aws

使用者指南

# Amazon Managed Service for Prometheus

# Amazon Managed Service for Prometheus: 使用者指南

# Table of Contents

# Amazon Managed Service for Prometheus 是什麼？

Amazon Managed Service for Prometheus 是無伺服器、且與 Prometheus 相容的監控服務，適用於容器指標，可讓您更輕鬆地大規模監控容器環境。透過 Amazon Managed Service for Prometheus，您可以使用目前用來監控容器化工作負載效能的相同開放原始碼 Prometheus 資料模型和查詢語言，並享有改良的可擴展性、可用性和安全性，而無需管理基礎設施。

Amazon Managed Service for Prometheus 會隨著工作負載向上擴展和向下縮減規模，自動擴展操作指標的擷取、儲存和查詢作業。它與 AWS 安全服務整合，可快速安全地存取資料。

Amazon Managed Service for Prometheus 專為使用多可用區 (Multi-AZ) 部署而設計。擷取至工作區的資料會跨相同區域中的三個可用區域進行複製。

Amazon Managed Service for Prometheus 會在 Amazon Elastic Kubernetes Service 和自我管理 Kubernetes 環境中執行的容器叢集。

透過 Amazon Managed Service for Prometheus，您可以使用與 Prometheus 搭配使用的相同開放原始碼 Prometheus 資料模型和 PromQL 查詢語言。工程團隊可以使用 PromQL 來篩選、彙總和警示指標，並快速獲得效能可見度，而不需要變更任何程式碼。Amazon Managed Service for Prometheus 提供彈性的查詢功能，無需支付營運成本和複雜性。

根據預設，擷取至工作區的指標會儲存 150 天，然後自動刪除。這個長度是可調整的配額。

## 支援地區

Amazon Managed Service for Prometheus 目前支援下列區域：

| 區域名稱 | 區域 | 端點 | 通訊協定 |
|---|---|---|---|
| 美國東部 (俄亥俄) | us-east-2 | aps.us-east-2.amazonaws.com | HTTPS |
| | | aps-workspaces.us-east-2.amazonaws.com | HTTPS |
| 美國東部 (維吉尼亞北部) | us-east-1 | aps.us-east-1.amazonaws.com | HTTPS |
| | | aps-workspaces.us-east-1.amazonaws.com | HTTPS |
| 美國西部 (奧勒岡) | us-west-2 | aps.us-west-2.amazonaws.com | HTTPS |

| 區域名稱 | 區域 | 端點 | 通訊協定 |
| --- | --- | --- | --- |
| | | aps-workspaces.us-west-2.amazonaws.com | HTTPS |
| 亞太區域 (孟買) | ap-south-1 | aps.ap-south-1.amazonaws.com | HTTPS |
| | | aps-workspaces.ap-south-1.amazonaws.com | HTTPS |
| 亞太區域 (首爾) | ap-northeast-2 | aps.ap-northeast-2.amazonaws.com | HTTPS |
| | | aps-workspaces.ap-northeast-2.amazonaws.com | HTTPS |
| 亞太區域 (新加坡) | ap-southeast-1 | aps.ap-southeast-1.amazonaws.com | HTTPS |
| | | aps-workspaces.ap-southeast-1.amazonaws.com | HTTPS |
| 亞太區域 (雪梨) | ap-southeast-2 | aps.ap-southeast-2.amazonaws.com | HTTPS |
| | | aps-workspaces.ap-southeast-2.amazonaws.com | HTTPS |
| 亞太區域 (東京) | ap-northeast-1 | aps.ap-northeast-1.amazonaws.com | HTTPS |
| | | aps-workspaces.ap-northeast-1.amazonaws.com | HTTPS |
| 歐洲 (法蘭克福) | eu-central-1 | aps.eu-central-1.amazonaws.com | HTTPS |
| | | aps-workspaces.eu-central-1.amazonaws.com | HTTPS |
| 歐洲 (愛爾蘭) | eu-west-1 | aps.eu-west-1.amazonaws.com | HTTPS |
| | | aps-workspaces.eu-west-1.amazonaws.com | HTTPS |
| 歐洲 (倫敦) | eu-west-2 | aps.eu-west-2.amazonaws.com | HTTPS |
| | | aps-workspaces.eu-west-2.amazonaws.com | HTTPS |

| 區域名稱 | 區域 | 端點 | 通訊協定 |
| --- | --- | --- | --- |
| 歐洲 (巴黎) | eu-west-3 | aps.eu-west-3.amazonaws.com | HTTPS |
| | | aps-workspaces.eu-west-3.amazonaws.com | HTTPS |
| 歐洲 (斯德哥爾摩) | eu-north-1 | aps.eu-north-1.amazonaws.com | HTTPS |
| | | aps-workspaces.eu-north-1.amazonaws.com | HTTPS |
| 南美洲 (聖保羅) | sa-east-1 | aps.sa-east-1.amazonaws.com | HTTPS |
| | | aps-workspaces.sa-east-1.amazonaws.com | HTTPS |

## 定價

擷取和儲存指標會產生費用。儲存費用是根據指標範例和中繼資料的壓縮大小而定。如需詳細資訊，請參閱《Amazon Managed Service for Prometheus 定價》。

您可以使用「Cost Explorer」和「 AWS 成本與用量報告」來監控您的費用。如需詳細資訊，請參閱使用 Cost Explorer 探索資料和什麼是 AWS 成本和使用量報告。

## 付費支援

如果您訂閱任何級別的 AWS 高級支持計劃，則您的高級支持將適用於 Prometheus 的 Amazon 託管服務。

# 開始使用

本節說明如何快速建立 Amazon Managed Service for Prometheus 工作區、設定這些工作區的 Prometheus 指標擷取作業，並查詢這些指標。

它還包括有關設置的信息 AWS 帳戶，以防您是新手 AWS。

主題



## 設定

完成本節中的任務以進行第一 AWS 次設置。如果您已經有 AWS 帳戶，請跳至建立工作區。

當您註冊時 AWS，您的 AWS 帳戶會自動存取中的所有服務 AWS，包括 Prometheus 的 Amazon 受管服務。不過，您只需針對所使用的服務付費。

主題



### 註冊一個 AWS 帳戶

如果您沒有 AWS 帳戶，請完成以下步驟來建立一個。

若要註冊成為 AWS 帳戶

1. 開啟 https://portal.aws.amazon.com/billing/signup。
2. 請遵循線上指示進行。

   部分註冊程序需接收來電，並在電話鍵盤輸入驗證碼。

當您註冊一個時 AWS 帳戶，將創建AWS 帳戶根使用者一個。根使用者有權存取該帳戶中的所有 AWS 服務 和資源。安全性最佳做法是將管理存取權指派給使用者，並僅使用 root 使用者來執行需要 root 使用者存取權的工作。

AWS 註冊過程完成後，會向您發送確認電子郵件。您可以隨時登錄 https://aws.amazon.com/ 並選擇我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

## 建立具有管理權限的使用者

註冊後，請保護您的 AWS 帳戶 AWS 帳戶根使用者 AWS IAM Identity Center、啟用和建立系統管理使用者，這樣您就不會將 root 使用者用於日常工作。

保護您的 AWS 帳戶根使用者

1. 選擇 Root 使用者並輸入您的 AWS 帳戶 電子郵件地址，以帳戶擁有者身分登入。AWS Management Console在下一頁中，輸入您的密碼。

   如需使用根使用者登入的說明，請參閱 AWS 登入 使用者指南中的以根使用者身分登入。
2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

   如需指示，請參閱《IAM 使用者指南》中的為 AWS 帳戶 根使用者啟用虛擬 MFA 裝置 (主控台)。

建立具有管理權限的使用者

1. 啟用 IAM Identity Center。

   如需指示，請參閱 AWS IAM Identity Center 使用者指南中的啟用 AWS IAM Identity Center。
2. 在 IAM 身分中心中，將管理存取權授予使用者。

   若要取得有關使用 IAM Identity Center 目錄 做為身分識別來源的自學課程，請參閱《使用指南》IAM Identity Center 目錄中的「以預設值設定使用AWS IAM Identity Center 者存取」。

以具有管理權限的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者簽署，請使用建立 IAM Identity Center 使用者時傳送至您電子郵件地址的簽署 URL。

如需使用 IAM 身分中心使用者登入的說明，請參閱使用AWS 登入 者指南中的登入 AWS 存取入口網站。

指派存取權給其他使用者

1. 在 IAM 身分中心中，建立遵循套用最低權限許可的最佳做法的權限集。

   如需指示，請參閱《AWS IAM Identity Center 使用指南》中的「建立權限集」。
2. 將使用者指派給群組，然後將單一登入存取權指派給群組。

   如需指示，請參閱《AWS IAM Identity Center 使用指南》中的「新增群組」。

# 建立工作區

工作區是專門用於儲存和查詢 Prometheus 指標的邏輯空間。工作區支援精細的存取控制，以授權其管理，例如更新、列出、說明和刪除，以及擷取和查詢指標。您可以在帳戶中的每個區域擁有一或多個工作區。

若要設定工作區，請遵循下列步驟。

> **Note**
>
> 如需有關建立工作區的詳細資訊，請參閱 建立工作區。

建立 Amazon Managed Service for Prometheus 工作區

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 針對「工作區別名」，輸入新工作區的別名。

   工作區別名是易記的名稱，可協助您識別工作區。名稱不必是唯一。兩個工作區可能具有相同的別名，但是所有工作區都將具有由 Amazon Managed Service for Prometheus 產生的唯一工作區 ID。
3. (選用) 若要將標籤新增至命名空間，請選擇新增標籤。

   之後，在 Key (索引鍵) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。

若要新增另一個標籤，請再次選擇 Add new tag (新增標籤)。

4. 選擇建立工作區。

   隨即顯示工作區詳細資訊頁面。這會針對遠端寫入和查詢顯示此工作區的資訊，包含狀態、ARN、工作區 ID 和端點 URL。

   最初，狀態可能是建立中。請等待狀態為「啟用中」，然後再繼續設定指標擷取。

   記下針對「端點 - 遠端寫入 URL」和「端點 - 查詢 URL」顯示的 URL。當您將 Prometheus 伺服器設定為將指標遠端寫入此工作區，以及查詢這些指標時，將會需要這些指標。

# 將 Prometheus 指標擷取至工作區

擷取指標的一種方法是使用獨立的 Prometheus 代理程式 (以代理程式模式執行的 Prometheus 執行個體) 從叢集抓取指標，然後將指標轉送至 Amazon Managed Service for Prometheus 以進行儲存和監控。本節說明如何透過使用 Helm 設定 Prometheus 代理程式的新執行個體，從 Amazon EKS 將指標擷取到 Amazon Managed Service for Prometheus 工作區。

如需其他將資料擷取至 Amazon Managed Service for Prometheus 方式的相關資訊，包含如何保護指標和建立高可用性指標，請參閱 擷取指標至您的工作區。

> **Note**
>
> 根據預設，擷取至工作區的指標會儲存 150 天，然後自動刪除。這個長度是可調整的配額。

本節中的指示可協助您快速啟動，並使用 Amazon Managed Service for Prometheus 執行。您在 Amazon EKS 叢集中設定新的 Prometheus 伺服器，而新伺服器會使用預設組態做為代理程式，將指標傳送至 Amazon Managed Service for Prometheus。此主題有以下先決條件：

- 您必須擁有 Amazon EKS 叢集，在其中收集新的 Prometheus 伺服器指標。
- 您必須使用 Helm CLI 3.0 或更新版本
- 您必須使用 Linux 或 macOS 電腦來執行以下各節中的步驟。

## 步驟 1：新增 Helm Chart 儲存庫

若要新增 Helm Chart 儲存庫，請輸入下列命令。如需有關這些命令的詳細資訊，請參閱 Helm 儲存庫。

```
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
```

## 步驟 2：建立 Prometheus 命名空間

輸入下列命令，為 Prometheus 伺服器和其他監控元件建立 Prometheus 命名空間。將 *prometheus-agent-namespace* 替換為您要用於此命名空間的名稱。

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

## 步驟 3：為服務帳戶設定 IAM 角色

透過此擷取方法，您需要在執行 Prometheus 代理程式的 Amazon EKS 叢集中，將 IAM 角色用於服務帳戶。

透過服務帳戶的 IAM 角色，您可以產生 IAM 角色與 Kubernetes 服務帳戶的關聯。然後，此服務帳戶可以為使用該服務帳戶之任何 Pod 中的容器提供 AWS 許可。如需詳細資訊，請參閱服務帳戶的 IAM 角色。

如果您尚未設定這些角色，請按照中的 自 Amazon EKS 叢集設定指標擷取作業的服務角色 指示設定角色。本節中的說明需要使用 `eksctl`。如需詳細資訊，請參閱 Amazon Elastic Kubernetes Service 入門 - `eksctl`。

> **Note**
>
> 如果您不在 EKS 上，或 AWS 僅使用訪問密鑰和密鑰訪問 Prometheus 的 Amazon 託管服務，則無法使用基於 Sigv4。`EKS-IAM-ROLE`

## 步驟 4：設定新伺服器並開始擷取指標

若要安裝新的 Prometheus 代理程式，並將指標傳送至您的 Amazon Managed Service for Prometheus 工作區，請依照這些步驟執行。

安裝新的 Prometheus 代理程式，並將指標傳送至您的 Amazon Managed Service for Prometheus 工作區

1. 使用文字編輯器建立名為 `my_prometheus_values_yaml` 的檔案，包含下列內容。

   - 將 *IAM_PROXY_PROMETHEUS_ROLE_ARN* 替換為您在 自 Amazon EKS 叢集設定指標擷取作業的服務角色 中建立 amp-iamproxy-ingest-role 的 ARN。
   - 將 *WORKSPACE_ID* 替換為您 Amazon Managed Service for Prometheus 工作區的 ID。
   - 將 *REGION* 替換為您 Amazon Managed Service for Prometheus 工作區的區域。

```
## 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. 輸入下列命令以建立 Prometheus 伺服器。

   - 將 *prometheus-chart-name* 替換為您的 Prometheus 版本名稱。
   - 將 *prometheus-agent-namespace* 替換為您 Prometheus 命名空間的名稱。

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

# 查詢 Prometheus 指標

既然已將指標擷取至工作區，便可對其進行查詢。常用的指標查詢方法是使用 Grafana 等服務來查詢指標。在本節中，您將學習如何使用 Amazon Managed Grafana 從 Amazon Managed Service for Prometheus 查詢指標。

> **Note**
>
> 若要瞭解其他查詢 Amazon Managed Service for Prometheus 指標或使用 Amazon Managed Service for Prometheus API 的方式，請參閱 查詢 Prometheus 指標。

您可以使用標準的 Prometheus 查詢語言 PromQL 執行查詢。如需有關 PromQL 和其語法的詳細資訊，請參閱 Prometheus 說明文件中的查詢 Prometheus。

Amazon Managed Grafana 是開放原始碼 Grafana 的全受管服務，可簡化與開放原始碼第三方 ISV 的連線，以及大規模視覺化和分析資料來源的 AWS 服務。

Amazon Managed Service for Prometheus 支援使用 Amazon Managed Grafana 查詢工作區中的指標。在 Amazon Managed Grafana 主控台中，您可以探索現有 Amazon Managed Service for Prometheus 帳戶，將 Amazon Managed Service for Prometheus 工作區新增為資料來源。Amazon Managed Grafana 管理存取 Amazon Managed Service for Prometheus 所需的身分驗證憑證組態。如需從 Amazon Managed Grafana 建立 Amazon Managed Service for Prometheus 連線的詳細指示，請參閱 Amazon Managed Grafana 使用者指南中的指示。

您也可以在 Amazon Managed Grafana 中檢視 Amazon Managed Service for Prometheus 警示。如需設定與警示整合的指示，請參閱 與 Amazon Managed Grafana 或開放原始碼 Grafana 整合警示。

> **Note**
>
> 如果您已將 Amazon Managed Grafana 工作區設定為使用私有 VPC，則必須將 Amazon Managed Service for Prometheus 工作區連線到相同 VPC。如需詳細資訊，請參閱 在私有 VPC 中連線至 Amazon Managed Grafana。

# 管理工作區

工作區是專門用於儲存和查詢 Prometheus 指標的邏輯空間。工作區支援精細的存取控制，以授權其管理，例如更新、列出、說明和刪除，以及擷取和查詢指標。您可以在帳戶中的每個地區擁有一或多個工作區。

使用本節的程序來建立和管理您的 Amazon Managed Service for Prometheus 工作區。

主題



## 建立工作區

請遵循下列步驟來建立 Amazon Managed Service for Prometheus 工作區。您可以選擇使用 AWS CLI 或適用於 Prometheus 主控台的 Amazon 託管服務。

> **Note**
>
> 如果您正在執行 Amazon EKS 叢集，也可以使用適用於 Kubernetes 的AWS 控制器來建立新的工作區。

使用建立工作區的步驟 AWS CLI

1. 輸入下列命令以建立工作區。此範例建立名為 `my-first-workspace` 的工作區，但您可視需要使用不同的別名 (或不使用)。工作區別名是易記的名稱，可協助您識別工作區。名稱不必是唯一。兩個工作區可以有相同的別名，但是所有工作區都有唯一由 Amazon Managed Service for Prometheus 產生的工作區 ID。

   (選擇性) 若要使用您自己的 KMS 金鑰來加密儲存在工作區中的資料，您可以將`kmsKeyArn`參數與要使用的金 AWS KMS 鑰一起加入。雖然 Prometheus 的 Amazon 受管服務不會向您收取使用客戶受管金鑰的費用，但可能會產生與金鑰相關的費用。 AWS Key Management Service如需有關 Amazon Managed Service for Prometheus 在工作區中加密資料，或是如何建立、管理和使用自己的客戶受管金鑰的詳細資訊，請參閱 靜態加密。

方括號 ([]) 中的參數為選用，請不要在命令中包含方括號。

```
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]
```

此命令會傳回下列資料：

- workspaceId 是此工作區的唯一 ID。記下此 ID。
- arn 是此工作區的 ARN。
- status 是工作區目前的狀態。在您立即建立工作區後，這將會是 CREATING。
- kmsKeyArn 是客戶受管金鑰，用來加密工作區資料 (如有提供)。

> **Note**
>
> 使用客戶受管金鑰建立的工作區無法使用 AWS 受管收集器進行擷取。
> 選擇要小心使用客戶管理的金鑰還是 AWS 擁有的金鑰。使用客戶管理金鑰建立的工作區之後無法轉換為使用 AWS 擁有的金鑰 (反之亦然)。

- tags 列出工作區的標籤 (若有)。

2. 如果您 create-workspace 命令傳回的狀態為 CREATING，則可輸入下列命令來判斷工作區何時已就緒。*my-workspace-id*以create-workspace命令傳回的值取代workspaceId。

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

當 describe-workspace 命令針對 status 傳回 ACTIVE 時，工作區已準備使用就緒。

使用 Amazon Managed Service for Prometheus 主控台建立工作區

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇建立。
3. 針對工作區別名，輸入新工作區的別名。

   工作區別名是易記的名稱，可協助您識別工作區。名稱不必是唯一。兩個工作區可以有相同的別名，但是所有工作區都有唯一由 Amazon Managed Service for Prometheus 產生的工作區 ID。

4. (選擇性) 若要使用您自己的 KMS 金鑰來加密儲存在工作區中的資料，您可以選取 [自訂加密設定]，然後選擇要使用的金 AWS KMS 鑰 (或建立新金鑰)。您可以從下拉式清單中選擇帳戶中的金鑰，或輸入您可存取的任何金鑰的 ARN。雖然 Prometheus 的 Amazon 受管服務不會向您收取使用客戶受管金鑰的費用，但可能會產生與金鑰相關的費用。 AWS Key Management Service

   如需有關 Amazon Managed Service for Prometheus 在工作區中加密資料，或是如何建立、管理和使用自己的客戶受管金鑰的詳細資訊，請參閱 靜態加密。

   > **Note**
   >
   > 使用客戶受管金鑰建立的工作區無法使用 AWS 受管收集器進行擷取。
   > 選擇要小心使用客戶管理的金鑰還是 AWS 擁有的金鑰。使用客戶管理金鑰建立的工作區之後無法轉換為使用 AWS 擁有的金鑰 (反之亦然)。

5. (選用) 若要將一個或多個標籤新增至工作區，請選擇新增標籤。之後，在 Key (索引鍵) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。

   若要新增另一個標籤，請再次選擇 Add new tag (新增標籤)。

6. 選擇建立工作區。

   隨即顯示工作區詳細資訊頁面。這會針對遠端寫入和查詢顯示此工作區的資訊，包含狀態、ARN、工作區 ID 和端點 URL。

   狀態會傳回 CREATING，直到工作區準備就緒。請等待狀態為啟用中，然後再繼續設定指標擷取。

   請記錄針對端點 - 遠端寫入 URL 和端點 - 查詢 URL 顯示的 URL。當您將 Prometheus 伺服器設定為將指標遠端寫入此工作區，以及查詢這些指標時，將會需要這些指標。

如需有關將指標擷取至工作區的資訊，請參閱 將 Prometheus 指標擷取至工作區。

## 編輯工作區

您可以編輯工作區以變更其別名。若要使用 AWS CLI變更工作區別名，請輸入下列命令。

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

使用 Amazon Managed Service for Prometheus 主控台編輯工作區

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇您要編輯工作區的工作區 ID，然後選擇編輯。
4. 輸入工作區的新別名，然後選擇儲存。

# 尋找工作區 ARN

您可以透過使用主控台或 AWS CLI，尋找 Amazon Managed Service for Prometheus 工作區 ARN。

使用 Amazon Managed Service for Prometheus 主控台尋找工作區 ARN

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇工作區的工作區 ID。

   工作區 ARN 會在 ARN 下顯示。

若要使用 AWS CLI 尋找工作區 ARN，請輸入下列指令。

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

在結果中尋找 `arn` 值。

# 刪除工作區

刪除工作區會刪除已擷取至其中的資料。

> **Note**
>
> 刪除 Prometheus 工作區的 Amazon 受管服務不會自動刪除正在抓取指標並將其傳送至工作區的任何 AWS 受管收集器。如需詳細資訊，請參閱 尋找並刪除湊集器。

若要使用刪除工作區 AWS CLI

使用下列命令：

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

使用 Amazon Managed Service for Prometheus 主控台刪除工作區

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇您要刪除工作區的工作區 ID，然後選擇刪除。
4. 在確認方塊中輸入 **delete**，然後選擇刪除。

# 擷取指標至您的工作區

您必須先將指標導入 Prometheus 工作區的 Amazon 受管服務，才能查詢或提醒這些指標。本節說明如何設定擷取指標至工作區。

> **Note**
>
> 根據預設，擷取至工作區的指標會儲存 150 天，然後自動刪除。此長度由可調配額控制。

有兩種方法可將指標擷取至 Amazon Managed Service for Prometheus 工作區。

- 使用 AWS 受管收集器 — 適用於 Prometheus 的 Amazon 受管服務提供全受管、無代理程式的抓取工具，可自動從您的 Amazon Elastic Kubernetes Service (Amazon EKS) 叢集擷取指標。抓取會自動從與 Prometheus 兼容的端點中提取指標。
- 使用客戶管理的收集器：您有許多管理自己收集器的選項。要使用的兩個最常用的收集器是安裝您自己的 Prometheus 執行個體、以代理程式模式執行，或使 AWS 用 Distro 來執行。 OpenTelemetry 下節會詳細說明這些內容。

  收集器會使用 Prometheus 遠端寫入功能，將指標傳送至 Amazon Managed Service for Prometheus。您可以使用 Prometheus 遠端寫入您自己的應用程式，將指標直接傳送至 Amazon Managed Service for Prometheus。有關直接使用遠端寫入和遠端寫入組態的更多詳細資訊，請參閱 Prometheus 說明文件中的 remote_write。

主題



# AWS 管理收集器

Amazon Managed Service for Prometheus 的常見使用案例是監控 Amazon Elastic Kubernetes Service (Amazon EKS) 管理的 Kubernetes 叢集。Kubernetes 叢集和 Amazon EKS 內執行的許多應用程式會自動匯出其指標，以供與 Prometheus 相容的湊集器存取。

> **Note**
>
> 在 Kubernetes 環境中執行的許多技術和應用程式都提供與 Prometheus 相容的指標。如需可用匯出工具的完整清單，請參閱 Prometheus 說明文件中的匯出工具和整合。

Amazon Managed Service for Prometheus 提供全受管、無代理程式的湊集器或收集器，可自動探索並提取與 Prometheus 相容的指標。您無需管理、安裝、修補或維護代理程式或湊集器。Amazon Managed Service for Prometheus 收集器可為您的 Amazon EKS 叢集提供可靠、穩定、高可用性、自動擴展的指標集合。適用於 Prometheus 的 Amazon 託管服務受管收集器可與 Amazon EKS 叢集搭配使用，包括 EC2 和 Fargate。

Amazon Managed Service for Prometheus 收集器會在建立湊集器時，為指定的每個子網路建立彈性網路介面 (ENI)。收集器會透過這些 ENI 湊集指標，並使用 `remote_write` 將資料推送到使用 VPC 端點的 Amazon Managed Service for Prometheus 工作區。湊集的資料永遠不會在公有網際網路上傳輸。

下列主題提供有關如何在 Amazon EKS 叢集中使用 Amazon Managed Service for Prometheus 收集器，以及所收集指標的詳細資訊。

主題



## 使用 AWS 受管理的收集器

若要使用 Amazon Managed Service for Prometheus 收集器，您必須建立一個湊集器，以探索並提取 Amazon EKS 叢集中的指標。

- 您可以建立湊集器作為 Amazon EKS 叢集建立作業的一部份。如需有關建立 Amazon EKS 叢集 (包括建立抓取工具) 的詳細資訊，請參閱 Amazon EKS 使用者指南中的建立 Amazon EKS 叢集。
- 您可以使用 AWS API 以程式設計方式建立自己的抓取工具，或使用 AWS CLI.

> **Note**
>
> 使用客戶受管金鑰建立的 Prometheus 工作區適用的 Amazon 受管服務無法使用受 AWS 管收集器進行擷取。

Amazon Managed Service for Prometheus 收集器會抓取與 Prometheus 相容的指標。如需 Prometheus 相容指標的詳細資訊，請參閱 什麼是與 Prometheus 相容的指標？。

下列主題說明如何建立、管理和設定湊集器。

主題

- 建立湊集器
- 設定 Amazon EKS 叢集
- 尋找並刪除湊集器
- 湊集器組態
- 對湊集器組態進行移難排解
- 湊集器限制

## 建立湊集器

Amazon Managed Service for Prometheus 收集器包含一個湊集器，可從 Amazon EKS 叢集中探索和收集指標。Amazon Managed Service for Prometheus 可為您管理湊集器，提供所需的可擴展性、安全性和可靠性，而無需自行管理任何執行個體、代理程式或湊集器。

當您透過 Amazon EKS 主控台建立 Amazon EKS 叢集時，將會自動為您建立湊集器。但是，在某些情況下，您可能需要自己建立湊集器。例如，如果您想要將 AWS 受管收集器新增至現有 Amazon EKS 叢集，或想要變更現有收集器的組態。

您可以使用 AWS API 或 AWS CLI.

您需先滿足幾項先決條件，才能建立自己的湊集器：

- 您必須已建立 Amazon EKS 叢集。
- 您的 Amazon EKS 叢集必須設定叢集端點存取控制以包含私有存取。它可以包括私有和公有，但必須包含私有。

> **Note**
>
> 該集群將通過其 Amazon 資源名稱（ARN）與抓取器相關聯。如果您刪除叢集，然後建立具有相同名稱的新叢集，ARN 將會重複用於新叢集。因此，抓取工具將嘗試收集新集群的指標。您可以分別刪除抓取工具以及刪除叢集。

### AWS API

#### 使用 AWS API 建立抓取工具

使用 `CreateScraper` API 作業可使用 AWS API 建立湊集器。以下範例會在 `us-west-2` 地區中建立湊集器。您需要使用自己的 ID 取代 AWS 帳戶、工作區、安全性和 Amazon EKS 叢集資訊，並提供用於抓取工具的組態。

> **Note**
>
> 您必須包含至少兩個子網路，至少位於兩個可用區域。

`scrapeConfiguration` 是一個 base64 編碼的 Prometheus 組態 YAML 檔案。您可以透過 `GetDefaultScraperConfiguration` API 作業下載一般用途設定。如需有關的格式的更多資訊`scrapeConfiguration`，請參閱湊集器組態。

```
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

若要使用建立抓取工具 AWS CLI

使用`create-scraper`指令建立抓取工具 AWS CLI。以下範例會在 `us-west-2` 地區中建立湊集器。您需要使用自己的 ID 取代 AWS 帳戶、工作區、安全性和 Amazon EKS 叢集資訊，並提供用於抓取工具的組態。

> **Note**
>
> 您必須包含至少兩個子網路，至少位於兩個可用區域。

`scrape-configuration` 是一個 base64 編碼的 Prometheus 組態 YAML 檔案。您可以使用`get-default-scraper-configuration`指令下載一般用途組態。如需有關的格式的更多資訊`scrape-configuration`，請參閱湊集器組態。

```
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'}"
```

下列是您可以與 AWS API 一併使用的完整湊集器作業清單：

- 使用 CreateScraperAPI 操作創建抓取工具。
- 使用 ListScrapersAPI 操作列出現有的抓取工具。
- 使用 DeleteScraperAPI 操作刪除抓取工具。

- 使用 DescribeScraperAPI 操作獲取有關抓取工具的更多詳細信息。
- 使用 GetDefaultScraperConfigurationAPI 操作獲取抓取工具的通用配置。

> **Note**
>
> 必須設定您要湊集的 Amazon EKS 叢集以讓 Amazon Managed Service for Prometheus 存取指標。下一個主題說明如何設定叢集。

創建抓取工具時的常見錯誤

以下是嘗試創建新抓取工具時最常見的問題。

- 所需的 AWS 資源不存在。指定的安全群組、子網路和 Amazon EKS 叢集必須存在。
- IP 位址空間不足。您必須在傳遞至 `CreateScraper` API 的每個子網路中至少有一個可用的 IP 位址。

## 設定 Amazon EKS 叢集

必須設定您的 Amazon EKS 叢集以讓湊集器存取指標。此組態有兩個選項：

- 使用 Amazon EKS 存取項目，自動為 Prometheus 收集器提供 Amazon 受管服務存取您叢集的存取權。
- 手動設定 Amazon EKS 叢集以進行受管指標抓取。

下列主題會更詳細地說明這些主題。

使用存取項目設定 Amazon EKS 以進行抓取工具存取

使用 Amazon EKS 的存取項目是讓 Prometheus 的 Amazon 受管服務存取權從叢集抓取指標的最簡單方法。

您要抓取的 Amazon EKS 叢集必須設定為允許 API 身份驗證。叢集驗證模式必須設定為`API`或`API_AND_CONFIG_MAP`。這可在叢集詳細資料的存取組態索引標籤上的 Amazon EKS 主控台中檢視。如需詳細資訊，請參閱 Amazon EKS 使用者指南中的允許 IAM 角色或使用者存取 Amazon EKS 叢集上的 Kubernetes 物件。

您可以在建立叢集時或建立叢集之後建立抓取工具：

- 建立叢集時 — 您可以在透過 Amazon EKS 主控台建立 Amazon EKS 叢集時設定此存取權 (依照指示在叢集中建立抓取工具)，並自動建立存取項目政策，讓 Prometheus 的 Amazon 受管服務可存取叢集指標。
- 在叢集建立後新增 — 如果您的 Amazon EKS 叢集已存在，請將身份驗證模式設定為API或API_AND_CONFIG_MAP，您透過 Prometheus API 或 CLI 的 Amazon 受管服務建立的任何抓取工具都會自動為您建立正確的存取輸入政策，而且抓取工具將可以存取您的叢集。

存取項目原則已建立

當您創建抓取工具並讓 Prometheus 的 Amazon 託管服務為您生成訪問輸入政策時，它會生成以下策略。如需有關存取項目的詳細資訊，請參閱 Amazon E KS 使用者指南中的允許 IAM 角色或使用者存取 Kubernetes。

```
{
    "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",
            "nonResourceURLs": [
                "/metrics"
            ],
            "verbs": [
                "get"
            ]
        }
    ]
}
```

手動配置 Amazon EKS 以進行刮板訪問

如果您偏好使用控制`aws-auth ConfigMap`對 kubernetes 叢集的存取，您仍然可以為 Prometheus 抓取工具授予 Amazon 受管服務存取您的指標。以下步驟將為 Prometheus 提供 Amazon 託管服務訪問權限，以從您的 Amazon EKS 集群抓取指標。

> **Note**
>
> 如需有關`ConfigMap`和存取項目的詳細資訊，請參閱 Amazon E KS 使用者指南中的允許 IAM 角色或使用者存取 Kubernetes。

此程序使用`kubectl`和 AWS CLI。如需有關安裝 `kubectl` 的資訊，請參閱《Amazon EKS 使用者指南》中的安裝 kubectl。

手動設定 Amazon EKS 叢集以進行受管指標抓取

1. 使用下列內文建立名為 `clusterrole-binding.yml` 的檔案：

```
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"]
---
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. 在叢集中執行下列命令：

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

這將建立叢集角色連結和規則。此範例使用 aps-collector-role 作為角色名稱和 aps-collector-user 作為使用者名稱。

3. 以下命令為您提供有關含有 ID 為 *scraper-id* 的資訊。這是您使用上一節命令建立的湊集器。

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

4. 在 describe-scraper 的結果中尋找 roleArn，其格式如下：

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

Amazon EKS 需要此 ARN 使用不同的格式。您必須調整傳回 ARN 的格式，以便在下一步中使用。編輯以比對此格式：

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

例如，此 ARN：

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

必須改寫為：

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

5. 使用上一個步驟中修改後的 `roleArn` 以及您的叢集名稱和區域，在叢集中執行下列命令：

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

這允許湊集器使用 `clusterrole-binding.yml` 檔案中建立的角色和使用者存取叢集。

## 尋找並刪除湊集器

您可以使用 AWS API 或列 AWS CLI 出帳戶中的抓取工具或刪除它們。

> **Note**
>
> 請確定您使用的是最新版本的 AWS CLI 或 SDK。最新版本為您提供最新的特色和功能，以及安全性更新。或者，使用 AWS Cloudshell，它可以自動提供始終 up-to-date 命令列體驗。

要列出您帳戶中的所有抓取工具，請使用 ListScrapersAPI 操作。

或者，使用 AWS CLI呼叫：

```
aws amp list-scrapers
```

ListScrapers 會傳回您帳戶中的所有湊集器，例如：

```
{
    "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"
                }
            }
        }
    ]
}
```

要刪除抓取工具，scraperId請使用該ListScrapers操作查找要刪除的抓取工具，然後使用該DeleteScraper操作將其刪除。

或者，使用 AWS CLI呼叫：

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

## 湊集器組態

您可以使用與 Prometheus 相容的湊集器組態控制湊集器如何探索和收集指標。例如，您可以變更將指標傳送至工作區的間隔。您也可以使用重新標籤來動態重新寫入指標的標籤。湊集器組態是一個 YAML 檔案，屬於湊集器定義的一部份。

建立新的湊集器時，您可以透過在 API 呼叫中提供 base64 編碼的 YAML 檔案來指定組態。您可以透過 Amazon Managed Service for Prometheus API 中的 GetDefaultScraperConfiguration 作業下載一般用途組態檔案。

若要修改湊集器的組態，請刪除湊集器並使用新組態重新建立。

### 支援的組態

如需有關抓取工具組態格式的資訊，包括可能值的詳細明細資訊，請參閱 Prometheus 文件中的組態。全域組態選項和 <scrape_config> 選項說明最常需要的選項。

由於 Amazon EKS 是唯一受支援的服務，因此唯一支援的服務探索設定 (<*_sd_config>) 是. <kubernetes_sd_config>

允許配置部分的完整列表：

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

這些區段中的限制會列在範例組態檔案之後。

### 範例組態檔案

以下是具有 30 秒湊集間隔的範例 YAML 組態檔。

```
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
```

下列是 AWS 受管理收集器的特定限制：

- 湊集間隔：湊集器組態無法指定少於 30 秒的湊集間隔。
- 目標：`static_config` 中的目標必須指定為 IP 地址。
- 授權 — 如果不需要授權，則省略。如果需要，授權必須是`Bearer`，並且必須指向該文件`/var/run/secrets/kubernetes.io/serviceaccount/token`。換句話說，如果使用，授權部分必須如下所示：

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

  > **Note**
  >
  > `type: Bearer`是預設值，因此可以省略。

## 對湊集器組態進行移難排解

Amazon Managed Service for Prometheus 收集器會自動探索和湊集指標。但是，若未在 Amazon Managed Service for Prometheus 工作區中看到您希望看到的指標，該如何進行疑難排解？

up 指標是一個有助益的工具。針對 Amazon Managed Service for Prometheus 收集器探索的每個端點，皆會自動分配此指標。此指標有三種狀態，可協助您疑難排解收集器內所發生的事情。

- up 不存在 - 若端點沒有 up 指標，則這表示收集器找不到端點。

  如果您確定端點已存在，則可能需要調整湊集組態。探索 `relabel_config` 可能需要調整，或者用於探索的 `role` 可能出現問題。

- up 已存在，但始終為 0 — 如果 up 已存在但為 0，則收集器能夠探索端點，但找不到任何與 Prometheus 相容的指標。

  在這種情況下，您可以嘗試直接對端點使用 `curl` 命令。您可以驗證您的詳細資料是否正確，例如您使用的通訊協定 (`http`或`https`)、端點或連接埠。您也可以檢查端點是否有效回`200`應，並遵循 Prometheus 格式。最後，響應的主體不能大於允許的最大大小。(如需 AWS 受管理之收集器的限制，請參閱下一節。)

- up 已存在且大於 0 — 若 up 已存在且大於 0，則指標會傳送至 Amazon Managed Service for Prometheus。

  驗證您正在尋找 Amazon Managed Service for Prometheus (或您的替代儀表板，例如 Amazon Managed Grafana) 中的正確指標。您可以再次使用 curl 來檢查`/metrics`端點中的預期數據。同時檢查您是否未超過其他限制，例如每個湊集器的端點數量。您可以使`count(up)`用檢查指標計數來檢查正在抓取的up指標端點數量。

## 湊集器限制

Amazon Managed Service for Prometheus 所提供全受管湊集器的限制較少。

- 區域：您的 EKS 叢集、受管理湊集器和 Amazon Managed Service for Prometheus workspace 必須位於相同 AWS 區域。
- 帳戶：您的 EKS 叢集、受管湊集器和 Amazon Managed Service for Prometheus 工作區都必須處於相同 AWS 帳戶。
- 收集器：各帳戶最多可以為每個區域提供 10 個 Amazon Managed Service for Prometheus 湊集器。

> **Note**
>
> 您可以透過請求增加配額來請求增加此限制。

- 指標回應：來自任何一個 `/metrics` 端點請求的回應主體不能超過 50 MB。
- 每個湊集器的端點：湊集器最多可以湊集 30,000 `/metrics` 個端點。
- 湊集間隔：湊集器組態無法指定少於 30 秒的湊集間隔。

## 什麼是與 Prometheus 相容的指標？

若要從您的應用程式和基礎設施中湊集 Prometheus 指標以用於 Amazon Managed Service for Prometheus，他們必須從與 Prometheus 相容的 `/metrics` 個端點中檢測並公開與 Prometheus 相容的指標。您可以建置自己的指標，但不必這樣做。Kubernetes (包括 Amazon EKS) 和許多其他程式庫和服務會直接建置這些指標。

將 Amazon EKS 中的指標匯出到與 Prometheus 相容的端點時，您可以讓 Amazon Managed Service for Prometheus 收集器自動湊集這些指標。

如需詳細資訊，請參閱下列主題：

- 如需有關將指標匯出為 Prometheus 指標的現有程式庫和服務詳細資訊，請參閱 Prometheus 說明文件中的匯出程式和整合。
- 如需有關從您自己的程式碼匯出與 Prometheus 相容指標的詳細資訊，請參閱 Prometheus 文件中的撰寫匯出程式。
- 如需有關如何設定 Amazon Managed Service for Prometheus 收集器以自動從 Amazon EKS 叢集湊集指標的詳細資訊，請參閱 使用 AWS 受管理的收集器。

## 客戶受管收集器

本節包含透過設定您自己的收集器來擷取資料的相關資訊，這些收集器會使用 Prometheus 遠端寫入將指標傳送至 Amazon Managed Service for Prometheus。

當您使用自己的收集器將指標傳送到 Amazon Managed Service for Prometheus 時，您負責保護指標的安全，並確保擷取程序符合您的可用性需求。

大多數客戶管理的收集器都使用下列其中一種工具：

- AWS 適用於 OpenTelemetry (ADOT) 的發行版 — ADOT 是完全支援、安全、生產就緒的開放原始碼發行版，可 OpenTelemetry 讓代理程式收集指標。您可以使用 ADOT 收集指標，並將其傳送至您的 Amazon Managed Service for Prometheus 工作區。有關 ADOT 收集器的更多信息，請參閱 AWS . OpenTelemetry
- Prometheus 代理程式 - 您可以設定自己的開放原始碼 Prometheus 伺服器執行個體 (以客服人員身分執行)，以收集指標並將其轉寄至 Amazon Managed Service for Prometheus。

下列主題說明如何使用這兩個工具，並包括設定自己收集器的一般資訊。

主題



# 保護您的指標擷取作業

Amazon Managed Service for Prometheus 提供協助您保護指標擷取的方法。

## AWS PrivateLink 與 Prometheus 的 Amazon 託管服務一起使用

將指標導入 Prometheus 的 Amazon 受管服務的網路流量，可透過公用網際網路端點或透過 VPC 端點完成。 AWS PrivateLink使用 AWS PrivateLink 可確保來自 VPC 的網路流量在 AWS 網路中受到保護，而無需透過公用網際網路。若要為 Prometheus 的 Amazon 受管服務建立 AWS PrivateLink VPC 端點，請參閱。使用 Amazon Managed Service for Prometheus 和介面 VPC 端點

## 身份驗證和授權

AWS Identity and Access Management (IAM) 是一種 Web 服務，可協助您安全地控制 AWS 資源存取。您可以使用 IAM 來控制能通過身分驗證 (登入) 和授權使用資源的 (具有許可) 的人員。Amazon Managed Service for Prometheus 與 IAM 整合，協助您確保資料安全。當您設定 Amazon Managed Service for Prometheus 時，您需要建立一些 IAM 角色，使其能夠從 Prometheus 伺服器擷取指標，並讓 Grafana 伺服器查詢存放在 Amazon Managed Service for Prometheus 工作區中的指標。如需有關 IAM 的相關資訊，請參閱什麼是 IAM？。

另一個可協助您為 Prometheus 設定 Amazon 受管服務的 AWS 安全功能是簽 AWS 名版本 4 簽署程序 (AWS Sigv4)。簽名版本 4 是將身份驗證信息添加到由 HTTP 發送的 AWS 請求的過程。為了安全起見，大多數的請求都 AWS 必須使用訪問密鑰進行簽名，該訪問密鑰包括訪問密鑰 ID 和秘密訪問密鑰。這兩種金鑰通常稱為您的安全憑證。如需有關 SigV4 的詳細資訊，請參閱簽章第 4 版簽署程序。

# 使用 AWS 發行版 OpenTelemetry 作為收集器

以下主題描述了將 AWS Distro 設置 OpenTelemetry為指標收集器的不同方法。

主題

- 使用遠端寫入設定 Amazon EC2 執行個體擷取的指標

## 使用 Amazon Elastic Kubernetes Service 叢集上的開放式遙測發行 AWS 版來設定指標擷取

本節 AWS 介紹如何將發行版 OpenTelemetry （ADOT）收集器配置為從 Prometheus 儀器的應用程序中抓取，並將指標發送到 Prometheus 的 Amazon 託管服務。有關 ADOT 收集器的更多信息，請參閱 AWS . OpenTelemetry

使用 ADOT 收集 Prometheus 指標涉及三個 OpenTelemetry 元件：Prometheus 接收器、Prometheus 遠端寫入匯出程式和 Sigv4 驗證延伸模組。

您可以使用現有的 Prometheus 組態，來組態 Prometheus 接收器來執行服務探索和指標抓取。Prometheus 接收器會以 Prometheus 展開格式抓取指標。您要抓取的任何應用程式或端點都應使用 Prometheus 用戶端程式庫進行組態。Prometheus 接收器在 Prometheus 說明文件中，支援組態中說明的全套 Prometheus 抓取和重新標籤組態。您可以將這些組態直接貼到您的 ADOT 收集器組態中。

Prometheus 遠端寫入匯出程式會使用 `remote_write` 端點將抓取的指標傳送至您的管理入口網站工作區。匯出資料的 HTTP 要求將使用 AWS Sigv4 (用於安全驗證的 AWS 通訊協定) 與 Sigv4 驗證延伸模組簽署。如需詳細資訊，請參閱簽章版本 4 簽署程序。

收集器會自動探索 Amazon EKS 上的 Prometheus 指標端點，並使用<kubernetes_sd_config>中找到的組態。

以下示範是在執行 Amazon Elastic Kubernetes Service 或自我管理 Kubernetes 的叢集上進行此組態的範例。若要執行這些步驟，您必須擁有預設認 AWS 證鏈結中任何可能選項的認 AWS 證。如需詳細資訊，請參閱 < 設定 AWS SDK for Go > 。此示範使用範例應用程式，用於程序的整合測試。範例應用程式會在 `/metrics` 端點公開指標，例如 Prometheus 用戶端程式庫。

必要條件

在開始下列擷取設定步驟之前，您必須為服務帳戶和信任政策設定 IAM 角色。

設定服務帳戶和信任政策的 IAM 角色

1. 遵循 自 Amazon EKS 叢集設定指標擷取作業的服務角色 中的步驟，為服務帳戶建立 IAM 角色。

   當 ADOT 收集器抓取並匯出指標時，將使用此角色。
2. 接下來，編輯信任政策。前往 https://console.aws.amazon.com/iam/ 開啟 IAM 主控台。

3. 在左側導覽窗格中，選擇 [角色]，然後尋找您amp-iamproxy-ingest-role在步驟 1 中建立的角色。
4. 選擇信任關係索引標籤，然後選擇編輯信任關係。
5. 在信任關係政策 JSON 中，取代 aws-amp 為 adot-col，然後選擇更新信任原則。結果信任政策應如下所示：

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

6. 選擇權限索引標籤，並確定已將下列權限政策附加至該角色。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:RemoteWrite",
                "aps:GetSeries",
                "aps:GetLabels",
                "aps:GetMetricMetadata"
            ],
            "Resource": "*"
        }
    ]
}
```

啟用 Prometheus 指標收集

> **Note**
>
> 當您在 Amazon EKS 中建立命名空間時，依預設 `alertmanager` 會停用節點匯出程式。

在 Amazon EKS 或 Kubernetes 叢集上啟用 Prometheus 集合

1. 從存儲庫中分叉並克隆示例應用程序，位於 aws-otel-community.

   然後，執行以下命令。

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

2. 將此映像推送到註冊表，例如 Amazon ECR 或 DockerHub.
3. 複製此 Kubernetes 組態並套用，在叢集中部署範例應用程式。通過在 `prometheus-sample-app.yaml` 檔案中替換 `{{PUBLIC_SAMPLE_APP_IMAGE}}` 將圖像變更為剛才推送的圖像。

   ```
   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. 輸入下列命令，確認已啟動範例應用程式。在命令的輸出中，您將在 NAME 欄中看到 `prometheus-sample-app`。

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

5. 啟動 ADOT 收集器的預設執行個體。若要執行此作業，請先輸入下列命令來拉取 ADOT 收集器的 Kubernetes 組態。

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

   然後編輯範本檔案，將您 `YOUR_ENDPOINT` 的 Amazon Managed Service for Prometheus 工作區以及您 `YOUR_REGION` 的區域替換 remote_write 端點。查看工作區詳細資料時，請使用在 Amazon Managed Service for Prometheus 主控台中顯示的 remote_write 端點。

您也必須將 Kubernetes 設定YOUR_ACCOUNT_ID的服務帳戶區段變更為您 AWS 的帳戶識別碼。

在此範例中，ADOT 收集器組態使用註解 (scrape=true) 來判斷要抓取哪些目標端點。這可讓 ADOT 收集器區分範例應用程式端點與叢集中的 kube 系統端點。若您要抓取不同範例應用程式，則可以將其從重新標籤組態中移除。

6. 輸入下列命令以部署 ADOT 收集器。

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

7. 輸入下列命令，確認已啟動 ADOT 收集器。在 NAMESPACE 欄中尋找 adot-col。

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

8. 使用記錄匯出程式確認管線是否正常運作。我們的範例範本已與記錄匯出程式整合。輸入下列命令：

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

範例應用程式中的某些抓取指標將依照以下範例所示：

```
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. 若要測試 Amazon Managed Service for Prometheus 是否收到指標，請使用 awscurl。此工具可讓您使用 AWS Sigv4 身份驗證透過命令列傳送 HTTP 請求，因此您必須在本機設定具有

正確許可的 AWS 登入資料，才能從 Prometheus 的 Amazon 受管服務進行查詢。如需有關安裝awscurl的指示，請參閱 awscurl。

在下列命令中，將 AMP_REGION 和 AMP_ENDPOINT 替換為 Amazon Managed Service for Prometheus 工作區的資訊。

```
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"]}]}}
```

如果您收到指標作為回應，則表示管道設定已成功，且指標已成功從範例應用程式傳播到 Amazon Managed Service for Prometheus。

清除

若要清理此示範，請輸入下列命令。

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

進階組態

Prometheus 接收器在 Prometheus 說明文件中，支援組態中說明的全套 Prometheus 抓取和重新標籤組態。您可以將這些組態直接貼到您的 ADOT 收集器組態中。

Prometheus 接收器的組態包括您的服務探索、抓取組態和重新標籤組態。接收器組態如下所示。

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

以下是範例組態。

```
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
```

如果您有現有的 Prometheus 組態，則必須將 $ 個字元替換為 $$ 以避免將值替換為環境變數。* 這對於 relabel_configurations 的替換值特別重要。例如，若您開始使用下列 relabel_configuration：

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

它將成為以下幾點：

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

Prometheus 遠端寫入匯出程式和 Sigv4 身分驗證延伸

Prometheus 遠端寫入匯出程式和 Sigv4 身分驗證延伸的設定較 Prometheus 接收器容易。在此管道階段已擷取指標，我們已準備好將這些資料匯出到 Amazon Managed Service for Prometheus。下列範例顯示成功組態與 Amazon Managed Service for Prometheus 通訊的最低需求。

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

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

此設定會傳送由 AWS Sigv4 使用預設認證鏈結中的認 AWS AWS 證簽署的 HTTPS 要求。如需詳細資訊，請參閱設定 AWS SDK for Go。您必須將服務指定為 aps。

無論採用何種部署方法，ADOT 收集器都必須能夠存取預設認 AWS 證鏈結中列出的其中一個選項。Sigv4 驗證延伸模組取決於 AWS SDK for Go 並使用它來擷取認證和驗證。您必須確保這些憑證有 Amazon Managed Service for Prometheus 的遠端寫入權限。

## 使用開放式遙測發行 AWS 版設定從 Amazon ECS 擷取指標

本節說明如何從 Amazon Elastic Container Service (Amazon ECS) 收集指標，並使用開放遙測發行 AWS 版 (ADOT) 將這些指標導入到適用於 Prometheus 的 Amazon 受管服務。這同時說明如何在 Amazon Managed Grafana 中將指標視覺化。

必要條件

> **Important**
>
> 在開始之前，您必須在具有預設設定的 AWS Fargate 叢集上擁有 Amazon ECS 環境、Amazon Managed Service for Prometheus 工作區，以及 Amazon Managed Grafana 工作區。我們假設您熟悉容器工作負載、Amazon Managed Service for Prometheus，以及 Amazon Managed Grafana。

如需詳細資訊，請參閱下列連結：

- 如需如何使用預設設定在 Fargate 叢集上建立 Amazon ECS 環境的詳細資訊，請參閱《Amazon ECS 開發人員指南》中的建立叢集。
- 如需如何建立 Amazon Managed Service for Prometheus 的詳細資訊，請參閱《Amazon Managed Service for Prometheus 使用者指南》中的建立工作區。
- 如需如何建立 Amazon Managed Grafana 工作區的詳細資訊，請參閱《Amazon Managed Grafana 使用者指南》中的建立工作區。

### 定義自訂的 ADOT 收集器容器映像

使用下列組態檔作為範本，定義您自己的 ADOT 收集器容器映像檔。將 *my-remote-URL* 和 *my-region* 替換為您的 `endpoint` 和 `region` 值。將組態儲存在一個名為 adot-config.yaml 檔案中的組態。

> **Note**
>
> 此組態使用 `sigv4auth` 延伸來驗證 Amazon Managed Service for Prometheus 的呼叫。如需有關設定的詳細資訊`sigv4auth`，請參閱驗證器-Sigv4 開啟。 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]
```

將您的 ADOT 收集器容器映像推送至 Amazon ECR 儲存庫

使用 Dockerfile 建立容器映像，然後將其推送至 Amazon Elastic Container Registry (ECR) 儲存庫。

1. 建立 Dockerfile 以複製和新增您的容器映像檔並將其新增至 OTEL Docker 映像檔中。

```
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. 建立 Amazon ECR 儲存庫。

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

3. 建立容器映像。

```
# build ADOT collector image:
```

```
docker build -t $COLLECTOR_REPOSITORY:ecs .
```

> **Note**
>
> 這假設您正在執行容器的相同環境中建構容器。若否，您可能需要在建立映像時使用 `--platform` 參數。

4. 登入 Amazon ECR 儲存庫。將 *my-region* 替換為您的 `region` 值。

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

5. 推送您的容器映像。

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

### 建立 Amazon ECS 任務定義來抓取 Amazon Managed Service for Prometheus

建立 Amazon ECS 任務定義來抓取 Amazon Managed Service for Prometheus。您的工作定義應包含名為 `adot-collector` 的容器和名為 `prometheus` 的容器。`prometheus` 產生指標，和 `adot-collector` 抓取 `prometheus`。

> **Note**
>
> Amazon Managed Service for Prometheus 以服務的形式執行，並從容器收集指標。在這種情況下，容器會以代理程式模式在本端執行 Prometheus，並將本端指標傳送至 Amazon Managed Service for Prometheus。

#### 範例：任務定義

以下為任務定義外觀的範例。您可以使用此範例作為建立您任務定義的範本。將 `adot-collector` 的 `image` 值替換為儲存庫 URL 和映像標籤 (`$COLLECTOR_REPOSITORY:ecs`)。將 `adot-collector` 和 `prometheus` 的 `region` 個值替換為 `region` 個值。

```
{
```

```
  "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"
}
```

將 AWS 受管政策 **AmazonPrometheusRemoteWriteAccess** 附加到任務的 IAM 角色

若要將抓取的指標傳送到適用於 Prometheus 的 Amazon 受管服務，您的 Amazon ECS 任務必須具有正確的許可，才能為您呼叫 API 操作。 AWS 您必須為任務建立 IAM 角色，並將 AmazonPrometheusRemoteWriteAccess 政策附加至 IAM 角色。如需有關建立此角色並附加政策的詳細資訊，請參閱為任務建立 IAM 角色和政策。

在您將 AmazonPrometheusRemoteWriteAccess 附加至 IAM 角色並將該角色用於您的任務之後，Amazon ECS 可以將您抓取的指標傳送到 Amazon Managed Service for Prometheus。

在 Amazon Managed Grafana 中視覺化您的指標

> **Important**
>
> 在開始之前，您必須在 Amazon ECS 任務定義中執行 Fargate 任務。否則，Amazon Managed Service for Prometheus 將無法使用您的指標。

1. 在 Amazon 受管的 Grafana 工作區中的導覽窗格中，選擇圖示 AWS 下的資料來源。
2. 在資料來源索引標籤上，針對服務選取 Amazon Managed Service for Prometheus，然後選擇您的預設區域。
3. 選擇 [新增資料來源]。
4. 使用 ecs 和 prometheus 個前綴查詢和檢視您的指標。

## 使用遠端寫入設定 Amazon EC2 執行個體擷取的指標

本節說明如何在 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體中使用遠端寫入執行 Prometheus 伺服器。這會說明如何從使用 Go 編寫的示範應用程式收集指標，並將其傳送到 Amazon Managed Service for Prometheus 工作區。

必要條件

> **Important**
>
> 在開始前，您必須安裝 Prometheus v2.26 或更高版本。我們假設您熟悉 Prometheus、Amazon EC2 和 Amazon Managed Service for Prometheus。有關如何安裝 Prometheus 的訊息，請參閱 Prometheus 網站上的入門。

如果您不熟悉 Amazon EC2 或 Amazon Managed Service for Prometheus，建議您先從閱讀以下各節開始：

- Amazon Elastic Compute Cloud 是什麼？
- Amazon Managed Service for Prometheus 是什麼？

### 建立 Amazon EC2 的 IAM 角色

若要串流指標，您必須先建立具有 AWS 受管政策的 IAM 角色AmazonPrometheusRemoteWriteAccess。然後，您可以啟動具有角色的執行個體，並將指標串流到 Amazon Managed Service for Prometheus 工作區。

1. 前往 https://console.aws.amazon.com/iam/ 開啟 IAM 主控台。
2. 從導覽窗格，選擇 Roles (角色)，然後選擇 Create role (建立角色)。
3. 對於信任的實體類型，選擇 AWS service (AWS 服務)。針對使用案例，選擇 EC2。選擇下一步：許可。
4. 在搜尋列中，輸入 AmazonPrometheusRemoteWriteAccess。針對「策略名稱」，選取 AmazonPrometheusRemoteWriteAccess，然後選擇「附加策略」。選擇 Next: Add Tags (下一步：新增標籤)。
5. (選用) 為您的 IAM 角色建立 IAM 標籤。選擇 [下一步：檢閱]。
6. 輸入您的角色名稱。選擇建立政策。

### 啟動 Amazon EC2 執行個體

若要啟動 Amazon EC2 執行個體，請遵循《適用於 Linux 執行個體的 Amazon Elastic Compute Cloud 使用者指南》中啟動執行個體的指示。

### 執行示範應用程式

建立 IAM 角色並啟動具有該角色的 EC2 執行個體之後，您可以執行示範應用程式來查看其運作狀態。

### 執行示範應用程式和測試指標

1. 使用下列範本建立名為 `main.go` 的 Go 檔案。

```
package main

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

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

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

2. 執行以下命令以安裝正確相依項目。

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

3. 執行示範應用程式。

```
go run main.go
```

展示應用程式應該在連接埠 8000 上運行，並顯示所有暴露的 Prometheus 指標。以下是這些指標的範例。

```
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
```

建立 Amazon Managed Service for Prometheus 工作區

若要建立 Amazon Managed Service for Prometheus 工作區，請按照建立工作區中的指示操作。

執行 Prometheus 伺服器

1. 使用下列範例 YAML 檔案作為範本，以建立名為 `prometheus.yaml` 的新檔案。對於`url`，請將我的*##*值取代為您的區域值，並*my-workspace-id*使用適用於 Prometheus 的 Amazon 代管服務為您產生的工作區 ID 來取代我的區域。針對 `region`，將 *my-region* 替換為您的地區值。

   範例：YAML 檔案

   ```
   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. 執行 Prometheus 伺服器，將示範應用程式的指標傳送至您的 Amazon Managed Service for Prometheus 工作區。

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

Prometheus 伺服器現在應該會將示範應用程式的指標傳送到您的 Amazon Managed Service for Prometheus 工作區。

# 使用 Prometheus 執行個體作為收集器

下列主題說明如何針對在代理程式模式下執行的 Prometheus 執行個體，設定作為指標的收集器的不同方法。

> **Warning**
>
> 通過啟用安全特徵，避免將 Prometheus 抓取端點暴露到公共網路。

如果您設定多個 Prometheus 執行個體來監控同一組指標，並將其傳送到單一 Amazon Managed Service for Prometheus 以獲得高可用性，則需要設定重複資料刪除功能。若未按照步驟設定重複資料刪除功能，則會向您收取傳送至 Amazon Managed Service for Prometheus 的所有資料樣本費用，包括重複樣本。如需有關設定重複資料刪除的指示，請參閱 將傳送至 Amazon Managed Service for Prometheus 的高可用性指標刪除重複資料。

主題

- 設定使用 Helm 從新的 Prometheus 伺服器擷取
- 在 EC2 上 Kubernetes 中設定現有 Prometheus 伺服器的擷取作業
- 在 Fargate 的 Kubernetes 從現有的 Prometheus 伺服器設定擷取作業

## 設定使用 Helm 從新的 Prometheus 伺服器擷取

本節中的指示可協助您快速啟動，並使用 Amazon Managed Service for Prometheus 執行。您在 Amazon EKS 叢集中設定新的 Prometheus 伺服器，新伺服器會使用預設組態將指標傳送至 Amazon Managed Service for Prometheus。此主題有以下先決條件：

- 您必須擁有 Amazon EKS 叢集，在其中收集新的 Prometheus 伺服器指標
- 您必須使用 Helm CLI 3.0 或更新版本
- 您必須使用 Linux 或 macOS 電腦來執行以下各節中的步驟

### 步驟 1：新增 Helm Chart 儲存庫

若要新增 Helm Chart 儲存庫，請輸入下列命令。如需有關這些命令的詳細資訊，請參閱 Helm 儲存庫。

```
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
```

步驟 2：建立 Prometheus 命名空間

輸入下列命令，為 Prometheus 伺服器和其他監控元件建立 Prometheus 命名空間。將 *prometheus-namespace* 替換為您希望此命名空間的名稱。

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

步驟 3：為服務帳戶設定 IAM 角色

若為我們正在記錄的入職方法，您需要在執行 Prometheus 伺服器的 Amazon EKS 叢集中使用服務帳戶的 IAM 角色。

透過服務帳戶的 IAM 角色，您可以產生 IAM 角色與 Kubernetes 服務帳戶的關聯。然後，此服務帳戶可以為使用該服務帳戶之任何 Pod 中的容器提供 AWS 許可。如需詳細資訊，請參閱服務帳戶的 IAM 角色。

如果您尚未設定這些角色，請按照中的 自 Amazon EKS 叢集設定指標擷取作業的服務角色 指示設定角色。本節中的說明需要使用 `eksctl`。如需詳細資訊，請參閱 Amazon Elastic Kubernetes Service 入門 - `eksctl`。

> **Note**
>
> 如果您不在 EKS 上，或 AWS 僅使用訪問密鑰和密鑰訪問 Prometheus 的 Amazon 託管服務，則無法使用基於 Sigv4。`EKS-IAM-ROLE`

步驟 4：設定新伺服器並開始擷取指標

若要安裝新的 Prometheus 伺服器，該伺服器會將指標傳送至您的 Amazon Managed Service for Prometheus 工作區，請按照下列步驟操作。

安裝新的 Prometheus 伺服器，以將指標傳送至您的 Amazon Managed Service for Prometheus 工作區

1. 使用文字編輯器建立名為 `my_prometheus_values_yaml` 的檔案，包含下列內容。

   - 以您在中建立的 ARN 取代 *IAM_PROMETHEUS_ROL_ARN*。amp-iamproxy-ingest-role自 Amazon EKS 叢集設定指標擷取作業的服務角色

- 將 *WORKSPACE_ID* 替換為您 Amazon Managed Service for Prometheus 工作區的 ID。
- 將 *REGION* 替換為您 Amazon Managed Service for Prometheus 工作區的區域。

```
## 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. 輸入下列命令以建立 Prometheus 伺服器。

- 請*prometheus-chart-name*以您的 Prometheus 版本名稱取代。
- 將 *prometheus-namespace* 替換為您的 Prometheus 命名空間的名稱。

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

> **Note**
>
> 您可以使用多種方式自訂 `helm install` 命令。如需詳細資訊，請參閱 Helm 文件中的 Helm 安裝。

# 在 EC2 上 Kubernetes 中設定現有 Prometheus 伺服器的擷取作業

Amazon Managed Service for Prometheus 支援擷取 Prometheus 伺服器的指標，位於執行 Amazon EKS 的叢集以及在 Amazon EC2 中執行的自我管理 Kubernetes。本節中的詳細說明適用於 Amazon EKS 叢集中的 Prometheus 伺服器。除了您將需要在 Kubernetes 叢集中自行設定服務帳戶的 OIDC 提供者和 IAM 角色以外，Amazon EC2 上的自我管理 Kubernetes 叢集步驟皆相同。

本節中的指示使用 Helm 做為 Kubernetes 套件管理員。

主題



## 步驟 1：為服務帳戶設定 IAM 角色

若為我們正在記錄的入職方法，您需要在執行 Prometheus 伺服器的 Amazon EKS 叢集中使用服務帳戶的 IAM 角色。這些角色也稱為服務角色。

透過服務角色，您可以產生 IAM 角色與 Kubernetes 服務帳戶的關聯。然後，此服務帳戶可以為使用該服務帳戶的任何網繭中的容器提供 AWS 權限。如需詳細資訊，請參閱服務帳戶的 IAM 角色。

如果您尚未設定這些角色，請按照中的 自 Amazon EKS 叢集設定指標擷取作業的服務角色 指示設定角色。

## 步驟 2：使用 Helm 升級您現有的 Prometheus 伺服器

本節中的說明包括設定遠端寫入和 sigv4 以進行驗證，並授權 Prometheus 伺服器遠端寫入 Amazon Managed Service for Prometheus 工作區。

使用 Prometheus 版本 2.26.0 或更新版本

如果您稍後使用 Helm Chart 與版本 2.26.0 或更新版本的 Prometheus 伺服器，請按照下列步驟操作。

使用 Helm Chart 從 Prometheus 伺服器設定遠端寫入

1. 在 Helm 組態檔案中建立一個新的遠端寫入區段：

   - 以您在中步驟 1：為服務帳戶設定 IAM 角色建立amp-iamproxy-ingest-role的 ARN 取${IAM_PROXY_PROMETHEUS_ROLE_ARN}代。角色 ARN 的格式應為 `arn:aws:iam::your account ID:role/amp-iamproxy-ingest-role`。

- 將 ${WORKSPACE_ID} 替換為 Amazon Managed Service for Prometheus 工作區 ID。
- 將 ${REGION} 替換為 Amazon Managed Service for Prometheus 工作區的地區 (例如 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. 使用 Helm 更新您現有的 Prometheus 伺服器組態：

- 將 prometheus-chart-name 替換為您的 Prometheus 版本名稱。
- 將 prometheus-namespace 替換為安裝 Prometheus 伺服器的 Kubernetes 命名空間。
- 將 my_prometheus_values_yaml 替換為 Helm 組態檔路徑。
- 將 current_helm_chart_version 替換為您 Prometheus 伺服器 Helm Chart 的目前版本。您可以通過使用 Helm list 命令找到目前圖表版本。

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

使用早期 Prometheus 的版本

若您正在使用 2.26.0 之前的 Prometheus 版本，請按照以下步驟操作。這些步驟使用並行方法，因為早期版本的 Prometheus 本身不支援簽 AWS 名版本 4 簽署程序 (Sigv4)。AWS

以下說明假設您使用 Helm 部署 Prometheus。

從 Prometheus 伺服器設定遠端寫入

1. 在您的 Prometheus 伺服器上，建立新的遠端寫入組態。首先，建立新的更新檔案。我們將呼叫該檔案 `amp_ingest_override_values.yaml`。

   將下列值新增至 YAML 檔案。

```
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
```

   將 `${REGION}` 替換為 Amazon Managed Service for Prometheus 工作區的地區。

以您在中步驟 1：為服務帳戶設定 IAM 角色建立amp-iamproxy-ingest-role的 ARN 取${SERVICE_ACCOUNT_IAM_INGEST_ROLE_ARN}代。角色 ARN 的格式應為 arn:aws:iam::*your account ID*:role/amp-iamproxy-ingest-role。

將 ${WORKSPACE_ID} 替換為工作區 ID。

2. 升級 Prometheus Helm Chart。首先，輸入下列命令尋找 Helm Chart 名稱。在此命令的輸出中，尋找名稱包含 prometheus 的圖表。

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

然後輸入下列命令。

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

替換*prometheus-helm-chart-name*為上一個命令中返回的 Prometheus 頭盔圖的名稱。將 *prometheus-namespace* 替換為命名空間名稱。

下載 Helm Chart

如果您尚未在本機下載 Helm Chart，您可以使用以下命令進行下載。

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

## 在 Fargate 的 Kubernetes 從現有的 Prometheus 伺服器設定擷取作業

Amazon Managed Service for Prometheus 支援在 Fargate 上執行的自我管理 Kubernetes 叢集中，從 Prometheus 伺服器擷取指標。若要從 Fargate 上執行 Amazon EKS 叢集中的 Prometheus 伺服器擷取指標，請覆寫名為 amp_ingest_override_values.yaml 組態檔中的預設組態，如下所示：

```
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
```

使用透過以下命令覆寫安裝 Prometheus：

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

請注意，在 Helm Chart 組態中，我們停用節點匯出程式和警示管理員以及執行 Prometheus 伺服器部署。

您可以使用以下範例測試查詢驗證安裝程序。

```
$ 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
```

## 正在設定 Amazon Managed Service for Prometheus 高可用性資料

若您將資料傳送到 Amazon Managed Service for Prometheus 時，將會跨地區中的 AWS 個可用區域複製，並從提供可擴展性、可用性和安全性的主機叢集提供給您。根據特定設定，您可能需要額外的高可用性失效安全。有兩種常見的方法可以為您的設定提供額外的高可用性安全性：

- 如果您有多個容器或執行個體具有相同資料，則可以將該資料傳送到 Amazon Managed Service for Prometheus，並自動刪除重複資料。這有助於確保將您的資料傳送到 Amazon Managed Service for Prometheus 工作區。

  如需有關刪除重複高可用性資料的詳細資訊，請參閱 將傳送至 Amazon Managed Service for Prometheus 的高可用性指標刪除重複資料。
- 若您想要確保資料有存取權，即使沒有 AWS 地區，您可將指標傳送給其他地區的第二個工作區。

  如需將指標資料傳送至多個工作區的詳細資訊，請參閱 跨區域可用性。

主題

- 將傳送至 Amazon Managed Service for Prometheus 的高可用性指標刪除重複資料
- 使用 Prometheus 將高可用性資料傳送至 Amazon Managed Service for Prometheus
- 將高可用性資料傳送至使用 Prometheus 操作員的 Amazon Managed Service for Prometheus
- 使 AWS 用適用於開放式遙測的發行版，將高可用性資料傳送至適用於 Prometheus 的 Amazon 受管服務
- 使用 Prometheus 社群 Helm Chart，將高可用性資料傳送至 Amazon Managed Service for Prometheus
- 常見問題：高可用性組態
- 跨區域可用性

## 將傳送至 Amazon Managed Service for Prometheus 的高可用性指標刪除重複資料

您可以將多位 Prometheus 客戶人員 (在客服人員模式中執行的 Prometheus 執行個體) 資料，傳送至 Amazon Managed Service for Prometheus 工作區。若其中一個執行個體正在記錄並傳送相同指標，您的資料的可用性將較高 (即使其中一位客服人員停止傳送資料，Amazon Managed Service for Prometheus 工作區仍將會收到其他執行個體的資料)。然而，您希望 Amazon Managed Service for Prometheus 工作區自動刪除重複指標，以便不想多次看到指標，進而防止多次收取資料擷取和儲存費用。

若要讓 Amazon Managed Service for Prometheus 自動從多個 Prometheus 代理程式刪除重複資料，您可以為傳送重複資料的一組代理程式提供單一叢集名稱，而每個執行個體都具有複本名稱。叢集名稱會將執行個體識別為具有共用資料，而複本名稱可讓 Amazon Managed Service for Prometheus 識別每個指標的來源。最終儲存的指標包含叢集標籤，但不包含複本，因此指標似乎來自單一來源。

> **Note**
>
> 某些版本的 Kubernetes (1.28 和 1.29) 可能會發出自己的量度並附有標籤。cluster這可能會導致針對 Prometheus 重複資料刪除的 Amazon 受管服務發生問題。如需詳細資訊，請參閱高可用性常見問題。

下列主題說明如何傳送資料並包含cluster和__replica__標籤，以便 Prometheus 的 Amazon 受管服務自動取消重複資料。

> **Important**
>
> 若您未設定重複資料刪除功能，則所有傳送至 Amazon Managed Service for Prometheus 的資料範例都需收費。這些資料範例包括重複的範例。

## 使用 Prometheus 將高可用性資料傳送至 Amazon Managed Service for Prometheus

若要使用 Prometheus 設定高可用性組態，您必須在高可用性群組的所有執行個體上套用外部標籤，以供 Amazon Managed Service for Prometheus 進行識別。使用 `cluster` 標籤識別 Prometheus 執行個體代理程式作為高可用性群組的一部份。使用 `__replica__` 標籤分別識別群組中的每個複本。您需要同時套用 `__replica__` 和 `cluster` 標籤，以便重複資料刪除工作。

> **Note**
>
> `__replica__` 標籤會在文字 `replica` 前後使用兩個底線符號進行格式化。

範例：程式碼片段

在下列程式碼片段中，`cluster` 標籤會識別 Prometheus 執行個體代理程式 `prom-team1`，而 `_replica_` 標籤會識別複本 `replica1` 和 `replica2`。

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

```
cluster: prom-team1
```

```
__replica__: replica2
```

由於 Amazon Managed Service for Prometheus 會使用這些標籤儲存來自高可用性複本的資料範例，因此在接受範例時會剝離 `replica` 標籤。這表示只有當前序列的 1：1 序列對應，而不是每個複本的系列。`cluster` 標籤會保留下來。

> **Note**
>
> 某些版本的 Kubernetes (1.28 和 1.29) 可能會發出自己的量度並附有標籤。`cluster`這可能會導致針對 Prometheus 重複資料刪除的 Amazon 受管服務發生問題。如需詳細資訊，請參閱高可用性常見問題。

## 將高可用性資料傳送至使用 Prometheus 操作員的 Amazon Managed Service for Prometheus

若要使用 Prometheus 操作員設定高可用性組態，您必須在高可用性群組的所有執行個體上套用外部標籤，以便 Amazon Managed Service for Prometheus 進行識別。您也必須在 Prometheus 操作員 Helm Chart 上設定屬性 `replicaExternalLabelName` 和 `externalLabels`。

範例：YAML 標題

在下列 YAML 標題中，`cluster` 會新增至 `externalLabel`，以將 Prometheus 執行個體代理程式識別為高可用性群組的一部份，而 `replicaExternalLabels` 會識別群組中的每個複本。

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

> **Note**
>
> 某些版本的 Kubernetes (1.28 和 1.29) 可能會發出自己的量度並附有標籤。`cluster`這可能會導致針對 Prometheus 重複資料刪除的 Amazon 受管服務發生問題。如需詳細資訊，請參閱高可用性常見問題。

## 使 AWS 用適用於開放式遙測的發行版，將高可用性資料傳送至適用於 Prometheus 的 Amazon 受管服務

AWS 開放遙測發行版（ADOT）是項目的安全和生產就緒分發。 OpenTelemetry ADOT 為您提供來源 API、程式庫和代理程式，因此您可以收集分散式追蹤和指標以進行應用程式監控。如需 ADOT 的相關資訊，請參閱關於開放式遙測 AWS 發行版。

若要使用高可用性組態設定 ADOT，您必須設定 ADOT 收集器容器映像檔，並將外部標籤`cluster`套用`__replica__`至 AWS Prometheus 遠端寫入匯出程式。此匯出程式會透過 `remote_write` 端點將您抓取的指標傳送到 Amazon Managed Service for Prometheus 工作區。當您在遠端寫入匯出程式上設定這些標籤時，可避免在執行備援複本時保留重複的指標。如需有關 AWS Prometheus 遠端寫入匯出器的詳細資訊，請參閱適用於 Prometheus 的 Amazon 受管服務的 Prometheus 遠端寫入匯出器入門。

> **Note**
>
> 某些版本的 Kubernetes (1.28 和 1.29) 可能會發出自己的量度並附有標籤。`cluster`這可能會導致針對 Prometheus 重複資料刪除的 Amazon 受管服務發生問題。如需詳細資訊，請參閱高可用性常見問題。

## 使用 Prometheus 社群 Helm Chart，將高可用性資料傳送至 Amazon Managed Service for Prometheus

若要使用 Prometheus 社群 Helm chart 設定高可用性組態，您必須在高可用性群組的所有執行個體上套用外部標籤，以便 Amazon Managed Service for Prometheus 進行識別。下面是如何從 Prometheus 社區 Helm Chart 將 `external_labels` 新增至 Prometheus 單一執行個體的範例。

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

> **Note**
>
> 如果您想要多個複本，您必須使用不同的複本值多次部署圖表，因為 Prometheus 社群 Helm Chart 不允許直接從控制器群組增加複本數量時動態設定複本值。如果您想要自動設定 `replica` 標籤，請使用 Prometheus 操作員 Helm Chart。

> **Note**
>
> 某些版本的 Kubernetes (1.28 和 1.29) 可能會發出自己的量度並附有標籤。`cluster`這可能會導致針對 Prometheus 重複資料刪除的 Amazon 受管服務發生問題。如需詳細資訊，請參閱高可用性常見問題。

## 常見問題：高可用性組態

我是否應該將值 __replica__ 包含到另一個標籤中以跟踪樣本點？

在高可用性設定中，Amazon Managed Service for Prometheus 可透過選擇 Prometheus 執行個體叢集中的領導者，以確保資料範例不會重複。若領導者複本停止傳送資料範例 30 秒，Amazon Managed Service for Prometheus 會自動將另一個 Prometheus 執行個體設為領導者複本，並從新領導者擷取資料，包括任何遺漏的資料。因此，答案為否，不建議執行。 這樣做可能會導致以下問題：

- 在選舉新領導者的期間，在 PromQL 中查詢 `count` 可能會傳回高於預期的值。
- 在選舉新領導者期間增加的 `active series` 數量，這會到達 `active series limits`。如需詳細資訊，請參閱 AMP 配額中的配額。

Kubernetes 似乎有它自己的集群標籤，並且不會刪除重複我的指標。我要如何修正這個情形？

在 Kubernetes 1.28 中引入`apiserver_storage_size_bytes`了一個新的量度，並帶有標籤。`cluster`這可能會導致 Prometheus 的 Amazon 受管服務中的重複資料刪除問題，這取決於標籤。`cluster`在 Kubernetes 1.3 中，標籤會重新命名為 `storage-cluster_id` (此標籤也會在稍後的 1.28 和 1.29 修補程式中重新命名)。如果您的叢集使用標籤發出此指`cluster`標，Prometheus 的 Amazon 受管服務無法刪除相關的時間序列。建議您將 Kubernetes 叢集升級至最新的修補版本，以避免發生此問題。或者，您可以在`apiserver_storage_size_bytes`指標上重新標記標`cluster`籤，然後再將其導入 Prometheus 的 Amazon 受管服務。

> **Note**
>
> 如需有關變更至 Kubernetes 的詳細資訊，請參閱在 Kubernetes 專案中將標籤叢集重新命名為 Storage_cluster_id。 GitHub

## 跨區域可用性

若要將跨區 AWS 域可用性新增至資料，您可以將指標傳送至跨區域的多個工作區。Prometheus 支持多個編寫器和跨區域編寫。

下列範例顯示如何設定在代理程式模式下執行的 Prometheus 伺服器，以便將指標傳送至位於不同區域中的兩個工作區。

```
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]
```

# 查詢 Prometheus 指標

既然已將指標擷取至工作區，便可對其進行查詢。您可以使用服務 (例如 Grafana) 來查詢指標，也可以使用 Amazon Managed Service for Prometheus API。

您可以使用標準的 Prometheus 查詢語言 PromQL 執行查詢。如需有關 PromQL 和其語法的詳細資訊，請參閱 Prometheus 說明文件中的查詢 Prometheus。

主題

- 保護您的指標查詢
- 設定 Amazon Managed Grafana，以搭配 Amazon Managed Service for Prometheus 使用
- 設定 Grafana 開放原始碼或 Grafana 企業版，以搭配 Amazon Managed Service for Prometheus 使用
- 使用 Amazon EKS 叢集中執行的 Grafana 查詢
- 使用與 Prometheus 相容的 API 查詢
- 在查詢 API 回應中查詢統計資訊

## 保護您的指標查詢

Amazon Managed Service for Prometheus 提供以下方法，協助您確保指標查詢安全性。

### AWS PrivateLink 與 Prometheus 的 Amazon 託管服務一起使用

在適用於 Prometheus 的 Amazon 受管服務中查詢指標的網路流量可透過公用網際網路端點或透過 VPC 端點完成。 AWS PrivateLink當您使用時 AWS PrivateLink，來自 VPC 的網路流量會在網路中受到保護，而無需透過公用 AWS 網際網路。若要為 Prometheus 的 Amazon 受管服務建立 AWS PrivateLink VPC 端點，請參閱。使用 Amazon Managed Service for Prometheus 和介面 VPC 端點

### 身份驗證和授權

AWS Identity and Access Management 是一種 Web 服務，可幫助您安全地控制對 AWS 資源的訪問。您可以使用 IAM 來控制能通過身分驗證 (登入) 和授權使用資源的 (具有許可) 的人員。Amazon Managed Service for Prometheus 與 IAM 整合，協助您確保資料安全。當您設定 Amazon Managed Service for Prometheus 時，您需要建立部分 IAM 角色，讓 Grafana 伺服器查詢儲存在 Amazon

Managed Service for Prometheus 工作區中的指標。如需有關 IAM 的詳細資訊，請參閱什麼是 IAM？。

另一個可協助您為 Prometheus 設定 Amazon 受管服務的 AWS 安全功能是簽 AWS 名版本 4 簽署程序 (AWS Sigv4)。簽名版本 4 是將身份驗證信息添加到由 HTTP 發送的 AWS 請求的過程。為了安全起見，大多數的請求都 AWS 必須使用訪問密鑰進行簽名，該訪問密鑰包括訪問密鑰 ID 和秘密訪問密鑰。這兩種金鑰通常稱為您的安全憑證。如需有關 SigV4 的詳細資訊，請參閱簽章第 4 版簽署程序。

# 設定 Amazon Managed Grafana，以搭配 Amazon Managed Service for Prometheus 使用

Amazon Managed Grafana 是開放原始碼 Grafana 的全受管服務，可簡化與開放原始碼第三方 ISV 的連線，以及大規模視覺化和分析資料來源的 AWS 服務。

Amazon Managed Service for Prometheus 支援使用 Amazon Managed Grafana 查詢工作區中的指標。在 Amazon Managed Grafana 主控台中，您可以探索現有 Amazon Managed Service for Prometheus 帳戶，將 Amazon Managed Service for Prometheus 工作區新增為資料來源。Amazon Managed Grafana 管理存取 Amazon Managed Service for Prometheus 所需的身分驗證憑證組態。如需從 Amazon Managed Grafana 建立 Amazon Managed Service for Prometheus 連線的詳細指示，請參閱 Amazon Managed Grafana 使用者指南中的指示。

您也可以在 Amazon Managed Grafana 中檢視 Amazon Managed Service for Prometheus 警示。如需設定與警示整合的指示，請參閱 與 Amazon Managed Grafana 或開放原始碼 Grafana 整合警示。

## 在私有 VPC 中連線至 Amazon Managed Grafana

Amazon Managed Service for Prometheus 提供 Amazon Managed Grafana 的服務端點，以在查詢指標和警示時進行連線。

您可以設定 Amazon Managed Grafana 以使用私有 VPC (如需在 Grafana 中設定私有 VPC 的詳細資訊，請參閱 Amazon Managed Grafana 使用者指南中的連線到 Amazon VPC)。根據設定，此 VPC 可能無法存取 Amazon Managed Service for Prometheus 服務端點。

若要將 Amazon Managed Service for Prometheus 作為資料來源新增到設定才能使用特定私有 VPC 的 Amazon Managed Grafana 工作區，您必須先建立 VPC 端點來將 Amazon Managed Service for Prometheus 連線到相同的 VPC。如需有關建立 VPC 端點的詳細資訊，請參閱 為 Amazon Managed Service for Prometheus 建立介面 VPC 端點。

# 設定 Grafana 開放原始碼或 Grafana 企業版，以搭配 Amazon Managed Service for Prometheus 使用

Amazon Managed Service for Prometheus 支援使用 Grafana 7.3.5 及更新版本來查詢工作區中的指標。7.3.5 及更高版本包括對 AWS 簽名版本 4（SIGv4）身份驗證的支持。

有關使用 tar.gz 或 zip 檔案設定獨立的 Grafana 指示，請參閱 Grafana 說明文件中的安裝 Grafana。如果您安裝新的獨立 Grafana，系統會提示您輸入使用者名稱和密碼。預設值為 **admin/admin**。首次登入後，系統會提示您變更密碼。如需詳細資訊，請參閱 Grafana 說明文件中的 Grafana 入門。

若要檢查您的 Grafana 版本，請輸入下列命令。

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

若要設定 Grafana 使用適用於 Prometheus 的 Amazon 受管服務，您必須登入具有AmazonPrometheusQueryAccess政策或aps:QueryMetrics、、aps:GetMetricMetadata和許可的帳戶。aps:GetSeries aps:GetLabels如需詳細資訊，請參閱 IAM 許可和政策。

## 設定 AWS SigV4

適用於 Prometheus 的 Amazon 受管服務可與 AWS Identity and Access Management (IAM) 搭配使用 IAM 登入資料保護對 Prometheus API 的所有呼叫。依預設，Grafana 中的 Prometheus 資料來源假定 Prometheus 不需要身份驗證。若要讓 Grafana 能夠利用 Amazon Managed Service for Prometheus 身份驗證和授權功能，您必須在 Grafana 資料來源中啟用 SigV4 身份驗證支援。當您使用自我管理的 Grafana 開放原始碼或 Grafana 企業伺服器時，請依照本頁面上的步驟操作。若您正在使用 Amazon Managed Grafana，則 SiGv4 身份驗證是完全自動化的。如需有關 Amazon Managed Grafana 的詳細資訊，請參閱什麼是 Amazon Managed Grafana？

若要在 Grafana 上啟用 SigV4，請在 AWS_SDK_LOAD_CONFIG 和 GF_AUTH_SIGV4_AUTH_ENABLED 環境變數設為 true 的情況下啟動 Grafana。GF_AUTH_SIGV4_AUTH_ENABLED 環境變數會覆寫 Grafana 的預設組態，以啟用 SigV4 支援。如需詳細資訊，請參閱 Grafana 說明文件中的組態 。

Linux

若要在 Linux 的獨立 Grafana 伺服器上啟用 SigV4，請輸入以下命令。

```
export AWS_SDK_LOAD_CONFIG=true
```

```
export GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

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

Windows

若要在 Windows 的獨立 Grafana 上使用 Windows 命令提示啟用 SigV4，請輸入下列命令。

```
set AWS_SDK_LOAD_CONFIG=true
```

```
set GF_AUTH_SIGV4_AUTH_ENABLED=true
```

```
cd grafana_install_directory
```

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

## 在 Grafana 中新增 Prometheus 資料來源

下列步驟說明如何在 Grafana 中設定 Prometheus 資料來源，以查詢您的 Amazon Managed Service for Prometheus 指標。

在您的 Grafana 伺服器中新增 Prometheus 資料來源

1. 開啟 Grafana 主控台。
2. 在組態下方，選擇資料來源。
3. 選擇新增資料來源。
4. 選擇 Prometheus。
5. 針對 HTTP URL，請在 Amazon Managed Service for Prometheus 主控台指定工作區詳細資訊頁面中顯示的端點 - 查詢 URL。
6. 由於 Prometheus 資料來源會自動附加該字串，因此請在剛指定的 HTTP URL 中移除附加至 URL 的 `/api/v1/query` 字串。

   正確的 URL 看起來應該像是 https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-1234a5b6-78cd-901e-2fgh-3i45j6k178l9。
7. 在驗證下，選取SigV4 驗證的切換功能以啟用。

8. 您可以直接在 Grafana 中指定長期憑證，或使用預設提供者鏈結來設定 SigV4 授權。直接指定長期憑證可讓您更快速地開始，而下列步驟會先提供這些指示。一旦您更熟悉與 Amazon Managed Service for Prometheus 搭配使用 Grafana，我們建議您使用預設提供者鏈結，因為這提供更佳的彈性和安全性。如需有關設定預設提供者鏈結的詳細資訊，請參閱指定憑證。

   - 若要直接使用長期憑證，請執行下列動作：

     a. 在 SigV4 身分驗證詳細資訊下，請針對身份驗證提供者選擇存取和密鑰。

     b. 針對存取金鑰 ID，輸入您的 AWS 存取金鑰 ID。

     c. 針對私密存取金鑰輸入您的 AWS 私密存取金鑰。

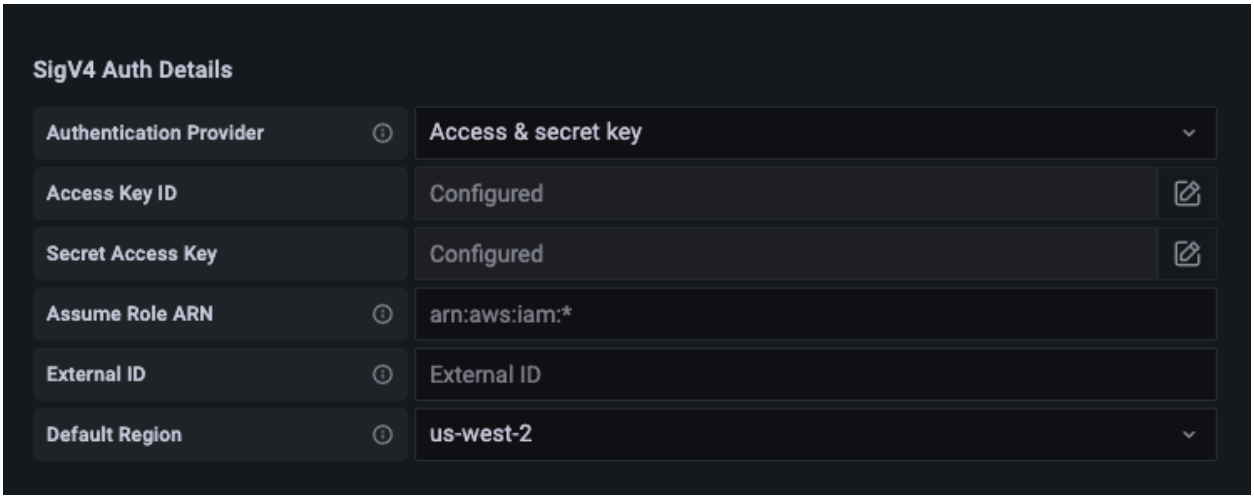
     d. 將假設角色 ARN 和外部 ID 欄位保留空白。

     e. 對於預設區域，請選擇 Amazon Managed Service for Prometheus 工作區的區域。此區域應與您在步驟 5 所列出 URL 中包含的區域相符。

     f. 選擇儲存並測試。

        您應該看到以下訊息：資料來源正在運作

        面的螢幕截取畫面顯示存取金鑰、密鑰 SigV4 身份驗證詳細資料設定。




   - 要改用預設提供者鏈結 (建議用於正式運作環境)，請執行下列步驟：

     a. 在 SigV4 身分驗證詳細資訊下，請針對身分驗證提供者選擇 AWS SDK 預設值。

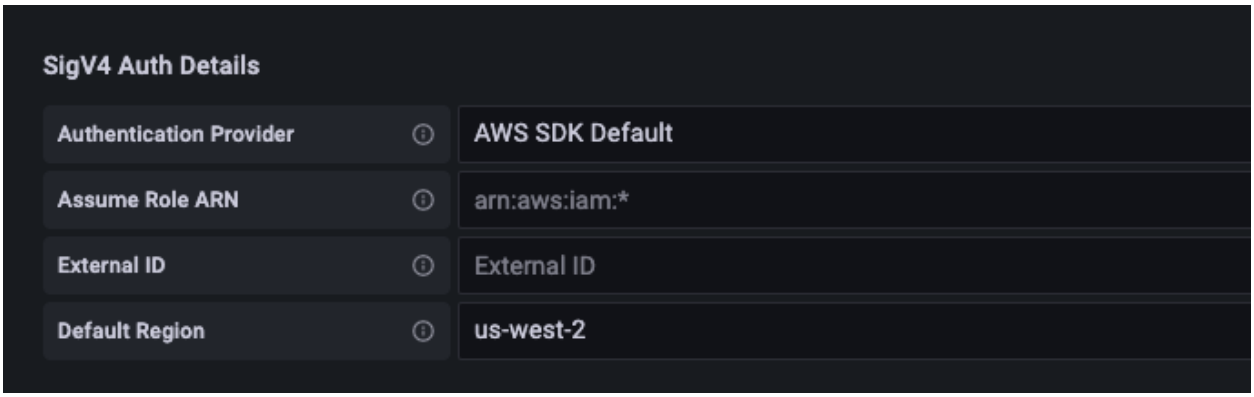
     b. 將假設角色 ARN 和外部 ID 欄位保留空白。

     c. 對於預設區域，請選擇 Amazon Managed Service for Prometheus 工作區的區域。此區域應與您在步驟 5 所列出 URL 中包含的區域相符。

     d. 選擇儲存並測試。

您應該看到以下訊息：資料來源正在運作

下列螢幕截取畫面顯示 SDK 預設 SigV4 身分驗證詳細資料設定。




9. 針對新的資料來源測試 PromQL 查詢：

   a. 選擇探索。

   b. 執行範例 PromQL 查詢，例如：

```
prometheus_tsdb_head_series
```

## 「儲存並測試」未運作時的疑難排解

在上一個程序中，如果您在選擇儲存並測試時看到錯誤，請檢查下列項目。

找不到 HTTP 錯誤

請確定 URL 中的工作區 ID 正確無誤。

禁止 HTTP 錯誤

此錯誤表示憑證無效。請檢查以下內容：

- 檢查預設區域中指定的區域是否正確。
- 檢查您的憑證是否有錯別字。
- 請確定您使用的認證具有原AmazonPrometheusQueryAccess則。如需詳細資訊，請參閱 IAM 許可和政策。
- 確保您正在使用的憑證可存取此 Amazon Managed Service for Prometheus 工作區。

HTTP 錯誤無效閘道

請查看 Grafana 伺服器日誌以解決此錯誤。如需詳細資訊，請參閱 Grafana 說明文件中的疑難排解。

如果您看到**`Error http: proxy error: NoCredentialProviders: no valid providers in chain`**，則預設認證提供者鏈結無法找到要使用的有效 AWS 認證。確認您已按照指定憑證中的說明設定您的憑證。若您要使用共用組態，請確認 `AWS_SDK_LOAD_CONFIG` 環境已設定為 `true`。

# 使用 Amazon EKS 叢集中執行的 Grafana 查詢

Amazon Managed Service for Prometheus 支援使用 Grafana 7.3.5 及更新版本，以及稍後在 Amazon Managed Service for Prometheus 工作區中查詢指標。7.3.5 及更高版本包括對 AWS 簽名版本 4（SIGv4）身份驗證的支持。

若要設定 Grafana 使用適用於 Prometheus 的 Amazon 受管服務，您必須登入具有AmazonPrometheusQueryAccess政策或`aps:QueryMetrics`、、`aps:GetMetricMetadata`和許可的帳戶。`aps:GetSeries` `aps:GetLabels`如需詳細資訊，請參閱 IAM 許可和政策。

## 設定第四 AWS 章

Grafana 新增了一項支援 AWS 簽名版本 4 (SIGv4) 驗證的新功能。如需詳細資訊，請參閱簽章版本 4 簽署程序。依預設，不會再 Grafana 伺服器上啟用此功能。若您正在使用 Helm 在 Kubernetes 叢集上部署 Grafana，則以下為啟用此功能的指示。

在您的 Grafana 7.3.5 或更新伺服器上啟用 SigV4

1. 建立新的更新檔案來覆寫您的 Grafana 組態，並將其命名 `amp_query_override_values.yaml`。
2. 將下列內容輸入檔案，然後儲存檔案。以正在執# Grafana 伺服器的 AWS 帳號 ID 取代帳號 ID。

```
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
```

在該 YAML 檔案內容中，amp-iamproxy-query-role 是您將在下一節中建立的角色名稱、設定服務帳戶的 IAM 角色。如果您已經建立用於查詢工作區的角色，則可以使用您自己的角色名稱替換此角色。

您稍後將在 使用 Helm 升級 Grafana 伺服器 中使用此檔案。

## 設定服務帳戶的 IAM 角色

如果您正在 Amazon EKS 叢集中使用 Grafana 伺服器，建議您針對服務帳戶使用 IAM 角色 (也稱為服務角色) 進行存取控制。當您執行此操作以將 IAM 角色與 Kubernetes 服務帳戶建立關聯時，服務帳戶就可以接著為使用該服務帳戶的任何網繭中的容器提供 AWS 權限。如需詳細資訊，請參閱服務帳戶的 IAM 角色。

如果您尚未設定這些服務角色以進行查詢，請遵循 設定服務帳戶的 IAM 角色，以查詢指標 中的指示來設定角色。

然後，您需要在信任關係的條件下新增 Grafana 服務帳戶。

在信任關係的條件下新增 Grafana 服務帳戶

1. 在終端機視窗中，判斷 Grafana 伺服器的命名空間和服務帳戶名稱。例如，您可以使用下列命令。

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

2. 在 Amazon EKS 主控台中，針對與 EKS 叢集相關聯的服務帳戶開啟 IAM 角色。
3. 選擇編輯信任關係。
4. 更新「條件」以包含 Grafana 命名空間，以及您在步驟 1 中的命令輸出中找到的 Grafana 服務帳戶名稱。以下是範例。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::account-id:oidc-provider/
oidc.eks.aws_region.amazonaws.com/id/openid"
      },
```

```
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "oidc.eks.region.amazonaws.com/id/openid:sub": [
            "system:serviceaccount:aws-amp:amp-iamproxy-query-service-account",
            "system:serviceaccount:grafana-namespace:grafana-service-account-name"
          ]
        }
      }
    }
  ]
}
```

5. 選擇更新信任政策。

## 使用 Helm 升級 Grafana 伺服器

此步驟會升級 Grafana 伺服器，以使用您在上一節中新增至 amp_query_override_values.yaml 檔案的項目。

執行下列命令。如需 Grafana 的 Helm 圖表的詳細資訊，請參閱 Grafana 社群 Kubernetes Helm Charts。

```
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
```

## 在 Grafana 中新增 Prometheus 資料來源

下列步驟說明如何在 Grafana 中設定 Prometheus 資料來源，以查詢您的 Amazon Managed Service for Prometheus 指標。

在您的 Grafana 伺服器中新增 Prometheus 資料來源

1. 開啟 Grafana 主控台。
2. 在組態下方，選擇資料來源。
3. 選擇新增資料來源。
4. 選擇 Prometheus。

5. 針對 HTTP URL，請在 Amazon Managed Service for Prometheus 主控台指定工作區詳細資訊頁面中顯示的端點 - 查詢 URL。
6. 由於 Prometheus 資料來源會自動附加該字串，因此請在剛指定的 HTTP URL 中移除附加至 URL 的 `/api/v1/query` 字串，
7. 在驗證下，選取 SigV4 驗證的切換功能以啟用。

   將假設角色 ARN 和外部 ID 欄位保留空白。然後針對預設區域，選取您的 Amazon Managed Service for Prometheus 工作區所在的區域。
8. 選擇儲存並測試。

   您應該看到以下訊息：資料來源正在運作
9. 針對新的資料來源測試 PromQL 查詢：

   a. 選擇探索。
   b. 執行範例 PromQL 查詢，例如：

   ```
   prometheus_tsdb_head_series
   ```

# 使用與 Prometheus 相容的 API 查詢

雖然使用 Amazon Managed Grafana 等工具是檢視和查詢指標最簡單的方式，但 Amazon Managed Service for Prometheus 也支援數個可用來查詢指標的與 Prometheus 相容 API。如需有關所有可用與 Prometheus 相容 API 的詳細資訊，請參閱 與 Prometheu 相容的 API。

當您使用這些 API 查詢指標時，必須使用簽 AWS 名版本 4 簽署程序來簽署要求。您可以設定AWS 簽章版本 4來簡化簽署程序。如需詳細資訊，請參閱 aws-sigv4-proxy。

透過 AWS Sigv4 代理伺服器簽署可以使用`awscurl`執行。下列主題使用 awscurl 查詢與 Prometheus 相容的 API 會逐步引導您使用 `awscurl` 完成設定 AWS SigV4。

## 使用 awscurl 查詢與 Prometheus 相容的 API

Amazon Managed Service for Prometheus 的 API 請求必須使用 SigV4 簽署。您可以使用 awscurl 來簡化查詢程序。

若要安裝 `awscurl`，您需要安裝 Python 3 和 pip 套件管理員。

在以 Linux 為基礎的執行個體上，下列命令會安裝 `awscurl`。

```
$ pip3 install awscurl
```

在 macOS 電腦上，下列命令會安裝 awscurl。

```
$ brew install awscurl
```

下列範例是awscurl查詢範例。以適合您使用案例的適當值取代*##*、*### ID* 和 *QUERY* 輸入：

```
# 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**
>
> 您的查詢字串必須是 url 編碼。

對於像這樣的查詢query=up，你可以得到如下結果：

```
{
  "status": "success",
  "data": {
    "resultType": "vector",
    "result": [
      {
        "metric": {
          "__name__": "up",
          "instance": "localhost:9090",
          "job": "prometheus",
          "monitor": "monitor"
        },
        "value": [
          1652452637.636,
```

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

為了 awscurl 簽署所提供的請求，您需要以下列其中一種方式傳送有效的憑證：

- 為 IAM 角色提供存取金鑰 ID 和密鑰。您可以在 https://console.aws.amazon.com/iam/ 中找到該角色的存取金鑰和密鑰。

  例如：

  ```
  $ 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>"
  ```

- 參考儲存在 .aws/credentials 和 /aws/config 檔案中的組態檔案。您也可以選擇指定將使用的設定檔名稱。如果未指定，將使用 default 檔案。例如：

  ```
  $ 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>"
  ```

- 使用與 EC2 執行個體相關聯的執行個體設定檔。

## 使用 awscurl 容器執行查詢請求

無法安裝不同版本的 Python 和相關的相依項目時，一個容器可以用來打包 awscurl 應用程式及其相依項目。下列範例使用 Docker 執行期進行部署 awscurl，但任何符合 OCI 規範的執行期和映像都可以運作。

```
$ 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"
```

# 在查詢 API 回應中查詢統計資訊

查詢定價是根據一個月內已執行查詢處理的查詢範例總數而定。query 或 queryRange API 的查詢回應包含有關已處理查詢範例的統計資料。當在請求中傳送查詢參數 stats=all 時，在 samples 物件中建立 stats 物件，並在回應中傳回 stats 資料。

samples 物件由下列屬性組成：

| 屬性 | 描述 |
| --- | --- |
| totalQueryableSamples | 已處理的查詢範例總數量。這是用於請款的信息。 |
| totalQueryableSamplesPerStep | 每個步驟處理的查詢範例數。這會依時期針對含時間戳記的陣列建構為其中一個陣列，以及在特定步驟中載入的範例數量。 |

在回應中包含 stats 資訊的範例請求和回應如下：

query 的範例：

GET

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

回應

```
{
    "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
                    ]
                ]
            }
        }
    }
}
```

queryRange 的範例：

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
```

回應

```
{
    "status": "success",
```

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

# 記錄規則和警示規則

Amazon Managed Service for Prometheus 支援兩種類型的規則，這些規則會進行定期評估：

- 記錄規則可讓您預先計算經常需要或計算上昂貴的運算式，並將其結果儲存為新的時間序列集。查詢預先計算的結果通常較需要時每次執行原始運算式快。
- 警示規則可讓您根據 PromQL 和閾值來定義警示條件。當規則觸發閾值時，系統會將通知傳送至警示管理員，該管理員會將通知轉寄到下游的接收者，例如 Amazon Simple Notification Service。

若要在 Amazon Managed Service for Prometheus 中使用規則，您需要建立一或多個用於定義規則的 YAML 規則檔案。Amazon Managed Service for Prometheus 規則檔案的格式，與獨立 Prometheus 中規則檔案的格式相同。如需詳細資訊，請參閱 Prometheus 說明文件中的定義記錄規則和警示規則。

您可以在工作區中擁有多個規則檔案。每個個別規則檔案包含在個別命名空間。擁有多個規則檔案可讓您將現有的 Prometheus 規則檔案匯入至工作區，而不需進行變更或合併。不同的規則群組命名空間也可以有不同的標籤。

規則排序

在規則檔案中，規則包含在規則群組中。規則檔案中單一規則群組內的規則一律會依照從上到下的順序進行評估。因此，在記錄規則中，一個記錄規則的結果可用於計算較新的記錄規則或相同規則群組中的警示規則。但是，由於您無法指定執行個別規則檔案的順序，因此無法使用一個記錄規則的結果來計算不同規則群組或不同規則檔案中的規則。

主題



# 必要的 IAM 許可

您必須授予使用者權限，才能在 Amazon Managed Service for Prometheus 中使用規則。建立具有下列權限的 AWS Identity and Access Management (IAM) 政策，並將政策指派給您的使用者、群組或角色。

> **Note**
>
> 如需有關 IAM 的詳細資訊，請參閱 Amazon Managed Service for Prometheus 的識別與存取管理。

授予使用規則存取權的政策

下列原則可授予使用帳戶中所有資源規則的存取權。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps: CreateRuleGroupsNamespace",
                "aps: ListRuleGroupsNamespaces",
                "aps: DescribeRuleGroupsNamespace",
                "aps: PutRuleGroupsNamespace",
                "aps: DeleteRuleGroupsNamespace",
            ],
            "Resource": "*"
        }
    ]
}
```

僅授予一個命名空間存取權的政策

您也可以建立只授予特定政策存取權的政策。下列範例政策僅提供 RuleGroupNamespace 指定的存取權。若要使用此政策，請將*<account>*、*<region>*、*<workspace-id>*和 *<namespace-name>*替換為您帳戶的適當值。

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

```
                "aps:GetLabels",
                "aps:CreateRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespaces",
                "aps:DescribeRuleGroupsNamespace",
                "aps:PutRuleGroupsNamespace",
                "aps:DeleteRuleGroupsNamespace"
            ],
            "Resource": [
                "arn:aws:aps:*:<account>:workspace/*",
                "arn:aws:aps:<region>:<account>:rulegroupnamespace/<workspace-
id>/<namespace-name>"
            ]
        }
    ]
}
```

# 建立規則檔案

若要在 Amazon Managed Service for Prometheus 中使用規則，您會建立定義規則的規則檔案。Amazon Managed Service for Prometheus 規則檔案的格式，與獨立 Prometheus 中規則檔案的格式相同。如需詳細資訊，請參閱定義記錄規則和警示規則。

以下是規則檔案的基本範例：

```
groups:
  - name: test
    rules:
    - record: metric:recording_rule
      expr: avg(rate(container_cpu_usage_seconds_total[5m]))
  - name: alert-test
    rules:
    - alert: metric:alerting_rule
      expr: avg(rate(container_cpu_usage_seconds_total[5m])) > 0
      for: 2m
```

如需更多警示規則範例，請參閱警示規則範例。

> **Note**
>
> 您可以在本機建立規則定義檔案，然後將其上傳到 Prometheus 的 Amazon 受管服務，也可以直接在 Prometheus 的 Amazon 受管服務主控台中建立、編輯和上傳定義。無論哪種方式，

都適用相同的格式規則。若要進一步瞭解如何上傳和編輯檔案，請參閱將規則組態檔案上傳至 Amazon Managed Service for Prometheus。

# 將規則組態檔案上傳至 Amazon Managed Service for Prometheus

一旦知道要對 rules 配置文件進行哪些更改，您可以在控制台中對其進行編輯，也可以使用控制台或上傳替代文件 AWS CLI。

> **Note**
>
> 如果您正在執行 Amazon EKS 叢集，也可以使用適用於 Kubernetes 的AWS 控制器上傳規則組態檔案。

使用適用於 Prometheus 的 Amazon 受管服務主控台編輯或取代您的規則組態，並建立命名空間

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇工作區的工作區 ID，然後選擇規則管理索引標籤。
4. 選擇新增命名空間。
5. 選擇選擇檔案，然後選取規則定義檔案。

   或者，您可以選取「定義組態」，直接在適用於 Prometheus 的 Amazon 受管服務主控台中建立和編輯規則定義檔案。這將建立一個範例預設定義檔案，您可以在上傳之前編輯該檔案。
6. (選用) 若要將標籤新增至命名空間，請選擇新增標籤。

   之後，在 Key (索引鍵) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。

   若要新增另一個標籤，請再次選擇新增標籤。
7. 選擇繼續。Amazon Managed Service for Prometheus 會建立名稱與所選規則檔案相同的新命名空間。

使用將警示管理員組態上載至新命名空間中的工作區 AWS CLI

1. Base64 會對警示管理員檔案的內容進行編碼。在 Linux 系統上，您可使用下列命令：

```
base64 input-file output-file
```

在 macOS 系統上，您可使用下列命令：

```
openssl base64 input-file output-file
```

2. 輸入以下其中一個命令，建立命名空間並上傳檔案。

   在 AWS CLI 版本 2 上，輸入：

   ```
   aws amp create-rule-groups-namespace --data file://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

   在 AWS CLI 版本 1 上，輸入：

   ```
   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. 警示管理員組態需要幾秒鐘才會變成啟用中。若要檢查狀態，請輸入以下命令：

   ```
   aws amp describe-rule-groups-namespace --workspace-id workspace_id --
   name namespace-name --region region
   ```

   如果 `status` 是 `ACTIVE`，則表示您的規則檔案已生效。

## 編輯規則組態檔

您可以上傳新的規則檔案來取代現有的組態，也可以直接在主控台中編輯目前的組態。或者，您可以下載目前檔案，在文字編輯器中進行編輯，然後上傳新版本。

使用 Amazon Managed Service for Prometheus 主控台編輯您的規則組態

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇工作區的工作區 ID，然後選擇規則管理索引標籤。
4. 選取您要編輯的規則組態檔案名稱。

5. (選擇性) 如果您要下載目前的規則組態檔案，請選擇「下載」或「複製」。
6. 選擇 [修改]，直接在主控台內編輯組態。完成時選擇「儲存」。

   或者，您也可以選擇「取代組態」來上傳新的組態檔。如果是這樣，請選取新的規則定義檔案，然後選擇「繼續」以上傳。

若要使用 AWS CLI 編輯規則組態檔案

1. Base64 會對規則檔案的內容進行編碼。在 Linux 系統上，您可使用下列命令：

   ```
   base64 input-file output-file
   ```

   在 macOS 系統上，您可使用下列命令：

   ```
   openssl base64 input-file output-file
   ```

2. 輸入以下其中一個命令以上傳新檔案。

   在 AWS CLI 版本 2 上，輸入：

   ```
   aws amp put-rule-groups-namespace --data file://path_to_base_64_output_file --
   name namespace-name  --workspace-id my-workspace-id --region region
   ```

   在 AWS CLI 版本 1 上，輸入：

   ```
   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. 規則需要幾秒鐘才會變成啟用中。若要檢查狀態，請輸入以下命令：

   ```
   aws amp describe-rule-groups-namespace --workspace-id workspace_id --
   name namespace-name --region region
   ```

   如果 `status` 是 `ACTIVE`，則表示您的規則檔案已生效。在此之前，此規則檔案的以前版本仍為啟用中。

# 尺規疑難排解

使用 CloudWatch 日誌 時，您可以進行警示管理員和尺規相關問題的疑難排解。本節包含尺規相關的疑難排解主題。

當日誌包含以下標尺失敗錯誤

```
{
    "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"
}
```

這表示執行規則時發生一些錯誤。

採取動作

使用錯誤訊息疑難排解規則執行。

# 警示管理員

若正在啟動 Amazon Managed Service for Prometheus 執行的警示規則，警示管理員處理已傳送的警示。這會刪除重複項目、分組，並將警示路由至下游接收者。Amazon Managed Service for Prometheus 僅支援 Amazon Simple Notification Service 作為接收者，且可將訊息傳送至相同帳戶中的 Amazon SNS 主題。您也可使用警示管理員將警示靜音和禁止。

警示管理員會在 Prometheus 中提供 Alertmanager 的相似功能。

您可以針對下列項目使用警示管理員的組態檔案：

- 分組：分組會將類似的警示收集到單一通知中。若許多系統立即無法執行且可能同步射擊數以百計的警示，則這會特別實用。例如，假設網路故障會導致許多節點同時無法執行。若已將這些警示類型分組，警示管理員會傳送單一通知給您。

  警示分組和分組通知的時間是由警示管理員組態檔案中的路由樹狀結構來設定。如需詳細資訊，請參閱<常式>。
- 抑制：若已發射某些其他警報，則會抑制某些警示的通知。例如，若發出的警示與無法觸及的叢集相關，則可組態警示管理員將此叢集所有其他警示靜音。這樣可以防止發生與實際問題無關的數百或數千個觸發警報的通知。如需如何撰寫抑制規則的詳細資訊，請參閱<inhibit_rule>。
- 靜音：靜音將警示靜音一段時間，例如在維護時段期間。將會檢查收到的警示與啟用中靜音的所有相等或一般表達式匹配程式相符。若要執行此作業，將不會傳送該警示的通知。

  若要建立靜音，請使用 `PutAlertManagerSilences` API。如需詳細資訊，請參閱 PutAlertManagerSilences。

Prometheus 範本

獨立 Prometheus 支持範本，使用分離的範本檔案。範本可在其他事物之間使用條件和格式資料。

在 Prometheus 的 Amazon 受管服務中，您可以將範本放在與警示管理員組態相同的警示管理員組態檔案中。

主題

- 必要的 IAM 許可
- 建立警示管理員組態檔案
- 設定您的警示接收器

- 將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus
- 與 Amazon Managed Grafana 或開放原始碼 Grafana 整合警示
- 疑難排解警示管理員

# 必要的 IAM 許可

您必須授予使用者權限，才能在 Amazon Managed Service for Prometheus 中使用規則。建立具有下列權限的 AWS Identity and Access Management (IAM) 政策，並將政策指派給您的使用者、群組或角色。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps: CreateAlertManagerDefinition",
                "aps: DescribeAlertManagerSilence",
                "aps: DescribeAlertManagerDefinition",
                "aps: PutAlertManagerDefinition",
                "aps: DeleteAlertManagerDefinition",
                "aps: ListAlerts",
                "aps: ListRules",
                "aps: ListAlertManagerReceivers",
                "aps: ListAlertManagerSilences",
                "aps: ListAlertManagerAlerts",
                "aps: ListAlertManagerAlertGroups",
                "aps: GetAlertManagerStatus",
                "aps: GetAlertManagerSilence",
                "aps: PutAlertManagerSilences",
                "aps: DeleteAlertManagerSilence",
                "aps: CreateAlertManagerAlerts"
            ],
            "Resource": "*"
        }
    ]
}
```

# 建立警示管理員組態檔案

若要在 Amazon Managed Service for Prometheus 中使用警示管理員和範本，請建立警示管理員組態 YAML 檔案。Amazon Managed Service for Prometheus 警示管理員檔案分成兩個主要部份：

- `template_files:` 包含用於接收者傳送訊息的範本。如需詳細資訊，請參閱 Prometheus 說明文件中的範本參考和範本範例。
- `alertmanager_config:` 包含警示管理員組態。這使用與獨立 Prometheus 中的警示管理員組態檔案相同的結構。如需詳細資訊，請參閱警示管理員說明文件中的組態。

> **Note**
>
> 上述 Prometheus 說明文件中描述的 `repeat_interval` 有額外的 Amazon Managed Service for Prometheus 限制。允許的值上限為五天。若您將其設為大於五天，這將視為五天且經過五天期間後將再次傳送通知。

> **Note**
>
> 您也可以直接在 Prometheus 的 Amazon 受管服務主控台中編輯組態檔案，但仍必須遵循此處指定的格式。如需上載或編輯組態檔案的詳細資訊，請參閱將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus。

在 Amazon Managed Service for Prometheus 中，您的警示管理員組態檔案必須在 YAML 檔案根目錄的 `alertmanager_config` 金鑰內包含所有警示管理員組態內容。

以下是警示管理員設定檔的基本範例：

```
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
```

目前唯一支援的接收器即 Amazon Simple Notification Service (Amazon SNS)。若您在組態中列出的其他接收者類型，則將會予以拒絕。

以下是其他同時使用 `template_files` 區塊和 `alertmanager_config` 區塊的警示管理員設定檔範例。

```
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
```

預設 Amazon SNS 範本區塊

除非您明確覆寫，否則預設 Amazon SNS 組態會使用下列範本。

```
{{ 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 }}
```

# 設定您的警示接收器

Amazon Managed Service for Prometheus 目前支援的唯一警示接收器即 Amazon Simple Notification Service (Amazon SNS)。如需詳細資訊，請參閱什麼是 Amazon SNS？。

主題

- (選用) 建立新的 Amazon SNS 主題
- 授予 Amazon Managed Service for Prometheus 的權限，以便將訊息傳送到您的 Amazon SNS 主題
- 在警示管理員組態檔案中指定您的 Amazon SNS 主題
- (選用) 設定警示管理員以將 JSON 輸出至 Amazon SNS
- (選用) 從 Amazon SNS 傳送至其他目的地
- SNS 接收者訊息驗證和截斷規則

## (選用) 建立新的 Amazon SNS 主題

您可以使用現有 Amazon SNS 主題或建立新主題。我們建議您使用「標準」類型的主題，以便將主題的警示轉寄至電子郵件、簡訊或 HTTP。

若要建立新的 Amazon SNS 主題作為您的警示管理員接收器，請按照步驟 1：建立主題中的步驟操作。請務必為主題類型選擇標準。

若您希望每次傳送訊息到該 Amazon SNS 主題時都接收電子郵件，請按照步驟 2：建立主題訂閱中的步驟進行操作。

## 授予 Amazon Managed Service for Prometheus 的權限，以便將訊息傳送到您的 Amazon SNS 主題

您必須授予 Amazon Managed Service for Prometheus 權限，以便將訊息傳送到您的 Amazon SNS 主題。以下的政策聲明包括一個 `Condition` 聲明，以幫助防止混淆代理人安全問題。`Condition`

聲明限制 Amazon SNS 主題的存取權，以便僅允許來自此特定帳戶和 Amazon Managed Service for Prometheus 工作區的作業。如需有關混淆代理人問題的詳細資訊，請參閱 預防跨服務混淆代理人。

授予 Amazon Managed Service for Prometheus 的許可，以訊息傳送到您的 Amazon SNS 主題

1. 在 https://console.aws.amazon.com/sns/v3/home 開啟 Amazon SNS 主控台。
2. 在導覽窗格中，選擇主題。
3. 選擇您與 Amazon Managed Service for Prometheus 搭配使用的主題名稱。
4. 選擇 [編輯] 。
5. 選擇存取政策，然後將下列政策陳述式新增至現有政策。

```
{
    "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"
}
```

[選用] 如果您的 SNS 主題已啟用服務端加密 (SSE)，則需要將下列權限新增至 "Action" 區塊中的 KMS 金鑰政策。如需詳細資訊，請參閱 SNS 主題的AWS KMS 許可。

```
    kms:GenerateDataKey
    kms:Decrypt
```

6. 選擇「Save changes (儲存變更)」。

> **Note**
>
> 依預設，Amazon SNS 在於 `AWS:SourceOwner` 上建立含條件的存取政策。如需詳細資訊，請參閱 SNS 存取政策。

> **Note**
>
> IAM 遵循最嚴格的政策第一條規則。在您的 SNS 主題中，如果政策區塊的限制比記錄的 Amazon SNS 政策區塊更嚴格，則不會授予主題政策的權限。若要評估您的原則並找出已授與的項目，請參閱政策評估邏輯。

## 預防跨服務混淆代理人

混淆代理人問題屬於安全性問題，其中沒有執行動作許可的實體可以強制具有更多許可的實體執行該動作。在中 AWS，跨服務模擬可能會導致混淆的副問題。在某個服務 (呼叫服務) 呼叫另一個服務 (被呼叫服務) 時，可能會發生跨服務模擬。可以操縱呼叫服務來使用其許可，以其不應有存取許可的方式對其他客戶的資源採取動作。為了防止這種情況發生， AWS 提供的工具可協助您透過已授予您帳戶中資源存取權的服務主體來保護所有服務的資料。

我們建議在資源政策中使用 `aws:SourceArn` 或 `aws:SourceAccount` 全域條件內容索引鍵，來限制 Amazon Managed Service for Prometheus 給予 Amazon SNS 對資源的許可。如果同時使用全域條件內容索引鍵，則在相同政策陳述式中使用 `aws:SourceAccount` 值和 `aws:SourceArn` 值中的帳戶時，必須使用相同的帳戶 ID。

`aws:SourceArn` 的值必須是 Amazon Managed Service for Prometheus 工作區的 ARN。

防範混淆代理人問題最有效的方法，是使用 `aws:SourceArn` 全域條件內容金鑰，以及資源的完整 ARN。如果不知道資源的完整 ARN，或者如果您指定了多個資源，請使用 `aws:SourceArn` 全域條件內容金鑰，同時使用萬用字元 (*) 表示 ARN 的未知部分。例如 `arn:aws:`*`servicename`*`::`*`123456789012`*`:*`。

授予 Amazon Managed Service for Prometheus 的權限，以便將訊息傳送到您的 Amazon SNS 主題中顯示的政策會顯示您可在 Amazon Managed Service for Prometheus 中使用 `aws:SourceArn` 和 `aws:SourceAccount` 全域條件內容索引鍵防止混淆代理人問題。

# 在警示管理員組態檔案中指定您的 Amazon SNS 主題

現在，您可以將 Amazon SNS 接收器新增至警示管理員組態。若要執行此動作，您必須知道 Amazon SNS 主題的 Amazon Resource Name (ARN)。

如需 Amazon SNS 接收器組態的詳細資訊，請參閱 Prometheus 組態文件中的<sns_configs>。

不支援的屬性

Amazon Managed Service for Prometheus 支援 Amazon SNS 作為警示接收器。但是，由於服務限制條件，因此並不支援 Amazon SNS 接收器的所有屬性。Amazon Managed Service for Prometheus 警示管理員組態檔案不允許下列屬性：

- `api_url:`：由於 Amazon Managed Service for Prometheus 為您設定 `api_url`，因此不允許此屬性。
- `Http_config`：此屬性可讓您設定外部代理程式。Amazon Managed Service for Prometheus 目前不支援此功能。

此外，需要 SigV4 設定才有「區域」屬性。未透過地區屬性，Amazon Managed Service for Prometheus 可進行權限請求的資訊不足。

將您的 Amazon SNS 主題設定為接收者的警示管理員

1. 如果您使用現有的警示管理員組態檔，請在文字編輯器中開啟。
2. 若 `receivers` 區塊中有非 Amazon SNS 的目前接收器，則將會移除。您可以在 `receivers` 區塊內將多個 Amazon SNS 主題設定為接收器，方法是將其放在個別 `sns_config` 區塊中。
3. 在 `receivers` 區段內新增下列 YAML 區塊。

```
- name: name_of_receiver
  sns_configs:
    - sigv4:
        region: region
      topic_arn: ARN_of_SNS_topic
      subject: somesubject
      attributes:
        key: somekey
        value: somevalue
```

若未指定 `subject`，依預設會使用含標籤名稱和值的預設範本產生主旨，這可能會導致 SNS 的值太長。若要變更套用至主旨的範本，請參閱本指南中的 (選用) 設定警示管理員以將 JSON 輸出至 Amazon SNS。

現在，您必須將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus。如需詳細資訊，請參閱 將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus。

## (選用) 設定警示管理員以將 JSON 輸出至 Amazon SNS

您可以將警示管理員設定為以 JSON 格式傳送警示，以便在 Web Hook 接收端點內 AWS Lambda 或在 Amazon SNS 下游處理這些警示。使用 Amazon Managed Service for Prometheus 警示管理員提供的預設範本，將會使用不可輕鬆清除的內文清單輸出訊息承載資料。您可以定義自訂範本而非使用預設範本，以 JSON 格式輸出訊息內容，以便在下游函數中更容易剖析。

若要以 JSON 格式將訊息從警示管理員輸出至 Amazon SNS，請更新警示管理員組態，以在 `template_files` 根區段中包含下列代碼：

```
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**
>
> 此版本會從英數字元資料建立 JSON。如果您的資料具有特殊字元，請在使用此範本之前對其進行編碼。

若要確認已在送出通知中使用此範本，則請在 `alertmanager_config` 區塊中依照下列方式參考：

```
alertmanager_config: |
  global:
  templates:
    - 'default_template'
```

> **Note**
>
> 此範本適用於整個郵件內文的 JSON 格式。此範本會覆寫整個訊息內文。如果您想要使用此特定範本，則無法覆寫訊息內文。任何手動完成的覆寫都會優先於範本。

如需更多相關資訊：

- 警示管理員組態檔案，請參閱 建立警示管理員組態檔案。
- 上傳您的組態檔案時，請參閱 將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus。

## (選用) 從 Amazon SNS 傳送至其他目的地

目前，Amazon Managed Service for Prometheus 只能將警示訊息直接傳送給 Amazon SNS。您可以將 Amazon SNS 設定為將這些訊息傳送到其他目的地，例如電子郵件、網路掛鉤、Slack 和。OpsGenie

### 電子郵件

若要設定 Amazon SNS 主題以將訊息輸出至電子郵件，請建立訂閱。在 Amazon SNS 主控台中，選擇「訂閱」索引標籤以開啟「訂閱」清單頁面。選擇建立訂閱，然後選取電子郵件。Amazon SNS 會將確認電子郵件傳送至所列出的電子郵件地址。接受確認後，您就能以電子郵件形式接收 Amazon SNS 通知，來自您訂閱的主題。如需詳細資訊，請參閱訂閱 Amazon SNS 主題。

### Webhook

若要設定 Amazon SNS 主題以將訊息輸出到 Webhook 端點，請建立訂閱。在 Amazon SNS 主控台中，選擇「訂閱」索引標籤以開啟「訂閱」清單頁面。選擇建立訂閱，然後選取 HTTP/HTTPS。建立訂閱之後，您必須遵循確認步驟來啟用訂閱。若為啟用中狀態，則 HTTP 端點則 HTTP 端點應收到 Amazon SNS 通知。如需詳細資訊，請參閱訂閱 Amazon SNS 主題。如需有關使用 Slack webhooks 將訊息發佈至不同目的地的詳細資訊，請參閱如何使用網路掛鉤將 Amazon SNS 訊息發佈到 Amazon Chime、Slack 或 Microsoft 團隊？

### Slack

若要將 Amazon SNS 主題設定為將訊息輸出至 Slack，您有兩種選擇。您可以與 Slack 的 email-to-channel 整合整合整合，如此 Slack 可以接受電子郵件訊息並將其轉寄至 Slack 通道，或者您也可以使用 Lambda 函數將 Amazon SNS 通知重寫為 Slack。如需有關轉寄電子郵件至鬆弛通道的詳細資訊，請參閱確認 Slack Webhook 的 AWS SNS 主題訂閱。如需有關建構 Lambda 函數以將 Amazon SNS 訊息轉換至 Slack 的詳細資訊，請參閱如何將 Amazon Managed Service for Prometheus 與 Slack 整合。

### OpsGenie

如需如何設定 Amazon SNS 主題以將訊息輸出至其中的相關資訊 OpsGenie，請參閱將選項與傳入的 Amazon SNS 整合。

## SNS 接收者訊息驗證和截斷規則

SNS 訊息將根據下列規則，在必要時由 SNS 接收者驗證、截斷或修改：

- 訊息包含非 utf 字元。
  - 「錯誤-不是有效的 UTF-8 編碼字串」將替代訊息。
  - 將新增一個訊息屬性，其鍵值為「截斷」且值為「True」。
  - 將新增一個訊息屬性，其鍵值為「修改」且值為「訊息：錯誤 - 不是有效的 UTF-8 編碼字串」的值。
- 訊息為空。
  - 「錯誤 - 消息不應空白」將替代訊息。
  - 將新增一個訊息屬性，其鍵值為「修改」且值為「訊息：錯誤 - 訊息不應為空白」。
- 訊息已被截斷。
  - 訊息將具有截斷的內容。

  - 將新增一個訊息屬性，其鍵值為「截斷」且值為「True」
  - 將新增一個訊息屬性，其鍵值為「已修改」，以及「訊息：錯誤 - 訊息已從 *X* KB 截斷，因為郵件超過 256 KB 的大小限制」。
- 主題不是 ASCII。
  - 「錯誤 - 包含不可打印的 ASCII 字符。」將替代主題。
  - 將新增一個訊息屬性，其鍵值為「已修改」且值為「主題：錯誤 - 包含非可列印的 ASCII 字元」。
- 主題已截斷。
  - 主題將具有截斷的內容。
  - 將新增一個訊息屬性，其鍵值為「已修改」，且「主題：錯誤 - 主題已從 *X* 個字元截斷，因為超過 100 個字元大小限制。」
- 訊息屬性有無效的鍵值/值。
  - 將移除無效的訊息屬性。
  - 一個消息屬性將被添加與「修改」的鍵和值「MessageAttribute：錯誤-*X* 的消息屬性已被刪除，因為無效 MessageAttributeKey 或。」 MessageAttributeValue
- 訊息屬性已截斷。
  - 其他訊息屬性將刪除。
  - 將新增一個訊息屬性，其鍵為「已修改」，並且已移除訊息屬性的值MessageAttribute：Error-*X*，因為它超過 256KB 的大小限制。

# 將警示管理員組態檔案上傳至 Amazon Managed Service for Prometheus

當您知道要對 Alert Manager 組態檔進行哪些變更後，您可以在主控台內編輯該檔案，也可以使用主控台或上傳取代檔案 AWS CLI。

> **Note**
>
> 如果您正在執行 Amazon EKS 叢集，也可以使用適用於 Kubernetes 的AWS 控制器上傳警示管理員組態檔案。

使用適用於 Prometheus 的 Amazon 受管服務主控台編輯或取代您的警示管理員組態

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 選擇頁面左上角的功能表圖示，然後選擇所有工作區。
3. 選擇工作區的工作區 ID，然後選擇警示管理員索引標籤。
4. 如果工作區尚無警示管理員定義，請選擇新增定義。

> **Note**
>
> 如果工作區具有您要取代的警示管理員定義，請改為選擇 [修改]。

5. 選取選擇檔案、選取警示管理員定義檔案，然後選擇繼續。

> **Note**
>
> 或者，您可以透過選擇 [建立定義] 選項，建立新檔案並直接在主控台中編輯它。這會建立您在上傳前編輯的範例預設組態。

使用 AWS CLI 將警示管理員組態上載至工作區

1. Base64 會對警示管理員檔案的內容進行編碼。在 Linux 系統上，您可使用下列命令：

```
base64 input-file output-file
```

在 macOS 系統上，您可使用下列命令：

```
openssl base64 input-file output-file
```

2. 若要上傳檔案，請輸入下列其中一個命令。

在 AWS CLI 版本 2 上，輸入：

```
aws amp create-alert-manager-definition --data file://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

在 AWS CLI 版本 1 上，輸入：

```
aws amp create-alert-manager-definition --data fileb://path_to_base_64_output_file
 --workspace-id my-workspace-id --region region
```

3. 警示管理員組態需要幾秒鐘才會變成啟用中。若要檢查狀態，請輸入以下命令：

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

如果 `status` 是 `ACTIVE`，則您的新警示管理員定義已生效。

使用將工作區的警示管理員組態取代 AWS CLI 為新的警示管理員組態

1. Base64 會對警示管理員檔案的內容進行編碼。在 Linux 系統上，您可使用下列命令：

```
base64 input-file output-file
```

在 macOS 系統上，您可使用下列命令：

```
openssl base64 input-file output-file
```

2. 若要上傳檔案，請輸入下列其中一個命令。

在 AWS CLI 版本 2 上，輸入：

```
aws amp put-alert-manager-definition --data file://path_to_base_64_output_file --
workspace-id my-workspace-id --region region
```

在 AWS CLI 版本 1 上，輸入：

```
aws amp put-alert-manager-definition --data fileb://path_to_base_64_output_file --
workspace-id my-workspace-id --region region
```

3. 新的警示管理員組態需要幾秒鐘才會變成啟用中。若要檢查狀態，請輸入以下命令：

```
aws amp describe-alert-manager-definition --workspace-id workspace_id --
region region
```

如果 `status` 是 `ACTIVE`，則您的新警示管理員定義已生效。在那之前，您先前的警示管理員組態仍為啟用中。

# 與 Amazon Managed Grafana 或開放原始碼 Grafana 整合警示

您在 Amazon Managed Service for Prometheus 內 Alertmanager 中建立的警示規則可以在 Amazon Managed Grafana 和 Grafana 中進行轉送和檢視，從而在單一環境中統一您的警示規則和警示。透過 Amazon Managed Grafana，您可檢視警示規則和產生的警示。

## 必要條件

在開始將 Amazon Managed Service for Prometheus 整合到 Amazon Managed Grafana 之前，您必須已完成下列先決條件：

- 您必須擁有現有的 AWS 帳戶 和 IAM 憑證，才能以程式設計方式建立 Amazon Managed Service for Prometheus 和 IAM 角色。

  如需有關建立 AWS 帳戶 和 IAM 憑證的詳細資訊，請參閱 設定。
- 您必須擁有 Amazon Managed Service for Prometheus 工作區，並將資料擷取至其中。若要設定新工作區，請參閱 建立工作區。您同時應該熟悉 Prometheus 概念，例如 Alertmanager 和尺規。如需有關這些主題的詳細資訊，請參閱 Prometheus 說明文件。
- 您已在 Amazon Managed Service for Prometheus 中設定 Alertmanager 組態和規則檔案。如需有關 Amazon Managed Service for Prometheus 中 Alertmanager 的詳細資訊，請參閱 警示管理員。如需規則的詳細資訊，請參閱記錄規則和警示規則。
- 您必須設定 Amazon Managed Grafana，或正在執行 Grafana 的開放原始碼版本。
  - 如果您使用的是 Amazon Managed Grafana，您必須使用 Grafana 提醒。如需詳細資訊，請參閱將舊版儀表板警示移轉至 Grafana 提醒。
  - 如果您使用的是 Grafana 開放原始碼版本，您必須執行 9.1 或更新版本。

> **Note**
>
> 您可以使用舊版 Grafana，但您必須啟用統一提醒 (Grafana 警示) 功能，而且您可能必須設定 sigv4 代理程式，才能從 Grafana 呼叫 Amazon Managed Service for Prometheus。如需更多詳細資訊，請參閱 設定 Grafana 開放原始碼或 Grafana 企業版，以搭配 Amazon Managed Service for Prometheus 使用。

- Amazon Managed Grafana 必須具備下列許可才能使用您的 Prometheus 資源。您必須將這些政策新增至中 https://docs.aws.amazon.com/grafana/latest/userguide/AMG-manage-permissions.html 所述的服務管理或客戶管理政策。
  - `aps:ListRules`
  - `aps:ListAlertManagerSilences`
  - `aps:ListAlertManagerAlerts`
  - `aps:GetAlertManagerStatus`
  - `aps:ListAlertManagerAlertGroups`
  - `aps:PutAlertManagerSilences`
  - `aps:DeleteAlertManagerSilence`

## 設定 Amazon Managed Grafana

如果您已經在 Amazon Managed Service for Prometheus 執行個體中設定規則和警示，則使用 Amazon Managed Grafana 作為這些警示儀表板的設定完全在 Amazon Managed Grafana 內完成。

將 Amazon Managed Grafana 設定為您的警示儀表板

1. 開啟您工作區的 Grafana 主控台。
2. 在「組態」下，選擇「資料來源」。
3. 建立或開啟您的 Prometheus 資料來源。如果您之前尚未設定 Prometheus 資料來源，請參閱以 在 Grafana 中新增 Prometheus 資料來源 取得更多資訊。
4. 在 Prometheus 資料來源中，選取「透過警示管理員使用者介面管理警示」。
5. 返回「資料來源」介面。
6. 建立新的警示管理員資料來源。
7. 在「警示管理員」 資料來源組態頁面中，新增下列設定：
   - 「建置」設定為 `Prometheus`。
   - 針對 URL 設定，請使用 Prometheus 工作區的 URL，移除工作區 ID 之後的所有內容，然後將 `/alertmanager` 附加到結尾。例如：*`https://aps-workspaces.us-east1.amazonaws.com/workspaces/ws-example-1234-5678-abcd-xyz00000001/alertmanager`*。
   - 在「驗證」下，開啟「SigV4Auth」。這告訴 Grafana 對請求使用AWS 身份驗證。

- 在 SigV4Auth 詳細資料下，對於「預設區域」，提供您 Prometheus 執行個體的區域，例如 `us-east-1`。
- 將「預設」選項設定為 `true`。

8. 選擇 Save and test (儲存並測試)。
9. 您的 Amazon Managed Service for Prometheus 警示現在應該已設定為與您的 Grafana 執行個體搭配使用。確認您可以在 Grafana 警示 頁面中看到來自 Amazon Managed Service for Prometheus 執行個的任何警示規則、警示群組 (包括啟用中警示) 和靜音。

# 疑難排解警示管理員

使用 CloudWatch 日誌 時，您可以進行警示管理員和尺規相關問題的疑難排解。本節包含警示管理員相關的疑難排解主題。

主題

- 空內容警告
- 非 ASCII 警告
- 無效的 key/value 警告
- 訊息限制警告
- 無資源型政策錯誤

## 空內容警告

日誌包含下列警告

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "message": {
       "log": "Message has been modified because the content was empty."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

這表示警示管理員範本將外傳警示解析為空白訊息。

採取動作

驗證您的警示管理員範本，並確保您擁有適用於所有接收者路徑的有效範本。

## 非 ASCII 警告

日誌包含下列警告

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "message": {
       "log": "Subject has been modified because it contains control or non-ASCII
 characters."
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

這表示主題具有非 ASCII 字元。

採取動作

移除範本主旨欄位中可能包含非 ASCII 字元標籤的參考。

## 無效的 `key/value` 警告

日誌包含下列警告

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "message": {
       "log": "MessageAttributes has been removed because of invalid key/value,
 numberOfRemovedAttributes=1"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

這表示由於鍵值/值無效，因此已移除某些訊息屬性。

採取動作

重新評估您用來填入訊息屬性的範本，並確定其解析為有效的 SNS 訊息屬性。如需驗證 Amazon SNS 主題的訊息的詳細資訊，請參閱驗證 SNS 主題

## 訊息限制警告

日誌包含下列警告

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "message": {
       "log": "Message has been truncated because it exceeds size limit,
 originSize=266K, truncatedSize=12K"
       "level": "WARN"
   },
   "component": "alertmanager"
}
```

這表示某些訊息大小過大。

採取動作

查看警示接收器訊息模板，然後重新調整以符合大小限制。

## 無資源型政策錯誤

日誌包含下列錯誤

```
{
   "workspaceId": "ws-efdc5b42-b051-11ec-b123-4567ac120002",
   "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"
}
```

這表示 Amazon Managed Service for Prometheus 沒有將警示提交至指定 SNS 主題的許可。

採取動作

驗證 Amazon SNS 主題的存取政策是否授予 Amazon Managed Service for Prometheus 將 SNS 訊息傳送至主題的能力。建立 SNS 存取政策，讓服務 `aps.amazonaws.com` (適用於 Prometheus 的亞馬

遜受管服務) 存取您的 Amazon SNS 主題。如需 SNS 存取政策的詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用存取政策語言和 Amazon SNS 存取控制範例案例。

# 日誌記錄和監控

您可以使用 Amazon CloudWatch 日誌記錄和監控功能來管理適用於 Prometheus 資源用量的 Amazon 受管服務。

- 使用 CloudWatch 度量 來監控 Amazon Managed Service for Prometheus。
- 使用 CloudWatch 日誌 查詢和檢視 Amazon Managed Service for Prometheus 警示管理器和尺規事件。

## CloudWatch 度量

適用於 Prometheus 的 Amazon 受管服務會將使用量指標出售給。 CloudWatch這些指標提供有關工作區使用率的可見性。您可以在中的AWS/Usage和AWS/Prometheus命名空間中找到付費量度。CloudWatch這些指標免 CloudWatch費提供。如需使用狀況測量結果的詳細資訊，請參閱CloudWatch使用狀況

| CloudWatch 量度名稱 | 資源名稱 | CloudWatch 名稱區 | 描述 |
|---|---|---|---|
| ResourceCount | IngestionRate | AWS/Usage | 範例擷取速率<br><br>單位：每秒計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| ResourceCount | ActiveSeries | AWS/Usage | 每個工作區的啟用中序列數量<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| ResourceCount | ActiveAlerts | AWS/Usage | 每個工作區的啟用中警示數量<br><br>單位：計數 |

| CloudWatch 量度名稱 | 資源名稱 | CloudWatch 名稱區 | 描述 |
| --- | --- | --- | --- |
| | | | 有效統計資訊：平均數、下限、上限、總和 |
| ResourceCount | SizeOf警示 | AWS/Usage | 工作區中所有警示的總大小，以位元組為單位<br><br>單位：位元組<br><br>有效統計資訊：平均數、下限、上限、總和 |
| ResourceCount | SuppressedAlerts | AWS/Usage | 每個工作區處於隱藏狀態的警示數量。警示可透過靜音或抑制來隱藏。<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| ResourceCount | UnprocessedAlerts | AWS/Usage | 每個工作區處於未處理狀態的警示數量。警示一旦收到警示，就會處於未處理狀態 AlertManager，但正在等待下一個彙總群組評估。<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |

| CloudWatch 量度名稱 | 資源名稱 | CloudWatch 名稱區 | 描述 |
| --- | --- | --- | --- |
| ResourceCount | AllAlerts | AWS/Usage | 每個工作區處於任何狀態的警示數量。<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| AlertManagerAlertsReceived | - | AWS/Prometheus | 警示管理員收到的成功警示總數<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| AlertManagerNotificationsFailed | - | AWS/Prometheus | 失敗警示傳送數量<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| AlertManagerNotificationsThrottled | - | AWS/Prometheus | 限流的警示數量<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |
| Discarded Samples* | - | AWS/Prometheus | 按原因排列的廢棄範例數量<br><br>單位：計數<br><br>有效統計資訊：平均數、下限、上限、總和 |

| CloudWatch 量度名稱 | 資源名稱 | CloudWatch 名稱區 | 描述 |
| --- | --- | --- | --- |
| RuleEvaluations | - | AWS/Prometheus | 規則評估總數量<br>單位：計數<br>有效統計資訊：平均數、下限、上限、總和 |
| RuleEvaluation失敗 | - | AWS/Prometheus | 間隔中的規則評估失敗次數<br>單位：計數<br>有效統計資訊：平均數、下限、上限、總和 |
| RuleGroup IterationsMissed | - | AWS/Prometheus | 間隔中缺少的規則群組迭代次數。<br>單位：計數<br>有效統計資訊：平均數、下限、上限、總和 |

* 導致捨棄範例的某些原因如下。

| 原因 | 意義 |
| --- | --- |
| greater_than_max_sample_age | 丟棄超過一小時的樣本。 |
| new-value-for-timestamp | 重複的樣本會以不同於先前記錄的時間戳記傳送。 |
| per_metric_series_limit | 使用者已達到每個度量限制的作用中序列。 |
| per_user_series_limit | 用戶已達到活動序列限制的總數。 |
| rate_limited | 攝入率有限。 |
| sample-out-of-order | 樣品按順序發送，無法處理。 |

| 原因 | 意義 |
| --- | --- |
| label_value_too_long | 標籤值超過允許的字元限制。 |
| max_label_names_per_series | 使用者已點擊每個度量的標籤名稱。 |
| missing_metric_name | 未提供測量結果名稱。 |
| metric_name_invalid | 提供的度量名稱無效。 |
| label_invalid | 提供的標籤無效。 |
| duplicate_label_names | 提供了重複的標籤名稱。 |

> **Note**
>
> 不存在或遺漏的指標與該指標為 0 的值相同。

> **Note**
>
> `RuleGroupIterationsMissed`、`RuleEvaluations`、和 `RuleEvaluationFailures` 具有下列結構的 `RuleGroup` 維度：
> *`RuleGroup####`*; *`RuleGroup`*

## 在 Prometheus 出價指標上設置 CloudWatch 警報

您可以使用警報監視 Prometheus 資源的使用情況。 CloudWatch

若要ActiveSeries在 Prometheus 的數量上設定鬧鐘

1. 選擇「繪圖量度」標籤，然後向下捲動至標ActiveSeries籤。

   在圖形化指標檢視中，只會顯示目前擷取的指標。
2. 在動作欄中選擇通知圖示。
3. 在指定指標和條件中，於條件值欄位中輸入門檻值條件，然後選擇下一步。
4. 在設定動作中，選取現有 SNS 主題，或建立新 SNS 主題以將通知傳送至其中。

5. 在新增名稱和說明中，新增警示名稱和選用說明。
6. 選擇 Create alarm (建立警示)。

# CloudWatch 日誌

適用於 Prometheus 的 Amazon 受管服務記錄了 Amazon 日誌中的日誌群組中的警示管理器和標尺錯誤和警告事件。 CloudWatch 如需有關警示管理員和尺規的詳細資訊，請參閱本指南中的警示管理員主題。您可以將工作區記錄資料發佈到記 CloudWatch 錄檔中的串流。您可以在 Amazon Managed Service for Prometheus 主控台或使用 AWS CLI，設定希望監控的日誌。您可以在 CloudWatch主控台中檢視或查詢這些記錄檔。如需有關在主控台中檢視 CloudWatch 記錄檔資料流的詳細資訊，請參閱使用指南 CloudWatch中的〈 CloudWatch 使用記錄群組和記錄串流〉。

CloudWatch 免費方案最多可在記錄中發佈 5Gb 的 CloudWatch 記錄。超過免費方案限額的記錄將根據CloudWatch 定價方案收費。

主題

- 設定 CloudWatch 記錄檔

## 設定 CloudWatch 記錄檔

適用於 Prometheus 的 Amazon 受管服務記錄了 Amazon 日誌中的日誌群組中的警示管理器和標尺錯誤和警告事件。 CloudWatch

您可以在適用於 Prometheus 主控台的 Amazon 受管服務中設定 CloudWatch 日誌記錄組態，或 AWS CLI 透過呼叫 API 請求在中設定日誌記錄組態。`create-logging-configuration`

先決條件

在呼叫之前`create-logging-configuration`，請將下列原則或同等權限附加至您將用來設定 CloudWatch 記錄的 ID 或角色。

```
{
    "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": "*"
        }
    ]
}
```

若要設定 CloudWatch 記錄檔

您可以使用 AWS 主控台或在 Prometheus 的 Amazon 受管服務中設定日誌記錄。 AWS CLI

Console

在 Amazon Managed Service for Prometheus 主控台中設定記錄

1. 導覽至工作區詳細資料面板中的「日誌」索引標籤。
2. 選擇「日誌」面板右上角的「管理日誌」。
3. 在「日誌層級」下拉式清單中選擇「全部」。
4. 在「日誌群組」下拉式清單中，選擇您要發佈日誌的日誌群組。

   您也可以在 CloudWatch 主控台中建立新的記錄群組。
5. 選擇儲存變更。

AWS CLI

您可以使用設定記錄組態 AWS CLI。

若要使用設定記錄 AWS CLI

- 使用 AWS CLI，執行下列命令。

```
aws amp create-logging-configuration --workspace-id my_workspace_ID
```

```
                                    --log-group-arn my-log-group-arn
```

## 限制

- 並非所有事件都記錄

  Amazon Managed Service for Prometheus 僅記錄處於 `warning` 或 `error` 層級的事件。

- 政策大小限制

  CloudWatch 記錄檔資源策略的長度限制為 5120 個字元。當 CloudWatch Logs 偵測到原則接近此大小限制時，會自動啟用以開頭的記錄群組`/aws/vendedlogs/`。

  當您建立啟用日誌記錄的警示規則時，Prometheus 的 Amazon 受管服務必須使用您指定的 CloudWatch 日誌群組更新您的日誌資源政策。若要避免達到 CloudWatch 記錄檔資源原則大小限制，請在 CloudWatch 記錄檔群組名稱前面加上`/aws/vendedlogs/`。當您在 Amazon Managed Service for Prometheus 主控台中建立日誌群組時，日誌群組名稱的前面會加上 `/aws/vendedlogs/`。如需詳細資訊，請參閱記錄 CloudWatch 檔使用指南中的啟用來自某些 AWS 服務的記錄。

# 瞭解並最佳化成本

下列常見問題及其答案可能有助於瞭解和最佳化 Amazon Managed Service for Prometheus 相關的成本。

## 什麼會導致我的成本？

針對多數客戶，指標擷取會形成多數成本。查詢使用量較高的客戶也會根據已處理的查詢樣本看到一些成本，而指標儲存量是整體成本的一小部份驅動因素。如需上述各項價格的詳細資訊，請參閱 Amazon Managed Service for Prometheus 產品頁面中的定價。

## 降低成本的最佳方法是什麼？ 如何降低擷取成本？

對於大多數客戶而言，擷取率 (而非指標的儲存量) 是大多數的成本。您可以降低收集頻率 (增加收集間隔) 或減少擷取啟用中序列的量，以降低擷取率。

您可以增加收集代理程式的收集 (抓取) 間隔：Prometheus 伺服器 (在代理程式模式下執行) 和發行 AWS 版 OpenTelemetry (ADOT) 收集器都支援組態。`scrape_interval`例如，將收集間隔從 30 秒增加到 60 秒，擷取的使用量會減少一半。

您也可以使用 `<relabel_config>` 篩選器傳送至 Amazon Managed Service for Prometheus 的指標。如需有關在 Prometheus 代理程式組態中重新標記的詳細資訊，請參閱 Prometheus 說明文件中的 https://prometheus.io/docs/prometheus/latest/configuration/configuration/#relabel_config。

## 降低查詢成本的最佳方法是什麼？

查詢費用是根據處理的樣本數量而定。您可以降低查詢頻率以降低查詢成本。

為了更容易瞭解對查詢成本貢獻最大的查詢，您可以與支援聯絡人聯繫歸檔票證。Amazon Managed Service for Prometheus 團隊可協助您瞭解對您成本貢獻最大的查詢。

## 如果我減少了指標的保留期間，這是否有助於減少總帳單？

您可以縮短保留期間，但後續無法大幅降低您的成本。

如果您想要縮短 (或增加) 保留期間，則可以將服務限制要求歸檔以變更 `Retention time for ingested data` 配額。

# 如何保持我的警報查詢成本較低？

警示會針對您的資料建立查詢，這會增加您的查詢成本。您可以使用以下策略來最佳化警示查詢，並降低成本。

- 將 Amazon 受管服務用於 Prometheus 警示 — 適用於 Prometheus 的 Amazon 受管服務外部的警示系統可能需要額外的查詢來增加彈性或高可用性，因為外部服務會查詢來自多個可用區域或區域的指標。這包括在 Grafana 中提醒高可用性。這可能會將您的成本乘以三倍或更多。適用於 Prometheus 的 Amazon 受管服務中的警示已經過最佳化，可透過最少的查詢數量為您提供高可用性和彈性。

  我們建議您在 Amazon 受管服務中針對 Prometheus 使用原生警示，而不是外部警示系統。
- 最佳化警示間隔 — 最佳化警示查詢的一種快速方法是增加自動重新整理間隔。如果您收到每分鐘查詢一次的警示，但每五分鐘只需要一次，則增加自動重新整理間隔可為您節省五倍的查詢費用。
- 使用最佳回顧 — 查詢中較大的回顧視窗會增加查詢的成本，因為它會提取更多資料。請確定 PromQL 查詢中的回溯視窗大小適合您需要警示的資料。例如，在下列規則中，運算式包含十分鐘的回顧視窗：

  ```
  - alert: metric:alerting_rule
    expr: avg(rate(container_cpu_usage_seconds_total[10m])) > 0
    for: 2m
  ```

  `expr`將變更為有`avg(rate(container_cpu_usage_seconds_total[5m])) > 0`助於降低查詢成本。

一般而言，請查看您的警示規則，並確定您正在針對服務的最佳指標發出警示。您可以輕鬆地在相同的指標或多個警示上建立重疊的警示，以提供相同資訊，尤其是隨著時間的推移新增警示時。如果您發現您經常看到同時發生的警報組，則可能是您可以優化警報，而不包括所有警報。

這些建議可以幫助您降低成本。最終，您必須通過創建正確的警報集來了解系統狀態來平衡成本。

如需有關在適用於 Prometheus 的 Amazon 受管服務中發出警示的詳細資訊，請參閱。警示管理員

# 我可以使用哪些指標來監控我的成本？

`IngestionRate`在 Amazon 中進行監控 CloudWatch 以跟踪您的攝入成本。如需有關在 CloudWatch 中監視 Prometheus 指標的 Amazon 受管服務的詳細資訊，請參閱。CloudWatch 度量

## 我可以隨時查閱我的帳單嗎？

會 AWS Cost and Usage Report 追蹤您的 AWS 使用情況，並在帳單週期內提供與您帳戶相關聯的預估費用。如需詳細資訊，請參閱什麼是 AWS 成本和使用量報告？「AWS 成本報表和使用報表用戶指南」中的

## 為什麼我的帳單在月初比月底高？

Amazon Managed Service for Prometheus 具有用於擷取的分層定價模式，因此導致初始用量的成本會提高。當您的用量達到更高的擷取層級時，成本較低，您的成本也會降低。如需有關定價的詳細資訊 (包括擷取層)，請參閱 Amazon Managed Service for Prometheus 產品頁面中的定價。

> **Note**
>
> - 層級僅供區域內使用，而非跨地區使用。區域內的使用量必須達到下一個等級，才能使用較低的費率。
> - 在中的組織中 AWS Organizations，層級使用量是按付款人帳戶計算，而非每個帳戶計算 (付款人帳戶一律為組織管理帳戶)。當組織中所有帳戶的總擷取量度 (在某個區域內) 達到下一個層級時，所有帳戶都會以較低的費率收費。

## 我刪除了 Prometheus 工作區的所有 Amazon 託管服務，但似乎仍然被收取費用。可能會發生什麼？

在這種情況下，一種可能性是您仍然 AWS 管理了抓取工具，這些抓取工具已設置為將指標發送到已刪除的工作區。按照說明進行操作尋找並刪除湊集器。

# 與其他 AWS 服務整合

Amazon Managed Service for Prometheus 與其他 AWS 個服務整合。本節說明與 Amazon Elastic Kubernetes Service (Amazon EKS) 成本監控 (搭配 Kubecost) 整合，以及使用 Terraform 模組透過 AWS 可觀察性加速器為您的 EKS 專案建立完整的可觀察性解決方案。

主題



## 與 Amazon EKS 成本監控整合

Amazon Managed Service for Prometheus 與 Amazon Elastic Kubernetes Service (Amazon EKS) 成本監控 (搭配 Kubecost) 整合，以執行成本分配計算，並提供有關最佳化 Kubernetes 叢集的見解。搭配 Kubecost 使用 Amazon Managed Service for Prometheus，您可以可靠地擴展成本監控以支援更大型的叢集。

與 Kubecost 整合可讓您精細掌握 Amazon EKS 叢集成本。您可以依據大多數 Kubernetes 內容彙總成本，從容器層級到叢集層級，甚至是多叢集層級。您可以跨容器或叢集產生報告，以追蹤顯示退款或退款用途的成本。

以下提供在單一或多叢集案例中與 Kubecost 整合的指示：

- 單一叢集整合：若要了解如何將 Amazon EKS 成本監控與單一叢集整合，請參閱 AWS 部落格文章 Integrating Kubecost with Amazon Managed Service for Prometheus。
- 多叢集整合：若要了解如何將 Amazon EKS 成本監控與多叢集整合，請參閱 AWS 部落格文章：Multi-cluster cost monitoring for Amazon EKS using Kubecost and Amazon Managed Service for Prometheus。

> **Note**
>
> 如需有關使用 Kubecost 的詳細資訊，請參閱 Amazon EKS 使用者指南 中的成本監控。

# 使用 AWS 可觀測性加速器

AWS 為您的 Amazon Elastic Kubernetes Service (Amazon EKS) 專案提供可觀察性工具，包含監控、記錄、警示和儀表板。這包含 Amazon Managed Service for Prometheus、Amazon Managed Grafana、適用於 OpenTelemetry 的 AWS Distro 和其他工具。為了協助您一起使用這些工具，AWS 會提供 Terraform 模組，以透過這些服務 (稱為 AWS 可觀測性加速器) 設定可觀測性。

AWS 可觀察性加速器提供監控基礎設施、NGINX 部署和其他方案的範例。本節提供監控 Amazon EKS 叢集內基礎設施的範例。

Terraform 範本和詳細說明可在 Terraform GitHub 頁面的 AWS 可觀察性加速器 頁面上找到。您也可閱讀 說明 AWS 可觀察性加速器的部落格文章。

## 必要條件

若要使用 AWS 可觀察性加速器，您必須擁有現有的 Amazon EKS 叢集和下列先決條件：

- AWS CLI：用於從命令行呼叫 AWS 功能。
- kubectl：用於從命令列控制您的 EKS 叢集。
- Terraform：用於自動建立此解決方案的資源。您必須擁有 IAM 角色的 AWS 提供者設定，該角色可擁有使用您的 AWS 帳戶建立和管理 Amazon Managed Service for Prometheus、Amazon Managed Grafana 和 IAM。如需有關如何設定 Terraform AWS 提供者的詳細資訊，請參閱 Terraform 說明文件中的AWS 提供者。

## 使用基礎設施監控範例

AWS 可觀測性加速器提供範例範本，這些範本使用隨附的 Terraform 模組來設定和組態 Amazon EKS 叢集的可觀測性。此範例示範使用 AWS 可觀測性加速器來設定基礎設施監控。如需有關使用此範本及其所包含其他功能的詳細資訊，請參閱 GitHub 上 Existing Cluster with the AWS Observability Accelerator base and Infrastructure monitoring。

使用基礎設施監控 Terraform 模組

1. 從您要在其中建立專案的資料夾中，使用以下命令複製儲存庫。

```
git clone https://github.com/aws-observability/terraform-aws-observability-accelerator.git
```

2. 使用以下命令初始化 Terraform。

```
cd examples/existing-cluster-with-base-and-infra

terraform init
```

3. 建立新 `terraform.tfvars` 檔案，如下列範例所示。為您的 Amazon EKS 叢集使用 AWS 區域和叢集 ID。

```
# (mandatory) AWS Region where your resources will be located
aws_region = "eu-west-1"

# (mandatory) EKS Cluster name
eks_cluster_id = "my-eks-cluster"
```

4. 若您尚無想要使用的工作區，請建立 Amazon Managed Grafana 工作區。如需有關如何建立新工作區的詳細資訊，請參閱 Amazon Managed Grafana 使用者指南 中的建立您的第一個工作區。

5. 在命令列中執行下列命令，為 Terraform 建立兩個變數以使用 Grafana 工作區。您需要將 *grafana-workspace-id* 替換為 Grafana 工作區的 ID。 。

```
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. [選用] 若要使用現有 Amazon Managed Service for Prometheus 工作區，請將 ID 新增至 `terraform.tfvars` 檔案，如下列範例所示，將 *prometheus-workspace-id* 替換為您的 Prometheus 工作區 ID。如果您未指定現有的工作區，則會為您建立新的 Prometheus 工作區。

```
# (optional) Leave it empty for a new workspace to be created
managed_prometheus_workspace_id = "prometheus-workspace-id"
```

7. 使用下列命令部署解決方案。

```
terraform apply -var-file=terraform.tfvars
```

這將在您的 AWS 帳戶中建立資源，包括以下內容：

- 全新 Amazon Managed Service for Prometheus 工作區 (除非您選擇使用現有的工作區)。
- Prometheus 工作區中的警示管理員組態、警示和規則。

- 您目前工作區中的全新 Amazon Managed Grafana 資料來源和儀表板。將會呼叫資料來源 `aws-observability-accelerator`。儀表板將列在「可觀測性加速器儀表板」下。
- 在所提供 Amazon EKS 叢集中設定 適用於 OpenTelemetry 的 AWS Distro，可將指標傳送至您的 Amazon Managed Service for Prometheus 工作區。

若要檢視新的儀表板，請在 Amazon Managed Grafana 工作區中開啟特定儀表板。如需有關使用 Amazon Managed Grafana 的詳細資訊，請參閱 Amazon Managed Grafana 使用者指南中的在 Grafana 工作區中工作。

# 整合適用於庫伯尼特的 AWS 控制器

Amazon Managed Service for Prometheus 與 Kubernetes 專用 AWS 控制器 (ACK) 整合，並支援管理您在 Amazon EKS 中的工作區、警示管理員和尺規資源。您可以將 AWS 控制器用於 Kubernetes 自訂資源定義 (CRD) 和原生 Kubernetes 物件，而不必定義叢集外部的任何資源。

本節說明如何在現有 Amazon EKS 叢集中為 Prometheus 的 Kubernetes 和 Amazon 受管服務設定 AWS 控制器。

您也可以閱讀介紹 Kubernetes AWS 控制器的部落格文章，以及介紹適用於 Prometheus 之 Amazon 受管服務的 ACK 控制器。

## 必要條件

在開始將適用於 Prometheus 的 Kubernetes AWS 控制器和 Amazon 受管服務與 Amazon EKS 叢集整合之前，您必須具備以下先決條件。

- 您必須擁有現有的 AWS 帳戶 和許可，才能以程式設計方式為 Prometheus 和 IAM 角色建立 Amazon 受管服務。
- 您必須擁有已啟用 OpenID Connect (OIDC) 的現有Amazon EKS 叢集。

  若您未啟用 OIDC，您可以使用下列命令來啟用。請記得將 *YOUR_CLUSTER_NAME* 和 *AWS_REGION* 替換為帳戶的正確值。

```
eksctl utils associate-iam-oidc-provider \
    --cluster ${YOUR_CLUSTER_NAME} --region ${AWS_REGION} \
    --approve
```

如需有關將 OIDC 與 Amazon EKS 搭配使用的詳細資訊，請參閱 Amazon EKS 使用者指南中的 OIDC 身分識別提供者身分驗證和建立 IAM OIDC 提供者。

- 您必須在 Amazon EKS 叢集中安裝 Amazon EBS CSI 驅動程式。
- 您必須已安裝 AWS CLI。的用 AWS CLI 於從命令列呼叫 AWS 功能。
- 必須安裝 Helm，Kubernetes 的套件管理員。
- 必須在您的 Amazon EKS 叢集中設定使用 Prometheus 的控制平面指標。
- 您必須擁有 Amazon Simple Notification Service (Amazon SNS) 的主題，您希望從新工作區傳送警示。請確認您已授予 Amazon Managed Service for Prometheus 權限，以將訊息傳送到該主題。

當您適當設定的 Amazon EKS 叢集時，您應該可以透過呼叫 `kubectl get --raw /metrics` 查看為 Prometheus 格式化的指標。現在您已準備好安裝 Kubernetes 服務 AWS 控制器的控制器，並使用它來部署適用於 Prometheus 資源的 Amazon 受管服務。

## 為 Kubernetes 部署具有 AWS 控制器的工作區

若要為 Prometheus 工作區部署新的 Amazon 受管服務，您需要安裝 Kubernetes 控制器的 AWS 控制器，然後使用該控制器來建立工作區。

使 AWS 用 Kubernetes 的控制器為 Prometheus 工作區部署新的 Amazon 受管服務

1. 使用下列命令來使用 Helm 安裝 Amazon Managed Service for Prometheus 服務控制器。如需詳細資訊，請參閱上的在 Kubernetes 的 AWS 控制器中安裝 ACK 控制器說明文件。 GitHub為您的系統使用正確的*##*，例如`us-east-1`。

```
export SERVICE=prometheusservice
export RELEASE_VERSION=`curl -sL https://api.github.com/repos/aws-controllers-k8s/
$SERVICE-controller/releases/latest | grep '"tag_name":' | cut -d'"' -f4`
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
```

   幾分鐘後，您應該會看到類似於以下內容的回應，表示成功。

```
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"
```

您可以選擇性地使用下列命令來驗證 Kubernetes 控 AWS 制器的控制器是否已成功安裝。

```
helm list --namespace $ACK_SYSTEM_NAMESPACE -o yaml
```

這將傳回與控制器 `ack-prometheusservice-controller` 有關的資訊，包含 `status: deployed`。

2. 使用下列內文建立稱為 `workspace.yaml` 的檔案。這將作為您正在建立的工作區組態使用。

```
apiVersion: prometheusservice.services.k8s.aws/v1alpha1
kind: Workspace
metadata:
  name: my-amp-workspace
spec:
  alias: my-amp-workspace
  tags:
    ClusterName: EKS-demo
```

3. 執行下列命令以建立工作區 (此命令取決於您在步驟 1 中設定的系統變數)。

```
kubectl apply -f workspace.yaml -n $ACK_SYSTEM_NAMESPACE
```

在幾分鐘之內，您應該能夠看到一個新的工作區，在您的帳戶中稱為 `my-amp-workspace`。

執行下列命令以檢視工作區的詳細資訊和狀態，包含工作區 ID。或者，您也可以在 Amazon Managed Service for Prometheus 主控台中檢視新的工作區。

```
kubectl describe workspace my-amp-workspace -n $ACK_SYSTEM_NAMESPACE
```

> **Note**
>
> 您也可以使用現有工作區，而不建立新工作區。

4. 建立兩個新的 yaml 檔案做為規則群組的組態，並 AlertManager 使用下列組態建立下一個檔案。

將此組態另存為 rulegroup.yaml。將 *WORKSPACE-ID* 替換為上一個步驟的工作區 ID。

```
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 }}"
```

將以下組態另存為 alertmanager.yaml。將 *WORKSPACE-ID* 替換為上一個步驟的工作區 ID。將主*# ARN ### ARN*，以便將通知傳送到您正在使用的*##*的 Amazon SNS 主題。 AWS 區域 請記住，Amazon Managed Service for Prometheus 必須有 Amazon SNS 主題的許可。

```
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**
>
> 若要深入瞭解這些組態檔案的格式，請參閱 RuleGroupsNamespaceData 和 AlertManagerDefinitionData。

5. 執行下列命令以建立規則群組和警示管理員組態 (此命令取決於您在步驟 1 中設定的系統變數)。

```
kubectl apply -f rulegroup.yaml -n $ACK_SYSTEM_NAMESPACE
kubectl apply -f alertmanager.yaml -n $ACK_SYSTEM_NAMESPACE
```

幾分鐘內將可進行這些變更。

> **Note**
>
> 若要更新資源，而不是建立資源，只要更新 yaml 檔案，然後再次執行 `kubectl apply` 命令即可。
> 若要刪除資源，請執行下列命令。取*ResourceType*代為您要刪除Workspace、AlertManagerDefinition或的資源類型RuleGroupNamespace。以要刪除的資源名稱取*ResourceName*代。
>
> ```
> kubectl delete ResourceType ResourceName -n $ACK_SYSTEM_NAMESPACE
> ```

這樣會完成新工作區的部署。下一節說明組態叢集以傳送指標至該工作區。

## 組態 Amazon EKS 叢集以寫入 Amazon Managed Service for Prometheus 工作區

本節說明如何使用 Helm 將 Amazon EKS 叢集中執行的 Prometheus 組態為遠端將指標寫入您在上一節建立的 Amazon Managed Service for Prometheus 工作區。

在此程序中，您將需要已建立的 IAM 角色名稱以用於擷取指標。如果您尚未這麼做，請參閱 自 Amazon EKS 叢集設定指標擷取作業的服務角色 以取得詳細資訊和指示。如果您遵循這些指示，將會呼叫 IAM 角色 `amp-iamproxy-ingest-role`。

為 Amazon EKS 叢集設定 Amazon EKS 組態遠端寫入

1. 使用下列命令來取得工作區的 `prometheusEndpoint`。將 *`WORKSPACE-ID`* 替換為上一節的工作區 ID。

   ```
   aws amp describe-workspace --workspace-id WORKSPACE-ID
   ```

   PromeTheusEndpoint 將在傳回結果中，並依照下列方式格式化：

   ```
   https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-a1b2c3d4-a123-b456-
   c789-ac1234567890/
   ```

   儲存此 URL，以便用於接下來的幾個步驟。

2. 使用下列內文建立新檔案，並將其稱為 `prometheus-config.yaml`。將*##*替換為您的帳戶 ID、將 *`WorkSpaceURL/`* 替換為您剛才找到的 URL，以及將 *`region`* 替換為您系統適用的 AWS 區域 。

   ```
   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. 使用下面的 Helm 命令，尋找 Prometheus 圖表和命名空間名稱以及的圖表版本。

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

根據到目前為止的步驟，Prometheus 圖表和命名空間都應該命名為 `prometheus`，並且圖表版本可能是 `15.2.0`

4. 使用、執行下列命令 *PrometheusChartNamePrometheusNamespace*，並在上一個步驟中*PrometheusChartVersion*找到。

```
helm upgrade PrometheusChartName prometheus-community/prometheus -
n PrometheusNamespace -f prometheus-config.yaml --version PrometheusChartVersion
```

幾分鐘後，您會看到升級成功的訊息。

5. 或者，透過 `awscurl` 查詢 Amazon Managed Service for Prometheus 端點，驗證是否成功傳送指標。將「*##*」取代為您 AWS 區域 正在使用的項目，並使用您在步驟 1 中找到的 *URL ###### #* URL」。

```
awscurl --service="aps" --region="Region" "workspaceURL/api/v1/query?
query=node_cpu_seconds_total"
```

您現在已經建立 Amazon Managed Service for Prometheus 工作區，並使用 YAML 檔案作為組態，從 Amazon EKS 叢集連線到該工作區。這些檔案，稱為自訂資源定義 (CRD)，在 Amazon EKS 叢集內運作中。您可以使用 Kubernetes AWS 控制器的控制器，直接從叢集管理適用於 Prometheus 資源的所有 Amazon 受管服務。

## 整合 CloudWatch 指標與 Firehose

本節說明如何儀器 Amazon 指 CloudWatch 標串流和使用 Amazon 資料 Firehose，AWS Lambda以及如何將指標導入 Prometheus 的 Amazon 受管服務。

您將使用 AWS Cloud Development Kit (CDK) 設定堆疊，以建立 Firehose 交付串流、Lambda 和 Amazon S3 儲存貯體，以展示完整的案例。

# 基礎設施

您必須做的第一件事是為此配方設定基礎設施。

CloudWatch 指標串流允許將串流指標資料轉送至 HTTP 端點或 Amazon S3 儲存貯體。

設定基礎設施將包括 4 個步驟：

- 組態先決條件
- 建立 Amazon Managed Service for Prometheus 工作區
- 安裝相依性
- 部署堆疊

先決條件

- 已在您的環境中安裝並設定。 AWS CLI
- 已在您的環境中安裝 AWS CDK Typescript。
- 已在您的環境中安裝 Node.js 和 Go。
- AWS 可觀察性 CloudWatch 指標出口商 github 存儲庫（`CWMetricsStreamExporter`）已克隆到您的本地計算機。

建立 Amazon Managed Service for Prometheus 工作區

1. 此配方中的示範應用程式將在 Amazon Managed Service for Prometheus 最上方執行。透過下列命令建立 Amazon Managed Service for Prometheus 工作區：

   ```
   aws amp create-workspace --alias prometheus-demo-recipe
   ```

2. 確定已使用下列指令建立您的工作區：

   ```
   aws amp list-workspaces
   ```

   如需有關 Amazon Managed Service for Prometheus 的詳細資訊，請參閱 Amazon Managed Service for Prometheus 使用者指南。

安裝相依項目

1. 安裝相依項目

   從 aws-o11y-recipes 儲存庫的根目錄中，使用以下指令將目錄變更為 CWMetricStreamExporter：

   ```
   cd sandbox/CWMetricStreamExporter
   ```

   現在，這將視為回溯的根目錄，並向前走。

2. 透過下列命令將目錄變更為 /cdk：

   ```
   cd cdk
   ```

3. 執行以下命令以安裝 CDK 相依性：

   ```
   npm install
   ```

4. 將目錄變更回儲存庫的根目錄，然後使用以下命令將目錄變更為 /lambda：

   ```
   cd lambda
   ```

5. 一旦進入 /lambda 資料夾後，使用以下命令安裝 Go 相依項目：

   ```
   go get
   ```

   現在已安裝所有相依性。

部署堆疊

1. 在儲存庫的根目錄中，開啟 config.yaml 並修改 Amazon Managed Service for Prometheus 工作區 URL，方法是將 {workspace} 替換為新建立的工作區 ID ，以及您 Amazon Managed Service for Prometheus 工作區所在的區域。

   例如，將以下項目修改為：

   ```
   AMP:
       remote_write_url: "https://aps-workspaces.us-east-2.amazonaws.com/workspaces/
   {workspaceId}/api/v1/remote_write"
   ```

```
    region: us-east-2
```

根據您的喜好變更 Firehose 交付串流和 Amazon S3 儲存貯體的名稱。

2. 要構建 AWS CDK 和 Lambda 代碼，請在回購的根目錄中運行以下讚揚：

```
npm run build
```

此構建步驟可確保構建 Go Lambda 二進製文件，並將 CDK 部署到. CloudFormation

3. 若要完成部署，請檢閱並接受堆疊所需的 IAM 變更。
4. （選用）若已透過執行下列命令建立堆疊，則可有所變化。

```
aws cloudformation list-stacks
```

名為 CDK Stack 的堆疊將會在清單中。

## 創建一個 Amazon CloudWatch 流

現在，您已經有了一個 lambda 函數來處理指標，您可以從 Amazon 創建指標流 CloudWatch。

若要建立 CloudWatch 量度資料流

1. 導覽至 CloudWatch 主控台，位於 https://console.aws.amazon.com/cloudwatch/home#metric-streams:streamsList，然後選取建立指標串流。
2. 選取所需指標，可以是所有指標，或僅從所選命名空間。
3. 在 Configuration 下方，選擇「選取您帳戶擁有的現有 Firehose」。
4. 您將會使用 CDK 先前建立的 Firehose。在「選取您的 Kinesis 資料 Firehose 串流」下拉式清單中，選取先前建立的串流。名稱將會像是 CdkStack-KinesisFirehoseStream123456AB-sample1234。
5. 將輸出格式變更為 JSON。
6. 為指標串流賦予對您有意義的名稱。
7. 選擇 Create metric stream (建立指標串流)。
8. (選用) 若要驗證 Lambda 函數調用，請導覽至 Lambda 主控台並選擇函數 KinesisMessageHandler。選取「監控」索引標籤和「記錄」子索引標籤，在「最近的呼叫」下應該會有要觸發的 Lambda 函數輸入項。

> **Note**
>
> 最多可能需要 5 分鐘才會開始在「監控」索引標籤中顯示調用。

您的指標現在正從 Amazon 流式傳輸 CloudWatch 到 Prometheus 的 Amazon 託管服務。

## 清除

您可能想要清除本範例中使用的資源。下列程序說明如何執行此作業。這會停止您建立的指標串流。

清理資源

1. 首先使用以下命令刪除 CloudFormation 堆棧：

```
cd cdk
cdk destroy
```

2. 移除 Amazon Managed Service for Prometheus 工作區：

```
aws amp delete-workspace --workspace-id \
    `aws amp list-workspaces --alias prometheus-sample-app --query
 'workspaces[0].workspaceId' --output text`
```

3. 最後，使用 Amazon 控 CloudWatch 制 CloudWatch 台刪除 Amazon 指標流。

# Amazon Managed Service for Prometheus 中的安全性

雲端安全是 AWS 最重視的一環。身為 AWS 的客戶，您將能從資料中心和網路架構中獲益，這些都是專為最重視安全的組織而設計的。

安全性是 AWS 與您共同肩負的責任。共同責任模型將其描述為雲端的安全性和雲端中的安全性：

- 雲端本身的安全 – AWS 負責保護在 AWS Cloud 中執行 AWS 服務的基礎設施。AWS 也提供您可安全使用的服務。第三方稽核人員會定期測試和驗證我們安全性的有效性，作為 AWS 合規計畫的一部分。若要了解適用於 Amazon Managed Service for Prometheus 的合規計劃，請參閱 AWS 合規計劃的服務範圍 。
- 雲端內部的安全 – 您的責任取決於所使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的請求和適用法律和法規。

本說明文件有助您瞭解如何在使用 Amazon Managed Service for Prometheus 時套用共同責任模型。下列主題說明如何設定 Amazon Managed Service for Prometheus 來符合您的安全與合規目標。您也將瞭解如何使用其他 AWS 服務，幫助您監控並保護 Amazon Managed Service for Prometheus 資源。

主題

# Amazon Managed Service for Prometheus 中的資料保護

AWS 共同責任模型適用於 Prometheus 的 Amazon 受管服務中的資料保護。如此模型中所述， AWS 負責保護執行所有 AWS 雲端. 您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務 的安全組態和管理任務。如需資料隱私權的詳細資訊，請參閱資料隱私權常見問答集。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 AWS 共同的責任模型和 GDPR 部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 登入資料並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源進行通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用設定 API 和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案，以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果您在透過命令列介面或 API 存取時需要經 AWS 過 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，請參閱聯邦資訊處理標準 (FIPS) 140-2 概觀。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您使用主控台、API 或 SDK 使用適 AWS 服務 用於 Prometheus 或其他人的 Amazon 受管服務時。 AWS CLI AWS 您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

主題

- Amazon Managed Service for Prometheus 收集的資料
- 靜態加密

## Amazon Managed Service for Prometheus 收集的資料

Amazon Managed Service for Prometheus 會收集並存放您設定的作業指標，以便從您帳戶中執行的 Prometheus 伺服器傳送至 Amazon Managed Service for Prometheus。資料包含以下內容：

- 指標值

- 有助於識別和分類資料的指標標籤 (或任意鍵值配對)
- 資料範例的時間戳記

唯一租用戶 ID 會隔離不同客戶的資料。這些 ID 限制可存取的客戶資料。客戶無法變更租用戶 ID。

適用 AWS Key Management Service 於 Prometheus 的 Amazon 受管服務會加密其使用 () 金鑰存放的資料。AWS KMS Amazon Managed Service for Prometheus 會管理這些金鑰。

> **Note**
>
> 適用於 Prometheus 的 Amazon 受管服務可支援建立用於加密資料的客戶受管金鑰。如需 Prometheus 專用 Amazon 受管服務預設使用的金鑰，以及如何使用您自己的客戶受管金鑰的詳細資訊，請參閱。靜態加密

傳輸中的資料會自動使用 HTTPS 進行加密。適用於 Prometheus 的 Amazon 受管服務可在內部使用 HTTPS 保護區域內可用區域之間的 AWS 連線。

## 靜態加密

根據預設，Prometheus 的 Amazon 受管服務會自動為您提供靜態加密，並使用 AWS 擁有的加密金鑰來執行此操作。

- AWS 擁有的金鑰 — 適用於 Prometheus 的 Amazon 受管服務會使用這些金鑰自動加密上傳至您工作區的資料。您無法檢視、管理或使用 AWS 擁有的金鑰，也無法稽核其使用情況。不過，您不需要採取任何動作或變更任何程式，即可保護加密您資料的金鑰。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的 AWS 擁有的金鑰。

加密靜態資料有助於減少保護敏感客戶資料 (例如個人可識別資訊) 的營運成本並降低複雜性。這可讓您建立符合嚴格加密合規性或管制需求的安全應用程式。

您也可以在建立工作區時，選擇使用客戶受管金鑰：

- 客戶受管金鑰：Amazon Managed Service for Prometheus 支援使用您建立、擁有並管理的對稱客戶受管金鑰來加密工作區中的資料。由於您可以完全控管此加密，因此能執行以下任務：
  - 建立和維護金鑰政策
  - 建立和維護 IAM 政策和授予操作
  - 啟用和停用金鑰政策

- 輪換金鑰密碼編譯資料
- 新增標籤
- 建立金鑰別名
- 安排金鑰供刪除

如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的客戶自管金鑰。

選擇要小心使用客戶管理的金鑰還是 AWS 擁有的金鑰。使用客戶管理金鑰建立的工作區之後無法轉換為使用 AWS 擁有的金鑰 (反之亦然)。

> **Note**
>
> 適用於 Prometheus 的 Amazon 受管服務會使用 AWS 擁有的金鑰自動啟用靜態加密，免費保護您的資料。
> 但是，使用客戶管理的金鑰需要 AWS KMS 支付費用。如需定價的詳細資訊，請參閱 AWS Key Management Service 定價。

如需有關的詳細資訊 AWS KMS，請參閱「什麼是 AWS Key Management Service？」

> **Note**
>
> 使用客戶受管金鑰建立的工作區無法使用 AWS 受管收集器進行擷取。

## Amazon Prometheus 託管服務如何在 AWS KMS

Amazon Managed Service for Prometheus 需要三個授權才能使用您的客戶受管金鑰。

當您使用客戶受管金鑰加密的 Prometheus 工作區建立 Amazon 受管服務時，Prometheus 的 Amazon 受管服務會將請求傳送至，代表您建立三個授權。CreateGrant AWS KMS中的授權可 AWS KMS 用於授予適用於 Prometheus 的 Amazon 受管服務存取您帳戶中的 KMS 金鑰，即使未直接代表您呼叫 (例如，存放從 Amazon EKS 叢集擷取的指標資料時) 也是如此。

Amazon Managed Service for Prometheus 需要授權才能使用您的客戶受管金鑰進行下列內部操作：

- 將DescribeKey要求傳送 AWS KMS 至，以確認建立工作區時提供的對稱客戶受管 KMS 金鑰是否有效。

- 傳送GenerateDataKey要求 AWS KMS 以產生由客戶管理金鑰加密的資料金鑰。
- 發送解密請求 AWS KMS 以解密加密的數據密鑰，以便可以使用它們來加密您的數據。

適用於 Prometheus 的 Amazon 受管服務會為 AWS KMS 金鑰建立三個授權，讓 Prometheus 的 Amazon 受管服務代表您使用金鑰。您可以透過變更金鑰政策、停用金鑰或撤銷授權來移除金鑰的存取權。在執行這些動作之前，應先充分了解這些動作的後果。您的工作區中可能會發生資料遺失。

如果您以任何方式移除任何授權的存取權，Amazon Managed Service for Prometheus 將無法存取使用客戶受管金鑰加密的任何資料，也無法儲存傳送至工作區的資料，而這會影響與該資料相關的操作。傳送至工作區的新資料將無法供存取，而且可能永久遺失。

> ⚠ Warning
>
> - 如果您停用金鑰，或在金鑰政策中移除 Amazon Managed Service for Prometheus 的存取權，則無法再存取工作區資料。傳送至工作區的新資料將無法供存取，而且可能永久遺失。
>
>  透過還原 Amazon Managed Service for Prometheus 對金鑰的存取權，就可以再次存取工作區資料並開始接收新資料。
> - 如果您撤銷授權，則無法重新建立該授權，且工作區中的資料會永久遺失。

## 步驟 1：建立客戶受管金鑰

您可以使用 AWS Management Console、或 AWS KMS API 建立對稱的客戶管理金鑰。只要您透過政策提供正確的存取權，金鑰與 Amazon Managed Service for Prometheus 工作區就不需要在相同帳戶中，如下所述。

建立對稱客戶受管金鑰

請依照《AWS Key Management Service 開發人員指南》中建立對稱客戶受管金鑰的步驟進行。

金鑰政策

金鑰政策會控制客戶受管金鑰的存取權限。每個客戶受管金鑰都必須只有一個金鑰政策，其中包含決定誰可以使用金鑰及其使用方式的陳述式。在建立客戶受管金鑰時，可以指定金鑰政策。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的管理客戶受管金鑰的存取。

若要將客戶受管金鑰與 Amazon Managed Service for Prometheus 工作區搭配使用，則必須在金鑰政策中允許下列 API 操作：

- kms:CreateGrant：新增客戶受管金鑰的授權。授權會控制對指定 KMS 金鑰的存取權，也就是允許存取 Amazon Managed Service for Prometheus 所需的授權操作。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的使用授權。

  這可讓 Amazon Managed Service for Prometheus 執行下列操作：

  - 呼叫 GenerateDataKey 以產生加密的資料金鑰並加以儲存，因為資料金鑰不會立即用來加密。
  - 呼叫 Decrypt 以使用儲存的加密資料金鑰來存取加密的資料。
- kms:DescribeKey：提供客戶受管金鑰的詳細資訊，讓 Amazon Managed Service for Prometheus 能夠驗證金鑰。

以下是您可針對 Amazon Managed Service for Prometheus 新增的政策陳述式範例：

```
 "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>
 ]
```

- 如需有關在政策中指定許可的詳細資訊，請參閱《AWS Key Management Service 開發人員指南》。
- 如需有關故障診斷金鑰存取的詳細資訊，請參閱《AWS Key Management Service 開發人員指南》。

### 步驟 2：為 Prometheus 的 Amazon 受管服務指定客戶受管金鑰

當您建立工作區時，可以透過輸入 KMS 金鑰 ARN 來指定客戶受管金鑰，Amazon Managed Service for Prometheus 會使用此 ARN 來加密工作區所儲存的資料。

### 步驟 3：從其他服務（例如 Amazon 受管的 Grafana）訪問數據

此步驟為選擇性步驟，只有當您需要從其他服務存取 Prometheus 資料的 Amazon 受管服務時，才需要此步驟。

您的加密數據無法從其他服務訪問，除非他們也有權使用密 AWS KMS 鑰。例如，如果您想要使用 Amazon 受管 Grafana 來建立資料儀表板或警示，您必須將金鑰的存取權授予 Amazon 受管的 Grafana。

讓 Amazon 受管的 Grafana 存取您的客戶受管金鑰

1. 在您的 Amazon 受管 Grafana 工作區清單中，選取您想要存取 Prometheus 之 Amazon 受管服務的工作區名稱。這會顯示有關 Amazon 受管的 Grafana 工作區的摘要資訊。
2. 請記下您的工作區所使用的 IAM 角色名稱。名稱的格式為`AmazonGrafanaServiceRole-<unique-id>`。主控台會顯示角色的完整 ARN。您將在稍後的步驟中在 AWS KMS 主控台中指定此名稱。
3. 在您的AWS KMS 客戶受管金鑰清單中，選擇您在建立 Prometheus 適用的 Amazon 受管服務工作區期間使用的客戶受管金鑰。這會開啟金鑰組態詳細資料頁面。
4. 選取 [金鑰使用者] 旁邊的 [新增] 按鈕。

5. 從名稱清單中，選擇您在上面提到的 Amazon 受管 Grafana IAM 角色。為了使其更容易找到，您也可以按名稱進行搜索。
6. 選擇 [新增]，將 IAM 角色新增至金鑰使用者清單。

您的 Amazon 受管 Grafana 工作區現在可以存取適用於 Prometheus 的 Amazon 受管服務工作區中的資料。您可以將其他使用者或角色新增至主要使用者，以便讓其他服務存取您的工作區。

## Amazon Managed Service for Prometheus 加密內容

加密內容是一組選用的金鑰值對，包含資料的其他相關內容資訊。

AWS KMS 使用加密內容作為其他驗證資料，以支援已驗證的加密。當您在加密資料的要求中包含加密內容時，會將加密內容 AWS KMS 繫結至加密的資料。若要解密資料，您必須在請求中包含相同的加密內容。

Amazon Managed Service for Prometheus 加密內容

適用於 Prometheus 的 Amazon 受管服務在所有密 AWS KMS 碼編譯作業中使用相同的加密內容，其中金鑰所在`aws:amp:arn`，值為工作區的 Amazon 資源名稱 (ARN)。

Example

```
"encryptionContext": {
    "aws:aps:arn": "arn:aws:aps:us-west-2:111122223333:workspace/ws-sample-1234-
abcd-56ef-7890abcd12ef"
}
```

使用加密內容進行監控

當您使用對稱的客戶受管金鑰加密工作區資料時，您也可以在稽核記錄和日誌中使用加密內容，以指出客戶受管金鑰的使用方式。加密內容也會出現在AWS CloudTrail 或 Amazon 日誌產生的 CloudWatch 日誌中。

使用加密內容控制對客戶受管金鑰的存取

您也可以在金鑰政策和 IAM 政策中，使用加密內容作為 `conditions` 來控制對於對稱客戶受管金鑰的存取。您也可以在授予中使用加密內容條件。

AmazonAmazon Managed Service for Prometheus 會在授權中，使用加密內容限制來控制對帳戶或區域中的客戶受管金鑰的存取權。授予條件會要求授予允許的操作使用指定的加密內容。

Example

以下是授予特定加密內容之客戶受管金鑰存取權的金鑰政策陳述式範例。此政策陳述式中的條件會要求具有指定加密內容的加密內容條件。

```
{
    "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"
          }
     }
}
```

## 監控 Amazon Managed Service for Prometheus 的加密金鑰

當您將 AWS KMS 客戶受管金鑰與 Prometheus 工作區的 Amazon 受管服務搭配使用時，您可以使用AWS CloudTrail或 Amazon CloudWatch Logs 來追蹤適用於 Prometheus 的 Amazon 受管服務傳送到的請求。 AWS KMS

下列範例是針對`CreateGrant`、`GenerateDataKey`、`Decrypt`和監控針對 Prometheus `DescribeKey` 的 Amazon 受管服務呼叫的 KMS 操作，以存取由客戶管理金鑰加密的資料的 AWS CloudTrail 事件：

### CreateGrant

當您使用 AWS KMS 客戶受管金鑰加密工作區時，Prometheus 的 Amazon 受管服務會代表您傳送三個`CreateGrant`請求，以存取您指定的 KMS 金鑰。Amazon Managed Service for Prometheus 建立的授權專屬於與 AWS KMS 客戶受管金鑰相關聯的資源。

以下範例事件會記錄 `CreateGrant` 操作：

```
{
    "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

當您為工作區啟用 AWS KMS 客戶受管金鑰時，Prometheus 的 Amazon 受管服務會建立唯一金鑰。它會將要GenerateDataKey求傳送至 AWS KMS 指定資源的 AWS KMS客戶管理金鑰。

下面的範例事件會記錄 GenerateDataKey 操作：

```
{
    "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

當查詢在加密的工作區上產生時，Amazon Managed Service for Prometheus 會呼叫 `Decrypt` 操作以使用儲存的加密資料金鑰來存取加密的資料。

下面的範例事件會記錄 `Decrypt` 操作：

```
{
    "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

Amazon Managed Service for Prometheus 會使用 `DescribeKey` 操作來驗證與工作區相關聯的 AWS KMS 客戶受管金鑰是否存在帳戶和區域中。

下面的範例事件會記錄 `DescribeKey` 操作：

```
{
    "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"
}
```

### 進一步了解

下列資源會提供有關靜態資料加密的詳細資訊。

- 如需 AWS Key Management Service 基本概念的詳細資訊，請參閱《AWS Key Management Service 開發人員指南》。
- 如需有關的安全性最佳做法的詳細資訊 AWS Key Management Service，請參閱開AWS Key Management Service 發人員指南。

## Amazon Managed Service for Prometheus 的識別與存取管理

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。AWS 服務 IAM 管理員可控制哪些人員可進行身分驗證 (登入) 並獲得授權 (具有許可) 以使用 Amazon Managed Service for Prometheus 資源。IAM 是您可以使用的 AWS 服務 ，無需額外付費。

主題

- 物件
- 使用身分驗證
- 使用政策管理存取權
- Amazon Managed Service for Prometheus 如何與 IAM 一併使用
- Amazon Managed Service for Prometheus 的身分型政策範例
- AWS 適用於 Prometheus 的 Amazon 受管服務的受管政策
- Amazon Managed Service for Prometheus 身分和存取的疑難排解

### 物件

您使用 AWS Identity and Access Management (IAM) 的方式會有所不同，具體取決於您在 Amazon Prometheus 受管服務中所做的工作。

服務使用者 – 如果您使用 Amazon Managed Service for Prometheus 執行任務，您的管理員會為您提供需要的憑證和許可。隨著您為了執行作業而使用的 Amazon Managed Service for Prometheus 功能數量變多，您可能會需要額外的許可。了解存取許可的管理方式可協助您向管理員請求正確的許可。如

果您無法存取 Amazon Managed Service for Prometheus 的功能，請參閱 Amazon Managed Service for Prometheus 身分和存取的疑難排解。

服務管理員：若您在公司負責管理 Amazon Managed Service for Prometheus 資源，您應該擁有 Amazon Managed Service for Prometheus 的完整存取權。您的任務是判斷服務使用者應該存取的 Amazon Managed Service for Prometheus 功能和資源。接著，您必須將請求提交給您的 IAM 管理員，來變更您服務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要深入瞭解貴公司如何可使用 IAM 和 Amazon Managed Service for Prometheus，請參閱 Amazon Managed Service for Prometheus 如何與 IAM 一併使用。

IAM 管理員：如果您是 IAM 管理員，則可能想要瞭解您可如何寫入政策的詳細資料，以管理 Amazon Managed Service for Prometheus 的存取權。若要檢視您可在 IAM 中使用的範例 Amazon Managed Service for Prometheus 以識別為基礎政策，請參閱 Amazon Managed Service for Prometheus 的身分型政策範例。

## 使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以 IAM 使用者身分或假設 IAM 角色進行驗證 (登入 AWS)。 AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。 AWS IAM Identity Center (IAM 身分中心) 使用者、貴公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料都是聯合身分識別的範例。您以聯合身分登入時，您的管理員先前已設定使用 IAM 角色的聯合身分。當您使 AWS 用同盟存取時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需有關登入的詳細資訊 AWS，請參閱《AWS 登入 使用指南》 AWS 帳戶中的如何登入您的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以使用您的認證以加密方式簽署您的要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱 IAM 使用者指南中的簽署 AWS API 請求。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如， AWS 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。如需更多資訊，請參閱 AWS IAM Identity Center 使用者指南中的多重要素驗證和 IAM 使用者指南中的在 AWS中使用多重要素驗證 (MFA)。

### AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入

來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需這些任務的完整清單，了解需以根使用者登入的任務，請參閱 IAM 使用者指南中的需要根使用者憑證的任務。

## 聯合身分

最佳作法是要求人類使用者 (包括需要系統管理員存取權的使用者) 使用與身分識別提供者的同盟，才能使用臨時登入資料進行存取 AWS 服務 。

聯合身分識別是來自企業使用者目錄的使用者、Web 身分識別提供者、Identi ty Center 目錄，或使用透過身分識別來源提供的認證進行存取 AWS 服務 的任何使用者。 AWS Directory Service同盟身分存取時 AWS 帳戶，他們會假設角色，而角色則提供臨時認證。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連線並同步到自己身分識別來源中的一組使用者和群組，以便在所有應用程式 AWS 帳戶 和應用程式中使用。如需 IAM Identity Center 的詳細資訊，請參閱 AWS IAM Identity Center 使用者指南中的什麼是 IAM Identity Center？。

## IAM 使用者和群組

IAM 使用者是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定許可。建議您盡可能依賴暫時憑證，而不是擁有建立長期憑證 (例如密碼和存取金鑰) 的 IAM 使用者。但是如果特定使用案例需要擁有長期憑證的 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱 IAM 使用者指南中的為需要長期憑證的使用案例定期輪換存取金鑰。

IAM 群組是一種指定 IAM 使用者集合的身分。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為 IAMAdmins 的群組，並給予該群組管理 IAM 資源的許可。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。如需進一步了解，請參閱IAM 使用者指南中的建立 IAM 使用者 (而非角色) 的時機。

## IAM 角色

IAM 角色是您 AWS 帳戶 內部具有特定許可的身分。它類似 IAM 使用者，但不與特定的人員相關聯。您可以切換角色，在中暫時擔任 IAM 角色。 AWS Management Console 您可以透過呼叫 AWS CLI 或 AWS API 作業或使用自訂 URL 來擔任角色。如需使用角色的方法詳細資訊，請參閱 IAM 使用者指南中的使用 IAM 角色。

使用暫時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 — 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱 IAM 使用者指南中的為第三方身分提供者建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的許可集。
- 暫時 IAM 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。
- 跨帳戶存取權：您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的主體) 存取您帳戶的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資源（而不是使用角色作為代理）。若要了解跨帳戶存取角色和以資源為基礎的政策之間的差異，請參閱 IAM 使用者指南中的 IAM 中的跨帳戶資源存取。
- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (Amazon S3) 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 轉寄存取工作階段 (FAS) — 當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱《轉發存取工作階段》。
  - 服務角色 – 服務角色是服務擔任的 IAM 角色，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱 IAM 使用者指南中的建立角色以委派許可給 AWS 服務服務。
  - 服務連結角色 — 服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 — 您可以使用 IAM 角色來管理在 EC2 執行個體上執行的應用程式以及發出 AWS CLI 或 AWS API 請求的臨時登入資料。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並提供給其所有應用程式，請建立連接至執行個體的執行個體設定檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得暫時憑證。如需詳細資訊，請參閱 IAM 使用者指南中的利用 IAM 角色來授予許可給 Amazon EC2 執行個體上執行的應用程式。

如需了解是否要使用 IAM 角色或 IAM 使用者，請參閱 IAM 使用者指南中的建立 IAM 角色 (而非使用者) 的時機。

# 使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS ，當與身分識別或資源相關聯時，會定義其權限。 AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以 JSON 文件的形式儲存在中。如需 JSON 政策文件結構和內容的詳細資訊，請參閱 IAM 使用者指南中的 JSON 政策概觀。

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

IAM 政策定義該動作的許可，無論您使用何種方法來執行操作。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該原則的使用者可以從 AWS Management Console AWS CLI、或 AWS API 取得角色資訊。

## 身分型政策

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分類型政策，請參閱 IAM 使用者指南中的建立 IAM 政策。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶. 受管政策包括 AWS 受管政策和客戶管理的策略。如需了解如何在受管政策及內嵌政策間選擇，請參閱 IAM 使用者指南中的在受管政策和內嵌政策間選擇。

## 資源型政策

資源型政策是連接到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

資源型政策是位於該服務中的內嵌政策。您無法在以資源為基礎的政策中使用 IAM 的 AWS 受管政策。

### 存取控制清單 (ACL)

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

Amazon S3 和 Amazon VPC 是支援 ACL 的服務範例。 AWS WAF如需進一步了解 ACL，請參閱 Amazon Simple Storage Service 開發人員指南中的存取控制清單 (ACL) 概觀。

### 其他政策類型

AWS 支援其他較不常見的原則類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 許可界限 – 許可範圍是一種進階功能，可供您設定身分型政策能授予 IAM 實體 (IAM 使用者或角色) 的最大許可。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 `Principal` 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政策中的明確拒絕都會覆寫該允許。如需許可界限的詳細資訊，請參閱 IAM 使用者指南中的 IAM 實體許可界限。
- 服務控制策略（SCP）— SCP 是 JSON 策略，用於指定中組織或組織單位（OU）的最大權限。AWS Organizations AWS Organizations 是一種用於分組和集中管理您企業擁 AWS 帳戶 有的多個服務。若您啟用組織中的所有功能，您可以將服務控制政策 (SCP) 套用到任何或所有帳戶。SCP 限制成員帳戶中實體的權限，包括每個 AWS 帳戶根使用者帳戶。如需 Organizations 和 SCP 的詳細資訊，請參閱 AWS Organizations 使用者指南中的 SCP 運作方式。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合使用者的暫時工作階段時，作為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參閱 IAM 使用者指南中的工作階段政策。

### 多種政策類型

將多種政策類型套用到請求時，其結果形成的許可會更為複雜、更加難以理解。要了解如何在涉及多個政策類型時 AWS 確定是否允許請求，請參閱《IAM 使用者指南》中的政策評估邏輯。

## Amazon Managed Service for Prometheus 如何與 IAM 一併使用

在您使用 IAM 管理 Amazon Managed Service for Prometheus 的存取權之前，瞭解哪些 IAM 功能可以與 Amazon Managed Service for Prometheus 搭配使用。

您可以搭配 Amazon Managed Service for Prometheus 使用的 IAM 功能

| IAM 功能 | Amazon Managed Service for Prometheus 支援 |
| --- | --- |
| 身分型政策 | 是 |
| 資源型政策 | 否 |
| 政策動作 | 是 |
| 政策資源 | 是 |
| 政策條件索引鍵 | 否 |
| ACL | 否 |
| ABAC (政策中的標籤) | 是 |
| 臨時憑證 | 是 |
| 轉送存取工作階段 (FAS) | 否 |
| 服務角色 | 否 |
| 服務連結角色 | 是 |

若要深入瞭解適用於 Prometheus 的 Amazon 受管服務和其他服 AWS 務如何與大多數 IAM 功能搭配使用，請參閱 IAM 使用者指南中的可與 IAM 搭配使用的AWS 服務。

## Amazon Managed Service for Prometheus 的身分型政策

| 支援身分型政策 | 是 |
| --- | --- |

身分型政策是可以連接到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分類型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。您無法在身分型政策中指定主體，因為這會套用至連接的使用者或角色。如要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素參考。

Amazon Managed Service for Prometheus 的身分型政策範例

若要檢視 Amazon Managed Service for Prometheus 以身分為基礎政策的範例，請參閱 Amazon Managed Service for Prometheus 的身分型政策範例。

## Amazon Managed Service for Prometheus 的資源型政策

| | |
|---|---|
| 支援以資源基礎的政策 | 否 |

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

如需啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，作為資源型政策的主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當主體和資源位於不同時 AWS 帳戶，受信任帳戶中的 IAM 管理員也必須授與主體實體 (使用者或角色) 權限，才能存取資源。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱 IAM 使用者指南中的 IAM 中的跨帳戶資源存取。

## Amazon Managed Service for Prometheus 政策動作

| | |
|---|---|
| 支援政策動作 | 是 |

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON 政策的 `Action` 元素描述您可以用來允許或拒絕政策中存取的動作。原則動作通常與關聯的 AWS API 作業具有相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

若要查看 Amazon Managed Service for Prometheus 動作的清單，請參閱《服務授權參考》中的 Amazon Managed Service in Prometheus 定義的動作。

Amazon Managed Service for Prometheus 中的政策動作會在動作之前使用下列前置字元：

```
aps
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [
      "aps:action1",
      "aps:action2"
         ]
```

若要檢視 Amazon Managed Service for Prometheus 以身分為基礎政策的範例，請參閱 Amazon Managed Service for Prometheus 的身分型政策範例。

## Amazon Managed Service for Prometheus 的政策資源

| | |
|---|---|
| 支援政策資源 | 是 |

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

`Resource` JSON 政策元素可指定要套用動作的物件。陳述式必須包含 `Resource` 或 `NotResource` 元素。最佳實務是使用其 Amazon Resource Name (ARN) 來指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (*) 來表示陳述式適用於所有資源。

```
"Resource": "*"
```

若要查看 Amazon Managed Service for Prometheus 資源類型及其 ARN 的清單，請參閱《服務授權參考》中的 Amazon Managed Service for Prometheus 定義的資源。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 Amazon Managed Service for Prometheus 定義的動作。

若要檢視 Amazon Managed Service for Prometheus 以身分為基礎政策的範例，請參閱 Amazon Managed Service for Prometheus 的身分型政策範例。

## Amazon Managed Service for Prometheus 的政策條件索引鍵

| | |
|---|---|
| 支援服務特定政策條件金鑰 | 否 |

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 (或 Condition 區塊) 可讓您指定使陳述式生效的條件。Condition 元素是選用項目。您可以建立使用條件運算子的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵， AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值，請使用邏輯OR運算來 AWS 評估條件。必須符合所有條件，才會授與陳述式的許可。

您也可以在指定條件時使用預留位置變數。例如，您可以只在使用者使用其 IAM 使用者名稱標記時，將存取資源的許可授予該 IAM 使用者。如需更多資訊，請參閱 IAM 使用者指南中的 IAM 政策元素：變數和標籤。

AWS 支援全域條件金鑰和服務特定條件金鑰。若要查看所有 AWS 全域條件金鑰，請參閱《IAM 使用者指南》中的AWS 全域條件內容金鑰。

若要查看 Amazon Managed Service for Prometheus 條件索引鍵的清單，請參閱《服務授權參考》中的 Amazon Managed Service in Prometheus 的條件索引鍵。若要了解您可以搭配哪些動作和資源使用條件索引鍵，請參閱 Amazon Managed Service for Prometheus 定義的動作。

若要檢視 Amazon Managed Service for Prometheus 以身分為基礎政策的範例，請參閱 Amazon Managed Service for Prometheus 的身分型政策範例。

## Amazon Managed Service for Prometheus 的存取控制清單 (ACL)

| | |
|---|---|
| 支援 ACL | 否 |

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

## 搭配 Amazon Managed Service for Prometheus 的屬性型存取控制 (ABAC)

| | |
|---|---|
| 支援 ABAC (政策中的標籤) | 是 |

屬性型存取控制 (ABAC) 是一種授權策略，可根據屬性來定義許可。在中 AWS，這些屬性稱為標籤。您可以將標籤附加到 IAM 實體 (使用者或角色) 和許多 AWS 資源。為實體和資源加上標籤是 ABAC 的第一步。您接著要設計 ABAC 政策，允許在主體的標籤與其嘗試存取的資源標籤相符時操作。

ABAC 在成長快速的環境中相當有幫助，並能在政策管理變得繁瑣時提供協助。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的條件元素中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需 ABAC 的詳細資訊，請參閱 IAM 使用者指南中的什麼是 ABAC?。如要查看含有設定 ABAC 步驟的教學課程，請參閱《IAM 使用者指南》中的使用屬性型存取控制 (ABAC)。

## 搭配 Amazon Managed Service for Prometheus 使用暫時憑證

| | |
|---|---|
| 支援臨時憑證 | 是 |

當您使用臨時憑據登錄時，某些 AWS 服務 不起作用。如需其他資訊，包括哪些 AWS 服務 與臨時登入資料搭配AWS 服務 使用，請參閱 IAM 使用者指南中的 IAM。

如果您使用除了使用者名稱和密碼以外的任何方法登入，則您正在 AWS Management Console 使用臨時認證。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取時，該程序會自動建立暫時認證。當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需切換角色的詳細資訊，請參閱 IAM 使用者指南中的切換至角色 (主控台)。

您可以使用 AWS CLI 或 AWS API 手動建立臨時登入資料。然後，您可以使用這些臨時登入資料來存取 AWS。 AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱 IAM 中的暫時性安全憑證。

## Amazon Managed Service for Prometheus 的轉送存取工作階段

| | |
|---|---|
| 支援轉寄存取工作階段 (FAS) | 否 |

當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱轉發存取工作階段。

### Amazon Managed Service for Prometheus 的服務角色

| | |
|---|---|
| 支援服務角色 | 否 |

服務角色是服務擔任的 IAM 角色，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱 IAM 使用者指南中的建立角色以委派許可給 AWS 服務服務。

> **Warning**
>
> 變更服務角色的權限可能會中斷 Amazon Managed Service for Prometheus 功能。只有在 Amazon Managed Service for Prometheus 提供指示時，才能編輯服務角色。

### Amazon Managed Service for Prometheus 連結角色

| | |
|---|---|
| 支援服務連結角色 | 是 |

服務連結角色是一種連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理 Amazon Managed Service for Prometheus 服務連結角色的詳細資訊，請參閱 使用 Amazon Managed Service for Prometheus 的服務連結角色。

## Amazon Managed Service for Prometheus 的身分型政策範例

依預設，使用者和角色不具備建立或修改 Amazon Managed Service for Prometheus 資源的許可。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 來執行工作。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

若要了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策。

如需有關 Amazon Managed Service for Prometheus 所定義動作和資源類型的資訊，包括每種資源類型的 ARN 格式，請參閱《服務授權參考》中的 Amazon Managed Service for Prometheus 的動作、資源和條件索引鍵。

主題

- 政策最佳實務
- 使用 Amazon Managed Service for Prometheus 主控台
- 允許使用者檢視他們自己的許可

## 政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 Amazon Managed Service for Prometheus 資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管原則並邁向最低權限權限 — 若要開始授與使用者和工作負載的權限，請使用可授與許多常見使用案例權限的AWS 受管理原則。它們可用在您的 AWS 帳戶. 建議您透過定義特定於您使用案例的 AWS 客戶管理政策，進一步降低使用權限。如需更多資訊，請參閱 IAM 使用者指南中的 AWS 受管政策或任務職能的AWS 受管政策。
- 套用最低權限許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的許可。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需使用 IAM 套用許可的更多相關資訊，請參閱 IAM 使用者指南中的 IAM 中的政策和許可。
- 使用 IAM 政策中的條件進一步限制存取權 – 您可以將條件新增至政策，以限制動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。您也可以使用條件來授與對服務動作的存取權 (如透過特定) 使用這些動作 AWS 服務，例如 AWS CloudFormation。如需詳細資訊，請參閱 IAM 使用者指南中的 IAM JSON 政策元素：條件。
- 使用 IAM Access Analyzer 驗證 IAM 政策，確保許可安全且可正常運作 – IAM Access Analyzer 驗證新政策和現有政策，確保這些政策遵從 IAM 政策語言 (JSON) 和 IAM 最佳實務。IAM Access Analyzer 提供 100 多項政策檢查及切實可行的建議，可協助您編寫安全且實用的政策。如需更多資訊，請參閱 IAM 使用者指南中的 IAM Access Analyzer 政策驗證。
- 需要多因素身份驗證 (MFA) — 如果您的案例需要 IAM 使用者或根使用者 AWS 帳戶，請開啟 MFA 以獲得額外的安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需更多資訊，請參閱 IAM 使用者指南中的設定 MFA 保護的 API 存取。

如需 IAM 中最佳實務的相關資訊，請參閱 IAM 使用者指南中的 IAM 安全最佳實務。

## 使用 Amazon Managed Service for Prometheus 主控台

若要存取 Amazon Managed Service for Prometheus 主控台，您必須擁有最基本的一組許可。這些許可必須允許您列出和檢視您 AWS 帳戶中 Amazon Managed Service for Prometheus 資源的詳細資料。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (使用者或角色) 而言，主控台就無法如預期運作。

您不需要為僅對 AWS CLI 或 AWS API 進行呼叫的使用者允許最低主控台權限。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。

為了確保使用者和角色仍然可以使用適用於 Prometheus 的 Amazon 受管服務主控台，請將 Prometheus 的 Amazon 受管服務`ConsoleAccess`或`ReadOnly` AWS 受管政策附加到實體。如需詳細資訊，請參閱《IAM 使用者指南》中的新增許可到使用者。

## 允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此原則包含在主控台上或以程式設計方式使用 AWS CLI 或 AWS API 完成此動作的權限。

```
{
    "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": "*"
        }
    ]
}
```

# AWS 適用於 Prometheus 的 Amazon 受管服務的受管政策

受 AWS 管理的策略是由建立和管理的獨立策略 AWS。 AWS 受管理的策略旨在為許多常見使用案例提供權限，以便您可以開始將權限指派給使用者、群組和角色。

請記住， AWS 受管理的政策可能不會為您的特定使用案例授與最低權限權限，因為這些權限可供所有 AWS 客戶使用。我們建議您定義使用案例專屬的客戶管理政策，以便進一步減少許可。

您無法變更受 AWS 管理策略中定義的權限。如果 AWS 更新 AWS 受管理原則中定義的權限，則此更新會影響附加原則的所有主體識別 (使用者、群組和角色)。 AWS 當新的啟動或新 AWS 服務 的 API 操作可用於現有服務時，最有可能更新 AWS 受管理策略。

如需詳細資訊，請參閱《IAM 使用者指南》中的 AWS 受管政策。

## AmazonPrometheusFullAccess

您可將 `AmazonPrometheusFullAccess` 政策連接到 IAM 身分。

許可詳細資訊

此政策包含以下許可。

- `aps`：允許完全存取 Amazon Managed Service for Prometheus
- `eks`：讓 Amazon Managed Service for Prometheus 服務讀取有關 Amazon EKS 叢集的資訊。若要允許在叢集中建立受管湊集器並探索指標，則這會相當必要。
- `ec2`：允許 Amazon Managed Service for Prometheus 服務讀取有關您的 Amazon EC2 網路的資訊。若要使用 Amazon EKS 指標的存取權建立受管湊集器，則這會相當必要。

- iam：允許主體為受管理的指標湊集器建立服務連結角色。

的內容AmazonPrometheusFullAccess如下：

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AllPrometheusActions",
   "Effect": "Allow",
   "Action": [
    "aps:*"
   ],
   "Resource": "*"
  },
  {
   "Sid": "DescribeCluster",
   "Effect": "Allow",
   "Action": [
    "eks:DescribeCluster",
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups"
   ],
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "aps.amazonaws.com"
     ]
    }
   },
   "Resource": "*"
  },
  {
   "Sid": "CreateServiceLinkedRole",
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "arn:aws:iam::*:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": "scraper.aps.amazonaws.com"
    }
   }
```

```
  }
 ]
}
```

## AmazonPrometheusConsoleFullAccess

您可將 AmazonPrometheusConsoleFullAccess 政策連接到 IAM 身分。

**許可詳細資訊**

此政策包含以下許可。

- aps：允許完全存取 Amazon Managed Service for Prometheus
- tag：允許主體在 Amazon Managed Service for Prometheus 主控台中查看標籤建議。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "TagSuggestions",
   "Effect": "Allow",
   "Action": [
    "tag:GetTagValues",
    "tag:GetTagKeys"
   ],
   "Resource": "*"
  },
  {
   "Sid": "PrometheusConsoleActions",
   "Effect": "Allow",
   "Action": [
    "aps:CreateWorkspace",
    "aps:DescribeWorkspace",
    "aps:UpdateWorkspaceAlias",
    "aps:DeleteWorkspace",
    "aps:ListWorkspaces",
    "aps:DescribeAlertManagerDefinition",
    "aps:DescribeRuleGroupsNamespace",
    "aps:CreateAlertManagerDefinition",
    "aps:CreateRuleGroupsNamespace",
    "aps:DeleteAlertManagerDefinition",
    "aps:DeleteRuleGroupsNamespace",
```

```
    "aps:ListRuleGroupsNamespaces",
    "aps:PutAlertManagerDefinition",
    "aps:PutRuleGroupsNamespace",
    "aps:TagResource",
    "aps:UntagResource",
    "aps:CreateLoggingConfiguration",
    "aps:UpdateLoggingConfiguration",
    "aps:DeleteLoggingConfiguration",
    "aps:DescribeLoggingConfiguration"
   ],
   "Resource": "*"
  }
 ]
}
```

## AmazonPrometheusRemoteWriteAccess

的內容AmazonPrometheusRemoteWriteAccess如下：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "aps:RemoteWrite"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

## AmazonPrometheusQueryAccess

的內容AmazonPrometheusQueryAccess如下：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "aps:GetLabels",
                "aps:GetMetricMetadata",
```

```
                "aps:GetSeries",
                "aps:QueryMetrics"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

## AWS 受管理的策略： AmazonPrometheusScraperServiceRolePolicy

您無法附加 AmazonPrometheusScraperServiceRolePolicy 到 IAM 實體。此政策會附加至服務連結角色，可讓 Amazon Managed Service for Prometheus 代表您執行動作。如需詳細資訊，請參閱 使用角色從 EKS 湊集指標。

此政策授予參與者許可，允許從 Amazon EKS 叢集讀取並寫入 Amazon Managed Service for Prometheus 工作區。

> **Note**
>
> 此使用者指南先前錯誤地稱為此政策 `AmazonPrometheusScraperServiceLinkedRolePolicy`

### 許可詳細資訊

此政策包含以下許可。

- `aps`：讓服務主體將指標寫入 Amazon Managed Service for Prometheus 工作區。
- `ec2`：讓服務主體讀取和修改網路組態，以連接到包含 Amazon EKS 叢集的網路。
- `eks`：讓服務主體存取您的 Amazon EKS 叢集。此為必要項目，以讓其可自動湊集指標。還允許主體在刪除刮板時清理 Amazon EKS 資源。

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

```
    "iam:DeleteRole"
   ],
   "Resource": "arn:aws:iam::*:role/aws-service-role/scraper.aps.amazonaws.com/
AWSServiceRoleForAmazonPrometheusScraper*"
  },
  {
   "Sid": "NetworkDiscovery",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ENIManagement",
   "Effect": "Allow",
   "Action": "ec2:CreateNetworkInterface",
   "Resource": "*",
   "Condition": {
    "ForAllValues:StringEquals": {
     "aws:TagKeys": [
      "AMPAgentlessScraper"
     ]
    }
   }
  },
  {
   "Sid": "TagManagement",
   "Effect": "Allow",
   "Action": "ec2:CreateTags",
   "Resource": "arn:aws:ec2:*:*:network-interface/*",
   "Condition": {
    "StringEquals": {
     "ec2:CreateAction": "CreateNetworkInterface"
    },
    "Null": {
     "aws:RequestTag/AMPAgentlessScraper": "false"
    }
   }
  },
  {
   "Sid": "ENIUpdating",
```

```
    "Effect": "Allow",
    "Action": [
     "ec2:DeleteNetworkInterface",
     "ec2:ModifyNetworkInterfaceAttribute"
    ],
    "Resource": "*",
    "Condition": {
     "Null": {
      "ec2:ResourceTag/AMPAgentlessScraper": "false"
     }
    }
   },
   {
    "Sid": "EKSAccess",
    "Effect": "Allow",
    "Action": "eks:DescribeCluster",
    "Resource": "arn:aws:eks:*:*:cluster/*"
   },
   {
    "Sid": "DeleteEKSAccessEntry",
    "Effect": "Allow",
    "Action": "eks:DeleteAccessEntry",
    "Resource": "arn:aws:eks:*:*:access-entry/*/role/*",
    "Condition": {
     "StringEquals": {
      "aws:PrincipalAccount": "${aws:ResourceAccount}"
     },
     "ArnLike": {
      "eks:principalArn": "arn:aws:iam::*:role/aws-service-role/
scraper.aps.amazonaws.com/AWSServiceRoleForAmazonPrometheusScraper*"
     }
    }
   },
   {
    "Sid": "APSWriting",
    "Effect": "Allow",
    "Action": "aps:RemoteWrite",
    "Resource": "arn:aws:aps:*:*:workspace/*",
    "Condition": {
     "StringEquals": {
      "aws:PrincipalAccount": "${aws:ResourceAccount}"
     }
    }
   }
```

```
 ]
}
```

## 適用於 Prometheus 的 Amazon 受管服務更新受管政策 AWS

檢視此服務開始追蹤這些變更後，適用於 Prometheus 的 Amazon 受管服務的受管政策更新詳細資訊。 AWS 如需有關此頁面變更的自動提示，請訂閱 Amazon Managed Service for Prometheus 文件歷史記錄頁面上的 RSS 摘要。

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| AmazonPrometheusScraperServiceRolePolicy – 更新現有政策 | 適用於 Prometheus 的 Amazon 受管服務增加了新的許可，以AmazonPrometheusScraperServiceRolePolicy支持使用 Amazon EKS 中的訪問條目。<br><br>包括管理 Amazon EKS 存取項目的許可，以便在刪除抓取工具時清理資源。<br><br>**Note**<br>使用者指南先前錯誤地稱為此政策 `AmazonPrometheusScraperServiceLinkedRolePolicy` | 2024年5月2日 |
| AmazonPrometheusFullAccess – 更新現有政策 | Amazon Managed Service for Prometheus 已新增 AmazonPrometheusFullAccess 許可，以支援在 | 2023 年 11 月 26 日 |

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| | Amazon EKS 叢集中為指標建立受管湊集器。<br><br>包含連線至 Amazon EKS 叢集、讀取 Amazon EC2 網路以及為湊集器建立服務連結角色的許可。 | |
| AmazonPrometheusScraperServiceLinkedRolePolicy – 新政策 | Amazon Managed Service for Prometheus 已新增服務連結角色政策，以自 Amazon EKS 容器讀取來允許自動湊集指標。<br><br>包括連線至 Amazon EKS 叢集、讀取 Amazon EC2 網路、建立和刪除標記為 `AMPAgentlessScraper` 的網路以及寫入 Amazon Managed Service for Prometheus 工作區的許可。 | 2023 年 11 月 26 日 |

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| AmazonPrometheusConsoleFullAccess – 更新現有政策 | 適用於 Prometheus 的 Amazon 受管服務新增許可，以支援日誌中AmazonPrometheusConsoleFullAccess的記錄警示管理員和統治者事件。 CloudWatch<br><br>已新增 `aps:CreateLoggingConfiguration` 、`aps:UpdateLoggingConfiguration` 、`aps:DeleteLoggingConfiguration` 、`aps:DescribeLoggingConfiguration` 許可。 | 2022 年 10 月 24 日 |

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| AmazonPrometheusConsoleFullAccess – 更新現有政策 | Amazon Managed Service for Prometheus 已新增 AmazonPrometheusConsoleFullAccess 許可，以支援全新 Amazon Managed Service for Prometheus 功能，以便使用此政策的使用者可在將標籤套用至 Amazon Managed Service for Prometheus 資源時，看到標籤建議清單。<br><br>已新增 `tag:GetTagKeys` 、`tag:GetTagValues` 、`aps:CreateAlertManagerDefinition` 、`aps:CreateRuleGroupsNamespace` 、`aps:DeleteAlertManagerDefinition` 、`aps:DeleteRuleGroupsNamespace` 、`aps:DescribeAlertManagerDefinition` 、`aps:DescribeRuleGroupsNamespace` 、`aps:ListRuleGroupsNamespaces` 、`aps:PutAlertManagerDefinition` 、`aps:PutRuleGroupsNamespace` 、`aps:TagRe | 2021 年 9 月 29 日 |

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| | source 和 aps:UntagResource 許可。 | |
| Amazon Managed Service for Prometheus 已開始追蹤變更 | 適用於 Prometheus 的 Amazon 受管服務開始追蹤其 AWS 受管政策的變更。 | 2021 年 9 月 15 日 |

# Amazon Managed Service for Prometheus 身分和存取的疑難排解

請使用以下資訊來協助您診斷和修正使用 Amazon Managed Service for Prometheus 和 IAM 時可能遇到的常見問題。

主題

- 我沒有在 Amazon Managed Service for Prometheus 中執行動作的權限。
- 我沒有授權執行 iam：PassRole
- 我想允許我的 AWS 帳戶以外的人訪問我的 Amazon Prometheus 資源託管服務

## 我沒有在 Amazon Managed Service for Prometheus 中執行動作的權限。

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

下列範例錯誤會在mateojackson IAM 使用者嘗試使用主控台檢視一個虛構 *my-example-widget* 資源的詳細資訊，但卻無虛構 aps:*GetWidget* 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 aps:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 mateojackson 使用者的政策，允許使用 aps:*GetWidget* 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

## 我沒有授權執行 iam：PassRole

如果您收到沒有執行 iam:PassRole 動作權限的錯誤，則必須更新政策以讓您將角色傳送給 Amazon Managed Service for Prometheus。

有些 AWS 服務 允許您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為 marymajor 的 IAM 使用者嘗試在 Amazon Managed Service for Prometheus 中執行動作時，發生下列範例錯誤。但是，動作請求服務具備服務角色授予的許可。Mary 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 iam:PassRole 動作。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

### 我想允許我的 AWS 帳戶以外的人訪問我的 Amazon Prometheus 資源託管服務

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解 Amazon Managed Service for Prometheus 是否支援這些功能，請參閱 Amazon Managed Service for Prometheus 如何與 IAM 一併使用。
- 若要了解如何提供對您所擁有資源 AWS 帳戶 的存取權，請參閱《IAM 使用者指南》中您擁有的另一 AWS 帳戶 個 IAM 使用者提供存取權限。
- 若要了解如何向第三方提供對資源的存取權 AWS 帳戶，請參閱 IAM 使用者指南中的提供第三方 AWS 帳戶 擁有的存取權。
- 如需了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的將存取權提供給在外部進行身分驗證的使用者 (聯合身分)。
- 若要了解跨帳戶存取使用角色和以資源為基礎的政策之間的差異，請參閱 IAM 使用者指南中的 IAM 中的跨帳戶資源存取。

# IAM 許可和政策

存取 Amazon Managed Service for Prometheus 動作和資料需要憑證。這些憑證必須擁有執行動作和存取 AWS 資源的許可，例如擷取有關雲端資源的 Amazon Managed Service for Prometheus data 資料。以下章節提供有關如何使用 AWS Identity and Access Management (IAM) 與 Amazon Managed

Service for Prometheus 的詳細資訊，藉由控制可存取的人員，協助確保您資源的安全。如需詳細資訊，請參閱 IAM 中的政策和許可。

## Amazon Managed Service for Prometheus 許可

下表顯示 Amazon Managed Service for Prometheus 動作及其所需的許可。這些動作可能還需要其他服務的許可，沒有詳細說明。

| 動作 | 所需的許可 |
| --- | --- |
| 建立警示。 | `aps:CreateAlertManagerAlerts` |
| 在工作區中建立警示管理員定義。如需更多詳細資訊，請參閱 警示管理員。 | `aps:CreateAlertManagerDefinition` |
| 在工作區中建立規則群組命名空間。如需更多詳細資訊，請參閱 記錄規則和警示規則。 | `aps:CreateRuleGroupsNamespace` |
| 建立 Amazon Managed Service for Prometheus 工作區。工作區是專用於儲存和查詢 Prometheus 指標的邏輯空間。 | `aps:CreateWorkspace` |
| 刪除工作區的警示管理員定義。 | `aps:DeleteAlertManagerDefinition` |
| 刪除警示靜音。 | `aps:DeleteAlertManagerSilence` |
| 刪除 Amazon Managed Service for Prometheus 工作區。 | `aps:DeleteWorkspace` |
| 擷取有關警示管理員定義的詳細資訊。 | `aps:DescribeAlertManagerDefinition` |
| 擷取有關規則群組命名空間的詳細資訊。 | `aps:DescribeRuleGroupsNamespace` |
| 擷取有關 Amazon Managed Service for Prometheus 工作區的詳細資訊。 | `aps:DescribeWorkspace` |
| 擷取警示靜音的詳細資訊。 | `aps:GetAlertManagerSilence` |
| 擷取工作區中警示管理員的狀態。 | `aps:GetAlertManagerStatus` |

| 動作 | 所需的許可 |
| --- | --- |
| 擷取標籤。 | aps:GetLabels |
| 擷取 Amazon Managed Service for Prometheus 指標的中繼資料。 | aps:GetMetricMetadata |
| 擷取時間序列資料。 | aps:GetSeries |
| 擷取警示管理員定義中定義的警示群組清單。 | aps:ListAlertManagerAlertGroups |
| 擷取警示管理員中定義的警示清單。 | aps:ListAlertManagerAlerts |
| 擷取已在警示管理員定義中定義的接收者清單。 | aps:ListAlertManagerReceivers |
| 擷取已定義警示靜音的清單。 | aps:ListAlertManagerSilences |
| 擷取啟用中警示的清單。 | aps:ListAlerts |
| 擷取工作區中規則群組命名空間的規則清單。 | aps:ListRules |
| 擷取工作區中規則群組命名空間的清單。 | aps:ListRuleGroupsNamespaces |
| 擷取與 Amazon Managed Service for Prometheus 資源相關聯的標籤。 | aps:ListTagsForResource |
| 擷取帳戶中存在的 Amazon Managed Service for Prometheus 工作區清單。 | aps:ListWorkspaces |
| 更新工作區中現有的警示管理員定義。 | aps:PutAlertManagerDefinition |
| 建立警示靜音。 | aps:PutAlertManagerSilences |
| 更新現有規則群組命名空間。 | aps:PutRuleGroupsNamespace |
| 在 Amazon Managed Service for Prometheus 指標執行查詢。 | aps:QueryMetrics |

| 動作 | 所需的許可 |
| --- | --- |
| 執行遠端寫入作業，以啟動 Prometheus 伺服器至 Amazon Managed Service for Prometheus 的指標串流。 | `aps:RemoteWrite` |
| 將標籤指派給 Amazon Managed Service in Prometheus 資源。 | `aps:TagResource` |
| 移除 Amazon Managed Service for Prometheus 資源的標籤。 | `aps:UntagResource` |
| 修改現有工作區的別名。 | `aps:UpdateWorkspaceAlias` |
| 建立記錄組態。 | `aps:CreateLoggingConfiguration` |
| 刪除記錄組態。 | `aps:DeleteLoggingConfiguration` |
| 說明工作區記錄組態。 | `aps:DescribeLoggingConfiguration` |
| 更新記錄組態。 | `aps:UpdateLoggingConfiguration` |

## 範例 IAM 政策

本節提供可以建立的其他自我管理政策範例。

下列 IAM 政策授予 Amazon Managed Service for Prometheus 的完整存取權，也可讓使用者探索 Amazon EKS 叢集並查看叢集相關詳細資訊。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aps:*",
                "eks:DescribeCluster",
                "eks:ListClusters"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

# Amazon Managed Service for Prometheus 的合規驗證

若要瞭解 AWS 服務 是否屬於特定規範遵循方案的範圍內，請參閱AWS 服務 遵循規範計劃方案中的，並選擇您感興趣的合規方案。如需一般資訊，請參閱AWS 規範計劃AWS。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱下載中的報告中的 AWS Artifact。

您在使用時的合規責任取決 AWS 服務 於您資料的敏感性、公司的合規目標以及適用的法律和法規。AWS 提供下列資源以協助遵循法規：

- 安全性與合規性快速入門指南 — 這些部署指南討論架構考量，並提供部署以安全性和合規性 AWS 為重點的基準環境的步驟。
- 在 Amazon Web Services 上架構 HIPAA 安全性與合規性 — 本白皮書說明公司如何使用建立符合 HIPAA 資格的應 AWS 用程式。

  > **Note**
  >
  > 並非所有人 AWS 服務 都符合 HIPAA 資格。如需詳細資訊，請參閱 HIPAA 資格服務參照。

- AWS 合規資源AWS — 此工作簿和指南集合可能適用於您的產業和所在地。
- AWS 客戶合規指南 — 透過合規的角度瞭解共同的責任模式。這份指南總結了在多個架構 (包括美國國家標準技術研究所 (NIST)、支付卡產業安全標準委員會 (PCI) 和國際標準化組織 (ISO)) 中，保 AWS 服務 護指引並對應至安全控制的最佳實務。
- 使用AWS Config 開發人員指南中的規則評估資源 — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub— 這 AWS 服務 提供了內部安全狀態的全面視圖 AWS。Security Hub 使用安全控制，可評估您的 AWS 資源並檢查您的法規遵循是否符合安全業界標準和最佳實務。如需支援的服務和控制清單，請參閱 Security Hub controls reference。
- Amazon GuardDuty — 透過監控環境中的 AWS 帳戶可疑和惡意活動，藉此 AWS 服務 偵測您的工作負載、容器和資料的潛在威脅。 GuardDuty 可協助您滿足特定合規性架構所要求的入侵偵測需求，例如 PCI DSS 等各種合規性需求。
- AWS Audit Manager— 這 AWS 服務 有助於您持續稽核您的 AWS 使用情況，以簡化您管理風險的方式，以及遵守法規和業界標準的方式。

# Amazon Managed Service for Prometheus 中的復原功能

AWS 全球基礎設施是以 AWS 區域與可用區域為中心建置的。AWS區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援聯網功能相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和可擴展性能力，均較單一或多個資料中心的傳統基礎設施還高。

如需有關 AWS 區域與可用區域的更多相關資訊，請參閱 AWS 全球基礎設施。

除了 AWS 全球基礎設施外，Amazon Managed Service for Prometheus 還提供數種功能來支援資料恢復能力和備份需求，包括支援高可用性資料。

# Amazon Managed Service for Prometheus 中的基礎設施安全性

作為受管服務，Amazon Managed Service for Prometheus 受到 AWS 全球網路安全的保護。如需有關 AWS 安全服務以及 AWS 如何保護基礎設施的詳細資訊，請參閱 AWS 雲端安全。若要使用基礎設施安全性的最佳實務來設計您的 AWS 環境，請參閱安全支柱 AWS 架構良好的框架中的基礎設施保護。

您可使用 AWS 發佈的 API 呼叫，透過網路存取 Amazon Managed Service for Prometheus。用戶端必須支援下列項目：

- Transport Layer Security (TLS)。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 具備完美轉送私密(PFS)的密碼套件，例如 DHE (Ephemeral Diffie-Hellman)或 ECDHE (Elliptic Curve Ephemeral Diffie-Hellman)。現代系統(如 Java 7 和更新版本)大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與 IAM 主體相關聯的私密存取金鑰來簽署。或者，您可以使用 AWS Security Token Service (AWS STS) 以產生暫時安全憑證以簽署請求。

# 使用 Amazon Managed Service for Prometheus 的服務連結角色

適用於 Prometheus 的 Amazon 受管服務使用 AWS Identity and Access Management (IAM) 服務連結角色。服務連結角色是一種專屬的 IAM 角色類型，可直接連結到 Amazon Managed Service for Prometheus。服務連結角色由 Amazon Managed Service for Prometheus 預先定義，並包含該服務需要代表您呼叫其他 AWS 服務的所有許可。

由於服務連結角色可更輕鬆設定 Amazon Managed Service for Prometheus，因此您不必手動新增必要的許可。Amazon Managed Service for Prometheus 定義其服務連結角色的許可，除非另有定義，

否則僅 Amazon Managed Service for Prometheus 可以擔任其角色。定義的許可包括信任政策和許可政策，且該許可政策無法附加至其他 IAM 實體。

## 使用角色從 EKS 湊集指標

使用適用於 Prometheus 受管收集器的 Amazon Managed Service 自動抓取指標時，AWSServiceRoleForAmazonPrometheusScraper服務連結角色可讓您更輕鬆地設定受管收集器，因為您不必手動新增必要的許可。Amazon Managed Service Prometheus 定義許可，且只有 Amazon Managed Service for Prometheus 可擔任角色。

如需關於支援服務連結角色的其他服務資訊，請參閱《可搭配 IAM 運作的AWS 服務》，尋找服務連結角色欄中顯示為是的服務。選擇具有連結的是，以檢視該服務的服務連結角色文件。

### Amazon Managed Service for Prometheus 的服務連結角色許可

適用於 Prometheus 的 Amazon 受管服務使用以前綴AWSServiceRoleForAmazonPrometheusScraper命名的服務連結角色，允許 Prometheus 的 Amazon 受管服務自動抓取 Amazon EKS 叢集中的指標。

服 AWSServiceRoleForAmazonPrometheusScraper 務連結角色會信任下列服務擔任該角色：

- `scraper.aps.amazonaws.com`

名為的角色許可政策AmazonPrometheusScraperServiceRolePolicy允許 Prometheus 的 Amazon 受管服務對指定的資源完成下列動作：

- 準備好並修改網路組態，以連接到包含 Amazon EKS 叢集的網路。
- 從 Amazon EKS 叢集讀取指標，並將指標寫入 Amazon Managed Service for Prometheus 工作區。

您必須設定許可，讓使用者、群組或角色建立服務連結角色。如需詳細資訊，請參閱 IAM 使用者指南中的服務連結角色許可。

### 建立 Amazon Managed Service for Prometheus 的服務連結角色

您不需要手動建立一個服務連結角色。當您在、或 AWS API 中使用適用於 Prometheus 的 Amazon EKS 或 Amazon 受管服務建立受管收集器執行個體時 AWS Management Console，適用於 Prometheus 的 Amazon 受管服務會為您建立服務連結角色。 AWS CLI

> **Important**
>
> 此服務連結角色可以顯示在您的帳戶，如果您於其他服務中完成一項動作時，可以使用支援此角色的功能。要了解更多信息，請參閱我的一個新角色出現 AWS 帳戶。

若您刪除此服務連結角色然後需要再次建立，便可在帳戶中使用相同程序重新建立角色。當您使用 Amazon EKS 或 Amazon Managed Service for Prometheus 建立受管收集器執行個體時，Amazon Managed Service for Prometheus 會再次為您建立服務連結角色。

### 正在編輯 Amazon Managed Service for Prometheus 的服務連結角色

適用於 Prometheus 的 Amazon 受管服務不允許您編輯服務連結 AWSServiceRoleForAmazonPrometheusScraper 角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。然而，您可使用 IAM 來編輯角色描述。如需詳細資訊，請參閱 IAM 使用者指南中的編輯服務連結角色。

### 正在刪除 Amazon Managed Service for Prometheus 的服務連結角色

您不需要手動刪除 AWSServiceRoleForAmazonPrometheusScraper 角色。當您刪除與 AWS Management Console、或 AWS API 中的角色相關聯的所有受管收集器執行個體時，Prometheus 的 Amazon 受管服務會清除資源並為您刪除服務連結角色。 AWS CLI

### Amazon Managed Service for Prometheus 服務連結角色支援的地區

Amazon Managed Service for Prometheus 會在提供服務的所有地區中，支援使用服務連結角色。如需詳細資訊，請參閱 支援地區。

## 使用 AWS CloudTrail 記錄 Amazon Managed Service for Prometheus API 呼叫

適用於 Prometheus 的 Amazon 受管服務與服務整合在一起 AWS CloudTrail，該服務可提供使用者在 Prometheus 的 Amazon 受管 AWS 服務中所採取的動作、角色或服務記錄。 CloudTrail 擷取適用於 Prometheus 之 Amazon 受管服務的所有 API 呼叫作為事件。擷取的呼叫包括從 Amazon Managed Service for Prometheus 主控台的呼叫，以及來自 Amazon Managed Service for Prometheus API 作業的程式碼呼叫。如果您建立追蹤，您可以啟用持續交付 CloudTrail 事件到 Amazon S3 儲存貯體，包括適用於 Prometheus 的 Amazon 受管服務的事件。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。使用收集的資訊 CloudTrail，您可以判斷針對 Prometheus 向

Amazon 受管服務提出的請求、提出請求的 IP 位址、提出請求的人員、提出要求的時間以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱使AWS CloudTrail 用者指南。

## Amazon Prometheus 託管服務信息 CloudTrail

CloudTrail 在您創建 AWS 帳戶時，您的帳戶已啟用。當活動在 Prometheus 的 Amazon 受管服務中發生時，該活動會與事件歷史記錄中的其他 AWS 服務 CloudTrail 事件一起記錄在事件中。您可以在帳戶中查看，搜索和下載最近的事 AWS 件。如需詳細資訊，請參閱使用 CloudTrail 事件歷程記錄檢視事件。

如需 AWS 帳戶中持續的事件記錄 (包括 Prometheus 的 Amazon 受管服務的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。根據預設，當您在主控台中建立追蹤時，追蹤會套用至所有 AWS 區域。追蹤記錄來自 AWS 分區中所有區域的事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄檔中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 從多個區域接收 CloudTrail 日誌文件並從多個帳戶接收 CloudTrail 日誌文件

Amazon Managed Service for Prometheus 支援記錄下列動作：

- CreateAlertManagerAlerts
- CreateAlertManagerDefinition
- CreateRuleGroupsNamespace
- CreateWorkspace
- DeleteAlertManagerDefinition
- DeleteAlertManagerSilence
- DeleteWorkspace
- DeleteRuleGroupsNamespace
- DescribeAlertManagerDefinition
- DescribeRulesGroupsNamespace
- DescribeWorkspace

- ListRuleGroupsNamespaces
- ListWorkspaces
- PutAlertManagerDefinition
- PutAlertManagerSilences
- PutRuleGroupsNamespace
- UpdateWorkspaceAlias

每一筆事件或日誌項目都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 要求是使用根使用者登入資料還是 AWS Identity and Access Management (IAM) 使用者登入資料提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱 CloudTrail userIdentity 元素。

## 了解 Amazon Managed Service for Prometheus 日誌檔案輸入項

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。 CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時間，請求參數等信息。 CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

範例： CreateWorkspace

下列範例顯示示範 CreateWorkspace動作的 CloudTrail 記錄項目。

```
{
    "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"
}
```

### 範例： CreateAlertManagerDefinition

下列範例顯示示範 CreateAlertManagerDefinition 動作的 CloudTrail 記錄項目。


```
{
    "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"
}
```

範例： CreateRuleGroupsNamespace

下列範例顯示示範 CreateRuleGroupsNamespace 動作的 CloudTrail 記錄項目。

```
{
    "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"
}
```


# 設定服務帳戶的 IAM 角色

透過服務帳戶的 IAM 角色，您可使用 Kubernetes 服務帳戶與 IAM 角色產生關聯。然後，此服務帳戶可以為使用該服務帳戶之任何 Pod 中的容器提供 AWS 許可。如需詳細資訊，請參閱服務帳戶的 IAM 角色。

服務帳戶的 IAM 角色也稱為服務角色。

在 Amazon Managed Service for Prometheus 中，使用服務角色可協助您取得在 Amazon Managed Service for Prometheus、Prometheus 伺服器和 Grafana 伺服器之間授權和驗證所需的角色。

先決條件

此頁面上的程序需要您已安裝 AWS CLI 和 EKSCTL 命令列介面。

## 自 Amazon EKS 叢集設定指標擷取作業的服務角色

若要在 Amazon EKS 叢集設定服務角色以讓 Amazon Managed Service for Prometheus 自 Prometheus 伺服器擷取指標，您必須登入具有下列許可的帳戶：

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

設定 Amazon Managed Service for Prometheus 擷取作業的服務角色

1. 建立名為 `createIRSA-AMPIngest.sh` 且具有下列內容的檔案。將 <my_amazon_eks_clustername> 替換為您的叢集名稱，並將 <my_prometheus_namespace> 替換為 Prometheus 命名空間。

```
#!/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. 請輸入下列命令，賦予指令碼必要權限。

```
chmod +x createIRSA-AMPIngest.sh
```

3. 執行指令碼。

## 設定服務帳戶的 IAM 角色，以查詢指標

若要設定服務帳戶的 IAM 角色 (服務角色) 以啟用從 Amazon Managed Service for Prometheus 工作區查詢指標，您必須登入具有下列許可的帳戶：

- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam:GetRole`
- `iam:AttachRolePolicy`
- `iam:GetOpenIDConnectProvider`

設定服務角色以查詢 Amazon Managed Service for Prometheus 指標；

1. 建立名為 `createIRSA-AMPQuery.sh` 且具有下列內容的檔案。將 <my_amazon_eks_clustername> 替換您的叢集名稱，並將 <my_prometheus_namespace> 替換為 Prometheus 命名空間。

```
#!/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. 請輸入下列命令，以提供指令碼取得必要的權限。

```
chmod +x createIRSA-AMPQuery.sh
```

3. 執行指令碼。

# 使用 Amazon Managed Service for Prometheus 和介面 VPC 端點

如果您使用 Amazon Virtual Private Cloud (Amazon VPC) 來託管您的 AWS 資源，可以在您的 VPC 與 Amazon Managed Service for Prometheus 之間建立私人連線。您可以使用這些連線來啟用 Amazon Managed Service for Prometheus 與 VPC 資源溝通而不經歷公有網際網路。

Amazon VPC 是一項 AWS 服務，您可用來在自己定義的虛擬網路中啟動 AWS 資源。您可利用 VPC 來控制您的網路設定，例如 IP 地址範圍、子網路、路由表和網路閘道。若要將 VPC 連接到 Amazon Managed Service for Prometheus，您會定義介面 VPC 端點以將 VPC 連接到 AWS 服務。端點提供 Amazon Managed Service for Prometheus 的可靠、可擴展連線，但不需要網路位址轉譯 (NAT) 執行個體或 VPN 連線。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的什麼是 Amazon VPC。

介面 VPC 端點是由 AWS PrivateLink 提供，AWS 技術可與私有 IP 位址搭配彈性網路介面使用，在兩個 AWS 服務之間啟用私有通訊。如需詳細資訊，請參閱最新 – AWS 服務的 AWS PrivateLink 部落格文章。

以下資訊適用於 Amazon VPC 的使用者。如需開始使用 Amazon VPC 的詳細資訊，請參閱《Amazon VPC 使用者指南》中的入門。

## 為 Amazon Managed Service for Prometheus 建立介面 VPC 端點

建立介面 VPC 端點，以開始使用 Amazon Managed Service for Prometheus。您可以從以下服務名稱端點中選擇：

- `com.amazonaws.region.aps-workspaces`

選擇此服務名稱，即可使用與 Prometheus 相容的 API。如需詳細資訊，請參閱《Amazon Managed Service for Prometheus 使用者指南》中的與 Prometheus 相容 API。

- `com.amazonaws.region.aps`

  選擇此服務名稱可執行工作區管理任務。如需詳細資訊，請參閱《Amazon Managed Service for Prometheus 使用者指南》中的 Amazon Managed Service for Prometheus API。

> **Note**
>
> 如果您在沒有直接網際網路存取的 VPC 中使用 remote_write，您同時必須為 AWS Security Token Service 建立介面 VPC 端點，以允許 sigv4 透過端點運作。如需建立 AWS STS VPC 端點的詳細資訊，請參閱《AWS Identity and Access Management 使用者指南》中的 使用 AWS STS 介面 VPC 端點。您必須設定 AWS STS 以使用區域化端點。

如需詳細資訊，包括建立介面 VPC 端點的逐步指示，請參閱《Amazon VPC 使用者指南》中的建立介面端點。

> **Note**
>
> 您可以使用 VPC 端點政策來控制您 Amazon Managed Service for Prometheus 介面 VPC 端點的存取權。如需詳細資訊，請參閱下一節。

如果您建立 Amazon Managed Service for Prometheus 的介面 VPC 端點，而且已有流動至您 VPC 所在工作區的資料，這些指標則會依預設透過介面 VPC 端點傳入。Amazon Managed Service for Prometheus 會使用公有端點或私有界面端點 (使用中) 來執行此任務。

## 控制 Amazon Managed Service for Prometheus VPC 端點的存取權

您可以使用 VPC 端點政策來控制 Amazon Managed Service for Prometheus 介面 VPC 端點的存取權。當您建立或修改端點時，VPC 端點政策是您連接至端點的 IAM 資源政策。如果您未在建立端點時連接政策，Amazon VPC 會以預設政策連接以允許完整存取服務。端點政策不會覆寫或取代 IAM 身分基礎政策或服務特定的政策。這個另行區分的政策會控制從端點到所指定之服務的存取。

如需詳細資訊，請參閱 《Amazon VPC 使用者指南》中的使用 VPC 端點控制服務的存取。

以下是 Amazon Managed Service for Prometheus 端點政策的範例。此政策允許角色為 PromUser 的使用者透過 VPC 連線到 Amazon Managed Service for Prometheus 以檢視工作區和規則群組，但不能檢視例如建立或刪除工作區。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonManagedPrometheusPermissions",
            "Effect": "Allow",
            "Action": [
                "aps:DescribeWorkspace",
                "aps:DescribeRuleGroupsNamespace",
                "aps:ListRuleGroupsNamespace",
                "aps:ListWorkspaces"
            ],
            "Resource": "arn:aws:aps:*:*:/workspaces*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::111122223333:role/PromUser"
                ]
            }
        }
    ]
}
```

下列範例顯示的原則僅允許來自指定 VPC 中指定 IP 位址的要求成功。來自其他 IP 位址的要求將會失敗。

```
{
    "Statement": [
        {
            "Action": "aps:*",
            "Effect": "Allow",
            "Principal": "*",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:VpcSourceIp": "192.0.2.123"
                },
        "StringEquals": {
                    "aws:SourceVpc": "vpc-555555555555"
```

```
                }
            }
        }
    ]
}
```

# 故障診斷

下列各節可幫助您對 Amazon Managed Service for Prometheus 相關問題進行移難排解。

主題



## 429 或超過限制錯誤

如果您看到類似下列範例的 429 錯誤，則您的請求已超過針對 Amazon Managed Service for Prometheus 的擷取配額。

```
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
```

如果您看到類似下列範例的 429 錯誤，則您的請求已超過工作區啟用中指標數量的 Amazon Managed Service for Prometheus 配額。

```
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
```

如果您看到類似下列範例的 400 個錯誤，表示您的請求已超過使用中時間序列的 Amazon Prometheus 受管服務配額。如需如何處理使用中時間序列配額的詳細資訊，請參閱啟用中序列預設值。

```
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)"
```

如需有關 Amazon Managed Service for Prometheus 服務配額以及請求如何增加的詳細資訊，請參閱 Amazon Managed Service for Prometheus Service Quotas

## 我看到重複的範例

如果您正在使用高可用性 Prometheus 群組，則需要在 Prometheus 執行個體上使用外部標籤來設定重複資料刪除。如需詳細資訊，請參閱 將傳送至 Amazon Managed Service for Prometheus 的高可用性指標刪除重複資料。

下一節將討論有關重複資料的其他問題。

## 我看到有關樣本時間戳記的錯誤

適用於 Prometheus 的 Amazon 受管服務會依序擷取資料，並預期每個樣本的時間戳記會比前一個範例晚。

如果您的資料未按順序送達，您可能會看到`out-of-order samples`或`duplicate sample for timestamp`、或的錯誤`samples with different value but same timestamp`。這些問題通常是因為將資料傳送至 Prometheus 的 Amazon 受管服務的用戶端設定不正確所造成。如果您使用的是在代理程式模式下執行的 Prometheus 用戶端，請檢查組態是否有重複序列名稱或重複目標的規則。如果您的指標直接提供時間戳記，請檢查它們是否錯誤。

有關如何運作的更多詳細信息，或檢查設置的方法，請參閱博客文章了解 Prometheus 的 Prometheus 從 Prometheus 實驗室中的重複樣本和 O ut-of-order 時間戳錯誤。

## 我看到與限制有關的錯誤訊息

> **Note**
>
> 適用於 Prometheus 的 Amazon 受管服務提供CloudWatch 用量指標，以監控 Prometheus 資源使用情況。使用使 CloudWatch用指標警報功能，您可以監控 Prometheus 資源和使用情況，以防止限制錯誤。

如果您看到下列其中一個錯誤訊息，便可請求增加其中一個 Amazon Managed Service for Prometheus 配額，以解決此問題。如需詳細資訊，請參閱 Amazon Managed Service for Prometheus Service Quotas。

- 超過每個使用者的 *<value>* 個序列限制，請聯繫管理員提高限制。
- 超過每個指標的 *<value>* 個序列限制，請聯繫管理員提高限制。
- 超過擷取速率限制 (...)
- 序列有太多標籤 (...) 序列:'%s'
- 查詢時間範圍超過限制（查詢長度：xxx、限制：yyy)
- 從擷取器擷取區塊時，查詢達到最大區塊數限制
- 超過限制。每個帳戶的最大工作區。

## 您的本端 Prometheus 伺服器輸出超過限制。

Amazon Managed Service for Prometheus 具有工作區可從 Prometheus 伺服器接收資料量的服務配額。若要尋找您 Prometheus 伺服器傳送至 Amazon Managed Service for Prometheus 的資料量，您可以在 Prometheus 伺服器上執行下列查詢。如果您發現 Prometheus 的輸出超過 Amazon Managed Service for Prometheus 限制，便可以請求增加對應的服務配額。如需詳細資訊，請參閱 Amazon Managed Service for Prometheus Service Quotas。

查詢您本端自行執行的 Prometheus 伺服器，尋找輸出限制。

| 資料類型 | 查詢使用 | |
|---|---|---|
| 目前啟用中序列 | `prometheus_tsdb_head_series` | |
| 目前擷取速率 | `rate(prometheus_tsdb_head_samples_appended_total[5m])` | |
| 每個度量名稱的作用中序列 M ost-to-least 清單 | `sort_desc(count by(__name__)({__name__!=""}))` | |
| 每個指標序列的標籤數量 | `group by(mylabelname)({__name__!=""})` | |

# 我的一些數據沒有出現

針對 Prometheus 傳送至 Amazon 受管服務的資料可能會因各種原因而遭到捨棄。下表顯示資料可能會被捨棄而非擷取的原因。

您可以使用 Amazon 追蹤資料被丟棄的數量和原因 CloudWatch。如需詳細資訊，請參閱 CloudWatch 度量。

| 原因 | 意義 |
| --- | --- |
| greater_than_max_sample_age | 捨棄比目前時間舊的記錄行 |
| new-value-for-timestamp | 針對重複範例傳送的時間戳記與先前記錄的時間戳記不同 |
| per_metric_series_limit | 使用者已到達每個指標限制的啟用中序列 |
| per_user_series_limit | 使用者已到達用啟用中序列總數限制 |
| rate_limited | 已限制擷取速率 |
| sample-out-of-order | 範例已寄出，無法處理 |
| label_value_too_long | 標籤值超過允許的字元限制 |
| max_label_names_per_series | 使用者已到達各指標的標籤名稱 |
| missing_metric_name | 未提供指標名稱 |
| metric_name_invalid | 提供的指標名稱無效 |
| label_invalid | 提供的標籤無效 |
| duplicate_label_names | 提供重複的標籤名稱 |

# 標記

標籤是一種自訂屬性標籤，可由您或 AWS 指派給 AWS 資源。每個 AWS 標籤都有兩個部分：

- 標籤鍵 (例如，CostCenter、Environment、Project 或 Secret )。標籤鍵會區分大小寫。
- 一個名為標籤值 (例如，111122223333、Production 或團隊名稱) 的選用欄位。忽略標籤值基本上等同於使用空字串。與標籤鍵相同，標籤值會區分大小寫。

這些合稱為鍵值組。每個工作區最多可以指派 50 個標籤。

標籤可協助您識別和整理 AWS 資源。許多 AWS 服務支援標記，因此您可以將相同標籤指派給不同服務的資源，以表示相關資源。例如，您可以將相同的標籤指派給您指派給 Amazon Managed Service for Prometheus 工作您可在此指派給 Amazon S3 儲存貯體。如需有關標記策略的詳細資訊，請參閱標記 AWS 資源。

在 Amazon Managed Service for Prometheus 中，可以標記工作區和規則群組的命名空間。您可以使用主控台、AWS CLI、 API 或 SDK 來新增、管理和移除這些資源的標籤。除了使用標籤識別、組織和追蹤含標籤的工作區和規則群組命名空間以外，您可使用 IAM 政策中的標籤來協助控制可檢視以及與 Amazon Managed Service for Prometheus 資源互動的人員。

標籤限制

以下基本限制適用於標籤：

- 每個資源的上限為 50 個標籤。
- 對於每一個資源，每個標籤金鑰必須是唯一的，且每個標籤金鑰只能有一個值。
- 最大標籤索引鍵長度為 128 個 UTF-8 形式的 Unicode 字元。
- 最大標籤值長度為 256 個 UTF-8 形式的 Unicode 字元。
- 如果您的標記結構描述用於多個 AWS 服務和資源，請記得，其他服務可能限制允許的字元。一般而言，允許的字元為字母、數字、使用 UTF-8 表示的空格，還有以下字元：. : + = @ _ / - (連字號)。
- 標籤鍵與值皆區分大小寫。做為最佳實務，請決定大寫標籤的策略，並一致地在所有資源類型中實作該策略。例如，決定要使用 Costcenter、costcenter 還是 CostCenter，並針對所有標籤使用相同的慣例。避免針對相似的標籤使用不一致的大小寫處理。
- 請勿使用 aws:、AWS: 或任何大小寫組合作為索引鍵或值的字首。因為僅預留給 AWS 使用。您不可編輯或刪除具此字首的標籤金鑰或值。具此字首的標籤不算在每一資源的標籤數限制內。

主題



# 標記工作區

使用本節的程序來處理 Amazon Managed Service for Prometheus 工作區的標籤。

主題



## 將標籤新增到工作區

新增標籤到 Amazon Managed Service for Prometheus 工作區後，可協助您識別和整理 AWS 資源並管理存取權。首先，將一或多個標籤 (金鑰值對) 新增到工作區。當您擁有標籤後，您可建立 IAM 政策，根據這些標籤管理工作區的存取權。您可以使用主控台或 AWS CLI，將標籤新增到 Amazon Managed Service for Prometheus 工作區。

> **Important**
>
> 將標籤新增至工作區可能會影響對該工作區的存取權。將標籤新增到工作區之前，務必檢閱任何可能會使用標籤控制資源存取權的 IAM 政策。

如需在建立政策時將標籤新增至 Amazon Managed Service for Prometheus 工作區的詳細資訊，請參閱 建立工作區。

主題

## 將標籤新增至工作區 (主控台)

您可以使用主控台將一個或多個標籤新增到 Amazon Managed Service for Prometheus 工作區。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇 [所有工作區]。
4. 選擇您要管理的工作區的工作區 ID。
5. 選擇 Tags (標籤) 索引標籤。
6. 若尚未將標籤新增至 Amazon Managed Service for Prometheus 工作區，請選擇 [建立標籤]。否則，請選擇 [管理標籤]。
7. 在 Key (金鑰) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。
8. (選用) 若要新增另一個標籤，再選擇 Add tag (新增標籤) 一次。
9. 當您完成新增標籤後，請選擇 Save changes (儲存變更)。

## 將標籤新增至工作區 (AWS CLI)

請依照下列步驟使用 AWS CLI，將標籤新增至 Amazon Managed Service for Prometheus 工作區。若要在建立標籤時，將其新增到工作區，請參閱 建立工作區。

在這些步驟中，我們假設您已經安裝新版 AWS CLI 或更新到最新版本。如需詳細資訊，請參閱安裝 AWS Command Line Interface。

在終端機或命令列上執行 tag-resource 命令，指定工作區的 Amazon Resource Name (ARN)，您希望新增標籤和想要新增的標籤索引鍵。您可以將超過一個標籤新增至工作區。例如，若要使用兩個標籤來標記名為 My-Workspace 的 Amazon Managed Service for Prometheus 工作區，一個名為 *Status* 的標籤金鑰，以及一個名為 *Team* 的標籤金鑰，兩個標籤金鑰的標籤值為 *Secret* 和 *My-Team* ：

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
--tags Status=Secret,Team=My-Team
```

如果成功，此命令不會傳回任何內容。

# 檢視工作區的標籤

標籤可協助您辨識和整理您的 AWS 資源和管理存取權。如需關於標記策略的詳細資訊，請參閱標記 AWS 資源。

## 檢視 Amazon Managed Service for Prometheus 工作區 (主控台)

您可以使用主控台來檢視與 Amazon Managed Service for Prometheus 工作區相關聯的標籤。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇 [所有工作區]。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇 Tags (標籤) 索引標籤。

## 檢視 Amazon Managed Service for Prometheus 工作區的標籤 (AWS CLI)

依照以下步驟，使用 AWS CLI 檢視工作區的 AWS 標籤。若未新增標籤，傳回的清單空白。

在終端機或命令列上執行 list-tags-for-resource 命令。例如，檢視工作區的標籤金鑰和標籤值清單：

```
aws amp list-tags-for-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring
```

若成功，此命令會傳回類似如下的資訊：

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

# 編輯工作區的標籤

您可以變更與工作區相關的標籤值。您也可以變更金鑰名稱，等於移除目前的標籤，並新增一個不同的新的名稱和相同的值作為其他金鑰。

> **Important**
>
> 編輯 Amazon Managed Service for Prometheus 工作區標籤時，可影響該工作區存取權。編輯儲存庫名稱 (金鑰) 或標籤的值之前，務必檢閱任何可能會使用標籤金鑰或值來控制資源存取權的 IAM 的 IAM 政策，例如儲存庫。

### 編輯 Amazon Managed Service for Prometheus 的標籤 (主控台)

您可以使用主控台來編輯與 Amazon Managed Service for Prometheus 工作區相關聯的標籤。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇 [所有工作區]。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇 Tags (標籤) 索引標籤。
6. 若未將標籤新增至工作區，請選擇 [建立標籤]。否則，請選擇 [管理標籤]。
7. 在 Key (金鑰) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。
8. (選用) 若要新增另一個標籤，再選擇 Add tag (新增標籤) 一次。
9. 當您完成新增標籤的作業時，請選擇 Save changes (儲存變更)。

### 編輯 Amazon Managed Service for Prometheus 工作區的標籤 (AWS CLI)

依照以下步驟使用 AWS CLI 更新工作區的標籤。您可以變更現有索引鍵的值或新增其他索引鍵。

在終端機或命令列執行 tag-resource 命令，指定您要更新標籤之 Amazon Managed Service for Prometheus 工作區的 Amazon Resource Name (ARN)，並指定標籤索引鍵和標籤值：

```
aws amp tag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tags Team=New-Team
```

## 將標籤從工作區移除

您可以移除一或多個與工作區相關聯的標籤。移除標籤不會從其他 AWS 資源刪除與該標籤相關聯的標籤。

> **Important**
>
> 移除 Amazon Managed Service for Prometheus 工作區的標籤後，可影響該工作區的存取權。從工作區移除標籤之前，務必檢閱任何可能會使用標籤金鑰或值來控制資源存取權的 IAM 政策，例如儲存庫。

## 從 Amazon Managed Service for Prometheus (主控台) 移除標籤

您可以使用主控台移除標籤和工作區之間的關聯。

1. 開啟 Amazon Managed Service for Prometheu 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇 所有工作區。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇 Tags (標籤) 索引標籤。
6. 選擇 Manage tags (管理標籤)。
7. 尋找要刪除的標籤，然後選擇 Remove (移除)。

## 從 Amazon Managed Service for Prometheus 工作區移除標籤 (AWS CLI)

依照以下步驟使用 AWS CLI 從資源移除標籤。移除標籤並不會將其刪除，只會移除標籤和工作區之間的關聯。

> **Note**
>
> 如果您刪除 Amazon Managed Service for Prometheus 工作區，所有標籤關聯皆會從刪除的工作區中移除。您不需要在刪除工作區之前移除標籤。

在終端機或命令列執行 untag-resource 命令，指定您要移除標籤之工作區的 Amazon Resource Name (ARN)，和您想移除之標籤的標籤索引鍵。例如，在名為 My-Workspace 的工作區中移除標籤，使用標籤索引鍵 *##*：

```
aws amp untag-resource --resource-arn arn:aws:aps:us-
west-2:123456789012:workspace/IDstring --tag-keys Status
```

若成功，此命令不會傳回任何內容。若要驗證與工作區相關聯的標籤，請執行 list-tags-for-resource 命令。

# 標記規則群組命名空間

使用本節的程序來處理 Amazon Managed Service for Prometheus 規則群組命名空間。

主題



## 將標籤新增到規則群組命名空間

新增標籤到 Amazon Managed Service for Prometheus 規則群組命名空間，可協助您識別和整理您的 AWS 資源並管理存取權。首先，將一或多個標籤 (金鑰值對) 新增到規則群組命名空間。當您擁有標籤後，可以根據這些標籤建立 IAM 政策，以管理專案的存取權。您可以使用主控台或 AWS CLI，將標籤新增至 Amazon Managed Service for Prometheus 規則群組命名空間。

> **Important**
>
> 將標記新增至規則群組命名空間可能會影響該規則群組命名空間的存取權。將標籤新增到專案之前，務必檢閱任何可能會使用標籤控制存取資源 (例如組建專案) 的 IAM 政策。

如需在建立政策時，將標籤新增至規則群組命名空間的詳細資訊，請參閱 建立規則檔案。

主題



### 將標籤新增至規則群組命名空間 (主控台)

您可以使用主控台為 Prometheus 規則群組命名空間的 Amazon 受管服務新增一或多個標籤。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇「所有工作區」。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇「規則管理」標籤。
6. 選擇命名空間名稱旁的按鈕，然後選擇「編輯」。
7. 選擇「建立標籤」，「新增標籤」。
8. 在 Key (金鑰) 中，輸入標籤的名稱。您可以在 Value (值) 中為標籤新增選用值。
9. (選用) 若要新增另一個標籤，請再選擇「Add tag (新增標籤)」一次。
10. 當您完成新增標籤的作業時，請選擇「Save changes (儲存變更)」。

## 將標記新增至規則群組命名空間 (AWS CLI)

請依照下列步驟使用 AWS CLI 將標籤新增至 Amazon Managed Service for Prometheus 規則群組命名空間。若要在建立標籤時將其新增至規則群組命名空間，請參閱 將規則組態檔案上傳至 Amazon Managed Service for Prometheus。

在這些步驟中，我們假設您已經安裝新版 AWS CLI 或更新到最新版本。如需詳細資訊，請參閱安裝 AWS Command Line Interface。

在終端機或命令列，執行 tag-resource 命令，為您要新增標籤的規則群組命名空間指定 Amazon Resource Name (ARN)，以及您想新增標籤的索引鍵和值。您可以將多個標記新增至規則群組命名空間。例如，若要使用兩個標籤來標記名為 My-Workspace 的 Amazon Managed Service for Prometheus 命名空間、名為 *Status* 的標籤金鑰 (標籤值為 *Secret*)，以及一個名為 *Team* 的標籤金鑰 (標籤值為 *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
```

如果成功，此命令不會傳回任何內容。

# 檢視規則群組命名空間標籤

標籤可協助您辨識和整理您的 AWS 資源和管理存取權。如需有關標記策略的詳細資訊，請參閱標記 AWS 資源。

## 檢視 Amazon Managed Service for Prometheus 規則群組命名空間 (主控台) 的標籤

您可以使用主控台來檢視與 Amazon Managed Service for Prometheus 規則群組命名空間相關聯的標籤。

1. 開啟 Amazon Managed Service in Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇「所有工作區」。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇「規則管理」標籤。
6. 選擇命名空間名稱。

## 檢視 Amazon Managed Service for Prometheus 工作區的標籤 (AWS CLI)

依照下列步驟使用 AWS CLI 以檢視規則群組命名空間的 AWS 標籤。如果沒有新增標籤，將會傳回空的清單。

在終端機或命令列上執行 list-tags-for-resource 命令。例如，檢視規則群組命名空間的標籤索引鍵和標籤值清單：

```
aws amp list-tags-for-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name
```

如果成功，此命令會傳回類似如下的資訊：

```
{
    "tags": {
        "Status": "Secret",
        "Team": "My-Team"
    }
}
```

# 編輯規則群組命名空間標籤

您可以變更與規則群組命名空間相關聯標籤的值。您也可以變更金鑰名稱，等於移除目前的標籤，並新增一個不同的新的名稱和相同的值作為其他金鑰。

> **Important**
>
> 編輯規則群組命名空間的標記可能會影響其存取權。編輯資源的名稱 (金鑰) 或標籤值之前，務必檢閱任何可能會使用標籤金鑰或值來控制存取資源的 IAM 政策。

## 編輯 Amazon Managed Service for Prometheus 規則群組命名空間 (主控台) 的標籤

您可以使用主控台來編輯與 Amazon Managed Service for Prometheus 規則群組命名空間相關聯的標籤。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇「所有工作區」。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇「規則管理」標籤。
6. 選擇命名空間的名稱。
7. 選擇「管理 」 和「新增標籤 」。
8. 若要變更既有標籤的值，請在「值」中輸入新值。
9. 若要新增其他標籤，請選擇「新增標籤」。
10. 當您完成新增和編輯標籤後，請選擇「儲存變更」。

## 編輯Amazon Managed Service for Prometheus 規則群組命名空間的標籤 (AWS CLI)

依照以下步驟使用 AWS CLI 更新規則群組命名空間的標籤。您可以變更現有索引鍵的值或新增其他索引鍵。

在終端機或命令列，執行 tag-resource 命令，指定您要更新標籤的 Amazon Resource Name (ARN)，並指定標籤索引鍵和標籤值：

```
aws amp tag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tags Team=New-Team
```

## 從規則群組命名空間移除標籤

您可以移除一或多個與規則群組命名空間相關聯的標籤。移除標籤不會從其他 AWS 資源刪除與該標籤相關聯的標籤。

> **Important**
>
> 移除資源的標籤可能會影響該資源的存取權。從資源移除標籤之前，務必檢閱任何可能會使用標籤金鑰或值來控制資源 (例如儲存庫) 的 IAM 政策。

### 從 Amazon Managed Service for Prometheus 規則群組命名空間 (主控台) 移除標籤

您可以使用主控台，移除標籤與規則群組命名空間之間的關聯。

1. 開啟 Amazon Managed Service for Prometheus 主控台，位於 https://console.aws.amazon.com/prometheus/。
2. 在導覽窗格中，選擇功能表圖示。
3. 選擇「所有工作區」。
4. 選擇您要管理工作區的工作區 ID。
5. 選擇「規則管理」標籤。
6. 選擇命名空間的名稱。
7. 選擇 Manage tags (管理標籤)。
8. 從您要刪除的標籤旁，選擇 Remove (移除)。
9. 完成之後，請選擇 Save changes (儲存變更)。

### 從 Amazon Managed Service for Prometheus 規則群組命名空間移除標籤 (AWS CLI)

依照以下步驟，使用 AWS CLI 從規則群組命名空間移除標籤。移除標籤並不會將其刪除，只會移除標籤和規則群組命名空間之間的關聯。

> **Note**
>
> 如果您刪除 Amazon Managed Service for Prometheus 規則群組命名空間，則所有標籤關聯都會從已刪除命名空間中移除。您不必在刪除命名空間之前移除標籤。

在終端機或命令執行 untag-resource 命令，指定您要移除標籤的規則群組命名空間 Amazon Resource Name (ARN)，以及您要移除標籤的標籤金鑰。例如，在名為 My-Workspace 的工作區中移除標籤金鑰為 *Status* 的標籤：

```
aws amp untag-resource --resource-arn rn:aws:aps:us-
west-2:123456789012:rulegroupsnamespace/IDstring/namespace_name --tag-keys Status
```

如果成功，此命令不會傳回任何內容。若要驗證與管道相關的標籤，請執行 list-tags-for-resource 命令。

# Amazon Managed Service for Prometheus Service Quotas

以下兩節說明與 Amazon Managed Service for Prometheus 相關的配額和限制。

## Service Quotas

Amazon Managed Service for Prometheus 具有以下配額。適用於 Prometheus 的 Amazon 受管服務會出售CloudWatch 使用量指標，以監控 Prometheus 資源使用情況。使用使 CloudWatch 用指標警報功能，您可以監控 Prometheus 資源和使用情況，以防止限制錯誤。

隨著專案和工作區的成長，您可能需要監控或請求增加的最常見配額為：每個工作區的啟用中序列、每個工作區的擷取率，以及每個工作區的擷取成組分解大小。

對於所有可調整的配額，您可透過選取 [可調整] 欄中的連結或請求增加配額，來請求提高配額。

動態套用 [每個工作區的啟用中序列] 限制。如需詳細資訊，請參閱 啟用中序列預設值。每個工作區的擷取率和每個工作區的擷取成組分解大小共同控制您可以將資料擷取至工作區的速度。如需更多資訊，請參閱 攝入節流。

> **Note**
>
> 除非另有說明，否則這些配額是每個工作區。

| 名稱 | 預設 | 可調整 | 描述 |
| --- | --- | --- | --- |
| 每個工作區含中繼資料的啟用中指標 | 每個受支援的地區：20,000 個 | 否 | 每個工作區具有中繼資料的唯一啟用中指標數量。 |
| 每個工作區的啟用中序列 | 每個受支援的區域：每 2 小時 10,000,000 | 是 | 每個工作區的唯一啟用中序列數量。如果在過去 2 小時內呈報範例，則該序列為啟用中。容量從 2M 到 10M 是根據最後 30 分鐘的使用量自動調整。 |

| 名稱 | 預設 | 可調整 | 描述 |
| --- | --- | --- | --- |
| 警示管理員定義檔案中的警示彙總群組大小 | 每個受支援的區域：1,000 | 是 | 警示管理員定義檔案中警示彙總群組的大小上限。group_by 的每個標籤值組合都會建立彙總群組。 |
| 警示管理員定義檔案大小 | 每個受支援的區域：1 MB | 否 | 警示管理員定義檔案的大小上限。 |
| 警報管理器中的警報裝載大小 | 每個受支援的地區：20 MB | 否 | 每個工作區之所有「警示管理員」警示的最大警示承載大小。警示大小取決於標籤和註釋。 |
| 警報管理器中的警報 | 每個受支援的區域：1,000 | 是 | 每個工作區同時警示管理員警示的最大數目。 |
| HA 追蹤器叢集 | 每個受支援的區域：500 | 否 | HA 追蹤器會追蹤每個工作區擷取樣本的叢集數量上限。 |
| 擷取每個工作區成組分解大小 | 每個受支援的區域：1,000,000 | 是 | 每個工作區每秒一次突發可擷取的最大範例數量。 |
| 每個工作區的擷取速率 | 每個受支援的區域：170,000 | 是 | 每個工作區每秒的指標範例擷取率。 |
| 警示管理員定義檔案中的抑制規則 | 每個受支援的區域：100 | 是 | 警示管理員定義檔案中抑制規則的數量上限。 |
| 標籤大小 | 每個受支援的區域：7 KB | 否 | 系列接受所有標籤和標籤值的最大組合大小。 |
| 每個公制系列的標籤 | 每個受支援的區域：70 | 是 | 每個公制系列的標籤數量。 |

| 名稱 | 預設 | 可調整 | 描述 |
| --- | --- | --- | --- |
| 中繼資料長度 | 每個受支援的區域：1 KB | 否 | 指標中繼資料可接受的最大長度。中繼資料泛指指標名稱、說明和單位。 |
| 每個指標的中繼資料 | 每個受支援的區域：10 | 否 | 每個指標的中繼資料數量。 |
| 警示管理員路由樹狀結構節點 | 每個受支援的區域：100 | 是 | 警示管理員路由樹狀結構中節點的數量上限。 |
| 每秒交易中的 API 作業數 | 每個受支援的區域：10 | 是 | 各區域每秒可以執行的 API 作業數量上限。這包括工作區 CRUD API、標記 API、規則群組命名空間 CRUD API，以及警示管理員定義 CRUD API。 |
| 查詢位元組以進行即時查詢 | 所有受支援的區域：5 GB | 否 | 單一即時查詢可掃描的最大位元組數。 |
| 範圍查詢的查詢位元組 | 所有受支援的區域：5 GB | 否 | 在單一範圍查詢中，每 24 小時間隔可掃描的最大位元組數。 |
| 擷取的查詢區塊 | 每個受支援的區域：20,000,000 | 否 | 單一查詢期間可掃描的區塊的最大數量。 |
| 範例查詢 | 每個受支援的區域：50,000,000 | 否 | 單一查詢期間可掃描的最大範例數量。 |
| 查詢序列擷取 | 每個受支援的區域：12,000,000 | 否 | 單一查詢期間可掃描的最大序列數量。 |
| 查詢時間範圍 (天) | 每個受支援的區域：32 | 否 | 任何 PromQL 查詢的最大時間範圍。 |

| 名稱 | 預設 | 可調整 | 描述 |
| --- | --- | --- | --- |
| 請求規模 | 每個受支援的區域：1 MB | 否 | 擷取或查詢的請求大小上限。 |
| 擷取資料的保留時間 (天) | 每個受支援的區域：150 | 是 | 工作區內資料保留的天數。系統會刪除超過此值的資料。您可以請求配額更改以增加或減少此值。 |
| 規則評估間隔 | 每個受支援的區域：30 秒 | 是 | 每個工作區規則群組的最小規則評估間隔。 |
| 規則群組命名空間定義檔大小 | 每個受支援的區域：1 MB | 否 | 規則群組命名空間定義檔案的大小上限。 |
| 每個工作區的規則 | 每個受支援的區域：2,000 | 是 | 每個工作區的規則數量上限。 |
| 警示管理員定義檔案中的範本 | 每個受支援的區域：100 | 是 | 警示管理員定義檔案中的範本數量上限。 |
| 每個帳戶每個區域的工作 | 每個受支援的區域：25 | 是 | 每個區域的工作區的數量上限。 |

## 啟用中序列預設值

Amazon Managed Service for Prometheus 可讓您預設使用最多啟用中時間序列的配額。

Amazon Managed Service for Prometheus 工作區會自動調整為您的擷取量。隨著使用量增加，Amazon Managed Service for Prometheus 會將您的時間序列容量自動增加至基準使用量的兩倍，直到預設配額為止。例如，如果過去 30 分鐘的平均啟用中時間序列為 350 萬，則您最多可以使用 700 萬個時間序列，而不需進行限流。

如果您需要先前基準的兩倍以上，Amazon Managed Service for Prometheus 會隨著擷取磁碟區增加而自動分配更多容量，以協助確保您的工作負載不會經歷持續的限流，直到到達您的配額為止。但是，

如果過去 30 分鐘超過先前基準的兩倍以上，還是會出現限流情況。為避免限流，Amazon Managed Service for Prometheus 建議在增加到先前啟用中時間序列的兩倍以上時，逐漸增加擷取。

> **Note**
>
> 啟用中時間序列的最小容量為 2 百萬，若您的序列小於 200 萬個，則不會出現限流。若要超過預設配額，您可請求增加配額。

## 攝入節流

適用於 Prometheus 的 Amazon 受管服務會根據您目前的限制，針對每個工作區進行節流擷取。這有助於維護工作區的效能。如果超過限制，您將`DiscardedSamples`在 CloudWatch 指標中看到（`rate_limited`原因）。您可以使 CloudWatch 用 Amazon 監控擷取，並建立警示，以便在您接近節流限制時發出警告。如需詳細資訊，請參閱 CloudWatch 度量。

適用於 Prometheus 的 Amazon 受管服務使用權杖儲存貯體演算法來實作擷取節流。使用此算法，您的帳戶擁有一個存儲區，其中包含特定數量的令牌。存儲桶中的令牌數量代表您在任何給定秒鐘的獲取限制。

每個擷取的資料樣本都會從值區中移除一個 Token。如果您的儲存貯體大小 (每個工作區的擷取成組分解大小) 為 1,000,000，您的工作區可以在一秒內擷取一百萬個資料樣本。如果要擷取的樣本超過一百萬個，則會進行節流，並且不會擷取任何更多記錄。其他數據樣本將被丟棄。

儲存貯體會以設定的速率自動補充。如果存儲桶低於其最大容量，則每秒會向其添加一組數量的令牌，直到達到其最大容量為止。如果在補充令牌到達時存儲桶已滿，則將其丟棄。值區的容量不能超過其最大數量的代幣。取樣擷取的重新填充率是由每個工作區的擷取率限制所設定。如果每個工作區的擷取率設定為 170,000，則值區的補充率為每秒 170,000 個代幣。

如果您的工作區一秒內擷取 1,000,000 個資料樣本，您的儲存貯體會立即減少為零個權杖。然後，桶每秒會重新填充 170,000 個代幣，直到達到 1,000,000 個代幣的最大容量為止。如果沒有更多的擷取，先前的空值區將會在 6 秒內回復為其最大容量。

> **Note**
>
> 擷取會在批次要求中進行。如果您有 100 個可用的令牌，並發送包含 101 個樣本的請求，則整個請求將被拒絕。適用於 Prometheus 的 Amazon 受管服務不接受部分請求。如果您正在編寫收集器，則可以管理重試次數 (使用較小的批次或經過一段時間後)。

您不需要等待值區已滿，您的工作區才能擷取更多資料樣本。您可以在添加到存儲桶中時使用令牌。如果您立即使用補充令牌，則存儲桶未達到其最大容量。例如，如果您耗盡儲存貯體，您可以繼續每秒內擷取 170,000 個資料樣本。只有當您每秒擷取少於 170,000 個資料樣本時，儲存貯體才能重新填充至最大容量。

# 對擷取資料的其他限制

針對擷取到工作區的資料，Amazon Managed Service for Prometheus 也有下列額外要求。這些不可調整。

- 拒絕擷取超過 1 小時的指標範例。
- 每個範例和中繼資料都必須有指標名稱。

# API 參考

本節列出 Amazon Managed Service for Prometheus 所支援的 API 作業和資料結構。

如需這些 API 作業及其系列、標籤和 API 請求配額的相關資訊，請參閱 《Amazon Managed Service for Prometheus 使用者指南》的 Amazon Managed Service for Prometheus 服務配額。

主題

- Amazon Managed Service for Prometheus API
- 與 Prometheu 相容的 API

## Amazon Managed Service for Prometheus API

適用於 Prometheus 的 Amazon 受管服務提供 API 操作，為 Prometheus 工作區建立和維護您的 Amazon 受管服務。這包括用於工作區，抓取器，警報管理器定義，規則組命名空間和日誌記錄的 API。

如需有關適用於 Prometheus API 的 Amazon 受管服務的詳細資訊，請參閱 Amazon Prometheus API 受管服務參考。

### 搭配 SDK 使用適用於 Prometheus 的 Amazon 託管服務 AWS

AWS 軟件開發套件（SDK）可用於許多流行的編程語言。每個 SDK 都提供 API、程式碼範例和文件，讓開發人員能夠更輕鬆地以慣用的語言建置 AWS 應用程式。如需依語言分類的 SDK 和工具清單，請參閱 AWS 開發人員中心中要建置的工具。 AWS

> **SDK 版本**
>
> 我們建議您使用最新版本的 AWS SDK，以及您在專案中使用的任何其他 SDK，並使 SDK 保持在最新狀態。 AWS SDK 提供您最新的特色和功能，以及安全性更新。

## 與 Prometheu 相容的 API

Amazon Managed Service for Prometheus 支援與 Prometheus 相容的 API相容的 API。

如需有關使用與 Prometheus 相容的 API 的詳細資訊，請參閱。使用與 Prometheus 相容的 API 查詢

主題



## CreateAlertManagerAlerts

`CreateAlertManagerAlerts` 作業會在工作區中建立警示。

有效的 HTTP 動詞：

`POST`

有效 URI：

`/workspaces/workspaceId/alertmanager/api/v2/alerts`

URL 查詢參數

`alerts` 物件陣列，其中每個物件代表一個警示。以下是警示物件路徑的範例：

```
[
```

```
  {
    "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"
  }
]
```

### 請求範例

```
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/"
  }
]
```

### 回應範例

```
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

DeleteSilence 刪除一個警示靜音。

有效的 HTTP 動詞：

DELETE

有效 URI：

/workspaces/*workspaceId*/alertmanager/api/v2/silence/*silenceID*

URL 查詢參數：無

### 請求範例

```
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
```

### 回應範例

```
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

GetAlertManagerStatus 擷取有關警示管理員狀態的資訊。

有效的 HTTP 動詞：

```
GET
```

有效 URI：

```
/workspaces/workspaceId/alertmanager/api/v2/status
```

URL 查詢參數：無

### 請求範例

```
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
```

### 回應範例

```
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

GetAlertManagerSilence 擷取有關一個警示靜音的資訊。

有效的 HTTP 動詞：

GET

有效 URI：

/workspaces/*workspaceId*/alertmanager/api/v2/silence/*silenceID*

URL 查詢參數：無

**請求範例**

```
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
```

**回應範例**

```
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

GetLabels 作業會擷取與時間序列相關聯的標籤。

有效的 HTTP 動詞：

GET, POST

有效 URI：

/workspaces/*workspaceId*/api/v1/labels

/workspaces/*workspaceId*/api/v1/label/*label-name*/values 此 URI 僅支援 GET 請求。

URL 查詢參數：

match[]=<series_selector> 重複序列選擇器引數，選擇要從中讀取標籤名稱的序列。選用。

start=<rfc3339 | unix_timestamp> 開始時間戳記。選用。

end=<rfc3339 | unix_timestamp> 結束時間戳記。選用。

### 樣品請求 **/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
```

### 範例回應 **/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",
```

```
        ...
    ]
}
```

**範例請求 /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
```

**範例回應 /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

此 GetMetricMetadata 作業會擷取目前從目標擷取指標的相關中繼資料。不會提供目標詳細資訊。

查詢結果的資料區段是由一個物件組成，其中每個索引鍵都是測量結果名稱，而每個值都是唯一的中繼資料物件清單，這些物件會顯示在所有目標的測量結果名稱。

有效的 HTTP 動詞：

- GET

有效 URI：

/workspaces/*workspaceId*/api/v1/metadata

URL 查詢參數：

limit=<number> 傳回的指標最大數量。

metric=<string> 用來篩選其中繼資料的指標名稱。如果將此項保留空白，則會擷取所有指標中繼資料。

### 請求範例

```
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
```

### 回應範例

```
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

此 GetSeries 作業會擷取符合特定標籤集的時間序列清單。

有效的 HTTP 動詞：

GET, POST

有效 URI：

/workspaces/*workspaceId*/api/v1/series

URL 查詢參數：

match[]=<series_selector> 重複序列選擇器參數，選擇要傳回的序列。至少必須提供一個 match[] 引數。

start=<rfc3339 | unix_timestamp> 開始時間戳記。選用

end=<rfc3339 | unix_timestamp> 結束時間戳記。選用

請求範例

```
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
```

回應範例

```
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

`ListAlerts` 作業會擷取工作區中目前啟用中的警示。

有效的 HTTP 動詞：

```
GET
```

有效 URI：

```
/workspaces/workspaceId/api/v1/alerts
```

### 請求範例

```
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
```

### 回應範例

```
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

ListAlertManagerAlerts 會擷取工作區警示管理員中目前觸發警示的相關資訊。

有效的 HTTP 動詞：

```
GET
```

有效 URI：

```
/workspaces/workspaceId/alertmanager/api/v2/alerts
```

### 請求範例

```
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
```

### 回應範例

```
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

此 `ListAlertManagerAlertGroups` 作業會擷取工作區警示管理員中所設定的警示群組清單。

有效的 HTTP 動詞：

`GET`

有效 URI：

`/workspaces/workspaceId/alertmanager/api/v2/alerts/groups`

URL 查詢參數：

`active` 布林值。如果為 true，則傳回的清單會包含作用中警示。預設值為 true。選用

`silenced` 布林值。如果為 true，則傳回的清單會包含靜音警示。預設值為 true。選用

`inhibited` 布林值。如果為 true，則傳回的清單表包括抑制警報。預設值為 true。選用

`filter` 字串陣列。篩選警示所依據的配對程式清單。選用

`receiver` 字串。一個規則表達式符合警示篩選依據的接收器。選用

### 請求範例

```
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
```

### 回應範例

```
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

ListAlertManagerReceivers 作業會擷取警示管理員中設定接收器的相關資訊。

有效的 HTTP 動詞：

```
GET
```

有效 URI：

```
/workspaces/workspaceId/alertmanager/api/v2/receivers
```

URL 查詢參數：無

**請求範例**

```
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
```

**回應範例**

```
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

ListAlertManagerSilences 作業會擷取工作區中設定的警示靜音相關資訊。

有效的 HTTP 動詞：

GET

有效 URI：

/workspaces/*workspaceId*/alertmanager/api/v2/silences

### 請求範例

```
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
```

### 回應範例

```
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

`ListRules` 會擷取有關在工作區中組態規則的資訊。

有效的 HTTP 動詞：

```
GET
```

有效 URI：

```
/workspaces/workspaceId/api/v1/rules
```

**請求範例**

```
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
```

**回應範例**

```
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

`PutAlertManagerSilences` 作業會建立新的警示靜音或更新現有警示靜音。

有效的 HTTP 動詞：

POST

有效 URI：

/workspaces/*workspaceId*/alertmanager/api/v2/silences

URL 查詢參數：

silence 代表靜音的物件。以下為其格式：

```
{
  "id": "string",
  "matchers": [
    {
      "name": "string",
      "value": "string",
      "isRegex": Boolean,
      "isEqual": Boolean
    }
  ],
  "startsAt": "timestamp",
  "endsAt": "timestamp",
  "createdBy": "string",
  "comment": "string"
}
```

請求範例

```
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"
}
```

回應範例

```
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

QueryMetrics 作業評估在單一時間點或在一定時間範圍內的即時查詢。

有效的 HTTP 動詞：

GET, POST

有效 URI：

/workspaces/*workspaceId*/api/v1/query 此 URI 會在單一時間點評估即時查詢。

/workspaces/*workspaceId*/api/v1/query_range 此 URI 會評估一段時間範圍內的即時查詢。

URL 查詢參數：

query=<string> Prometheus 表達式查詢字串。用於 query 和 query_range。

time=<rfc3339 | unix_timestamp> (選用) 若您在單一時間點使用 query 立即查詢，則評估時間戳記。

timeout=<duration> (選用) 評估逾時。預設為和由-query.timeout旗標的值加上限。用於 query 和 query_range。

start=<rfc3339 | unix_timestamp> 若您正在使用 query_range 查詢時間範圍，則開始時間戳記。

end=<rfc3339 | unix_timestamp> 若您正在使用 query_range 查詢時間範圍，則結束時間戳記。

step=<duration | float> 查詢解析度步驟寬度 (duration 格式或 float 秒數)。只有在您正在使用 query_range 查詢時間範圍，並在此類查詢必要時才可使用。

Duration (持續時間)

與 Prometheus 相容 API 的 duration，後續立即接著下列其中一個單位：

- ms 毫秒
- s 秒
- m 分鐘
- h 小時
- d 天，假設一天總是 24 小時
- w 週，假設一周總是 7 天
- y 年，假設一年總是 365 天

請求範例

```
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
```

回應範例

```
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

RemoteWrite 作業會使用標準化格式將指標自 Prometheus 伺服器寫入遠端 URL。通常，您將使用現有的用戶端（例如 Prometheus 伺服器）來呼叫此作業。

有效的 HTTP 動詞：

POST

有效 URI：

/workspaces/*workspaceId*/api/v1/remote_write

URL 查詢參數：

無

RemoteWrite 擷取速率為每秒 70,000 個樣本，擷取突發大小為 1,000,000 個樣本。

請求範例

```
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**
>
> 有關請求主體語法，請參閱 https://github.com/prometheus/prometheus/blob/1c624c58ca934f618be737b4995e22051f5724c1/prompb/remote.pb.go#L64 的協議緩衝區定義。

### 回應範例

```
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
```

# Amazon Managed Service for Prometheus 使用者指南的文件歷史記錄

下表說明 Amazon Managed Service for Prometheus 使用者指南中的重要說明文件更新。如需有關此說明文件更新的通知，您可以訂閱 RSS 摘要。

| 變更 | 描述 | 日期 |
| --- | --- | --- |
| 在控制台中添加了對規則定義文件和警報管理器配置文件的編輯 | 適用於 Prometheus 的 Amazon 受管服務新增支援，可從適用於 Prometheus 的 Amazon 受管服務主控台中編輯警示管理員組態檔案和規則定義檔案。 | 2024年5月16日 |
| 使用 Amazon EKS 的訪問條目添加了更簡單的 AWS 託管收集器設置 | 適用於 Prometheus 的 Amazon 受管服務新增了對 Amazon EKS 存取項目的支援，以簡化受管收集器的設定。AWS受AmazonPrometheusScraperServiceRolePolicy管理之收集器的 AWS 受管理原則會更新，以允許刪除不再使用的存取項目。 | 2024年5月2日 |
| 將 AWS API 移至單獨的 API 參考指南 | 適用於 Prometheus AWS API 的 Amazon 託管服務現在可以在他們自己的參考中使用，即 Prometheus 的 Amazon 管理服務 API 參考。與 Prometheus 相容的 API 繼續記錄在 Amazon Prometheus 受管服務使用者指南中。 | 2024年2月7日 |
| 新增了用於工作區加密的客戶受管金鑰 | Amazon Managed Service for Prometheus 新增了客戶受管 | 2023 年 12 月 21 日 |

| | | |
|---|---|---|
| | 金鑰的支援，可用於工作區加密。如需詳細資訊，請參閱靜態加密。 | |
| 添加了新的權限 AmazonPrometheusFullAccess | 為AmazonPrometheusFullAccess受管政策新增許可，以支援為 Amazon EKS 叢集建立 AWS 受管收集器。 | 2023 年 11 月 26 日 |
| 添加了新的受管策略，AmazonPrometheusScraperServiceLinkedRolePolicy | 新增受管政策，AmazonPrometheusScraperServiceLinkedRolePolicy讓 AWS 受管收集器從 Amazon EKS 叢集收集指標。 | 2023 年 11 月 26 日 |
| 新增 AWS 受管理的收集器做為擷取方法 | Amazon Managed Service for Prometheus 新增支援 AWS 個受管收集器。 | 2023 年 11 月 26 日 |
| 增加支援與 Amazon Managed Grafana 整合 | Amazon Managed Service for Prometheus 新增支援與 Amazon Managed Grafana 警示整合。 | 2022 年 11 月 23 日 |
| 添加了新的權限 AmazonPrometheusConsoleFullAccess | 已新增AmazonPrometheusConsoleFullAccess受管理原則的新權限，以支援記錄 CloudWatch 檔中的記錄警示管理員和量尺事件。 | 2022 年 10 月 24 日 |
| 增加 Amazon EKS 可觀測性解決方案。 | 適用於 Prometheus 的 Amazon 受管服務新增了使用可觀察性加速器的新解決方案 AWS 。如需詳細資訊，請參閱使用 AWS 可觀測性加速器。 | 2022 年 10 月 14 日 |

| | | |
|---|---|---|
| 增加支援整合至 Amazon EKS 成本監控。 | Amazon Managed Service for Prometheus 增加支援整合至 Amazon EKS 成本監控。如需詳細資訊，請參閱與 Amazon EKS 成本監控整合。 | 2022 年 9 月 22 日 |
| 在 Amazon 日誌中啟動了對警報管理器和標尺日 CloudWatch 誌的支持。 | 適用於 Prometheus 的 Amazon 託管服務啟動了對 Amazon 日誌中警報管理器和標尺錯誤日誌的支持。 CloudWatch 如需詳細資訊，請參閱 Amazon CloudWatch 日誌。 | 2022 年 9 月 1 日 |
| 已增加自訂儲存保留支援。 | Amazon Managed Service for Prometheus 透過修改該工作區的配額，為每個工作區新增自訂儲存保留支援。如需有關 Amazon Managed Service for Prometheus 配額的詳細資訊，請參閱服務配額。 | 2022 年 8 月 12 日 |
| 向 Amazon 添加了使用指標 CloudWatch。 | 適用於 Prometheus 的 Amazon 受管服務增加了將使用指標傳送到 Amazon 的支援。 CloudWatch如需詳細資訊，請參閱 Amazon CloudWatch 指標。 | 2022 年 5 月 6 日 |
| 增加支援歐洲 (倫敦) 區域。 | Amazon Managed Service for Prometheus 增加支援歐洲 (倫敦) 區域。 | 2022 年 5 月 4 日 |

| | | |
|---|---|---|
| Amazon Managed Service for Prometheus 一般可用，並新增支援規則和警示管理員。 | Amazon Managed Service for Prometheus 一般可用。這也支援規則和警示管理員。如需詳細資訊，請參閱記錄規則和警示規則及警示管理員和範本化。 | 2021 年 9 月 29 日 |
| 已新增標記支援。 | Amazon Managed Service for Prometheus 支援標記 Amazon Managed Service for Prometheus 工作區。 | 2021 年 9 月 7 日 |
| 增加啟用中序列和擷取速率配額。 | 啟用中序列配額已增加到 1,000,000，而擷取速率配額已增加到每秒 70,000 個範例。 | 2021 年 2 月 22 日 |
| Amazon Managed Service for Prometheus 預覽版本。 | 已核發 Amazon Managed Service for Prometheus 預覽。 | 2020 年 12 月 15 日 |

# AWS 詞彙表

如需最新的 AWS 術語，請參閱《AWS 詞彙表 參考》中的 AWS 詞彙表。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。