aws

使用者指南

EventBridge 排程器

EventBridge 排程器: 使用者指南

Table of Contents

什麼是亞馬遜 EventBridge 調度程序？

Amazon EventBridge Scheduler 是無伺服器排程器，可讓您從單一中央受管服務建立、執行和管理任務。 EventBridge 排程器具備高度擴充能力，可讓您排程數百萬個可叫用 270 項以上AWS服務的工作，以及 6,000 項以上的 API 作業。 EventBridge Scheduler 不需要佈建和管理基礎架構，或與多項服務整合，讓您能夠大規模交付排程並降低維護成本。

EventBridge Scheduler 可靠地提供您的工作，內建機制可根據下游目標的可用性調整排程。使用 EventBridge Scheduler，您可以使用循環模式的 cron 和速率運算式來建立排程，或設定一次性呼叫。您可以設定彈性的傳遞時間範圍、定義重試限制，以及設定失敗觸發程序的最長保留時間。

主題

EventBridge 排程器的主要功能

EventBridge Scheduler 提供下列重要功能，您可以使用這些功能來設定目標和調整排程。

- 範本化目標 — EventBridge 排程器支援範本化目標，以使用 Amazon SQS、Amazon SNS、Lambda 和執行常見的 API 操作 EventBridge。透過預先定義的目標，您可以使用 EventBridge 排程器主控台、 EventBridge 排程器 SDK 或AWS CLI.
- 通用目標 — S EventBridge cheduler 提供通用目標參數 (UTP)，可用來建立自訂觸發程序，以排程定位 270 項以上的AWS服務和 6,000 多個 API 作業。透過 UTP，您可以使用 EventBridge 排程器主控台、 EventBridge 排程器 SDK 或AWS CLI.
- 彈性的時間範圍 — S EventBridge cheduler 支援彈性的時間範圍，可讓您分散排程，並針對不需要精確排程目標呼叫的使用案例改善觸發器的可靠性。
- 重試 — EventBridge 排程器提供 at-least-once 事件傳遞至目標，表示至少有一個傳遞成功，並從目標回應。 EventBridge 排程器可讓您設定失敗工作排程的重試次數。 EventBridge 排程器會嘗試延遲重試失敗的工作，以改善排程的可靠性並確保目標可用。

存取 EventBridge 排程器

您可以透過 EventBridge 排程 EventBridge 器主控台、 EventBridge 排程器 SDK 或直接使用排 EventBridge 程器 API 來使用排程器。AWS CLI

設置 Amazon EventBridge 調度程序

您必須先完成下列步驟，才能使用「 EventBridge 排程器」。

主題

註冊成為 AWS

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。

建立 IAM 使用者

若要建立管理員使用者，請選擇下列其中一個選項。

選擇一種管理管理員的方式	到	By	您也可以
在 IAM Identity Center (建議)	使用短期憑證存取 AWS。 這與安全性最佳實務一致。有關最佳實務的資訊，請參閱 IAM 使用者指南中的 IAM 安全最佳實務。	請遵循 AWS IAM Identity Center 使用者指南的入門中的說明。	AWS IAM Identity Center 在《使用AWS Command Line Interface 者指南》中設定 AWS CLI 要使用的，以設定程式設計方式存取。
在 IAM 中 (不建議使用)	使用長期憑證存取 AWS。	請遵循 IAM 使用者指南中建立您的第一個 IAM 管理員使用者和使用者群組的說明。	請參閱 IAM 使用者指南中的管理 IAM 使用者的存取金鑰，設定程式設計存取。

使用受管政策

在上一個步驟中，您可以使用登入資料設定 IAM 使用者以存取您的 AWS 資源。在大多數情況下，若要安全地使用 EventBridge Scheduler，建議您建立僅具有必要權限的個別使用者、群組或角色，以便使用 EventBridge Scheduler。 EventBridge Scheduler 針對常見使用案例支援下列受管理的原則。

- the section called “AmazonEventBridgeSchedulerFullAccess”— 使用主控台和 API 授予 EventBridge 排程器的完整存取權。
- the section called “AmazonEventBridgeSchedulerReadOnlyAccess”— 授與 EventBridge 排程器的唯讀存取權。

您可以將這些受管政策附加到 IAM 主體，方式與上一步驟中附加AdministratorAccess政策的方式相同。如需使用身分識別型 IAM 政策管理 EventBridge 排程器存取權的詳細資訊，請參閱。the section called “使用身分型政策”

設定執行角色

執行角色是 EventBridge 排程器為了代表您與其他人互動而假設 AWS 服務 的 IAM 角色。您可以將權限原則附加至此角色，以授與 EventBridge 排程器呼叫目標的存取權。

您也可以在使用主控台建立新排程時建立新的執行角色。如果您使用主控台， EventBridge Scheduler 會根據您選擇的目標，代表您建立具有權限的角色。當 EventBridge Scheduler 為您建立角色時，角色的信任原則包含條件金鑰，這些金鑰會限制哪些主體可以代表您擔任該角色。這樣可以防止潛在混淆的副安全問題。

下列步驟說明如何建立新的執行角色，以及如何授與 EventBridge 排程器存取權以叫用目標。本主題說明常用範本化目標的權限。如需為其他目標新增權限的資訊，請參閱the section called “使用範本化目標”。

若要使用建立執行角色 AWS CLI

1. 複製下列假設角色 JSON 原則，並將其儲存為本機`Scheduler-Execution-Role.json`。此信任原則允許 EventBridge 排程器代表您擔任該角色。

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

> **Important**
>
> T 在生產環境中設置執行角色，我們建議實施其他保護措施以防止混淆的副問題。如需詳細資訊和原則範例，請參閱the section called “預防混淆代理人”。

2. 在 AWS Command Line Interface (AWS CLI) 中，輸入下列指令以建立新角色。取代*`SchedulerExecutionRole`*為您要賦予此角色的名稱。

```
$ aws iam create-role --role-name SchedulerExecutionRole --assume-role-policy-
document file://Scheduler-Execution-Role.json
```

如果成功，您將看到以下輸出：

```
{
    "Role": {
        "Path": "/",
        "RoleName": "Scheduler-Execution-Role",
        "RoleId": "BR1L2DZK3K4CTL5ZF9EIL",
        "Arn": "arn:aws:iam::123456789012:role/SchedulerExecutionRole",
        "CreateDate": "2022-03-10T18:45:01+00:00",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "scheduler.amazonaws.com"
                    },
                    "Action": "sts:AssumeRole"
                }
            ]
        }
    }
}
```

3. 若要建立允許「 EventBridge 排程器」呼叫目標的新原則，請選擇下列其中一個通用目標。複製 JSON 權限原則，並將其儲存為`.json`檔案在本機。

   Amazon SQS – SendMessage

   以下內容允許 EventBridge 排程器對帳戶中的所有 Amazon SQS 佇列呼叫`sqs:SendMessage`動作。

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

```
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Amazon SNS – Publish

以下內容允許 EventBridge 排程器`sns:Publish`針對您帳戶中的所有 Amazon SNS 主題呼叫動作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sns:Publish"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Lambda – Invoke

以下內容可讓 EventBridge 排程器呼叫您帳戶中所有 Lambda 函數的`lambda:InvokeFunction`動作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

4. 執行下列命令以建立新的權限原則。取代*PolicyName*為您要提供此原則的名稱。

```
$ aws iam create-policy --policy-name PolicyName --policy-document file://
PermissionPolicy.json
```

如果成功，你會看到下面的輸出。請注意政策 ARN。您在下一個步驟中使用此 ARN 將原則附加到我們的執行角色。

```
{
    "Policy": {
        "PolicyName": "PolicyName",
        "CreateDate": "2022-03-015T19:31:18.620Z",
        "AttachmentCount": 0,
        "IsAttachable": true,
        "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
        "DefaultVersionId": "v1",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:policy/PolicyName",
        "UpdateDate": "2022-03-015T19:31:18.620Z"
    }
}
```

5. 執行下列命令，將原則附加至您的執行角色。取代*your-policy-arn*為您在上一個步驟中建立之原則的 ARN。*SchedulerExecutionRole*以執行角色的名稱取代。

```
$ aws iam attach-role-policy --policy-arn your-policy-arn --role-
name SchedulerExecutionRole
```

作`attach-role-policy`業不會在命令列上傳回應。

設定目標

在建立「 EventBridge 排程器」排程之前，您至少需要一個目標來呼叫排程。您可以使用現有 AWS 資源，也可以建立新資源。下列步驟說明如何使 AWS CloudFormation用建立新的標準 Amazon SQS 佇列。

若要建立新的 Amazon SQS 佇列

1. 複製以下 JSON AWS CloudFormation 範本並將其儲存為本機`SchedulerTargetSQS.json`。

```
{
   "AWSTemplateFormatVersion": "2010-09-09",
   "Resources": {
      "MyQueue": {
         "Type": "AWS::SQS::Queue",
         "Properties": {
            "QueueName": "MyQueue"
             }
         }
      },
   "Outputs": {
      "QueueName": {
         "Description": "The name of the queue",
         "Value": {
            "Fn::GetAtt": [
               "MyQueue",
               "QueueName"
            ]
         }
      },
      "QueueURL": {
         "Description": "The URL of the queue",
         "Value": {
            "Ref": "MyQueue"
         }
      },
      "QueueARN": {
         "Description": "The ARN of the queue",
         "Value": {
            "Fn::GetAtt": [
               "MyQueue",
               "Arn"
            ]
         }
      }
   }
}
```

2. 從中 AWS CLI執行下列命令以從`Scheduler-Target-SQS.json`範本建立 AWS CloudFormation 堆疊。

```
$ aws cloudformation create-stack --stack-name Scheduler-Target-SQS --template-body
 file://Scheduler-Target-SQS.json
```

如果成功，您將看到以下輸出：

```
{
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/Scheduler-
Target-SQS/1d2af345-a121-12eb-abc1-012e34567890"
}
```

3. 執行下列命令以檢視 AWS CloudFormation 堆疊的摘要資訊。此資訊包括堆疊的狀態以及範本中指定的輸出。

```
$ aws cloudformation describe-stacks --stack-name Scheduler-Target-SQS
```

如果成功，命令會建立 Amazon SQS 佇列並傳回下列輸出：

```
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/
Scheduler-Target-SQS/1d2af345-a121-12eb-abc1-012e34567890",
            "StackName": "Scheduler-Target-SQS",
            "CreationTime": "2022-03-17T16:21:29.442000+00:00",
            "RollbackConfiguration": {},
            "StackStatus": "CREATE_COMPLETE",
            "DisableRollback": false,
            "NotificationARNs": [],
            "Outputs": [
                {
                    "OutputKey": "QueueName",
                    "OutputValue": "MyQueue",
                    "Description": "The name of the queue"
                },
                {
                    "OutputKey": "QueueARN",
                    "OutputValue": "arn:aws:sqs:us-west-2:123456789012:MyQueue",
                    "Description": "The ARN of the queue"
                },
                {
                    "OutputKey": "QueueURL",
```

```
                    "OutputValue": "https://sqs.us-
west-2.amazonaws.com/123456789012/MyQueue",
                    "Description": "The URL of the queue"
                }
            ],
            "Tags": [],
            "EnableTerminationProtection": false,
            "DriftInformation": {
                "StackDriftStatus": "NOT_CHECKED"
            }
        }
    ]
}
```

在本指南稍後，您將使用的值QueueARN將佇列設定為 EventBridge 排程器的目標。

後續步驟？

完成設定步驟後，請使用入門指南建立第一個 EventBridge 排程器排程器並叫用目標。

開始使用 EventBridge 排程器

本主題說明建立新的 EventBridge 排程器排程。您可以使用 EventBridge 排程器主控台、AWS Command Line Interface (AWS CLI) 或AWS開發套件來建立具有範本化 Amazon SQS 目標的排程。然後，您將設定記錄、設定重試次數，以及設定失敗工作的保留時間上限。建立排程之後，您將驗證排程是否成功叫用目標，並將訊息傳送至目標佇列。

> **Note**
>
> 若要遵循本指南，建議您使用中所述的最低必要許可設定 IAM 使用者the section called “使用身分型政策”。建立並設定使用者之後，請執行下列命令來設定您的存取認證。您需要您的存取金鑰 ID 和秘密存取金鑰，才能設定AWS CLI.
>
> ```
> $ aws configure
> AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
> AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
> Default region name [None]: us-west-2
> Default output format [None]: json
> ```
>
> 如需有關不同方式設定認證的詳細資訊，請參閱第 2 版AWS Command Line Interface使用者指南中的組態設定和優先順序。

主題

先決條件

在嘗試本節中的步驟之前，您必須執行下列動作：

- 完成中所述的任務 設定

使用 EventBridge 排程器主控台建立排程

使用主控台建立新排程

1. 登入AWS Management Console，然後選擇下列連結以開啟 EventBridge 主控台的 [EventBridge 排程器] 區段：https://us-west-2.console.aws.amazon.com/scheduler/home?region=us-west-2#home

   > **Note**
   >
   > 您可以使用AWS 區域AWS Management Console的區域選擇器切換您的。

2. 在排程頁面上，選擇建立排程。
3. 在指定排程詳細資訊頁面的排程名稱和描述區段中，執行以下動作：

   a. 在排程名稱中，輸入排程的名稱，例如：**`MyTestSchedule`**

   b. 在「說明-選用」中，輸入排程的說明。例如：**`My first schedule`**。

   c. 在「排程」群組中，從下拉式選項中選擇排程群組。如果您之前沒有建立任何排程群組，您可以選擇排程的`default`群組。若要建立新的排程群組，請在主控台說明中選擇 [建立您自己的排程] 連結。您可以使用排程群組，為不同群組的排程加上標籤。

4. 在「排程樣式」區段中，執行下列操作：

   a. 對於「複本」，請選擇下列其中一個陣列選項。組態選項會根據您選取的樣式而變更。

      - 一次性排程 — 一次性排程只會在您指定的日期和時間呼叫目標一次。

        在「日期和時間」中，以`YYYY/MM/DD`格式輸入有效日期。然後，指定 24 小時`hh:mm`格式的時間戳記。最後，從下拉選項中選擇一個時區。

      - 週期性排程 — 週期性排程會以您使用cron運算式或速率運算式指定的速率叫用目標。

        選擇以 CRON 為基礎的排程，以使用運算式來設定排程cron。o 使用費率表示式，選擇以比率為基準的排程，然後在「值」中輸入正數，然後從下拉式選項中選擇「單位」。

      如需使用 cron 和速率運算式的詳細資訊，請參閱排程類型。

   b. 對於彈性時間範圍，請選擇關閉以關閉選項，或從下拉式清單中選擇其中一個預先定義的時間範圍。例如，如果您選擇 15 分鐘並設定週期性排程，每小時調用目標一次，則排程會在每小時一開始的 15 分鐘內執行。

5.

> **Note**
>
> 彈性時間範圍功能不適用於一次性排程。

如果您在上一個步驟中選擇「週期性」排程，請在「時間範圍」區段中指定時區，並選擇性地設定排程的開始日期和時間，以及結束日期和時間。沒有開始日期的循環排程將在創建並可用後立即開始。沒有結束日期的週期性排程將繼續無限期地調用它的目標。

6. 選擇下一步。
7. 在 「選取目標」 頁面上，執行下列動作：

   a. 選取範本化目標，然後選擇目標 API。在此範例中，我們將選擇 Amazon SQS **`SendMessage`** 範本化目標。

   b. 在此SendMessage段落`arn:aws:sqs:`*`us-west-2`*`:`*`123456789012`*`:`*`TestQueue`*中，對於 SQS 佇列，請從下拉式清單中選擇現有的 Amazon SQS 佇列 ARN。若要建立新佇列，請選擇「建立新 SQS 佇列」以導覽至 Amazon SQS 主控台。完成建立佇列之後，請返回「 EventBridge 排程器」主控台並重新整理下拉式清單。您的新佇列 ARN 隨即出現，您可以選取該佇列。

   c. 針對 Target，輸入您希望 EventBridge 排程器傳遞至目標的承載。在此範例中，我們會將下列訊息傳送至目標佇列：**`Hello, it's EventBridge Scheduler.`**

8. 選擇 [下一步]，然後在 [設定]-選用頁面上執行下列動作：
9.

   a. 在「排程狀態」區段中，對於「啟用排程」，使用開關切換功能開啟或關閉。依預設，「 EventBridge 排程器」會啟用您的排程。

   b. 在「排程完成後的動作」區段中，設定排程器在 EventBridge 排程完成後採取的動作：

   - 如果您要自動刪除排程，請選擇「刪除」。對於一次性排程，這會在排程呼叫目標一次之後發生。對於週期性排程，這會在排程的最後一次計劃叫用之後發生。如需自動刪除的更多資訊，請參閱the section called “排程完成後刪除”。
   - 如果您不希望「排程器」在 EventBridge 排程完成後採取任何動作，請選擇「無」，或不選擇值。

   c. 在 [重試原則和無效字母佇列 (DLQ)] 區段中，針對 [重試] 原則，開啟 [重試] 以設定排程的重試原則。使用重試原則時，如果排程無法呼叫其目標， EventBridge 排程器會重新執行排程。一旦設定此功能，您就必須設定排程的最長保留時間和重試次數。

d. 對於事件的最長保留時間-選用，請輸入 EventBridge 排程器必須保留未處理事件的最大小時數和最小時數。

> **Note**
>
> 最大值為 24 小時。

e. 針對重試次數上限，輸入目標傳回錯誤時， EventBridge 排程器重試排程的次數上限。

> **Note**
>
> 最大值為重試 185 次。

f. 對於無效字母佇列 (DLQ)，請從下列選項中選擇：

- 無 — 如果您不想配置 DLQ，請選擇此選項。
- 在我的AWS帳戶中選取一個 Amazon SQS 佇列做為 DLQ — 選擇此選項，然後從下拉式清單中選取佇列 ARN，設定 DLQ 與您要建立排程的AWS 帳戶相同。
- 將其他AWS帳戶中的 Amazon SQS 佇列指定為 DLQ — 如果佇列位於另一個佇列中，請選擇此選項，然後將佇列設定的 ARN 輸入為 DLQ。AWS 帳戶您必須輸入佇列的正確 ARN，才能使用此選項。

g. 在 [加密] 區段中，選擇 [自訂加密設定 (進階)] 以使用客戶管理的 KMS 金鑰來加密您的目標輸入。如果選擇此選項，請輸入現有的 KMS 金鑰 ARN，或選擇 [建立 AWS KMS 金鑰] 以導覽至AWS KMS主控台。如需 EventBridge Scheduler 如何加密靜態資料的詳細資訊，請參閱the section called “靜態加密”。

h. 對於「權限」，請選擇「使用現有角色」，然後從下拉式清單中選取您在設定程序期間建立的角色。您也可以選擇前往 IAM 主控台來建立新角色。

如果您想要 [EventBridge 排程器] 為您建立新的執行角色，請改為選擇 [為此排程建立新角色]。接著輸入角色名稱。如果您選擇此選項，「 EventBridge 排程器」會將範本化目標所需的必要權限新增至角色。

10. 選擇下一步。
11. 在檢閱和建立排程頁面上，檢閱排程的詳細資訊。在每個區段中選擇編輯，即可返回該步驟並編輯其詳細資訊。
12. 選擇 [建立排程] 以完成建立新排程。您可以在排程頁面檢視新建立和現有的排程。在狀態欄底下，確認您的新排程狀態為已啟用。

13. 若要確認您的排程是否呼叫 Amazon SQS 目標，請開啟 Amazon SQS 主控台並執行下列動作：

    a. 從「佇列」清單中選擇目標佇列。
    b. 選擇傳送及接收訊息。
    c. 在 [傳送及接收訊息] 頁面的 [接收訊息] 下，選擇 [輪詢郵件]，以擷取排程傳送至目標佇列的測試訊息。

使用建立排程 AWS CLI

下列範例顯示如何使用命AWS CLI令`create-schedule`建立具有範本化 Amazon SQS 目標的 EventBridge 排程器排程。以您的資訊取代下列參數的預留位置值：

- -name — 輸入排程的名稱。
- RoleArn— 輸入您要與排程產生關聯之執行角色的 ARN。
- 收銀 — 輸入目標的 ARN。在這種情況下，目標是 Amazon SQS 佇列。
- 輸入 — 輸入 EventBridge 排程器傳送至目標佇列的訊息。

```
$ aws scheduler create-schedule --name sqs-templated-schedule --schedule-expression
 'rate(5 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn": "QUEUE_ARN", "Input": "TEST_PAYLOAD" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

使用 EventBridge 排程器 SDK 建立排程

在下列範例中，您可以使用 EventBridge 排程器 SDK 建立具有範本化 Amazon SQS 目標的 EventBridge 排程器排程。

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

flex_window = { "Mode": "OFF" }

sqs_templated = {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<QUEUE_ARN>",
```

```
    "Input": "Message for scheduleArn: '<aws.scheduler.schedule-arn>', scheduledTime:
 '<aws.scheduler.scheduled-time>'"
}

scheduler.create_schedule(
    Name="sqs-python-templated",
    ScheduleExpression="rate(5 minutes)",
    Target=sqs_templated,
    FlexibleTimeWindow=flex_window)
```

Example Java 開發套件

```
package com.example;

import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.scheduler.SchedulerClient;
import software.amazon.awssdk.services.scheduler.model.*;


public class MySchedulerApp {

    public static void main(String[] args) {

        final SchedulerClient client = SchedulerClient.builder()
                .region(Region.US_WEST_2)
                .build();

        Target sqsTarget = Target.builder()
                .roleArn("<ROLE_ARN>")
                .arn("<QUEUE_ARN>")
                .input("Message for scheduleArn: '<aws.scheduler.schedule-arn>',
 scheduledTime: '<aws.scheduler.scheduled-time>'")
                .build();

        CreateScheduleRequest createScheduleRequest = CreateScheduleRequest.builder()
                .name("<SCHEDULE NAME>")
                .scheduleExpression("rate(10 minutes)")
                .target(sqsTarget)
                .flexibleTimeWindow(FlexibleTimeWindow.builder()
                        .mode(FlexibleTimeWindowMode.OFF)
                        .build())
                .build();
```

```
        client.createSchedule(createScheduleRequest);
        System.out.println("Created schedule with rate expression and an Amazon SQS
 templated target");
    }
}
```

後續步驟？

- 如需有關使用主控台或 EventBridge Scheduler SDK 管理排程的詳細資訊，請參閱管理排程。AWS CLI
- 如需有關如何設定範本化目標以及如何使用通用目標參數的詳細資訊，請參閱管理目標。
- 如需有關 EventBridge 排程器資料類型和 API 作業的詳細資訊，請參閱EventBridge 排程器 API 參考。

EventBridge 排程器上的排程類型

下列主題說明 Amazon EventBridge Scheduler 支援的不同排程類型，以及排程 EventBridge 器如何處理日光節約時間，以及在不同時區中排程。設定排程時，您可以從三種排程型態中進行選擇：以費率為基準、以 Cronn 為基準及一次性排程。

以費率為基準與以客戶為基準的排程都是重複產生排程。您可以針對要設定的排程類型，使用排程運算式來設定每個週期性排程類型，並指定 EventBridge Scheduler 評估運算式的時區。

一次性排程是只呼叫目標一次的排程。您可以透過指定「排程器」評估 EventBridge 排程的時間、日期和時區來設定一次性排程。

> **Note**
>
> EventBridge 排程器上的所有排程類型都會以 60 秒精確度呼叫其目標。這意味著，如果您將計劃設置為在運行時`1:00`，它將在`1:00:00`和之間調用目標 API `1:00:59`。

您可以使用下列各節來瞭解如何設定每個週期性排程類型的排程運算式，以及如何在 EventBridge Scheduler 上設定一次性排程。

主題

以速率為基礎的

以費率為基準的排程會在您為排程指定的開始日期之後開始，並以您定義的一般費率執行，直到排程的結束日期為止。您可以使用以費率為基礎的排程來設定最常見的經常性排程使用案例。例如，如果您希望排程每 15 分鐘、每兩小時或每五天呼叫一次目標，則可以使用以費率為基準的排程來達成此目標。您可以使用費率運算式來設定以比率為基礎的排程。

使用以費率為基礎的明細表時，您可以使用`StartDate`性質來設定明細表的第一次出現。如果您未提供以費率`StartDate`為基準的排程，則排程會立即開始呼叫目標。

費率運算式有兩個必填欄位，並以空格分隔，如下所示。

語法

```
rate(value unit)
```

value

正數。

單位

您希望排程呼叫其目標的時間單位。

有效輸入：minutes| hours | days

範例

下列範例顯示如何將費率運算式與AWS CLIcreate-schedule命令搭配使用，以設定以速率為基礎的排程。此範例會建立每五分鐘執行一次的排程，並使用範本化的SqsParameters目標類型將訊息傳遞至 Amazon SQS 佇列。

由於此範例並未設定--start-date參數的值，因此排程會在您建立並啟動它之後立即開始呼叫其目標。

```
$ aws scheduler create-schedule --schedule-expression 'rate(5 minutes)' --
name schedule-name \
--target '{"RoleArn": "role-arn", "Arn": "QUEUE_ARN", "Input": "TEST_PAYLOAD" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

以 Cron 為基礎的排程

Cron 運算式會建立精細的週期性排程，並在您選擇的特定時間執行。 EventBridge Scheduler 支援以世界協調時間 (UTC) 或您在建立排程時指定的時區設定以 Cron 為基礎的排程。使用以 Cron 為基礎的排程，您可以更好地控制排程執行的時間和頻率。當您需要排程器的費率運算式不支援的自訂週期排程時，請使用以 Cron 為基礎的 EventBridge 排程。例如，您可以建立在上午 8:00 執行的以 Cron 為基礎的排程。太平洋標準時間為每個月的第一個星期一。您可以使用 cron 運算式來設定以 C ron 為基礎的排程。

Cron 運算式包含五個以空格分隔的必要欄位：分鐘、小時 day-of-month、月 day-of-week，以及一個選擇性欄位 (年份)，如下所示。

語法

```
cron(minutes hours day-of-month month day-of-week year)
```

欄位	Values (數值)	Wildcards (萬用字元)
分鐘	0-59	, - * /
小時	0-23	, - * /
D ay-of-month	1-31	, - * ? / L W
月	1-12 或 JAN-DEC	, - * /
D ay-of-week	1-7 或 SUN-SAT	, - * ? L #
年	1970-2199	, - * /

萬用字元

- , (逗號) 萬用字元包含額外的值。在 Month (月) 欄位，JAN、FEB、MAR 包括 January (一月)、February (二月) 與 March (三月)。
- - (破折號) 萬用字元用於指定範圍。在 Day (日) 欄位，1-15 包含指定月份的 1 至 15 號。
- * (星號) 包含欄位中所有的值。在 Hours (小時) 欄位，* 包含每個小時。您不能在 D ay-of-month 和 D ay-of-week 欄位中使用 *。若您在其中一個欄位使用它，您必須在另一個欄位使用 ?。
- / (斜線) 萬用字元用於指定增量。在 Minutes (分鐘) 欄位，您可以輸入 1/10 指定每十分鐘的間隔，從小時的第一分鐘開始 (例如第 11、第 21、第 31 分鐘等)。
- ? (問號) 萬用字元用於表示不限定任何一個。在 D 字ay-of-month 段中，您可以輸入 7，如果一周中的任何一天可以接受，則可以輸入？ 在 D 字ay-of-week段中。
- D ay-of-month 或 D ay-of-week 欄位中的 L 萬用字元會指定月份或週的最後一天。
- D ay-of-month 欄位中的**W**萬用字元指定工作日。在 D ay-of-month 欄位中，**3W**指定最接近月份第三天的星期幾。

- D ay-of-week 欄位中的 # 萬用字元會指定一個月內星期中指定日期的特定執行個體。例如，3#2 代表則該月的第二個星期二：3 是指星期二，因為它是每週的第三天，2 指的是一個月內該類型的第二天。

> **Note**
>
> 如果您使用 '#' 字元，則只能在 day-of-week 欄位中定義一個運算式。例如："3#1,6#3" 是無效的，因為它被轉譯為兩個表達式。

範例

下列範例顯示如何使用 cron 運算式搭配命AWS CLIcreate-schedule令來設定 cron 型排程。此範例會建立一個排程，該排程在 2022 年至 2023 年期間每個月最後一個星期五上午 10:15 UTC+0 執行，並使用範本化的目標類型將訊息傳遞至 Amazon SQS 佇列。SqsParameters

```
$ aws scheduler create-schedule --schedule-expression "cron(15 10 ? * 6L 2022-2023)" --
name schedule-name \
--target '{"RoleArn": "role-arn", "Arn": "QUEUE_ARN", "Input": "TEST_PAYLOAD" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

一次性排程

一次性排程只會在您使用有效日期和時間戳記指定的日期和時間呼叫目標一次。 EventBridge 排程器支援在世界協調時間 (UTC) 或您在建立排程時指定的時區中排程。

> **Note**
>
> 一次性排程在完成執行並呼叫其目標後，仍會計入您的帳戶配額。我們建議您在完成執行後刪除一次性排程。

您可以使用 at 運算式設定一次性排程。at 運算式包含您希望 EventBridge 排程器呼叫排程的日期和時間，如下所示。

語法

```
at(yyyy-mm-ddThh:mm:ss)
```

當您設定一次性排程時，排 EventBridge 程器會忽略EndDate您為排程所指定的，StartDate而且會忽略您指定的

範例

下列範例顯示如何使用 at 運算式搭配命AWS CLIcreate-schedule令來設定一次性排程。此範例會建立在 2022 年 11 月 20 日 UTC-8 下午 1 點執行一次的排程，並使用範本化SqsParameters的目標類型將訊息傳遞至 Amazon SQS 佇列。

```
$ aws scheduler create-schedule --schedule-expression "at(2022-11-20T13:00:00)" --
name schedule-name \
--target '{"RoleArn": "role-arn", "Arn": "QUEUE_ARN", "Input": "TEST_PAYLOAD" }' \
--schedule-expression-timezone "America/Los_Angeles"
--flexible-time-window '{ "Mode": "OFF"}'
```

EventBridge 排程器上的時區

EventBridge 排程器支援在您指定的任何時區中設定 cron 型和一次性排程。 EventBridge 排程器會使用由網際網路編號指派授權單位 (IANA) 維護的時區資料庫。

透過AWS CLI，您可以設定要 EventBridge Scheduler 使用--schedule-expression-timezone參數評估排程的時區。例如，以下命令會建立以 Cron 為基礎的排程，以便在美洲/紐約每天早上 8:30 叫用範本化的 Amazon SQS SendMessage 目標。

```
$ aws scheduler create-schedule --schedule-expression "cron(30 8 * * ? *)" --name
 schedule-in-est \
    --target '{"RoleArn": "role-arn", "Arn": "QUEUE_ARN", "Input": "This schedule runs
 in the America/New_York time zone." }' \
    --schedule-expression-timezone "America/New_York"
    --flexible-time-window '{ "Mode": "OFF"}'
```

EventBridge 排程器上的日光節約時間

EventBridge 排程器會自動調整您的日光節約時間排程。當時間在 Spring 中向前移動時，如果 cron 表達式落在不存在的日期和時間上，則會跳過您的計劃調用。當秋季時間向後移動時，您的排程只會執行一次，而不會重複呼叫。以下調用通常發生在指定的日期和時間。

EventBridge 排程器會根據您在建立排程時指定的時區來調整排程。如果您在 America/New_York 中設定排程，則您的排程會在該時區的時間變更時進行調整，而當西海岸時間變更時，美洲/洛杉磯的排程會在三小時後調整。

對於以費率`days`為基礎的排程，例如`rate(1 days)`，`days`代表時鐘 24 小時的持續時間。這表示當日光節約時間導致一天縮短至 23 小時，或延長至 25 小時， EventBridge Scheduler 仍會在排程最後一次呼叫後 24 小時評估比率運算式。

> **Note**
>
> 根據當地的規則和規定，某些時區不會遵守夏令時。如果您在未遵守日光節約時間的時區中建立 EventBridge 排程，Scheduler 不會調整您的排程。節省日光的時間調整不適用於世界協調時間 (UTC) 的排程。

範例

假設您在美國/洛杉磯使用下列 cron 運算式建立排程的案例：。`cron(30 2 * * ? *)`此排程會在指定時區的每天上午 2:30 執行。

- 向前春天 — 當春季時間從上午 1:59 向前移至凌晨 3:00 時， EventBridge 排程器會略過當天的排程呼叫，並在次日恢復正常執行排程。
- 倒退 — 當秋季時間從上午 2:59 向後移至凌晨 2:00 時， EventBridge 排程器只會在工作班次發生前的上午 2:30 執行排程一次，但不會在時間偏移之後的上午 2:30 再次重複排程呼叫。

管理排程

排程是您使用 Amazon EventBridge 排程器建立、設定和管理的主要資源。

每個排程都有一個排程運算式，可決定排程執行的時間和頻率。 EventBridge 排程器支援三種排程類型：費率、排程和一次性排程。如需不同排程類型的詳細資訊，請參閱排程類型。

建立排程時，您可以設定要呼叫之排程的目標。Target 是每次排程執行時， EventBridge 排程器都會代表您呼叫的 API 作業。 EventBridge Scheduler 支援兩種類型的目標：範本化目標會在核心服務群組間呼叫通用 API 作業，以及可用來呼叫超過 270 個服務的 6,000 項作業的通用目標參數 (UTP)。如需設定目標的詳細資訊，請參閱管理目標。

您可以使用兩種主要機制：重試原則和無效字母佇列 (DL Q)，設定 EventBridge 排程器無法順利將事件傳遞至目標時，如何處理失敗。重試原則決定 EventBridge 排程器必須重試失敗事件的次數，以及保留未處理事件的時間長度。DLQ 是標準的 Amazon SQS 佇列 EventBridge 排程器，用於在重試政策用盡後將失敗事件傳遞至。您可以使用 DLQ 來疑難排程或其下游目標的問題。如需有關的更多資訊，請參閱the section called “設定無效字母佇列”。

在本節中，您可以找到使用主控台AWS CLI和 EventBridge 排程器 SDK 管理 EventBridge 排程器排程的範例。

主題

變更排程狀態

EventBridge 排程器排程有兩種狀態：已啟用和停用。下列範例會使`UpdateSchedule`用停用每五分鐘觸發一次並叫用 Lambda 目標的排程。

使用時`UpdateSchedule`，您必須提供所有必要的參數。 EventBridge 排程器會以您提供的資訊取代您的排程。如果您未指定先前設定的參數，則預設值為`null`。

Example AWS CLI

```
$ aws scheduler update-schedule --name lambda-universal --schedule-expression 'rate(5
 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"arn:aws:scheduler:::aws-sdk:lambda:invoke"
 "Input": "{\"FunctionName\":\"arn:aws:lambda:REGION:123456789012:function:HelloWorld
\",\"InvocationType\":\"Event\",\"Payload\":\"{\\\"message\\\":\\\"testing function\\
\"}\"}" }' \
--flexible-time-window '{ "Mode": "OFF"}' \
--state DISABLED
```

```
{
    "ScheduleArn": "arn:aws:scheduler:us-west-2:123456789012:schedule/default/lambda-
universal"
}
```

下列範例使用 Python 開發套件和UpdateSchedule作業來停用使用範本化目標鎖定 Amazon SQS 的排程。

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

sqs_templated = {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<QUEUE_ARN>",
    "Input": "{}"}

flex_window = { "Mode": "OFF" }

scheduler.update_schedule(Name="your-schedule",
    ScheduleExpression="rate(5 minutes)",
    Target=sqs_templated,
    FlexibleTimeWindow=flex_window,
    State='DISABLED')
```

設定彈性時間範圍

當您使用彈性時間範圍設定排程時，EventBridge Scheduler 會在您設定的時間範圍內呼叫目標。這在不需要精確排程呼叫目標的情況下非常有用。設定彈性的時間範圍可透過分散目標呼叫來改善排程的可靠性。

例如，如果您為每小時執行的排程設定 15 分鐘彈性時間範圍，則會在排定時間後的 15 分鐘內呼叫目標。以下AWS CLI和 EventBridge Scheduler SDK 範例用於`UpdateSchedule`為每小時執行一次的排程設定 15 分鐘彈性時間範圍。

> **Note**
>
> 您必須指定是否要設定彈性時間範圍。如果您不想設定此選項，請指定`OFF`。如果將值設定為`FLEXIBLE`，則必須指定排程執行的最大時段。

Example AWS CLI

```
$ aws scheduler update-schedule --name lambda-universal --schedule-expression 'rate(1
 hour)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"arn:aws:scheduler:::aws-sdk:lambda:invoke"
 "Input": "{\"FunctionName\":\"arn:aws:lambda:REGION:123456789012:function:HelloWorld
\",\"InvocationType\":\"Event\",\"Payload\":\"{\\\"message\\\":\\\"testing function\\
\"}\"}" }' \
--flexible-time-window '{ "Mode": "FLEXIBLE", "MaximumWindowInMinutes": 15} \
```

```
{
    "ScheduleArn": "arn:aws:scheduler:us-west-2:123456789012:schedule/lambda-universal"
}
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

sqs_templated = {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<QUEUE_ARN>",
    "Input": "{}"}
```

```
flex_window = { "Mode": "FLEXIBLE", "MaximumWindowInMinutes": 15}

scheduler.update_schedule(Name="your-schedule",
    ScheduleExpression="rate(1 hour)",
    Target=sqs_templated,
    FlexibleTimeWindow=flex_window)
```

設定排程的無效字母佇列

亞馬遜 EventBridge 排程器使用 Amazon 簡單佇列服務支援無效字母佇列 (DLQ)。當排程無法叫用其目標時，S EventBridge cheduler 會將包含叫用詳細資訊以及從目標接收到的任何回應傳送至您指定的 Amazon SQS 標準佇列的 JSON 承載。

下列主題將此 JSON 稱為無效字母事件。無效字母事件可讓您疑難排解排程或目標的問題。如果您為排程設定重試原則， EventBridge Scheduler 會傳送已耗盡您設定的重試次數上限的無效字母事件。

下列主題說明如何將 Amazon SQS 佇列設定為排程的 DLQ、設定將訊息傳遞至 Amazon SQS 所需的許可 EventBridge 排程器，以及從 DLQ 接收無效字母事件。

主題

- 建立 Amazon SQS 佇列
- 設定執行角色權限
- 指定一個無效字母佇列
- 擷取無效字母事件

建立 Amazon SQS 佇列

在為排程設定 DLQ 之前，您必須先建立標準 Amazon SQS 佇列。如需使用 Amazon SQS 主控台建立佇列的說明，請參閱 Amazon Simple Queue Service 開發人員指南中的建立 Amazon SQS 佇列。

> **Note**
>
> EventBridge 排程器不支援使用 FIFO 佇列作為排程的 DLQ。

使用下列AWS CLI指令建立標準佇列。

```
$ aws sqs create-queue --queue-name queue-name
```

如果成功，您會在輸出結果QueueURL中看到。

```
{
    "QueueUrl": "https://sqs.us-west-2.amazonaws.com/123456789012/scheduler-dlq-test"
}
```

建立佇列之後，請記下佇列 ARN。當您為 EventBridge 排程器排程指定 DLQ 時，您將需要 ARN。您可以在 Amazon SQS 主控台或使用get-queue-attributesAWS CLI指令來尋找佇列 ARN。

```
$ aws sqs get-queue-attributes --queue-url your-dlq-url --attribute-names QueueArn
```

如果成功，您將會在輸出結果中看到佇列的 ARN。

```
{
    "Attributes": {
        "QueueArn": "arn:aws:sqs:us-west-2:123456789012:scheduler-dlq-test"
    }
}
```

在下一節中，您將新增必要的許可到排程執行角色，以允許 EventBridge 排程器向 Amazon SQS 傳遞無效字母事件。

設定執行角色權限

若要讓 EventBridge 排程器將無效信件事件傳遞給 Amazon SQS，您的排程執行角色需要下列權限政策。如需有關將新權限原則附加至排程執行角色的詳細資訊，請參閱設定執行角色。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sqs:SendMessage"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> 如果您使用 EventBridge 排程器呼叫 Amazon SQS API 目標，您的排程執行角色可能已附加必要的許可。

在下一節中，您將會使用 EventBridge 排程器主控台，並為您的排程指定 DLQ。

指定一個無效字母佇列

若要指定 DLQ，請使用 [EventBridge 排程器] 主控台或更新現有排程，或建立新排程。AWS CLI

Console

若要使用主控台指定 DLQ

1. 登入AWS Management Console，然後選擇下列連結以開啟主 EventBridge 控台的「 EventBridge 排程器」區段：https://console.aws.amazon.com/scheduler/home
2. 在「 EventBridge 排程器」主控台上，建立新排程，或從排程清單中選擇要編輯的現有排程。
3. 在 [設定] 頁面上，對於無效字母佇列 (DLQ)，執行下列其中一個動作：
   - 選擇在我的AWS帳戶中選取 Amazon SQS 佇列做為 DLQ，然後從下拉式清單中選擇 DLQ 的佇列 ARN。
   - 選擇將其他AWS帳戶中的 Amazon SQS 佇列指定為 DLQ，然後輸入 DLQ 的佇列 ARN。如果您在其他AWS帳戶中選擇佇列， EventBridge Scheduler 主控台將無法在下拉式清單中顯示佇列 ARN。
4. 檢閱您的選取項目，然後選擇 [建立排程] 或 [儲存排程] 以完成 DLQ 的設定。
5. (選擇性) 若要檢視排程的 DLQ 詳細資料，請從清單中選擇排程名稱，然後選擇「排程詳細資訊」頁面上的無效字母佇列標籤。

AWS CLI

若要使用更新現有排程AWS CLI

- 使用`update-schedule`指令更新排程。將您先前建立的 Amazon SQS 佇列指定為 DLQ。指定您連接所需 Amazon SQS 許可的 IAM 角色 ARN 作為執行角色。以您的資訊取代所有其他預留位置值。

```
$ aws scheduler update-schedule --name existing-schedule \
    --schedule-expression 'rate(5 minutes)' \
    --target '{"DeadLetterConfig": {"Arn": "DLQ_ARN"}, "RoleArn": "ROLE_ARN",
 "Arn":"QUEUE_ARN", "Input": "Hello world!" }' \
    --flexible-time-window '{ "Mode": "OFF"}'
```

若要使用 DLQ 建立新排程AWS CLI

- 使用指create-schedule令建立明細表。將所有預留位置值取代為您的資訊。

```
$ aws scheduler create-schedule --name new-schedule \
    --schedule-expression 'rate(5 minutes)' \
    --target '{"DeadLetterConfig": {"Arn": "DLQ_ARN"}, "RoleArn": "ROLE_ARN",
 "Arn":"QUEUE_ARN", "Input": "Hello world!" }' \
    --flexible-time-window '{ "Mode": "OFF"}'
```

在下一節中，您將會AWS CLI使用從 DLQ 接收無效字母事件。

擷取無效字母事件

使用receive-message命令 (如下所示) 從 DLQ 擷取無效字母事件。您可以使用--max-number-of-messages屬性設定要擷取的訊息數目。

```
$ aws sqs receive-message --queue-url your-dlq-url --attribute-names All --message-attribute-names All --max-number-of-messages 1
```

如果成功，您會看到類似如下的輸出。

```
{
    "Messages": [
        {
            "MessageId": "2aeg3510-fe3a-4f5a-ab6a-6906560eaf7e",
            "ReceiptHandle": "AQEBkNKTdOMrWgHKPoITRBwrPoK3eCSZIcZwVqCY0BZ
+FfTcORFpopJbtCqj36VbBTlHreM8+qM/m5jcwqSlAlGmIJO/hYmMgn/
+dwIty9izE7HnpvRhhEyHxbeTZ5V05RbeasYaBdNyi9WLcnAHviDh6MebLXXNWoFyYNsxdwJuG0f/
w3htX6r3dxpXvvFNPGoQb8ihY37+u0gtsbuIwhLtUSmE8rbldEEwiUfi3IJ1zEZpUS77n/k1GWrMrnYg0Gx/
BuaLzOrFi2F738XI/
Hnh45uv3ca6OYwS1ojPQ1LtX2URg1haV5884FYlaRvY8jRlpCZabTkYRTZKSXG5KNgYZnHpmsspii6JNkjitYVFKPo0H91w
            "MD5OfBody": "07adc3fc889d6107d8bb8fda42fe0573",
```

```
            "Body": "{\"MessageBody\":\"Hello, world!",\"QueueUrl\":\"https://sqs.us-
west-2.amazonaws.com/123456789012/does-not-exist\"}",
            "Attributes": {
                "SenderId": "AROA2DZE3W4CTL5ZR7EIN:ff00212d8c453aaaae644bc6846d4723",
                "ApproximateFirstReceiveTimestamp": "1652499058144",
                "ApproximateReceiveCount": "2",
                "SentTimestamp": "1652490733042"
            },
            "MD5OfMessageAttributes": "f72c1d78100860e00403d849831d4895",
            "MessageAttributes": {
                "ERROR_CODE": {
                    "StringValue": "AWS.SimpleQueueService.NonExistentQueue",
                    "DataType": "String"
                },
                "ERROR_MESSAGE": {
                    "StringValue": "The specified queue does not exist for this wsdl
 version.",
                    "DataType": "String"
                },
                "EXECUTION_ID": {
                    "StringValue": "ad06616e51cdf74a",
                    "DataType": "String"
                },
                "EXHAUSTED_RETRY_CONDITION": {
                    "StringValue": "MaximumEventAgeInSeconds",
                    "DataType": "String"
                }
                "IS_PAYLOAD_TRUNCATED": {
                    "StringValue": "false",
                    "DataType": "String"
                },
                "RETRY_ATTEMPTS": {
                    "StringValue": "0",
                    "DataType": "String"
                },
                "SCHEDULED_TIME": {
                    "StringValue": "2022-05-14T01:12:00Z",
                    "DataType": "String"
                },
                "SCHEDULE_ARN": {
                    "StringValue": "arn:aws:scheduler:us-west-2:123456789012:schedule/
DLQ-test",
                    "DataType": "String"
                },
```

```
                "TARGET_ARN": {
                    "StringValue": "arn:aws:scheduler:::aws-sdk:sqs:sendMessage",
                    "DataType": "String"
                }
            }
        }
    ]
}
```

請注意無效字母事件中的下列屬性，以協助您識別和疑難排解目標 inovtion 失敗的可能原因。

- **ERROR_CODE**— 包含 EventBridge 排程器從目標服務 API 接收的錯誤代碼。在上述範例中，Amazon SQS 傳回的錯誤碼為`AWS.SimpleQueueService.NonExistentQueue`。如果排程因為 EventBridge 排程器問題而無法叫用目標，您將會看到下列錯誤碼：`AWS.Scheduler.InternalServerError`。
- **ERROR_MESSAGE**— 包含 EventBridge 排程器從目標服務 API 接收的錯誤訊息。在上述範例中，Amazon SQS 傳回的錯誤訊息為`The specified queue does not exist for this wsdl version`。如果排程因為 EventBridge 排程器問題而失敗，您將會看到下列錯誤訊息：`Unexpected error occurred while processing the request`。
- **TARGET_ARN**— 排程呼叫之目標的 ARN，採用下列服務 ARN 格`arn:aws:scheduler:::aws-sdk:`*`service`*`:`*`apiAction`*式：
- **EXHAUSTED_RETRY_CONDITION**— 指出事件傳遞至 DLQ 的原因。如果來自目標 API 的錯誤是可重試的錯誤，而不是永久錯誤，則此屬性將存在。`MaximumRetryAttempts`如果 S EventBridge cheduler 在超過您為排程設定的重試嘗試次數上限之後傳送至 DLQ，或者`MaximumEventAgeInSeconds`，如果事件早於您在排程上設定的保留天數上限且仍無法傳遞，則屬性可以包含這些值。

在前面的例子中，我們可以根據錯誤代碼和錯誤消息來確定我們為排程指定的目標隊列不存在。

刪除排程

您可以設定自動刪除或手動刪除個別排程來刪除排程。您可以使用下列主題，瞭解如何使用這兩種方法刪除排程，以及為何可以選擇其中一種方法。

主題

- 排程完成後刪除
- 手動刪除

排程完成後刪除

如果您想要避免在排程器上個別管理您的排程資源，請設定 EventBridge 排程完成後自動刪除。在您一次建立數千個排程且需要彈性以隨需擴充排程數量的應用程式中，自動刪除可確保您未達到指定區域中排程數量的帳戶配額。

當您設定排程的自動刪除時， EventBridge 排程器會在排程上次呼叫之後刪除排程。對於一次性排程，這會在排程呼叫其目標一次之後發生。對於您使用費率 (或 cron) 運算式設定的週期性排程，您的排程會在上次呼叫後刪除。週期性排程的最後一次呼叫是最接近`EndDate`您指定的呼叫。如果您設定具有自動刪除的排程，但未指定值`EndDate`，則「 EventBridge 排程器」不會自動刪除排程。

您可以在第一次建立排程時設定自動刪除，或更新現有排程的偏好設定。下列步驟說明如何設定現有排程的自動刪除。

AWS Management Console

1. 在 https://console.aws.amazon.com/scheduler/ 開啟 EventBridge 排程器主控台。
2. 從排程清單中選取您要編輯的排程，然後選擇編輯。
3. 從左側的導覽清單中，選擇 [設定]。
4. 在「排程完成後的動作」區段中，從下拉式清單中選取「刪除」，然後儲存變更。

AWS CLI

1. 開啟新的提示視窗。
2. 使用更新排程AWS CLI命令來更新下列所示的現有排程。指令會將設定`--action-after-completion`為`DELETE`。此範例假設您已在 JSON 檔案中在本機定義目標組態。若要更新排程，您必須提供目標，以及要為現有排程設定的任何其他排程參數。

   這是週期性排程，每小時調用一次的速率。因此，您可以在設定`--action-after-completion`參數時指定結束日期。

```
$ aws scheduler update-schedule --name schedule-name
 \
--action-after-completion 'DELETE' \
--schedule-expression 'rate(1 hour)' \
--end-date '2024-01-01T00:00:00'
--target file://target-configuration.json \
--flexible-time-window '{ "Mode": "OFF"}' \
```

手動刪除

當您不再需要排程時，可以使用此DeleteSchedule作業將其刪除。

Example AWS CLI

```
$ aws scheduler delete-schedule --name your-schedule
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

scheduler.delete_schedule(Name="your-schedule")
```

後續步驟？

- 如需如何為 Lambda 和 Step Functions 數設定範本化目標的詳細資訊，以及如何使用通用目標參數，請參閱管理目標。
- 如需有關 EventBridge 排程器資料類型和 API 作業的詳細資訊，請參閱EventBridge 排程器 API 參考。

管理排程群組

排程群組是您用來組織 EventBridge 排程的 Amazon 排程器資源。

您AWS 帳戶附帶一個default調度程序組。您可以將新排程與default群組或您建立和管理的排程群組相關聯。您最多可以在「」中建立 500 個排程群組AWS 帳戶。使用 EventBridge 「排程器」，您可以套用標籤來組織排程群組，而非個別排程。

標籤是由區分大小寫的鍵和您定義的區分大小寫值組成的標籤。您可以建立標籤，以依據用途、擁有者或環境等準則對明細表進行分類。例如，您可以使用下列標籤來識別明細表所屬的環境：environment:*production*

> **Important**
>
> 請勿在標籤中加入個人身分識別資訊 (PII) 或其他機密或敏感資訊。許多 AWS 服務都可以存取標籤，包括帳單。標籤不適用於私人或敏感資料。

排程群組有兩種可能的狀態：「作用中」和「刪除」。

第一次建立群組時，ACTIVE依預設為該群組。您可以將排程新增至ACTIVE群組。刪除群組時，狀態會變更為，DELETING直到「 EventBridge 排程器」完成刪除相關排程為止。 EventBridge 排程器刪除群組中的排程後，您的帳戶中就無法再使用該群組。

請使用下列主題來建立明細表群組，並將標記套用至該群組。您也會將排程與群組產生關聯。最後，您將刪除該群組。

主題

- 建立排程群組
- 刪除排程群組
- 相關資源

建立排程群組

使用明細表群組和標籤來組織共用一般用途或屬於相同環境的明細表。在下列步驟中，您會建立新明細表群組，並使用標籤為其加上標示。然後，您可以將新排程與該群組產生關聯。

> **Note**
>
> 建立群組後，您就無法從該群組中移除排程，也無法將排程與其他群組建立關聯。只有在您第一次建立排程時，才能將排程與群組相關聯。

步驟 1：建立新的排程群組

下列主題說明如何建立新的明細表群組，並使用下列標籤為其加上標籤：`environment:development`

AWS Management Console

若要使用建立新群組 AWS Management Console

1. 登錄到AWS Management Console並打開亞馬遜 EventBridge 控制台 https://console.aws.amazon.com/events/.
2. 在左側導覽窗格中，選擇 [排程群組]。
3. 在 [排程群組] 頁面上，選擇 [建立排程群組]。
4. 在「排程群組詳細資訊」區段中，輸入群組的名稱做為「名稱」。例如：**`TestGroup`**。
5. 在「標籤」區段中，執行下列動作：
   a. 選擇 Add new tag (新增標籤)。
   b. 在金鑰中，輸入您要指派給此金鑰的名稱。在本自學課程中，若要標示此明細表群組所屬的環境，請輸入**`environment`**。
   c. 在值-選擇性中，輸入您要指派給此機碼的值。在此自學課程中，請輸入環境金鑰的值**`development`**。

   > **Note**
   >
   > 您可以在建立群組後新增其他標記。
6. 若要完成，請選擇「建立排程群組」。您的新群組會出現在「排程群組」清單中。
7. (選擇性) 若要編輯群組或管理其標記，請選取新群組的核取方塊，然後選擇「編輯」。

> **Note**
>
> 您無法編輯`default`排程群組。

AWS CLI

若要使用建立新群組 AWS CLI

1. 開啟新的命令提示視窗。
2. 在 AWS Command Line Interface (AWS CLI) 中，輸入下列`create-schedule-group`指令以建立新群組。此指令會建立具有下列標籤的群組：`environment:development`。您可以使用此標籤或類似的標籤系統，根據明細表群組所屬的環境標示明細表群組。

   使用您的資訊取代明細表名稱和標籤關鍵字和值。

```
$ aws scheduler create-schedule-group --name TestGroup --tags
 Key=environment,Value=development
```

依預設，您的新群組`ACTIVE`處於狀態。您現在可以將新排程與您建立的新群組相關聯。

步驟二：將排程與群組產生關聯

請遵循下列步驟，將新排程與您在上一個步驟中建立的群組產生關聯。

AWS Management Console

若要使排程與群組產生關聯，請使用 AWS Management Console

1. 登錄到AWS Management Console並打開亞馬遜 EventBridge 控制台 https://console.aws.amazon.com/events/.
2. 在左側導覽窗格中，選擇左側導覽窗格中的 [排程]。
3. 從「排程」表格中，選擇「建立排程」以建立新排程。
4. 在 [指定排程詳細資訊] 頁面上，對於 [排程] 群組，從下拉式清單中選取新群組的名稱。例如，選取`TestGroup`。
5. 指定排程模式、目標、設定，然後在「檢閱並儲存排程」頁面上檢閱您的選擇。如需設定新排程的詳細資訊，請參閱入門。

6. 若要完成並儲存排程，請選擇 [儲存排程]。

AWS CLI

若要使排程與群組產生關聯，請使用 AWS CLI

1. 開啟新的命令提示視窗。
2. 在 AWS Command Line Interface (AWS CLI) 中，輸入以下`create-schedule`命令。這會建立排程，並將其與上一個名為的步驟中的群組產生關聯`sqs-test-schedule`。此排程會使用範本化的 Amazon SQS 目標類型來叫用作業。`SendMessage`以您的資訊取代排程名稱、目標和群組名稱。

```
$ aws scheduler create-schedule --name sqs-test-schedule --schedule-expression
 'rate(5 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn": "QUEUE_ARN", "Input": "TEST_PAYLOAD" }'
 \
--group-name TestGroup
--flexible-time-window '{ "Mode": "OFF"}'
```

您的新排程現在會與`TestGroup`排程群組產生關聯。

刪除排程群組

在下文中，您可以瞭解如何使用AWS Management Console和刪除排程群組AWS Command Line Interface。刪除群組時，該群組會處於`DELETING`狀態，直到「 EventBridge 排程器」刪除群組中的所有排程為止。 EventBridge 排程器刪除群組中的排程後，您的帳戶中就無法再使用該群組。

> **Note**
>
> 建立群組後，您就無法從該群組中移除排程，也無法將排程與其他群組建立關聯。只有在您第一次建立排程時，才能將排程與群組相關聯。

AWS Management Console

若要使用刪除群組 AWS Management Console

1. 登錄到AWS Management Console並打開亞馬遜 EventBridge 控制台 https://console.aws.amazon.com/events/.
2. 在左側導覽窗格中，選擇左側導覽窗格中的 [排程群組]。
3. 在 [排程群組] 頁面上，從目前的現有群組清單中AWS 區域，找出要刪除的群組。如果沒有看到您要尋找的群組，請選擇其他群組AWS 區域。

> **Note**
>
> 您無法刪除或編輯預設群組。

4. 選取要刪除之群組的核取方塊。
5. 選擇 刪除 。
6. 在「刪除排程群組」對話方塊中，輸入要確認選擇的群組名稱，然後選擇「刪除」。
7. 在「排程群組」清單中，「狀態」欄會變更，表示您的群組現在正在「刪除」。群組會保持此狀態，直到「 EventBridge 排程器」刪除與群組相關聯的所有排程為止。
8. 若要重新整理清單並確認群組已刪除，請選擇「重新整理」圖示。

AWS CLI

若要使用刪除群組 AWS CLI

1. 開啟新的命令提示視窗。
2. 在 AWS Command Line Interface (AWS CLI) 中，輸入下列delete-schedule-group指令以刪除排程群組。--name以您的資訊取代的值。

```
$ aws scheduler delete-schedule-group --name TestGroup
```

如果成功，此AWS CLI操作不會返回響應。

3. 若要確認群組是否處於狀DELETING態，請執行下列get-schedule-group命令。

```
$ aws scheduler get-schedule-group --name TestGroup
```

如果成功，您會收到類似下列內容的輸出：

```
{
    "Arn": "arn:aws::scheduler:us-west-2:123456789012:schedule-group/TestGroup",
    "CreationDate": "2023-01-01T09:00:00.000000-07:00",
    "LastModificationDate": "2023-01-01T09:00:00.000000-07:00",
    "Name": "TestGroup",
    "State": "DELETING"
}
```

EventBridge 排程器會在刪除與群組相關聯的排程後刪除群組。如果您`get-schedule-group`再次執行，您會收到下列`ResourceNotFoundException`回應：

```
An error occurred (ResourceNotFoundException) when calling the GetScheduleGroup
 operation: Schedule group TestGroup does not exist.
```

相關資源

如需排程群組的詳細資訊，請參閱下列資源：

- CreateScheduleGroupEventBridge 排程器 API 參考中的作業。
- DeleteScheduleGroupEventBridge 排程器 API 參考中的作業。

管理目標

下列主題說明如何搭配排程器使用範本化和通用目標，並提供您可以使用 EventBridge 排 EventBridge 程器的通用目標參數設定的支援AWS服務清單。

範本化目標是跨一組核心AWS服務 (例如 Amazon SQS、Lambda 和 Step Functions) 的一組常見 API 操作。例如，您可以透過提供函數 ARN 或 Amazon SQS 的作業與目標的佇列 ARN 來鎖定 Lambda 的叫用 API `SendMessage`作業。

通用目標是一組可自訂的參數，可讓您為許多AWS服務叫用更廣泛的 API 作業集。例如，您可以使用 EventBridge 排程器的通用目標參數 (UTP)，使用該`CreateQueue`作業建立新的 Amazon SQS 佇列。

若要設定範本化或通用目標，您的排程必須具有呼叫您設定為目標的 API 作業的權限。您可將所需許可附加到您排程的執行角色。例如，若要以 Amazon SQS 的`SendMessage`作業為目標，則會授與執行角色執行`sqs:SendMessage`動作的權限。在大多數情況下，您可以使用目標服務支援的AWS受管理策略來新增必要的權限。不過，您也可以建立自己的客戶管理政策，或將內嵌權限新增至附加至執行角色的現有政策。下列主題示範為範本化和通用目標類型新增權限的範例。

如需有關設定排程的執行角色的詳細資訊，請參閱the section called “設定執行角色”。

主題

使用範本化目標

範本化目標是跨一組核心 AWS 服務 (例如 Amazon SQS、Lambda 和 Step Functions) 的一組通用 API 操作。例如，您可以透過提供函數 ARN 來鎖定 Lambda 的`Invoke`作業，或使用佇列 ARN 提供 Amazon SQS 的`SendMessage`作業。若要設定範本化目標，您還必須授與排程執行角色的權限，才能執行目標 API 作業。

若要使用 AWS CLI 或其中一個 S EventBridge cheduler SDK 以程式設計方式設定範本目標，您需要指定執行角色的 ARN、目標資源的 ARN、您希望 EventBridge 排程器傳遞至目標的選擇性輸入，以及某些範本化目標的唯一參數集，以及該目標的其他組態選項。當您指定範本化目標資源的 ARN 時，

EventBridge 排程器會自動假設您要呼叫該服務支援的 API 作業。如果您希望 EventBridge 排程器針對服務以不同的 API 作業為目標，則必須將目標設定為通用目標。

以下是 EventBridge Scheduler 支援的所有樣板化目標的完整清單，以及每個目標的唯一相關參數集 (如果適用)。選擇每個參數集的連結，以查看「EventBridge 排程器 API 參考」中的必要欄位和選用欄位。

- CodeBuild – StartBuild
- CodePipeline – StartPipelineExecution
- Amazon ECS — RunTask
  - Parameters: EcsParameters
- EventBridge – PutEvents
  - Parameters: EventBridgeParameters
- Amazon Inspector — StartAssessmentRun
- kinesis：PutRecord
  - Parameters: KinesisParameters
- Firehose — PutRecord
- Lambda – Invoke
- SageMaker – StartPipelineExecution
  - Parameters: SageMakerPipelineParameters
- Amazon SNS — Publish
- Amazon SQS：SendMessage
  - Parameters: SqsParameters
- Step Functions — StartExecution

使用下列範例來瞭解如何設定不同的範本化目標，以及每個描述目標的必要 IAM 許可。

Amazon SQS **SendMessage**

Example 執行角色的權限原則

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

```
    "Statement": [
        {
            "Action": [
                "sqs:SendMessage"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Example AWS CLI

```
$ aws scheduler create-schedule --name sqs-templated --schedule-expression 'rate(5
 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"QUEUE_ARN", "Input": "Message for scheduleArn:
 '<aws.scheduler.schedule-arn>', scheduledTime: '<aws.scheduler.scheduled-time>'" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

flex_window = { "Mode": "OFF" }

sqs_templated = {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<QUEUE_ARN>",
    "Input": "Message for scheduleArn: '<aws.scheduler.schedule-arn>', scheduledTime:
 '<aws.scheduler.scheduled-time>'"
}

scheduler.create_schedule(
    Name="sqs-python-templated",
    ScheduleExpression="rate(5 minutes)",
    Target=sqs_templated,
    FlexibleTimeWindow=flex_window)
```

Example Java 開發套件

```
package com.example;
```

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.scheduler.SchedulerClient;
import software.amazon.awssdk.services.scheduler.model.*;


public class MySchedulerApp {

    public static void main(String[] args) {

        final SchedulerClient client = SchedulerClient.builder()
                .region(Region.US_WEST_2)
                .build();

        Target sqsTarget = Target.builder()
                .roleArn("<ROLE_ARN>")
                .arn("<QUEUE_ARN>")
                .input("Message for scheduleArn: '<aws.scheduler.schedule-arn>',
 scheduledTime: '<aws.scheduler.scheduled-time>'")
                .build();

        CreateScheduleRequest createScheduleRequest = CreateScheduleRequest.builder()
                .name("<SCHEDULE NAME>")
                .scheduleExpression("rate(10 minutes)")
                .target(sqsTarget)
                .flexibleTimeWindow(FlexibleTimeWindow.builder()
                        .mode(FlexibleTimeWindowMode.OFF)
                        .build())
                .build();

        client.createSchedule(createScheduleRequest);
        System.out.println("Created schedule with rate expression and an Amazon SQS
 templated target");
    }
}
```

Lambda **Invoke**

Example 執行角色的權限原則

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

```
        {
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Example AWS CLI

```
$ aws scheduler create-schedule --name lambda-templated-schedule --schedule-expression
 'rate(5 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"FUNCTION_ARN", "Input": "{ \"Payload\":
 \"TEST_PAYLOAD\" }" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

flex_window = { "Mode": "OFF" }

lambda_templated = {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<LAMBDA_ARN>",
    "Input": "{ 'Payload': 'TEST_PAYLOAD' }"}
}

scheduler.create_schedule(
    Name="lambda-python-templated",
    ScheduleExpression="rate(5 minutes)",
    Target=lambda_templated,
    FlexibleTimeWindow=flex_window)
```

Example Java 開發套件

```
package com.example;

import software.amazon.awssdk.regions.Region;
```

```
import software.amazon.awssdk.services.scheduler.SchedulerClient;
import software.amazon.awssdk.services.scheduler.model.*;


public class MySchedulerApp {

    public static void main(String[] args) {

        final SchedulerClient client = SchedulerClient.builder()
                .region(Region.US_WEST_2)
                .build();

        Target lambdaTarget = Target.builder()
                .roleArn("<ROLE_ARN>")
                .arn("<Lambda ARN>")
                .input("{ 'Payload': 'TEST_PAYLOAD' }")
                .build();

        CreateScheduleRequest createScheduleRequest = CreateScheduleRequest.builder()
                .name("<SCHEDULE_NAME>")
                .scheduleExpression("rate(10 minutes)")
                .target(lambdaTarget)
                .flexibleTimeWindow(FlexibleTimeWindow.builder()
                        .mode(FlexibleTimeWindowMode.OFF)
                        .build())
                .clientToken("<Token GUID>")
                .build();

        client.createSchedule(createScheduleRequest);
        System.out.println("Created schedule with rate expression and Lambda templated
 target");
    }
}
```

Step Functions **`StartExecution`**

Example 執行角色的權限原則

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

```
                "states:StartExecution"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Example AWS CLI

```
$ aws scheduler create-schedule --name sfn-templated-schedule --schedule-expression
 'rate(5 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"STATE_MACHINE_ARN", "Input": "{ \"Payload\":
 \"TEST_PAYLOAD\" }" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

flex_window = { "Mode": "OFF" }

sfn_templated= {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "<STATE_MACHINE_ARN>",
    "Input": "{ 'Payload': 'TEST_PAYLOAD' }"
}

scheduler.create_schedule(Name="sfn-python-templated",
    ScheduleExpression="rate(5 minutes)",
    Target=sfn_templated,
    FlexibleTimeWindow=flex_window)
```

Example Java 開發套件

```
package com.example;

import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.scheduler.SchedulerClient;
import software.amazon.awssdk.services.scheduler.model.*;
```

```
public class MySchedulerApp {

    public static void main(String[] args) {

        final SchedulerClient client = SchedulerClient.builder()
                .region(Region.US_WEST_2)
                .build();

        Target stepFunctionsTarget = Target.builder()
                .roleArn("<ROLE_ARN>")
                .arn("<STATE_MACHINE_ARN>")
                .input("{ 'Payload': 'TEST_PAYLOAD' }")
                .build();

        CreateScheduleRequest createScheduleRequest = CreateScheduleRequest.builder()
                .name("<SCHEDULE_NAME>")
                .scheduleExpression("rate(10 minutes)")
                .target(stepFunctionsTarget)
                .flexibleTimeWindow(FlexibleTimeWindow.builder()
                        .mode(FlexibleTimeWindowMode.OFF)
                        .build())
                .clientToken("<Token GUID>")
                .build();

        client.createSchedule(createScheduleRequest);
        System.out.println("Created schedule with rate expression and Step Function
 templated target");
    }
}
```

使用通用目標

通用目標是一組可自訂的參數，可讓您針對許多AWS服務叫用更廣泛的 API 作業集。例如，您可以使用通用目標參數 (UTP)，使用該CreateQueue作業建立新的 Amazon SQS 佇列。

若要使用或其中一個 EventBridge 排程器 SDK 設定排程的通用目標，您必須指定下列資訊：AWS CLI

- RoleArn— 您要用於目標之執行角色的 ARN。您指定的執行角色必須具有呼叫您要排程鎖定之 API 作業的權限。
- Arn — 完整的服務 ARN，包括您要定位的 API 作業，格式如下：。arn:aws:scheduler:::aws-sdk:*service*:*apiAction*

例如，對於 Amazon SQS，您指定的服務名稱為`arn:aws:scheduler:::aws-sdk:sqs:sendMessage`。

- 輸入 — 您使用 EventBridge 排程器傳送至目標 API 的要求參數來指定格式良好的 JSON。您在中設定的 JSON 參數和形狀`Input`由排程叫用的服務 API 決定。若要尋找此資訊，請參閱您要鎖定之服務的 API 參考。

不支援動作

EventBridge 排程器不支援以下列首碼清單開頭的唯讀 API 動`GET`作，例如一般作業：

```
get
describe
list
poll
receive
search
scan
query
select
read
lookup
discover
validate
batchGet
batchDescribe
batchRead
transactGet
adminGet
adminList
testMigration
retrieve
testConnection
translateDocument
isAuthorized
isAuthorizedWithToken
invokeModel
```

例如，GetQueueUrlAPI 動作的服務 ARN 如下所示：`arn:aws:scheduler:::aws-sdk:sqs:getQueueURL`由於 API 動作以`get`前綴開頭，因此「 EventBridge 排程器」不支援此目標。同樣地，不支援 Amazon MQ 動`ListBrokers`作做為目標，因為作業具有前置詞。`list`

使用通用目標的範例

您在排程Input欄位中傳遞的參數取決於您要叫用的服務 API 接受的要求參數。例如，若要以 Lambda 為目標 Invoke，您可以設定 AWS LambdaAPI 參考中列出的參數。這包括您可以傳遞給 Lambda 函數的選用 JSON 承載資料。

若要判斷您可以為不同 API 設定的參數，請參閱該服務的 API 參考資料。與 Lambda 類似Invoke，某些 API 接受 URI 參數以及請求主體有效負載。在這種情況下，您可以在排程中指定 URI 路徑參數以及 JSON 承載Input。

下列範例說明如何使用通用目標來叫用 Lambda、Amazon SQS 和 Step Functions 的常見 API 作業。

Example Lambda

```
$ aws scheduler create-schedule --name lambda-universal-schedule --schedule-expression
 'rate(5 minutes)' \
--target '{"RoleArn": "ROLE_ARN", "Arn":"arn:aws:scheduler:::aws-sdk:lambda:invoke"
 "Input": "{\"FunctionName\":\"arn:aws:lambda:REGION:123456789012:function:HelloWorld
\",\"InvocationType\":\"Event\",\"Payload\":\"{\\\"message\\\":\\\"testing function\\
\"}\"}" }' \
--flexible-time-window '{ "Mode": "OFF"}'
```

Example Amazon SQS

```
import boto3
scheduler = boto3.client('scheduler')

flex_window = { "Mode": "OFF" }

sqs_universal= {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "arn:aws:scheduler:::aws-sdk:sqs:sendMessage",
    "Input": "{\"MessageBody\":\"My message\",\"QueueUrl\":\"<QUEUE_URL>\"}"}
}

scheduler.create_schedule(
    Name="sqs-sdk-test",
    ScheduleExpression="rate(5 minutes)",
    Target=sqs_universal,
    FlexibleTimeWindow=flex_window)
```

Example Step Functions

```
package com.example;

import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.scheduler.SchedulerClient;
import software.amazon.awssdk.services.scheduler.model.*;


public class MySchedulerApp {

    public static void main(String[] args) {

        final SchedulerClient client = SchedulerClient.builder()
                .region(Region.US_WEST_2)
                .build();

        Target stepFunctionsUniversalTarget = Target.builder()
                .roleArn("<ROLE_ARN>")
                .arn("arn:aws:scheduler:::aws-sdk:sfn:startExecution")
                .input("{\"Input\":\"{}\",\"StateMachineArn\":\"<STATE_MACHINE_ARN>
\"}")
                .build();

        CreateScheduleRequest createScheduleRequest = CreateScheduleRequest.builder()
                .name("<SCHEDULE_NAME>")
                .scheduleExpression("rate(10 minutes)")
                .target(stepFunctionsUniversalTarget)
                .flexibleTimeWindow(FlexibleTimeWindow.builder()
                        .mode(FlexibleTimeWindowMode.OFF)
                        .build())
                .clientToken("<Token GUID>")
                .build();

        client.createSchedule(createScheduleRequest);
        System.out.println("Created schedule with rate expression and Step Function
 universal target");
    }
}
```

新增上下文屬性

在傳遞給目標的承載中使用下列關鍵字，以收集有關排程的中繼資料。 EventBridge 排程器會在排程叫用目標時，以各自的值取代每個關鍵字。

- **<aws.scheduler.schedule-arn>**— 排程的 ARN。
- **<aws.scheduler.scheduled-time>**— 您指定的排程呼叫其目標的時間，例如，2022-03-22T18:59:43Z。
- **<aws.scheduler.execution-id>**— EventBridge 排程器為每次嘗試呼叫目標指派的唯一識別碼，d32c5kddcf5bb8c3例如。
- **<aws.scheduler.attempt-number>**— 識別目前呼叫之嘗試編號的計數器，1例如。

此範例顯示建立每五分鐘觸發一次的排程，並將 Amazon SQS SendMessage 作業作為通用目標呼叫。郵件內文包含的值schedule-time。

Example AWS CLI

```
$ aws scheduler create-schedule --name your-schedule \
    --schedule-expression 'rate(5 minutes)' \
    --target '{"RoleArn": "ROLE_ARN", \
        "Arn": "arn:aws:scheduler:::aws-sdk:sqs:sendMessage", \
        "Input": "{\"MessageBody\":\"<aws.scheduler.scheduled-time>\",\"QueueUrl\":
\"https://sqs.us-west-2.amazonaws.com/123456789012/scheduler-cli-test\"}"}' \
    --flexible-time-window '{ "Mode": "OFF"}'
```

Example Python SDK

```
import boto3
scheduler = boto3.client('scheduler')

sqs_universal= {
    "RoleArn": "<ROLE_ARN>",
    "Arn": "arn:aws:scheduler:::aws-sdk:sqs:sendMessage",
    "Input": "{\"MessageBody\":\"<aws.scheduler.scheduled-time>\",\"QueueUrl\":
\"https://sqs.us-west-2.amazonaws.com/123456789012/scheduler-cli-test\"}"
}

flex_window = { "Mode": "OFF" }
```

```
scheduler.update_schedule(Name="your-schedule",
    ScheduleExpression="rate(5 minutes)",
    Target=sqs_universal,
    FlexibleTimeWindow=flex_window)
```

後續步驟？

如需有關 EventBridge 排程器資料類型和 API 作業的詳細資訊，請參閱EventBridge 排程器 API 參考。

Amazon EventBridge 調度程序中的

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，這些架構是為了滿足對安全性最敏感的組織的需求而建置的。

安全是 AWS 與您之間共同承擔的責任。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：

- 雲端安全性 — AWS 負責保護中執行 AWS 服務的基礎架構 AWS 雲端。 AWS 還為您提供可以安全使用的服務。若要了解適用於 Amazon EventBridge Scheduler 的合規計劃，請參閱合規計劃AWS服務範圍內的合規計劃AWS的服務。
- 雲端中的安全性 — 您的責任取決於您使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的要求和適用法律和法規。

本文件可協助您瞭解如何在使用「 EventBridge 排程器」時套用共同的責任模型。下列主題說明如何設定 EventBridge 排程器以符合您的安全性和合規性目標。您也會學到如何使用其他可 AWS 協助您監視和保護 EventBridge 排程器資源的服務。

主題

- 管理 Amazon EventBridge 排程器的存取
- Amazon EventBridge 排程器中的資料保護
- Amazon EventBridge 排程器的合規驗證
- Amazon EventBridge 排程器的彈性
- Amazon EventBridge 排程器中的基礎設施安

管理 Amazon EventBridge 排程器的存取

AWS Identity and Access Management (IAM) 可協助管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM 管理員控制哪些人可以驗證 (登入) 和授權 (具有權限) 以使用 EventBridge 排程器資源。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

- EventBridge 排程器如何與 IAM 搭配使用
- 使用身分型政策
- 預防混淆代理人
- 疑難排解 Amazon EventBridge 排程器身分和存取

物件

根據您在 EventBridge 排程器中執行的工作而定，使用方式 AWS Identity and Access Management (IAM) 會有所不同。

服務使用者 — 如果您使用 EventBridge Scheduler 服務執行工作，則管理員會為您提供所需的認證和權限。當您使用更多 EventBridge 排程器功能來完成工作時，您可能需要其他權限。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 EventBridge 排程器中的功能，請參閱疑難排解 Amazon EventBridge 排程器身分和存取。

服務管理員 — 如果您負責公司的 EventBridge 排程器資源，您可能擁有 EventBridge Scheduler 的完整存取權。決定您的服務使用者應該存取哪些 EventBridge Scheduler 功能和資源是您的工作。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搭配 EventBridge 排程器使用 IAM，請參閱 EventBridge 排程器如何與 IAM 搭配使用。

IAM 管理員 — 如果您是 IAM 管理員，您可能想要瞭解如何撰寫政策以管理 EventBridge 排程器存取權限的詳細資訊。若要檢視可在 IAM 中使用的 EventBridge 排程器身分型政策範例，請參閱。使用身分型政策

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。

EventBridge 排程器如何與 IAM 搭配使用

在您使用 IAM 管理 EventBridge 排程器的存取權限之前，請先了解哪些 IAM 功能可搭配 EventBridge 排程器使用。

您可以搭配 Amazon EventBridge 排程器使用的 IAM 功能

IAM 功能	EventBridge 排程器支援
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 (服務特定)	是
ACL	否
ABAC(政策中的標籤)	部分
臨時憑證	是
主體許可	是
服務角色	是
服務連結角色	否

若要取得 EventBridge 排程器和其他 AWS 服務如何搭配大多數 IAM 功能運作的高階檢視，請參閱 IAM 使用者指南中的搭配 IAM 使用的AWS 服務。

排程器的身分型原則 EventBridge

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。

排程器的身分識別原則範例 EventBridge

若要檢視以 EventBridge 排程器身分識別為基礎的原則範例，請參閱。使用身分型政策

排程 EventBridge 器內的資源型政策

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。

EventBridge 排程器的原則動作

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，哪個主體在什麼条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 EventBridge 排程器動作清單，請參閱服務授權參考中由 Amazon EventBridge 排程器定義的動作。

EventBridge 排程器中的原則動作會在動作之前使用下列前置詞：

```
scheduler
```

如需在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [
      "scheduler:action1",
      "scheduler:action2"
]
```

您也可以使用萬用字元 (*) 來指定多個動作。例如，若要指定開頭是 `List` 文字的所有動作，請包含以下動作：

```
"Action": [
   "scheduler:List*"
]
```

EventBridge 排程器的原則資源

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 EventBridge 排程器資源類型及其 ARN 的清單，請參閱服務授權參考中由 Amazon EventBridge 排程器定義的資源。若要了解可以使用哪些動作指定每個資源的 ARN，請參閱 Amazon EventBridge 排程器定義的動作。

若要檢視以 EventBridge 排程器身分識別為基礎的原則範例，請參閱。使用身分型政策

EventBridge 排程器的原則條件金鑰

支援服務特定政策條件金鑰	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。

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 EventBridge 排程器條件金鑰清單，請參閱服務授權參考中的 Amazon EventBridge 排程器的條件金鑰。若要了解可以使用條件金鑰的動作和資源，請參閱 Amazon EventBridge 排程器定義的動作。

若要檢視以 EventBridge 排程器身分識別為基礎的原則範例，請參閱。使用身分型政策

排程器中的 EventBridge 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。

含 EventBridge 排程器的 ABAC

支援 ABAC (政策中的標籤)	部分

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 EventBridge 排程器使用臨時認證

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，而非使用長期存取金鑰。如需詳細資訊，請參閱 IAM 中的暫時性安全憑證。

排程 EventBridge 器的跨服務主體權限

支援轉寄存取工作階段 (FAS)	是

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》。

EventBridge 排程器的服務角色

支援服務角色	是

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 [EventBridge 排程器] 功能。只有當 EventBridge 排程器提供指引時，才編輯服務角色。

排程 EventBridge 器的服務連結角色

支援服務連結角色。	否

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。

如需建立或管理服務連結角色的詳細資訊，請參閱可搭配 IAM 運作的AWS 服務。在表格中尋找服務，其中包含服務連結角色欄中的 Yes。選擇是連結，以檢視該服務的服務連結角色文件。

使用身分型政策

根據預設，使用者和角色沒有建立或修改 EventBridge 排程器資源的權限。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。

有關 EventBridge 排程器定義的動作和資源類型的詳細資訊，包括每個資源類型的 ARN 格式，請參閱服務授權參考中的 Amazon S EventBridge cheduler 的動作、資源和條件金鑰。

主題

- 政策最佳實務
- EventBridge 排程器權限
- AWS EventBridge 排程器的受管理原則
- EventBridge 排程器的客戶管理政策
- AWS 受管理策略更新

政策最佳實務

以身分識別為基礎的原則會決定某人是否可以在您的帳戶中建立、存取或刪除 EventBridge Scheduler 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。

EventBridge 排程器權限

若要讓 IAM 主體 (使用者、群組或角色) 在 S EventBridge cheduler 中建立排程，並透過主控台或 API 存取 S EventBridge cheduler 資源，主體必須在其權限原則中新增一組許可。您可以根據主參與者的工作功能來設定這些權限。例如，只使用 EventBridge Scheduler 主控台檢視現有排程清單的使用者或角色，不需要具備呼叫 `CreateSchedule` API 作業所需的權限。我們建議您量身定制基於身份的權限，以僅提供最低權限的存取。

下列清單顯示 EventBridge 排程器的資源及其對應的支援動作。

- 排程
  - `scheduler:ListSchedules`
  - `scheduler:GetSchedule`
  - `scheduler:CreateSchedule`
  - `scheduler:UpdateSchedule`
  - `scheduler:DeleteSchedule`
- 排程群組
  - `scheduler:ListScheduleGroups`
  - `scheduler:GetScheduleGroup`
  - `scheduler:CreateScheduleGroup`
  - `scheduler:DeleteScheduleGroup`

- scheduler:ListTagsForResource
- scheduler:TagResource
- scheduler:UntagResource

您可以使用 EventBridge 排程器權限來建立自己的客戶管理政策，以搭配 EventBridge Scheduler 使用。您也可以使用下一節所述的 AWS 受管理策略來授與常見使用案例的必要權限，而不必管理自己的策略。

AWS EventBridge 排程器的受管理原則

AWS 透過提供可 AWS 建立和管理員的獨立 IAM 政策，解決許多常見使用案例。受管或預定義政策會針對常用案例授予必要的許可，因此您無須調查需要哪些許可。如需詳細資訊，請參閱《IAM 使用者指南》中的 AWS 受管政策。您可以附加至帳戶中使用者的下列 AWS 受管理政策是 EventBridge Scheduler 專用的：

- the section called “AmazonEventBridgeSchedulerFullAccess”— 使用主控台和 API 授予 EventBridge 排程器的完整存取權。
- the section called “AmazonEventBridgeSchedulerReadOnlyAccess”— 授與 EventBridge 排程器的唯讀存取權。

AmazonEventBridgeSchedulerFullAccess

AmazonEventBridgeSchedulerFullAccess受管理的原則會授與對排程和 EventBridge 排程群組使用所有排程器動作的權限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "scheduler:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/*",
            "Condition": {
                "StringLike": {
```

```
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
            }
        }
    ]
}
```

AmazonEventBridgeSchedulerReadOnlyAccess

AmazonEventBridgeSchedulerReadOnlyAccess受管理的原則會授與唯讀權限，以檢視排程和排程群組的詳細資料。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "scheduler:ListSchedules",
                "scheduler:ListScheduleGroups",
                "scheduler:GetSchedule",
                "scheduler:GetScheduleGroup",
                "scheduler:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

EventBridge 排程器的客戶管理政策

使用下列範例為 EventBridge Scheduler 建立您自己的客戶管理政策。客戶管理的政策可讓您根據主參與者的工作職能，僅授與應用程式和使用者所需的動作和資源的權限。

主題

- 範例：CreateSchedule
- 範例：GetSchedule
- 範例：UpdateSchedule
- 範例：DeleteScheduleGroup

範例：**CreateSchedule**

建立新排程時，您可以選擇是使用或客戶管理的金鑰來加密 EventBridge Scheduler 上的資料。AWS 擁有的金鑰

下列原則可讓主體建立排程，並使用 AWS 擁有的金鑰. 使用 AWS 擁有的金鑰, 為您 AWS 管理 on AWS Key Management Service (AWS KMS) 資源，因此您不需要額外的權限即可與之互動 AWS KMS。

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Action":
            [
                "scheduler:CreateSchedule"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-
schedule-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/*",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
            }
        }
    ]
}
```

使用下列原則可允許主體建立排程，並使用 AWS KMS 客戶管理的金鑰進行加密。若要使用客戶管理的金鑰，主體必須具有存取您帳戶中 AWS KMS 資源的權限。此原則會授與單一指定 KMS 金鑰的存取權，以用來加密 EventBridge 排程器上的資料。或者，您可以使用萬用字元 (*) 字元來授與帳戶中所有金鑰的存取權，或是符合指定名稱模式的子集。

```
{
    "Version": "2012-10-17"
    "Statement":
    [
        {
            "Action":
            [
                "scheduler:CreateSchedule"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-
schedule-name"
            ]
        },
        {
            "Action":
            [
                "kms:DescribeKey",
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:kms:us-west-2:123456789012:key/my-key-id"
            ],
            "Conditions": {
                "StringLike": {
                    "kms:ViaService": "scheduler.amazonaws.com",
                    "kms:EncryptionContext:aws:scheduler:schedule:arn":
 "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-schedule-name"
            }
        }
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/*",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
```

```
            }
        }
    ]
}
```

範例：**GetSchedule**

使用下列原則可允許主參與者取得排程的相關資訊。

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Action":
            [
                "scheduler:GetSchedule"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-
schedule-name"
            ]
        }
    ]
}
```

範例：**UpdateSchedule**

使用下列原則可允許主參與者呼叫scheduler:UpdateSchedule動作來更新排程。類似於CreateSchedule，該策略取決於排程是使用 AWS KMS AWS 擁有的金鑰 還是客戶管理的金鑰進行加密。對於使用設定的排程 AWS 擁有的金鑰，請使用下列原則：

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

```
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-
schedule-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/*",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
            }
        }
    ]
}
```

對於使用客戶管理金鑰設定的排程，請使用下列原則。此原則包含可讓主體存取您帳戶中資 AWS KMS 源的其他權限：

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Action":
            [
                "scheduler:UpdateSchedule"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-
schedule-name
        },
        {
            "Action":
            [
                "kms:DescribeKey",
```

```
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Effect": "Allow",
            "Resource":
            [
                "arn:aws:kms:us-west-2:123456789012:key/my-key-id"
            ],
            "Conditions": {
                "StringLike": {
                    "kms:ViaService": "scheduler.amazonaws.com",
                    "kms:EncryptionContext:aws:scheduler:schedule:arn":
 "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-schedule-name"
            }
        }
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/*",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
            }
        }
    ]
}
```

範例：**DeleteScheduleGroup**

使用下列原則可允許主體刪除排程群組。刪除群組時，也會刪除與該群組相關聯的排程。刪除群組的主參與者必須擁有刪除與該群組相關聯之排程的權限。此原則會授與對指定排程群組以及群組中所有排程呼叫`scheduler:DeleteScheduleGroup`動作的主要權限：

> **Note**
>
> EventBridge 排程器不支援指定個別排程的資源層級權限。例如，以下聲明無效，不應包含在您的政策中：
> `"Resource": "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-schedule-name"`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "scheduler:DeleteSchedule",
            "Resource": "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/*"
        },
        {
            "Effect": "Allow",
            "Action": "scheduler:DeleteScheduleGroup",
            "Resource": "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/*",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "scheduler.amazonaws.com"
                }
            }
        }
    ]
}
```

AWS 受管理策略更新

變更	描述	日期
the section called “AmazonEventBridgeSchedulerFullAccess”— 新的受管理策略	EventBridge Scheduler 新增了對新的受管理政策的支援，可授與使用者對所有資源 (包括排程和排程群組) 的完整存取權。	2022 年 11 月 10 日
the section called “AmazonEventBridgeSchedulerReadOnlyAccess”— 新的受管理策略	EventBridge Scheduler 新增了對新的受管理策略的支援，該策略授與使用者對所有資源 (包括排程和排程群組) 的唯讀存取權。	2022 年 11 月 10 日

變更	描述	日期
EventBridge 排程器開始追蹤變更	EventBridge 排程器開始追蹤其 AWS 受管理原則的變更。	2022 年 11 月 10 日

預防混淆代理人

混淆代理人問題屬於安全性議題，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。為了預防這種情況， AWS 提供的工具可協助您保護所有服務的資料，而這些服務主體已獲得您帳戶中資源的存取權。

我們建議您在排程執行角色中使用aws:SourceArn和aws:SourceAccount全域條件內容索引鍵，以限制 EventBridge Scheduler 為其他服務提供存取資源的權限。如果您想要僅允許一個資源與跨服務存取相關聯，則請使用 aws:SourceArn。如果您想要允許該帳戶中的任何資源與跨服務使用相關聯，請使用 aws:SourceAccount。

防範混淆代理人問題的最有效方法是使用 aws:SourceArn 全域條件內容索引鍵，以及資源的完整 ARN。下列條件範圍為個別排程群組：arn:aws:scheduler:*:*123456789012*:schedule-group/*your-schedule-group*

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 全域內容條件索引鍵搭配萬用字元 (*) 來表示 ARN 的未知部分。例如：arn:aws:scheduler:*:*123456789012*:schedule-group/*。

的值aws:SourceArn必須是您要限定此條件範圍的「 EventBridge 排程器」排程群組 ARN。

> **Important**
>
> 請勿將aws:SourceArn陳述式的範圍限定為特定排程或排程名稱前置詞。您指定的 ARN 必須是排程群組。

下列範例顯示如何在執行角色信任原則中使用aws:SourceArn和aws:SourceAccount全域條件內容索引鍵，以避免混淆的副問題：

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "scheduler.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012",
                    "aws:SourceArn": "arn:aws:scheduler:us-
west-2:123456789012:schedule-group/your-schedule-group"
                }
            }
        }
    ]
}
```

疑難排解 Amazon EventBridge 排程器身分和存取

使用下列資訊可協助您診斷並修正使用 EventBridge 排程器和 IAM 時可能會遇到的常見問題。

主題

- 我沒有授權在 EventBridge 排程器中執行動作
- 我沒有授權執行 iam：PassRole
- 我想允許我以外的人訪問我 AWS 帳戶 的 EventBridge 調度程序資源

我沒有授權在 EventBridge 排程器中執行動作

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 scheduler:*GetWidget* 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 scheduler:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 Mateo 政策，允許他使用 scheduler:*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

如果您收到未獲授權執行`iam:PassRole`動作的錯誤訊息，則必須更新您的原則，以允許您將角色傳遞給 EventBridge Scheduler。

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。

當名為的 IAM 使用者`marymajor`嘗試使用主控台在 EventBridge Scheduler 中執行動作時，就會發生下列範例錯誤。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。

我想允許我以外的人訪問我 AWS 帳戶 的 EventBridge 調度程序資源

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：

- 若要瞭解 EventBridge 排程器是否支援這些功能，請參閱 EventBridge 排程器如何與 IAM 搭配使用。
- 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。
- 若要了解如何將資源存取權提供給第三方 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。

Amazon EventBridge 排程器中的資料保護

AWS 共同責任模型適用於 Amazon EventBridge Scheduler 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 AWS SDK AWS 服務 使用 EventBridge 排程器或其他排程器時。 AWS CLI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

靜態加密

本節說明 Amazon EventBridge 排程器如何加密和解密靜態資料。靜態資料是儲存在 EventBridge 排程器和服務基礎元件中的資料。 EventBridge 排程器與 AWS Key Management Service (AWS KMS) 整合，可使用 AWS KMS key. EventBridge 排程器支援兩種類型的 KMS 金鑰：AWS 擁有的金鑰和客戶受管金鑰。

> **Note**
>
> EventBridge 排程器僅支援使用對稱加密 KMS 金鑰。

AWS 擁有的金鑰是 AWS 服務擁有和管理的 KMS 金鑰，可在多個 AWS 帳戶中使用。雖然 AWS 擁有的金鑰 EventBridge 排程器使用不會儲存在您的 AWS 帳戶中，但 EventBridge Scheduler 會使用它們來保護您的資料和資源。依預設， EventBridge Scheduler 會使用擁有的金鑰加密和解密您的所有 AWS 資料。您不需要管理您 AWS 擁有的金鑰 或其存取原則。當 EventBridge Scheduler 用 AWS 擁有的金鑰 來保護您的資料時，您不會產生任何費用，且其使用量不會計入您帳戶中 AWS KMS 配額的一部分。

客戶受管金鑰是儲存在您建立、擁有和管理的 AWS 帳戶中的 KMS 金鑰。如果您的特定使用案例要求您控制和稽核 S EventBridge cheduler 上保護資料的加密金鑰，您可以使用客戶管理的金鑰。如果您選擇客戶管理的金鑰，則必須管理您的金鑰政策。客戶受管金鑰會衍生每月費用，以及超出免費方案部分的使用費用。使用客戶管理的金鑰也會算作AWS KMS 配額的一部分。如需定價的詳細資訊，請參閱AWS Key Management Service 定價。

主題

- 加密成品
- 管理 KMS 金鑰
- CloudTrail 事件範例

加密成品

下表說明 EventBridge 排程器在靜態時加密的不同類型資料，以及每個類別支援的 KMS 金鑰類型。

資料類型	描述	AWS 擁有的金鑰	客戶管理的金鑰
有效載荷 (最高 256KB)	設定要傳遞至目標的排程時，您在排程`TargetInput` 參數中指定的資料。	支援	支援
識別碼和狀態	排程的唯一名稱和狀態 (啟用、停用)。	支援	不支援

資料類型	描述	AWS 擁有的金鑰	客戶管理的金鑰
Scheduling configuration (排程組態)	排程運算式，例如週期性排程的速率或 cron 運算式，以及一次性呼叫的時間戳記，以及排程的開始日期、結束日期和時區。	支援	不支援
目標組態	目標的 Amazon 資源名稱 (ARN) 以及其他目標相關組態詳細資料。	支援	不支援
調用和失敗行為配置	彈性的時間範圍組態、排程的重試原則，以及失敗傳送所使用的無效字母佇列詳細資訊。	支援	不支援

EventBridge Scheduler 只有在加密和解密目標承載時，才會使用您的客戶受管金鑰，如上表所述。如果您選擇使用客戶管理的金鑰， EventBridge Scheduler 會對承載進行兩次加密和解密：一次使用預設值 AWS 擁有的金鑰，另一次使用您指定的客戶管理金鑰。對於所有其他資料類型， EventBridge Scheduler 只會使用預設值 AWS 擁有的金鑰 來保護您的靜態資料。

請the section called “管理 KMS 金鑰”參閱以下章節，瞭解如何管理 IAM 資源和金鑰政策，才能搭配 EventBridge Scheduler 使用客戶受管金鑰。

管理 KMS 金鑰

您可以選擇性地提供客戶管理的金鑰，以加密和解密排程傳送至其目標的承載。 EventBridge 排程器會加密和解密您的承載，最多可達 256KB 的資料。使用客戶管理金鑰會產生每月費用和超出免費方案的費用。使用客戶管理的金鑰會計入AWS KMS 配額的一部分。如需定價的詳細資訊，請參閱AWS Key Management Service 定價

EventBridge 排程器會使用與主體相關聯的 IAM 許可，以建立排程來加密您的資料。這表示您必須將必要的 AWS KMS 相關權限附加至呼叫 EventBridge Scheduler API 的使用者或角色。此外，

EventBridge 排程器會使用以資源為基礎的政策來解密您的資料。這表示與排程相關聯的執行角色也必須具有必要的 AWS KMS 相關權限，才能在解密資料時呼叫 AWS KMS API。

> **Note**
>
> EventBridge 排程器不支援使用臨時權限的授權。

請參閱以下章節，瞭解如何管理 AWS KMS 金鑰政策以及在 EventBridge Scheduler 上使用客戶受管金鑰所需的 IAM 許可。

主題

- 新增 IAM 許可
- 管理金鑰原則

新增 IAM 許可

若要使用客戶受管金鑰，您必須將下列許可新增至建立排程的身分型 IAM 主體，以及與排程相關聯的執行角色。

客戶受管金鑰的身分識別權限

建立 EventBridge 排程時，您必須將下列 AWS KMS 動作新增至與任何主體 (使用者、群組或角色) 相關聯的權限原則 (使用者、群組或角色)。

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

                # Required to pass the execution role
                "iam:PassRole",

                "kms:DescribeKey",
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*",
            "Effect": "Allow"
```

```
        },
    ]
}
```

- **kms:DescribeKey**— 驗證您提供的金鑰是否為對稱加密 KMS 金鑰的必要項目。
- **kms:GenerateDataKey**— 需要才能產生 EventBridge Scheduler 用來執行用戶端加密的資料金鑰。
- **kms:Decrypt**— 必須解密 EventBridge 排程器與加密資料一起儲存的加密資料金鑰。

客戶受管金鑰的執行角色權限

您必須將下列動作新增至排程的執行角色權限原則，以提供 EventBridge Scheduler 的存取權，以便在解密資料時呼叫 AWS KMS API。

```
{
    "Version": "2012-10-17",
    "Statement" : [
    {
      "Sid" : "Allow EventBridge Scheduler to decrypt data using a customer managed
 key",
      "Effect" : "Allow",
      "Action" : [
        "kms:Decrypt"

      ],
      "Resource": "arn:aws:kms:your-region:123456789012:key/your-key-id"
    }
  ]
}
```

- **kms:Decrypt**— 必須解密 EventBridge 排程器與加密資料一起儲存的加密資料金鑰。

如果您在建立新排程時使用 EventBridge Scheduler 主控台建立新的執行角色， EventBridge Scheduler 會自動將必要的權限附加至您的執行角色。但是，如果您選擇現有的執行角色，則必須將必要的權限新增至角色，才能使用客戶管理的金鑰。

管理金鑰原則

根據預設 AWS KMS，當您使用建立客戶受管金鑰時，金鑰具有下列金鑰原則，可讓您存取排程的執行角色。

```
{
    "Id": "key-policy-1",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Provide required IAM Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        }
    ]
}
```

或者，您可以將金鑰原則的範圍限制為僅提供執行角色的存取權。如果您只想將客戶管理的金鑰與 EventBridge Scheduler 資源搭配使用，您可以這麼做。請使用下列金鑰原則範例來限制哪些 EventBridge Scheduler 資源可以使用您的金鑰。

```
{
    "Id": "key-policy-2",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Provide required IAM Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::695325144837:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/schedule-execution-role"
            },
            "Action": [
                "kms:Decrypt"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

CloudTrail 事件範例

AWS CloudTrail 擷取所有 API 呼叫事件。這包括每當 EventBridge Scheduler 使用客戶管理金鑰解密您的資料時，都會呼叫 API 呼叫。下列範例顯示 CloudTrail 事件項目，示範 EventBridge Scheduler 使用客戶管理金鑰的`kms:Decrypt`動作。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
      "type": "AssumedRole",
      "principalId": "ABCDEABCD1AB12ABABAB0:70abcd123a123a12345a1aa12aa1bc12",
      "arn": "arn:aws:sts::123456789012:assumed-role/execution-
role/70abcd123a123a12345a1aa12aa1bc12",
      "accountId": "123456789012",
      "accessKeyId": "ABCDEFGHI1JKLMNOP2Q3",
      "sessionContext": {
          "sessionIssuer": {
              "type": "Role",
              "principalId": "ABCDEABCD1AB12ABABAB0",
              "arn": "arn:aws:iam::123456789012:role/execution-role",
              "accountId": "123456789012",
              "userName": "execution-role"
          },
          "webIdFederationData": {},
          "attributes": {
              "creationDate": "2022-10-31T21:03:15Z",
              "mfaAuthenticated": "false"
          }
      }
    },
    "eventTime": "2022-10-31T21:03:15Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "eu-north-1",
    "sourceIPAddress": "13.50.87.173",
    "userAgent": "aws-sdk-java/2.17.295 Linux/4.14.291-218.527.amzn2.x86_64 OpenJDK_64-
Bit_Server_VM/11.0.17+9-LTS Java/11.0.17 kotlin/1.3.72-release-468 (1.3.72) vendor/
```

```
Amazon.com_Inc. md/internal exec-env/AWS_ECS_FARGATE io/sync http/Apache cfg/retry-
mode/standard AwsCrypto/2.4.0",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-west-2:123456789012:key/2321abab-2110-12ab-a123-
a2b34c5abc67",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "encryptionContext": {
            "aws:scheduler:schedule:arn": "arn:aws:scheduler:us-
west-2:123456789012:schedule/default/execution-role"
        }
    },
    "responseElements": null,
    "requestID": "request-id",
    "eventID": "event-id",
    "readOnly": true,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:123456789012:key/2321abab-2110-12ab-a123-
a2b34c5abc67"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
      "tlsVersion": "TLSv1.3",
      "cipherSuite": "TLS_AES_256_GCM_SHA384",
      "clientProvidedHostHeader": "kms.us-west-2.amazonaws.com"
  }
}
```

傳輸中加密

EventBridge 排程器會在您傳輸網路時加密傳輸中的資料。傳輸層安全性 (TLS) 會在您呼叫任何 EventBridge 排程器 API 作業時加密您的資料，以及 EventBridge 排程器呼叫您的排程時呼叫任何目標 API 時。依預設， EventBridge 排程器會在加密傳輸中的資料時使用 TLS 1.2。您不需要在傳輸過程中設定加密，也無法在使用 EventBridge 排程器時選擇不同的 TLS 版本。

使用 EventBridge 排程器 API — 當您執行 API 作業 (例如) 時`CreateSchedule`， EventBridge 排程器會加密整個 HTTP 要求，包括要求主體和標頭。 EventBridge 排程器也會加密您從我們的 API 收到的整個回應物件。

使用目標 API — EventBridge 排程器呼叫您的排程時，會呼叫您在建立排程時指定的目標 API。將事件傳送至目標時， EventBridge Scheduler 會加密整個要求，包括要求主體和所有標頭，以及從目標接收到的回應。

Amazon EventBridge 排程器的合規驗證

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 可協助您因應各種合規性需求，例如 PCI DSS，滿足特定合規性架構所規定的入侵偵測需求。
- 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 EventBridge 排程器的彈性

AWS 全球基礎架構是圍繞 AWS 區域 和可用區域建立的。 AWS 區域 提供多個實體分離和隔離的可用區域，這些區域透過低延遲、高輸送量和高度備援的網路連線。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、容錯能力和擴展能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域 和可用區域的詳細資訊，請參閱AWS 全域基礎結構。

除了 AWS 全球基礎架構之外， EventBridge Scheduler 還提供多種功能，協助支援您的資料恢復能力和備份需求。

Amazon EventBridge 排程器中的基礎設施安

作為受管服務，Amazon EventBridge 排程器受到 AWS 全球網路安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎結構的詳細資訊，請參閱AWS 雲端安全 若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱安全性支柱架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的 API 呼叫透過網路存取 EventBridge 排程器。使用者端必須支援下列專案：

- 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。

亞馬遜 EventBridge 排程器的監控和指標

監控是維護 Amazon EventBridge Scheduler 及其他AWS解決方案的可靠性、可用性和可用性所不可或缺。 AWS提供了以下監控工具來監看 EventBridge Scheduler，這些工具會在發生錯誤時回報，並自動適時採取動作：

- Amazon 會即時 CloudWatch監控您的AWS資源，以及您在上即時執行的資源，以及您AWS在上即時 您可以收集和追蹤指標、建立自訂儀表板，以及設定警示，在特定指標達到您指定的閾值時通知您或採取動作。如需詳細資訊，請參閱 Amazon Amazon Amazon Amazon Amazon CloudWatch 使用者指南。
- AWS CloudTrail 擷取您 AWS 帳戶發出或代表發出的 API 呼叫和相關事件，並傳送記錄檔案至您指定的 Simple Storage Service (Amazon S3) 儲存貯體。您可以找出哪些使用者和帳戶呼叫 AWS、發出呼叫的來源 IP 地址，以及呼叫的發生時間。如需詳細資訊，請參閱《AWS CloudTrail 使用者指南》。

主題

- 用 Amazon 監控 Amazon EventBridge 調度 CloudWatch
- 使用記錄亞馬遜 EventBridge 排程器 API 呼叫AWS CloudTrail

用 Amazon 監控 Amazon EventBridge 調度 CloudWatch

您可以使用監控 Amazon EventBridge 排程器 CloudWatch，該排程器會收集原始資料並將其處理為可讀且接近即時的指標。 EventBridge 排程器會針對所有排程發出一組量度，並針對具有關聯無效字母佇列 (DLQ) 的排程發出一組額外的量度。如果您為排程設定 DLQ，排程器會在 EventBridge 排程用盡其重試原則時發佈其他指標。

這些統計資料會保留 15 個月，因此您可以存取歷史資訊，並更好地瞭解排程失敗的原因，並針對潛在問題進行疑難排解。您也可以設定留意特定閾值的警示，當滿足這些閾值時傳送通知或採取動作。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。

主題

- 條款
- 維度
- 存取 指標
- 指標清單

- EventBridge 排程器 使用量度

條款

命名空間

命名空間是 AWS 服務指 CloudWatch 標的容器。若為「 EventBridge 排程器」，命名空間為`AWS/Scheduler`。

CloudWatch 度量

CloudWatch 量度代表特定於的一組時間順序的資料點。 CloudWatch

維度

維度是一組名稱值對，是指標身分的一部分。

單位

統計資料具有測量單位。對於 EventBridge 「排程器」，單位包括計數。

維度

本節說明中「 EventBridge 排程器」量度的 CloudWatch 維度群組 CloudWatch。

維度	描述
ScheduleGroup	您要使用檢視其量度的排程群組 CloudWatch。如果您尚未建立任何群組， EventBridge Scheduler 會將您的排程與`default`群組建立關聯。

存取 指標

本節說明如何存取特定「排程器」 EventBridge 排程 CloudWatch 的效能測量結果。

檢視維度的效能測量結果

1. 在主控台上開啟「測量結果」頁 CloudWatch 面。

2. 使用「 AWS 地區」選取器選擇排程的「地區」
3. 選擇排程器命名空間。
4. 在「所有測量結果」標籤中，選擇維度，例如「排程群組量度」。若要查看您在所選區域中建立的所有排程的量度，請選擇「帳戶指標」。
5. 選擇維 CloudWatch 度的量度。例如，「InvocationAttempt計數」或「計InvocationDropped數」，然後選擇「圖表搜尋」。
6. 選擇「圖形測量結果」頁籤，檢視「 EventBridge 排程器」測量結果的效能統計

指標清單

下表列出所有「排程器」 EventBridge 排程的量度，以及您已設定 DLQ 之排程的其他量度。

所有排程的量度

命名空間	指標	單位	描述
AWS/Scheduler	InvocationAttemptCount	計數	為每次調用嘗試發出。使用此量度可檢查 EventBridge Scheduler 是否嘗試呼叫您的排程，以及查看呼叫何時接近您的帳戶配額。
AWS/Scheduler	TargetErrorCount	計數	當 EventBridge 排程器呼叫目標 API 之後，目標傳回例外狀況時發出。使用此選項可檢查傳遞至目標失敗的時間。
AWS/Scheduler	TargetErrorThrottledCount	計數	當目標調用由於目標 API 節流而失敗時發出。當基本原因是 Scheduler 進行的目標 API 節流呼叫時，使用

命名空間	指標	單位	描述
			此功能來診斷傳遞失敗 EventBridge
AWS/Scheduler	InvocationThrottleCount	計數	當排程器因為超出「 EventBridge 排程器」設定的服務配額而限制目標叫用時發出。 EventBridge 使用此選項來判斷您何時已超過 EventBridge 排程器配額。如需服務配額的詳細資訊，請參閱配額。
AWS/Scheduler	InvocationDroppedCount	計數	當排程器在 EventBridge 排程的重試原則用盡之後，停止嘗試呼叫目標時發出。如需重試原則的詳細資訊，請參閱EventBridge 排程器 API 參考RetryPolicy中的。

使用 DLQ 的排程量度

命名空間	指標	單位	描述
AWS/Scheduler	InvocationsSentToDeadLetterCount	計數	為每次成功傳遞至排程的 DLQ 而發出。使用此選項可判斷何時傳送事件至 DLQ，然後檢查傳送至排

命名空間	指標	單位	描述
			程 DLQ 的事件，以取得可協助您判斷失敗原因的其他詳細資料。

命名空間	指標	單位	描述
AWS/Scheduler	InvocationsFailedToBeSentToDeadLetterCount	計數	當 EventBridge 排程器無法將事件傳遞至 DLQ 時發出。使用這兩個度量來判斷 EventBridge 排程器無法將事件傳送至 DLQ 的原因，並修改 DLQ 組態以解決問題。 以下是指定為 DLQ 的 Amazon SQS 佇列不存在時的指InvocationsFailedToBeSentToDeadLetterCount_<error_code> 標範例：InvocationsFailedToBeSentToDeadLetterCount_ **AWS.SimpleQueueService.NonE**
AWS/Scheduler	InvocationsFailedToBeSentToDeadLetterCount_<error_code>	計數	

命名空間	指標	單位	描述
			xistentQueue
AWS/Scheduler	InvocationsSentToDeadLetterCount_Truncated_MessageSizeExceeded	計數	當傳送至 DLQ 的事件有效負載超過 Amazon SQS 允許的大小上限，且 EventBridge 排程器會截斷您在排程屬性中指定的承載時發出Input。

EventBridge 排程器 使用量度

CloudWatch 收集跟踪某些 AWS 資源使用情況的指標。這些量度對應於 AWS 服務配額。追蹤這些指標可協助您主動管理配額。使用下列指標來判斷您何時已超過 EventBridge 排程器配額。如需服務配額的詳細資訊，請參閱配額。

這些測量結果包含在AWS/Usage命名空間中AWS/Scheduler，而不是每分鐘收集一次。

目前，此命名空間中唯一 CloudWatch 發佈的測量結果名稱是CallCount。此指標會與維度 Resource、Service 和 Type 一起發佈。Resource 維度指定要追蹤之 API 操作的名稱。

例如，具有下列維度的CallCount量度表示在您的帳戶中呼叫 EventBridge 排程器 CreateSchedule API 作業的次數：

- 「服務」：「排程器」
- 「類型」:「API」
- 「資源」：「CreateSchedule」

CallCount 指標沒有指定的單位。指標最實用的統計資訊是 SUM，代表 1 分鐘期間的總操作計數。

指標

指標	描述		
CallCount	在您的帳戶中執行的指定操作數目。		

維度

維度	描述		
Service	包含資源的 AWS 服務名稱。 對於 EventBridge 排程器 使用狀況測量結果，此維度的值為Scheduler。		
Class	正在追蹤的資源類別。 EventBridge 排程器 API 使用量度使用此維度的值為None。		
Type	正在追蹤的資源類型。 目前，當 Service 維度為 Scheduler ，Type 的唯一有效值為 API。		
Resource	API 操作的名稱。有效值包括以下項目： • CreateSchedule • CreateScheduleGroup • DeleteSchedule • DeleteScheduleGroup • GetSchedule • GetScheduleGroup • ListScheduleGroups		

維度	描述		
	• ListSchedulesCallCount • ListTagsForResource • TagResource • UntagResource • UpdateSchedule		

使用記錄亞馬遜 EventBridge 排程器 API 呼叫AWS CloudTrail

Amazon EventBridge Scheduler 已與整合AWS CloudTrail，這項服務可提供由使用者、角色或 EventBridge 排程器中AWS服務所採取之動作的記錄。 CloudTrail 擷取 EventBridge 排程的 API 呼叫擷取為事件。擷取的呼叫包括來自 EventBridge 排程器主控台的呼叫，以及針對 EventBridge 排程器 API 操作的程式碼呼叫。如果您建立追蹤，就可以將 CloudTrail 事件持續交付至 Amazon S3 S3 儲存貯體，包括 EventBridge 排程器的事件。即使沒設定追蹤，您依然可以在 CloudTrail 主控台的事件歷史記錄中檢視最新的事件。您可以使用 CloudTrail收集的資訊來判斷向 S EventBridge cheduler 發出的請求，以及發出請求的 IP 地址、人員、時間和其他詳細資訊。

若要進一步了解 CloudTrail，請參閱使AWS CloudTrail用者指南。

EventBridge 排程器資訊 CloudTrail

CloudTrail 當您建立帳戶AWS 帳戶時，系統即會在中啟用。此外， EventBridge Scheduler 中發生活動時，系統便會將該活動記錄至 CloudTrail 事件，並將其他AWS服務事件記錄到事件中。您可以檢視、搜尋和下載 AWS 帳戶 的最新事件。如需詳細資訊，請參閱使用 CloudTrail 事件歷史記錄檢視事件。

若要持續記錄您的事件AWS 帳戶，包括 EventBridge 排程的事件，請建立追蹤。線索能 CloudTrail 將日誌檔案交付至 Amazon S3 S3 S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存貯體。此外，您還能設定其他AWS服務，以進一步分析和處理 CloudTrail 日誌中收集的事件資料，並採取相應動作。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
- 設定的 Amazon SNS 通通通通通通 CloudTrail

- 接收多個區域的 CloudTrail 日誌檔案及接收多個帳戶的 CloudTrail 日誌檔案

所有 EventBridge 排程器動作都會由 Amazon 排程器 API 參考記錄， CloudTrail 並記錄在 Amazon EventBridge 排程器 API 參考中。例如，呼叫`UpdateSchedule`和`DeleteSchedule`動作會`CreateSchedule`在 CloudTrail 記錄檔中產生項目。

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：

- 該請求是否透過根或 AWS Identity and Access Management (IAM) 使用者憑證來提出。
- 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。
- 該請求是否由另一項 AWS 服務提出。

如需詳細資訊，請參閱 CloudTrail userIdentity 元素。

了解 EventBridge 排程器日誌檔案項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付至您指定的 Amazon S3 儲存貯體。 CloudTrail 日誌檔案包含一個或多個日誌檔案項目 一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。 CloudTrail 日誌檔案並非依公 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

Amazon EventBridge 排程器的配額

您的 AWS 帳戶有每項 AWS 服務的預設配額 (先前稱為限制)。除非另有說明，否則每個配額都是區域特定規定。您可以要求增加某些配額，有些配額無法增加。

若要檢視 EventBridge 排程器的配額，請開啟「Service Quotas」主控台。在導覽窗格中，選擇「AWS 服務」，然後選取「EventBridge 排程器」。

若要請求提高配額，請參閱《Service Quotas 使用者指南》中的請求提高配額。如果 Service Quotas 中尚未提供配額，請使用增加服務配額表單。

> **Note**
>
> EventBridge 排程器的`CreateSchedule`、`UpdateSchedule`、`GetSchedule`和每秒`DeleteSchedule`交易 (TPS) 配額最多可調整數千個 TPS。調用節流配額最多可調節到數萬個 TPS。

您的 AWS 帳戶具有下列與 EventBridge 排程器相關的配額。

名稱	預設	可調整	描述
CreateSchedule 請求率	每個受支援的區域：50	是	每秒 CreateSchedule 要求數上限。當您達到此配額時， EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
CreateScheduleGroup 請求率	每個受支援的區域：10	是	每秒 CreateScheduleGroup 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。

名稱	預設	可調整	描述
DeleteSchedule 請求率	每個受支援的區域：50	是	每秒 DeleteSchedule 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
DeleteScheduleGroup 請求率	每個受支援的區域：10	是	每秒 DeleteScheduleGroup 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
GetSchedule 請求率	每個受支援的區域：50	是	每秒 GetSchedule 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
GetScheduleGroup 請求率	每個受支援的區域：10	是	每秒 GetScheduleGroup 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
調用限流每秒交易的限制	每個受支援的區域：500	是	調用是傳遞到定義目標的計劃有效負載。達到上限之後，就會限流呼叫，也就是說，雖然仍會繼續呼叫，但是會延遲一些。

名稱	預設	可調整	描述
ListScheduleGroups 請求率	每個受支援的區域：10	是	每秒 ListScheduleGroups 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
ListSchedules 請求率	每個受支援的區域：50	是	每秒 ListSchedules 要求數上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。
ListTagsForResource 請求率	每個受支援的區域：10	是	列出與「排程器」資源相關聯的標籤。
排程群組數	每個受支援的區域：500	是	每個區域的排程群組數目上限。
排程數	每個受支援的區域：1,000,000	是	每個區域的最大排程數目。此配額包括已完成執行的一次性排程。建議您在一次性排程完成執行並呼叫目標之後刪除它們。
TagResource 請求率	每個受支援的區域：1	是	將一或多個標籤 (鍵值配對) 指定給指定的排程器資源。
UntagResource 請求率	每個受支援的區域：1	是	從指定的「排程器」資源移除一或多個標籤。

名稱	預設	可調整	描述
UpdateSchedule 請求率	每個受支援的區域：50	是	每秒 UpdateSchedule 要求數上限。當您達到此配額時， EventBridge 排程器會在剩餘的間隔內拒絕此作業的要求。

如需 EventBridge 排程器配額和服務端點的詳細資訊，請參閱AWS 一般參考指南中的 Amazon EventBridge 排程器端點和配額。

此版本的說明文件 EventBridge 排程器

下表說明此版本版本的說明文件。 EventBridge 排程器。

變更	描述	日期
執行角色的變化和混淆副預防	此更新描述當您在角色的權限原則中實作混淆副預防時，如何將執行角色套用至排程群組資源的變更。 • the section called “預防混淆代理人”	2023 年 9 月 7 日
完成後自動刪除排程	EventBridge 排程器支援自動刪除。當您設定自動刪除時，EventBridge 排程器會在上次計劃的呼叫後刪除您的排程。 • the section called “排程完成後刪除”	2023 年 8 月 2 日
更新了使用通用目標的主題	更新了支持的服務列表 EventBridge 排程器可以定位並與之整合。此版本更新的說明此版本。GETAPI 操作，並包括對通用目標示例的改進，以及對整個指南的其他小改進。 • the section called “使用通用目標”	2023 年 3 月 17 日
更新沒有開始日期之以費率為基準的排程資訊	添加了參數。 EventBridge 排程器會處理以速率為基礎的排程 (如果您未指定)`StartDate`。	2023 年 3 月 17 日

	• the section called “以速率為基礎的”	
管理排程器群組的新主題	增加了有關如何創建調度程序組的新章節 EventBridge 排程器。使用本章瞭解如何建立群組、將排程新增至群組、套用標籤以便更輕鬆地管理和 monirot EventBridge 調度程序資源，最後刪除一個組。 • 管理排程群組	2023 年 3 月 17 日
夏令時和時區的新主題	添加了描述如何新的部分 EventBridge 排程器會處理日光節約時間，以及如何在不同的時區建立排程。 • the section called “日光節約時間” • the section called “时区”	2022 年 11 月 17 日
有關指標的新主題	已新增描述量度的新主題 EventBridge 排程器發佈至 CloudWatch。您可以使用這些指標來監視呼叫失敗，並瞭解如何解決排程的問題。 • the section called “使用監控 CloudWatch”	2022 年 11 月 15 日
初始版本	初始版本 EventBridge 排程程式。	2022 年 11 月 10 日

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。