2.20.1
Content-Type: application/x-protobuf
Content-Encoding: snappy
X-Prometheus-Remote-Write-Version: 0.1.0

body
```

> **Note**
>
> Para a sintaxe do corpo da solicitação, consulte a definição do buffer de protocolo em https://github.com/prometheus/prometheus/blob/1c624c58ca934f618be737b4995e22051f5724c1/prompb/remote.pb.go #L64.

Exemplo de resposta

```
HTTP/1.1 200 OK
x-amzn-RequestId: 12345678-abcd-4442-b8c5-262b45e9b535
Content-Length:0
Connection: keep-alive
Date: Tue, 01 Dec 2020 19:37:25 GMT
```

```
Content-Type: application/json
Server: amazon
vary: Origin
```

Guia do usuário do histórico de documentos do Amazon Managed Service for Prometheus

A tabela a seguir descreve as atualizações importantes da documentação no Guia do usuário do Amazon Managed Service for Prometheus. Para receber notificações sobre atualizações dessa documentação, é possível inscrever-se em um feed RSS.

Alteração	Descrição	Data
Suporte lançado para PagerDuty	O Amazon Managed Service for Prometheus adiciona suporte à integração que permite fluxos PagerDuty de trabalho automatizados de resposta a incidentes e garante que alertas críticos cheguem aos membros certos da equipe no momento certo. Para obter mais informaçõ es, consulte Usar PagerDuty como receptor de alertas.	29 de agosto de 2025
Adicionado suporte a políticas baseadas em recursos	As seguintes ações de API estão disponíveis: • DeleteResourcePolicy • DescribeResourcePolicy • PutResourcePolicy	15 de agosto de 2025
Atualização da política AmazonPrometheusCo nsoleFullAccess gerenciada do IAM.	A AmazonPrometheusCo nsoleFullAccesspolítica foi atualizada. As permissões `aps:CreateQueryLog gingConfiguration` , `aps:UpdateQueryLog gingConfiguration` ,	5 de maio de 2025

	`aps:DeleteQueryLoggingConfiguration` e `aps:DescribeQueryLoggingConfiguration` foram adicionadas à política.	
Foi adicionada ao console a edição de arquivos de definição de regras e arquivos de configuração do gerenciador de alertas	O Amazon Managed Service for Prometheus adicionou suporte à edição de arquivos de configuração do gerenciador de alertas e arquivos de definição de regras pelo console do Amazon Managed Service for Prometheus.	16 de maio de 2024
Foi adicionada uma configuração mais simples de coletor AWS gerenciado com entradas de acesso para o Amazon EKS	O Amazon Managed Service for Prometheus adicionou suporte a entradas de acesso do Amazon EKS para simplificar a configuração de coletores gerenciados pela AWS. A política AmazonPrometheusScraperServiceRolePolicygerenciada para coletores AWS gerenciados é atualizada para permitir a exclusão de entradas de acesso que não são mais usadas.	2 de maio de 2024

Mova a AWS API para um guia de referência de API separado	O Amazon Managed Service para AWS APIs Prometheus agora está disponível em sua própria referência, a Amazon Managed Service for Prometheus API Reference. Os compatíveis com o Prometheus APIs continuam sendo documentados no Guia do usuário do Amazon Managed Service for Prometheus.	7 de fevereiro de 2024
Chaves gerenciadas pelo cliente adicionadas para criptografia do espaço de trabalho	O Amazon Managed Service for Prometheus adiciona suporte para chaves gerenciadas pelo cliente para criptografia do espaço de trabalho. Para obter mais informações, consulte Criptografia em repouso.	21 de dezembro de 2023
Foram adicionadas novas permissões ao AmazonPrometheusFullAccess	Foram adicionadas novas permissões à política AmazonPrometheusFullAccessgerenciada para apoiar a criação de coletores AWS gerenciados para clusters do Amazon EKS.	26 de novembro de 2023
Foi adicionada uma nova política gerenciada, AmazonPrometheusScraperServiceLinkedRolePolicy	Foi adicionada uma nova política gerenciada AmazonPrometheusScraperServiceLinkedRolePolicypara que coletores AWS gerenciados coletem métricas de clusters do Amazon EKS.	26 de novembro de 2023

Coletores AWS gerenciados adicionados como método de ingestão	O Amazon Managed Service for Prometheus adiciona suporte para coletores gerenciados pela AWS.	26 de novembro de 2023
Suporte adicionado para integração com o Amazon Managed Grafana	O Amazon Managed Service for Prometheus adiciona suporte para integração com alertas Amazon Managed Grafana.	23 de novembro de 2022
Foram adicionadas novas permissões ao AmazonPrometheusConsoleFullAccess	Foram adicionadas novas permissões à política AmazonPrometheusConsoleFullAccessgerenciada para dar suporte ao registro de eventos do gerenciador de alertas e da régua no CloudWatch Logs.	24 de outubro de 2022
Foi adicionada a solução de observabilidade Amazon EKS.	O Amazon Managed Service for Prometheus adiciona uma nova solução AWS usando o Observability Accelerator. Para obter mais informações, consulte Uso do acelerador de observabilidade AWS.	14 de outubro de 2022
Suporte adicional para integração ao monitoramento de custos do Amazon EKS.	O Amazon Managed Service for Prometheus adiciona suporte para integração ao monitoramento de custos do Amazon EKS. Para obter mais informações, consulte Integração ao monitoramento de custos do Amazon EKS.	22 de setembro de 2022

Lançou o suporte para registros do Alert Manager e do Ruler no Amazon CloudWatch Logs.	O Amazon Managed Service for Prometheus lança suporte para registros de erros do Alert Manager e do Ruler no Amazon Logs. CloudWatch Para obter mais informações, consulte Amazon CloudWatch Logs.	1º de setembro de 2022
Foi adicionado suporte de retenção de armazenamento personalizado.	O Amazon Managed Service for Prometheus adiciona suporte personalizado à retenção de armazenam ento, por espaço de trabalho, modificando a cota desse espaço de trabalho. Para obter mais informações sobre cotas no Amazon Managed Service for Prometheus, consulte Service Quotas.	12 de agosto de 2022
Métricas de uso adicionadas à Amazon CloudWatch.	O Amazon Managed Service for Prometheus adiciona suporte ao envio de métricas de uso para a Amazon. CloudWatch Para obter mais informações, consulte as CloudWatchmétricas da Amazon.	6 de maio de 2022
Adicionado suporte para a região Europa (Londres).	O Amazon Managed Service for Prometheus adiciona suporte para a região Europa (Londres).	4 de maio de 2022

O Amazon Managed Service for Prometheus está disponível ao público em geral e adiciona suporte ao gerenciador de regras e alertas.	O Amazon Managed Service for Prometheus já está disponível ao público em geral. Ele também oferece suporte ao gerenciador de regras e alertas. Para obter mais informações, consulte Regras de gravação e regras de alerta e Gerenciador de alertas e modelos.	29 de setembro de 2021
Suporte de tag adicionado.	O Amazon Managed Service for Prometheus oferece suporte à marcação com tag de espaços de trabalho do Amazon Managed Service for Prometheus.	7 de setembro de 2021
As cotas de séries ativas e de taxa de ingestão aumentaram.	A cota da série ativa aumentou para 1.000.000 e a cota da taxa de ingestão aumentou para 70.000 amostras por segundo.	22 de fevereiro de 2021
Prévia do lançamento do Amazon Managed Service for Prometheus.	A prévia do Amazon Managed Service for Prometheus foi lançada.	15 de dezembro de 2020

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.