



使用者指南

# EventBridge 排程器



# EventBridge 排程器: 使用者指南

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

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

# Table of Contents

什麼是 EventBridge Scheduler ? .....	1
EventBridge 排程器的主要功能 .....	1
存取 EventBridge 排程器 .....	1
設定 .....	2
註冊 AWS .....	2
建立 IAM 使用者 .....	2
使用 受管政策 .....	3
設定執行角色 .....	4
設定目標 .....	8
後續步驟? .....	10
開始使用 .....	11
先決條件 .....	11
使用主控台 .....	12
使用 AWS CLI .....	15
使用 SDK .....	15
後續步驟? .....	16
排程類型 .....	18
以速率為基礎的排程 .....	18
語法 .....	19
範例 .....	19
以 Cron 為基礎的排程 .....	19
語法 .....	20
範例 .....	21
一次性排程 .....	21
語法 .....	21
範例 .....	22
時區 .....	22
日光節約時間 .....	22
管理排程 .....	24
變更排程狀態 .....	24
設定彈性時段 .....	25
設定 DLQ .....	27
建立 Amazon SQS 佇列 .....	27
設定執行角色許可 .....	28

指定無效字母佇列 .....	29
擷取無效字母事件 .....	30
刪除排程 .....	33
排程完成後刪除 .....	33
手動刪除 .....	34
後續步驟? .....	34
管理排程群組 .....	35
建立排程群組 .....	35
步驟一：建立新的排程群組 .....	36
關聯排程 .....	37
刪除排程群組 .....	38
相關資源 .....	40
管理目標 .....	41
使用範本目標 .....	41
Amazon SQS SendMessage .....	43
Lambda Invoke .....	45
步驟函數 StartExecution .....	47
使用通用目標 .....	49
不支援的動作 .....	49
範例 .....	50
新增內容屬性 .....	52
後續步驟? .....	53
AWS PrivateLink .....	54
考量事項 .....	54
建立介面端點 .....	54
建立端點政策 .....	54
安全 .....	56
管理存取 .....	56
目標對象 .....	57
使用身分驗證 .....	57
使用政策管理存取權 .....	58
與 IAM 整合 .....	59
使用身分型政策 .....	64
預防混淆代理人 .....	74
疑難排解 .....	76
資料保護 .....	77

靜態加密 .....	78
傳輸中加密 .....	84
法規遵循驗證 .....	84
恢復能力 .....	84
基礎設施安全性 .....	85
監控和指標 .....	86
使用 CloudWatch 進行監控 .....	86
條款 .....	87
維度 .....	87
存取 指標 .....	87
指標清單 .....	88
用量指標 .....	92
使用 CloudTrail 日誌進行監控 .....	95
CloudTrail 中的 EventBridge 排程器資訊 .....	96
了解 EventBridge Scheduler 日誌檔案項目 .....	96
配額 .....	97
對配額進行故障診斷 .....	106
ServiceQuotaExceededException .....	106
疑難排解 .....	108
目標錯誤 .....	108
常見原因： .....	108
疑難排解步驟 .....	108
角色許可 .....	110
常見原因 .....	110
徵狀 .....	110
疑難排解步驟 .....	110
Service Quotas .....	112
識別配額問題 .....	113
解決配額問題 .....	113
模式和觸發時間 .....	113
常見原因 .....	113
疑難排解步驟 .....	113
建立模式 .....	115
常見問題 .....	115
疑難排解步驟 .....	115
我的目標是否被觸發？ .....	115

---

範本化與通用目標 .....	116
無效的通用目標輸入 .....	116
徵狀 .....	116
範例 .....	116
如何解決 .....	117
觸發非預期調用的排程更新 .....	118
停用或啟用一次性排程 .....	118
文件歷史紀錄 .....	119
.....	cxxii

# 什麼是 Amazon EventBridge Scheduler ？

Amazon EventBridge 排程器是無伺服器排程器，可讓您從單一受管的中央服務建立、執行及管理任務。EventBridge Scheduler 具有高度可擴展性，可讓您排程數百萬個可叫用超過 270 個 AWS 服務和超過 6,000 個 API 操作的任務。EventBridge Scheduler 不需要佈建和管理基礎設施，也不需要與多個服務整合，可讓您大規模交付排程並降低維護成本。

EventBridge Scheduler 可靠地交付您的任務，並內建機制，可根據下游目標的可用性調整您的排程。使用 EventBridge 排程器，您可以使用週期性模式的 Cron 和 Rate 表達式來建立排程，或設定一次性調用。您可以設定彈性的交付時段、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](#)
- [存取 EventBridge 排程器](#)

## EventBridge 排程器的主要功能

EventBridge Scheduler 提供下列主要功能，可用來設定目標和擴展排程。

- 範本化目標 – EventBridge Scheduler 支援範本化目標，以使用 Amazon SQS、Amazon SNS、Lambda 和 EventBridge 執行常見的 API 操作。使用預先定義的目標，您可以使用 EventBridge Scheduler 主控台、EventBridge Scheduler SDK 或快速設定排程 AWS CLI。
- 通用目標 – EventBridge Scheduler 提供通用目標參數 (UTP)，可用來建立自訂觸發，以排程超過 270 個 AWS 服務和超過 6,000 個 API 操作為目標。透過 UTP，您可以使用 EventBridge Scheduler 主控台、EventBridge Scheduler SDK 或來設定自訂觸發 AWS CLI。
- 彈性時段 – EventBridge Scheduler 支援彈性時段，可讓您分散排程，並針對不需要精確排程調用目標的使用案例改善觸發程序的可靠性。
- 重試 - EventBridge Scheduler at-least-once 事件交付至目標，這表示至少有一個交付成功與來自目標的回應。EventBridge Scheduler 可讓您為失敗的任務設定排程的重試次數。EventBridge Scheduler 會重試失敗的任務，並延遲嘗試改善排程的可靠性，並確保目標可用。

## 存取 EventBridge 排程器

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

# 設定 Amazon EventBridge 排程器

您必須先完成下列步驟，才能使用 EventBridge 排程器。

## 主題

- [註冊 AWS](#)
- [建立 IAM 使用者](#)
- [使用 受管政策](#)
- [設定執行角色](#)
- [設定目標](#)
- [後續步驟？](#)

## 註冊 AWS

如果您沒有 AWS 帳戶，請完成下列步驟來建立一個。

### 註冊 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或簡訊，並在電話鍵盤輸入驗證碼。

當您註冊時 AWS 帳戶，AWS 帳戶根使用者會建立。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為安全最佳實務，請將管理存取權指派給使用者，並且僅使用根使用者來執行[需要根使用者存取權的任務](#)。

## 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	根據	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 <a href="#">IAM 安全最佳實務</a> 。	請遵循 AWS IAM Identity Center 使用者指南的 <a href="#">入門</a> 中的說明。	透過在 AWS Command Line Interface 使用者指南中設定 <a href="#">AWS CLI 以使用來設定 AWS IAM Identity Center</a> 程式設計存取。
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 <a href="#">建立 IAM 使用者以進行緊急存取</a> 的指示。	請依照《IAM 使用者指南》中的 <a href="#">管理 IAM 使用者的存取金鑰</a> 設定以程式設計方式存取。

## 使用 受管政策

在上一個步驟中，您會使用登入資料設定 IAM 使用者來存取您的 AWS 資源。在大多數情況下，若要安全地使用 EventBridge 排程器，建議您建立單獨的使用者、群組或角色，只具有使用 EventBridge 排程器的必要許可。EventBridge 排程器支援下列常見使用案例的受管政策。

- AmazonEventBridgeSchedulerFullAccess – 使用主控台和 API 授予 EventBridge 排程器的完整存取權。
- AmazonEventBridgeSchedulerReadOnlyAccess – 授予 EventBridge 排程器的唯讀存取權。

您可以將這些受管政策連接到您的 IAM 主體，就像您在上一個步驟中連接 AdministratorAccess 政策一樣。如需使用身分型 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 排程器擔任的 IAM 角色，以便 AWS 服務 代表您與其他 互動。您可以將許可政策連接至此角色，以授予 EventBridge 排程器調用目標的存取權。

當您使用主控台建立新的[排程時，也可以建立新的](#)執行角色。如果您使用 主控台，EventBridge 排程器會根據您選擇的目標，以 許可代表您建立角色。當 EventBridge 排程器為您建立角色時，角色的信任政策包含[條件索引鍵](#)，以限制哪些委託人可以代表您擔任該角色。這可防止潛在的[混淆代理人安全問題](#)。

下列步驟說明如何建立新的執行角色，以及如何授予 EventBridge 排程器調用目標的存取權。本主題說明熱門範本目標的許可。如需新增其他目標許可的詳細資訊，請參閱 [the section called “使用範本目標”](#)。

使用 建立執行角色 AWS CLI

1. 複製下列擔任角色 JSON 政策，並將其儲存為 Scheduler-Execution-Role.json。此信任政策允許 EventBridge 排程器代表您擔任該角色。

JSON

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

### Important

若要在生產環境中設定執行角色，建議您實作額外的保護措施，以防止混淆代理人問題。如需詳細資訊和範例政策，請參閱 [the section called “預防混淆代理人”](#)。

2. 從 AWS Command Line Interface (AWS CLI) 輸入下列命令來建立新的角色。*SchedulerExecutionRole* 將取代為您要提供此角色的名稱。

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

如果成功，您會看到下列輸出：

```
{
  "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. 若要建立新的政策以允許 EventBridge 排程器叫用目標，請選擇下列其中一個常見目標。複製 JSON 許可政策並將其儲存為 .json 檔案。

### Amazon SQS – SendMessage

以下允許 EventBridge 排程器在您帳戶中的所有 Amazon SQS 佇列上呼叫 `sqs:SendMessage` 動作。

#### JSON

```
{
```

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

### Amazon SNS – Publish

以下允許 EventBridge 排程器呼叫您帳戶中所有 Amazon SNS 主題 `sns:Publish` 的動作。

JSON

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

### Lambda – Invoke

以下允許 EventBridge Scheduler 呼叫您帳戶中所有 Lambda 函數 `lambda:InvokeFunction` 的動作。

JSON

```
{
```

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

- 執行下列命令來建立新的許可政策。*PolicyName* 將取代為您要提供此政策的名稱。

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

如果成功，您會看到下列輸出。請注意政策 ARN。您可以在下一個步驟中使用此 ARN，將政策連接至我們的執行角色。

```
{
  "Policy": {
    "PolicyName": "PolicyName",
    "CreateDate": "2022-03-01T19:31:18.620Z",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
    "DefaultVersionId": "v1",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:policy/PolicyName",
    "UpdateDate": "2022-03-01T19:31:18.620Z"
  }
}
```

- 執行下列命令，將政策連接至您的執行角色。*your-policy-arn* 以您在上一個步驟中建立的政策 ARN 取代。*SchedulerExecutionRole* 以執行角色的名稱取代。

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

attach-role-policy 操作不會在命令列傳回回應。

## 設定目標

建立 EventBridge 排程器排程之前，您至少需要一個目標才能叫用排程。您可以使用現有的 AWS 資源，或建立新的資源。下列步驟說明如何使用 建立新的標準 Amazon SQS 佇列 CloudFormation。

### 建立新的 Amazon SQS 佇列

1. 複製下列 JSON CloudFormation 範本，並將其儲存為 Scheduler-Target-SQS.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 AWS CLI 執行下列命令，從 Scheduler-Target-SQS.json 範本建立 CloudFormation 堆疊。

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

如果成功，您會看到下列輸出：

```
{  
  "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 CloudFormation 堆疊的摘要資訊。此資訊包含堆疊的狀態和範本中指定的輸出。

```
$ 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"
}
}
]
```

稍後在本指南中，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](#)指南來建立您的第一個 EventBridge 排程器排程器並叫用目標。

# EventBridge 排程器入門

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 SDKs 來建立具有範本 Amazon SQS 目標的排程。然後，您將設定記錄、設定重試，並設定失敗任務的最長保留時間。建立排程後，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，建議您設定具有 [中所述最低必要許可的 IAM 使用者](#) [the section called “使用身分型政策”](#)。在您建立和設定使用者之後，請執行下列命令來設定您的存取憑證。您需要存取金鑰 ID 和私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
```

如需不同登入資料設定方式的詳細資訊，請參閱《第 AWS Command Line Interface 2 版使用者指南》中的 [組態設定和優先順序](#)。

## 主題

- [先決條件](#)
- [使用 EventBridge 排程器主控台建立排程](#)
- [使用 建立排程 AWS CLI](#)
- [使用 EventBridge 排程器 SDKs 建立排程](#)
- [後續步驟？](#)

## 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 管理主控台，然後選擇以下連結，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 管理主控台的區域選擇器 **AWS 區域** 來切換您的。

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. 針對描述 - 選用，輸入排程的描述。例如：**My first schedule**。
  - c. 對於排程群組，請從下拉式清單選項中選擇排程群組。如果您先前尚未建立任何排程群組，您可以選擇排程的default群組。若要建立新的排程群組，請在主控台描述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. 在排程模式區段中，執行下列動作：
  - a. 針對發生，選擇下列其中一個模式選項。組態選項會根據您選擇的模式而變更。
    - 一次性排程 – 一次性排程只會在您指定的日期和時間叫用目標一次。

針對日期和時間，以 YYYY/MM/DD 格式輸入有效的日期。然後，以 24 小時hh:mm格式指定時間戳記。最後，從下拉式清單選項中選擇時區。
    - 週期性排程 – 週期性排程會以您使用cron表達式或速率表達式指定的速率叫用目標。

選擇以 Cron 為基礎的排程，以使用cron表達式設定排程。若要使用速率表達式，請選擇以速率為基礎的排程，並輸入值的正數，然後從下拉式清單選項中選擇單位。


如需使用 Cron 和 Rate 表達式的詳細資訊，請參閱 [排程類型](#)。
  - b. 對於彈性時段，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段。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. 如果您在上一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。沒有開始日期的週期性排程會在建立和可用時立即開始。沒有結束日期的週期性排程將繼續無限期地調用其目標。
6. 選擇 Next (下一步)。
7. 在選取目標頁面上，執行下列動作：
  - a. 選取範本目標，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 SendMessage 區段中，針對 SQS 佇列，`arn:aws:sqs:us-west-2:123456789012:TestQueue` 從下拉式清單中選擇現有的 Amazon SQS 佇列 ARN，例如。若要建立新的佇列，請選擇建立新的 SQS 佇列以導覽至 Amazon SQS 主控台。建立佇列完成後，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，並且可以選取。
  - c. 針對目標，輸入您希望 EventBridge Scheduler 交付至目標的承載。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，針對啟用排程，使用開關開啟或關閉功能。根據預設，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 EventBridge Scheduler 在排程完成後採取的動作：
    - 如果您想要自動刪除排程，請選擇刪除。對於一次性排程，這會在排程調用目標一次之後發生。對於週期性排程，這發生在排程的最後一次計劃調用之後。如需自動刪除的詳細資訊，請參閱[the section called “排程完成後刪除”](#)。
    - 如果您不希望 EventBridge Scheduler 在排程完成後採取任何動作，請選擇 NONE，或不選擇值。
  - c. 在重試政策和無效字母佇列 (DLQ) 區段中，針對重試政策，開啟重試以設定排程的重試政策。設定好重試政策後，如果排程無法調用其目標，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. 針對事件的最長存留期 - 選用，輸入 EventBridge 排程器必須保留未處理事件的最大小時數 (h) 和分鐘數 (m)。

**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 排程器重新嘗試執行排程的次數上限。

 Note

最大值為重試 185 次。

- f. 針對無效佇列 (DLQ)，請從下列選項中選擇：
  - 無 – 如果您不想設定 DLQ，請選擇此選項。
  - 在我的 AWS 帳戶中選取 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 AWS 帳戶與您建立排程的 DLQ 相同的 DLQ。
  - 將其他 AWS 帳戶中的 Amazon SQS 佇列指定為 DLQ – 選擇此選項，然後輸入佇列的 ARN，如果佇列位於另一個 DLQ AWS 帳戶。您必須輸入佇列的確切 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），以使用客戶受管 KMS 金鑰來加密您的目標輸入。如果您選擇此選項，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. 針對許可，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 主控台來建立新的角色。

如果您希望 EventBridge Scheduler 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 Scheduler 會將範本目標所需的許可新增至角色。

10. 選擇 Next (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的排程。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. 若要驗證您的排程是否叫用 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. 在傳送和接收訊息頁面的接收訊息下，選擇輪詢訊息，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

下列範例示範如何使用 AWS CLI 命令 [create-schedule](#)，以建立具有範本 Amazon SQS 目標的 EventBridge 排程器排程。將下列參數的預留位置值取代為您的資訊：

- `--name` – 輸入排程的名稱。
- `RoleArn` – 輸入您要與排程建立關聯的執行角色 ARN。
- `Arn` – 輸入目標的 ARN。在此情況下，目標為 Amazon SQS 佇列。
- `輸入` – 輸入 EventBridge Scheduler 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 SDKs 建立排程

在下列範例中，您可以使用 EventBridge 排程器 SDKs 建立具有範本 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？

- 如需使用 主控台 AWS CLI 或 EventBridge 排程器 SDK 管理排程的詳細資訊，請參閱 [管理排程](#)。

- 如需如何設定範本目標並了解如何使用通用目標參數的詳細資訊，請參閱[管理目標](#)。
- 如需 EventBridge 排程器資料類型和 API 操作的詳細資訊，請參閱 [EventBridge 排程器 API 參考](#)。

# EventBridge 排程器中的排程類型

下列主題說明 Amazon EventBridge 排程器支援的不同排程類型，以及 EventBridge 排程器如何處理日光節約時間和在不同時區排程。您可以在設定排程時從三種排程類型中選擇：以速率為基礎、以 Cron 為基礎和一次性排程。

速率型和 Cron 型排程都是週期性排程。您可以使用您要設定之排程類型的排程表達式來設定每個週期性排程類型，並指定 EventBridge 排程器評估表達式的時區。

一次性排程是只叫用目標一次的排程。您可以在指定 EventBridge 排程器評估排程的時間、日期和時區時設定一次性排程。

## Note

EventBridge 排程器上的所有排程類型會以 60 秒的精確度叫用其目標。這表示如果您將排程設定為在執行 1:00，則假設未設定彈性時段 1:00:59，它會在 1:00:00 和 之間叫用目標 API。

使用下列各節來了解如何為每個週期性排程類型設定排程表達式，以及如何在 EventBridge 排程器上設定一次性排程。

## 主題

- [以速率為基礎的排程](#)
- [以 Cron 為基礎的排程](#)
- [一次性排程](#)
- [EventBridge 排程器上的時區](#)
- [EventBridge 排程器上的日光節約時間](#)

## 以速率為基礎的排程

以速率為基礎的排程會在您為排程指定的開始日期之後開始，並以您定義的一般速率執行，直到排程的結束日期為止。您可以使用以速率為基礎的排程來設定最常見的週期性排程使用案例。例如，如果您想要每 15 分鐘、每 2 小時或每 5 天叫用其目標的排程，您可以使用以速率為基礎的排程來達成此目標。您可以使用速率表達式設定以速率為基礎的排程。

透過以速率為基礎的排程，您可以使用 `StartDate` 屬性來設定排程的第一次出現。如果您未提供 `StartDate`，您的排程會立即開始叫用目標。

速率表達式有兩個必要欄位以空格分隔，如下所示。

## 語法

```
rate(value unit)
```

**value**

正數。

**單位**

您希望排程調用其目標的時間單位。

有效輸入：minutes | hours | days

## 範例

下列範例示範如何搭配 AWS CLI `create-schedule` 命令使用速率表達式來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 `SqsParameters` 目標類型將訊息傳送到 Amazon SQS 佇列。

由於此範例不會設定 `--start-date` 參數的值，因此排程會在您建立並啟用後立即開始調用其目標。

```
$ 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 表達式會建立精細的週期性排程，在您選擇的特定時間執行。EventBridge 排程器支援在國際標準時間 (UTC) 或您建立排程時指定的時區中設定 cron 型排程。使用 cron 型排程，您可以更好地控制排程執行的時間和頻率。當您需要其中一個 EventBridge 排程器速率表達式不支援的自訂週期排程時，請使用 cron 型排程。例如，您可以建立在上午 8:00 執行的 cron 型排程。太平洋標準時間為每個月的第一個星期一。您可以使用 Cron 表達式設定 Cron 型排程。

Cron 表達式包含以空格分隔的五個必要欄位：分鐘、小時、day-of-month、月中日、day-of-week，以及每年一個選用欄位，如下所示。

## 語法

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

欄位	Values (數值)	萬用字元
分鐘	0-59	, - * /
小時	0-23	, - * /
月中的日	1-31	, - * ? / L W
月	1-12 或 JAN-DEC	, - * /
週中的日	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。在月欄位，JAN、FEB、MAR 包括一月、二月與三月。
- - (破折號) 萬用字元用於指定範圍。在日欄位，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。您無法在月中的特定一天和週中的特定一天兩個欄位同時使用 \*。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。在月中的日欄位，您可以輸入 7，如果您不在意這個月的 7 號是星期幾，就可以在月中的日欄位中輸入 ?。
- L 萬用字元在 Day-of-month (月中的日) 或 Day-of-week (週中的日) 欄位可指定月份或週的最後一天。
- W 萬用字元在 Day-of-month (月中的日) 欄位可指定工作日。在 Day-of-month (月中的日) 欄位，3W 指定的是月份中最接近第三個工作日的日子。

- # 萬用字元在 Day-of-week (週中的日) 欄位可指定某個月中某週特定日子的特定執行個體。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，則只能在星期幾欄位中定義一個表達式。例如："3#1,6#3" 是無效的，因為它被轉譯為兩個表達式。

## 範例

下列範例示範如何使用 cron 表達式搭配 AWS CLI `create-schedule` 命令來設定以 cron 為基礎的排程。此範例會建立在 2022 年至 2023 年期間每個月最後一個星期五上午 10:15 UTC+0 執行的排程，並使用範本 `SqsParameters` 目標類型將訊息傳送到 Amazon SQS 佇列。

```
$ 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叫用目標一次。EventBridge 排程器支援以國際標準時間 (UTC) 或您在建立排程時指定的時區進行排程。

#### Note

一次性排程在完成執行並叫用其目標之後，仍會計入您的帳戶配額。我們建議在完成執行後[刪除](#)您的一次性排程。

您可以在表達式中使用 `at` 設定一次性排程。`at` 表達式包含您希望 EventBridge 排程器調用排程的日期和時間，如下所示。

## 語法

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

當您設定一次性排程時，EventBridge 排程器會忽略 `StartDate`，而 `EndDate` 您會為排程指定。

## 範例

下列範例示範如何在表達式中使用搭配 AWS CLI `create-schedule` 命令來設定一次性排程。此範例會建立排程，在 2022 年 11 月 20 日下午 1 點 UTC-8 執行一次，並使用範本 `SqsParameters` 目標類型將訊息傳送到 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設定 cron 型和一次性排程。EventBridge 排程器使用國際網路指派號碼授權機構 (IANA) 維護的 [時區資料庫](#)。

使用 AWS CLI，您可以設定您希望 EventBridge 排程器使用 `--schedule-expression-timezone` 參數評估排程的時區。例如，下列命令會建立 cron 型排程，每天上午 8:30 在 `America/New_York` 中調用範本 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排程以節省日光時間。當時間在 Spring 中向前轉移時，如果 Cron 表達式落在不存在的日期和時間，則會略過您的排程調用。當時間在秋天向後轉移時，您的排程只會執行一次，而不會重複其調用。下列叫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，您的排程會在該時區的時間變更時調整，而 `America/Los_Angeles` 中的排程會在三小時後在西岸的時間變更時調整。

對於使用 days 做為單位的速率型排程，例如 `rate(1 days)`，days 代表時鐘上的 24 小時持續時間。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 小時時，EventBridge 排程器仍會在排程的最後一次調用後 24 小時評估速率表達式。

### Note

根據當地規則和法規，某些時區不會遵守日光節約時間。如果您在未遵守日光節約時間的時區建立排程，EventBridge 排程器不會調整您的排程。日光節約時間調整不適用於通用協調時間 (UTC) 中的排程。

## 範例

假設您在 `America/Los_Angeles` 中使用下列 cron 表達式建立排程的情況：`cron(30 2 * * ? *)`。此排程會在指定時區的每天上午 2:30 執行。

- Spring-forward – 當時間在 Spring 中從上午 1:59 向前移動到凌晨 3:00 時，EventBridge 排程器會跳過當天的排程調用，並在第二天繼續正常執行排程。
- 備用 – 當時間從 2:59 a.m. 向後偏移到 2:00 a.m. 時，EventBridge 排程器只會在輪班發生前的 2:30 a.m. 執行一次排程，但不會在時間輪班後 2:30 a.m. 再次重複排程調用。

# 在 EventBridge 排程器中管理排程

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

每個排程都有排程表達式，可決定排程執行的時間和頻率。EventBridge 排程器支援三種類型的排程：速率、cron 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，您可以設定要叫用的排程目標。目標是每次排程執行時 EventBridge 排程器代表您呼叫的 API 操作。EventBridge Scheduler 支援兩種類型的目標：範本目標會呼叫跨核心服務群組的常見 API 操作，以及通用目標參數 (UTP)，您可以用來呼叫跨 270 多個服務呼叫超過 6,000 個操作。如需設定目標的詳細資訊，請參閱 [管理目標](#)。

當 EventBridge 排程器無法成功將事件交付至目標時，您可以使用兩種主要機制來設定排程處理失敗的方式：重試政策和無效字母佇列 (DLQ)。重試政策會決定 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 Amazon SQS 佇列 EventBridge 排程器，用於在重試政策用盡之後，將失敗的事件交付至。您可以使用 DLQ 來疑難排解排程或其下游目標的問題。如需的詳細資訊，請參閱 [the section called “設定 DLQ”](#)。

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

## 主題

- [在 EventBridge 排程器中變更排程狀態](#)
- [在 EventBridge 排程器中設定彈性時段](#)
- [在 EventBridge 排程器中設定排程的無效字母佇列](#)
- [在 EventBridge 排程器中刪除排程](#)
- [後續步驟？](#)

## 在 EventBridge 排程器中變更排程狀態

EventBridge 排程器排程有兩個狀態：啟用和停用。下列範例使用 UpdateSchedule 停用每五分鐘觸發並叫用 Lambda 目標的排程。

使用 UpdateSchedule，您必須提供所有必要的參數。EventBridge 排程器會將您的排程取代為您提供的資訊。如果您未指定先前設定的參數，則預設為 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 SDK 和 UpdateSchedule 操作來停用使用範本目標以 Amazon SQS 為目標的排程。

## 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')
```

## 在 EventBridge 排程器中設定彈性時段

當您使用彈性時段設定排程時，EventBridge 排程器會在您設定的時段內叫用目標。這在不需要精確排程調用目標的情況下非常有用。設定彈性時段可透過分散您的目標調用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 分鐘的彈性時段，則會在排程時間後 15 分鐘內叫用目標。以下 AWS CLI 和 EventBridge 排程器 SDK 範例使用 UpdateSchedule 為每小時執行一次的排程設定 15 分鐘的彈性時段。

### Note

您必須指定是否要設定彈性時段。如果您不想設定此選項，請指定 OFF。如果您將值設定為 FLEXIBLE，則必須指定排程將執行的最大時段。

### 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)
```

## 在 EventBridge 排程器中設定排程的無效字母佇列

Amazon EventBridge 排程器支援使用 Amazon Simple Queue Service 的無效字母佇列 (DLQ)。當排程無法調用其目標時，EventBridge 排程器會將包含調用詳細資訊和從目標收到的任何回應的 JSON 承載交付到您指定的 Amazon SQS 標準佇列。

下列主題將此 JSON 稱為無效字母事件。無效字母事件可讓您對排程或目標的問題進行故障診斷。如果您為排程設定重試政策，EventBridge 排程器會交付已耗盡所設定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 EventBridge 排程器傳遞訊息至 Amazon SQS 所需的許可，以及從 DLQ 接收無效字母事件。

### 主題

- [建立 Amazon SQS 佇列](#)
- [設定執行角色許可](#)
- [指定無效字母佇列](#)
- [擷取無效字母事件](#)

## 建立 Amazon SQS 佇列

為排程設定 DLQ 之前，您必須建立標準 Amazon SQS 佇列。如需使用 Amazon SQS 主控台建立佇列的指示，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 佇列做為排程的 DLQ。

使用下列 AWS CLI 命令來建立標準佇列。

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

如果成功，您會在輸出 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-attributes](#) AWS CLI 命令來尋找佇列 ARN。

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

如果成功，您會在輸出中看到佇列 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您的排程執行角色，以允許 EventBridge 排程器將無效字母事件交付至 Amazon SQS。

## 設定執行角色許可

若要讓 EventBridge 排程器將無效字母事件交付至 Amazon SQS，您的排程執行角色需要下列許可政策。如需將新許可政策連接至排程執行角色的詳細資訊，請參閱[設定執行角色](#)。

JSON

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

**Note**

如果您使用 EventBridge 排程器叫用 Amazon SQS API 目標，您的排程執行角色可能已連接必要的許可。

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

## 指定無效字母佇列

若要指定 DLQ，請使用 EventBridge 排程器主控台或 AWS CLI 來更新現有排程，或建立新的排程。

### Console

#### 使用主控台指定 DLQ

1. 登入 AWS 管理主控台，然後選擇以下連結以開啟 EventBridge 主控台的 EventBridge 排程器區段：<https://console.aws.amazon.com/scheduler/home>
2. 在 EventBridge 排程器主控台上，建立新的排程，或從要編輯的排程清單中選擇現有的排程。
3. 在設定頁面上，針對無效字母佇列 (DLQ)，執行下列其中一項操作：
  - 選擇選取我 AWS 帳戶中的 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。如果您在另一個 AWS 帳戶中選擇佇列，EventBridge 排程器主控台將無法在下拉式清單中顯示佇列 ARNs。
4. 檢閱您的選擇，然後選擇建立排程或儲存排程以完成設定 DLQ。
5. (選用) 若要檢視排程的 DLQ 詳細資訊，請從清單中選擇排程的名稱，然後選擇排程詳細資訊頁面上的無效字母佇列索引標籤。

### AWS CLI

#### 使用更新現有的排程 AWS CLI

- 使用 `update-schedule` 命令來更新您的排程。指定您先前建立的 Amazon SQS 佇列做為 DLQ。指定您連接必要 Amazon SQS 許可的 IAM 角色 ARN 做為執行角色。將所有其他預留位置值取代為您的資訊。

```
$ 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" }'
```

使用 建立具有 DLQ 的新排程 AWS CLI

- 使用 [create-schedule](#) 命令來建立排程。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，您將使用 AWS CLI 接收來自 DLQ 的無效字母事件。

## 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量。

```
$ 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，您會看到類似以下的輸出。

```
{
  "Messages": [
    {
      "MessageId": "2aeg3510-fe3a-4f5a-ab6a-6906560eaf7e",
      "ReceiptHandle": "AQEBkNKTD0MrWgHKPoITRBwrPoK3eCSZICzWvqCY0BZ
+FFtC0RFpopJbtCqj36VbBT1HreM8+qM/m5jcwqS1A1GmIJ0/hYmMgn/
+dwIty9izE7HnpvRhhEyHxbeTZ5V05RbeasYaBdNyi9WLcnAHviDh6MebLXXNWoFyYnsxdwJuG0f/
w3htX6r3dpxXvvFNPGoQb8ihY37+u0gtsbuIwhLtUSmE8rblDEEwiUfi3IJ1zEZpUS77n/k1GWrMrnYg0Gx/
BuaLz0rFi2F738XI/
Hnh45uv3ca60YwS1ojPQ1LtX2URg1haV5884FY1aRvY8jR1pCZabTkYRTZKSXG5KNgYZnHpmsspii6JNkjitYVFKPo0H91w
"MD50fBody": "07adc3fc889d6107d8bb8fda42fe0573",
```

```

"Body": "{\"MessageBody\": \"Hello, world!\", \"QueueUrl\": \"https://sqs.us-
west-2.amazonaws.com/123456789012/does-not-exist\"}\",
  "Attributes": {
    "SenderId": "ARO0A2DZE3W4CTL5ZR7EIN: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"
        }
    }
}
```

在無效字母事件中記下下列屬性，以協助您識別目標調用失敗的可能原因並進行故障診斷。

- **ERROR\_CODE** – 包含 EventBridge 排程器從目標服務 API 收到的錯誤代碼。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 排程器的問題而無法叫用目標，您將改為看到下列錯誤碼：`AWS.Scheduler.InternalServerError`。
- **ERROR\_MESSAGE** – 包含 EventBridge 排程器從目標服務 API 收到的錯誤訊息。在上述範例中，Amazon SQS 傳回的錯誤訊息為 `The specified queue does not exist for this wsd1 version`。如果排程因為 EventBridge 排程器問題而失敗，您將改為看到下列錯誤訊息：`Unexpected error occurred while processing the request`。
- **TARGET\_ARN** – 您的排程調用的目標 ARN，採用下列服務 ARN 格式：`arn:aws:scheduler::aws-sdk:service:apiAction`。
- **EXHAUSTED\_RETRY\_CONDITION** – 指出將事件交付至 DLQ 的原因。如果目標 API 的錯誤是可重試的錯誤，而不是永久錯誤，則會出現此屬性。MaximumRetryAttempts 如果 EventBridge 排程器在超過您為排程設定的重試嘗試次數上限之後將其傳送至 DLQMaximumEventAgeInSeconds，則屬性可以包含這些值；如果事件比您在排程上設定的存留期上限更舊，且仍然無法交付，則屬性可以包含這些值。

在上述範例中，我們可以根據錯誤代碼和錯誤訊息，判斷我們為排程指定的目標佇列不存在。

#### Note

如果您使用[通用目標](#)，請注意 EventBridge 排程器不會在排程建立時間驗證 Input 欄位的內容。具有無效輸入參數的排程將成功建立，但每次調用都會失敗。DLQ 訊息將包含來自目標服務的錯誤代碼和訊息，這可協助您識別無效的參數。如需詳細資訊，請參閱[無效的通用目標輸入組態](#)。

## 在 EventBridge 排程器中刪除排程

您可以透過設定自動刪除或手動刪除個別排程來刪除排程。使用下列主題來了解如何使用兩種方法刪除排程，以及為什麼您可以選擇一種方法。

### 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 EventBridge 排程器上個別管理排程資源，請在排程完成後設定自動刪除。在一次建立數千個排程且需要彈性擴展隨需排程數量的應用程式中，自動刪除可以確保您不會達到指定區域中[排程數量](#)的帳戶配額。

當您設定排程的自動刪除時，EventBridge 排程器會在其最後一次目標調用後刪除排程。對於一次性排程，這會在排程調用其目標一次之後發生。對於您使用 Rate 或 cron、運算式設定的週期性排程，您的排程會在最後一次調用後刪除。週期性排程的最後一次調用是最接近 [EndDate](#) 您指定的調用。如果您使用自動刪除設定排程，但未指定的值 `EndDate`，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 管理主控台

1. 開啟位於 <https://console.aws.amazon.com/scheduler/> 的 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 DELETE，然後儲存您的變更。

### AWS CLI

1. 開啟新的提示視窗。
2. 使用 [update-schedule](#) AWS CLI 命令來更新現有排程，如下所示。命令會將 `--action-after-completion` 設定為 DELETE。此範例假設您已在 JSON 檔案中於本機定義目標組態。若要更新排程，您必須提供目標，以及您想要為現有排程設定的任何其他排程參數。

這是週期性排程，速率為每小時 1 次調用。因此，您在設定 `--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，您可以使用 [DeleteSchedule](#) 操作將其刪除。

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？

- 如需如何設定 Lambda 和 Step Functions 範本化目標的詳細資訊，以及了解如何使用通用目標參數，請參閱 [管理目標](#)。
- 如需 EventBridge 排程器資料類型和 API 操作的詳細資訊，請參閱 [EventBridge 排程器 API 參考](#)。

## 在 EventBridge Scheduler 中管理排程群組

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

您的 AWS 帳戶 隨附 default 排程器群組。您可以將新的排程與 default 群組建立關聯，或與您建立和管理的排程群組建立關聯。您可以在 中建立最多 [500 個排程群組](#) AWS 帳戶。使用 EventBridge Scheduler，您可以透過套用 [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索引鍵和您定義的區分大小寫值所組成的標籤。您可以建立標籤，以根據目標、擁有者或環境等條件來分類排程。例如，您可以使用下列標籤來識別排程所屬的環境：`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](#)：ACTIVE 和 DELETING。

第一次建立群組時，預設為 ACTIVE。您可以將排程新增至 ACTIVE 群組。當您刪除群組時，狀態會變更為 DELETING 直到 EventBridge Scheduler 完成刪除相關聯的排程為止。EventBridge Scheduler 刪除群組中的排程後，您的帳戶就不再提供該群組。

使用下列主題來建立排程群組，並將標籤套用至其中。您也將建立排程與群組的關聯。最後，您將刪除群組。

### 主題

- [在 EventBridge Scheduler 中建立排程群組](#)
- [在 EventBridge 排程器中刪除排程群組](#)
- [相關資源](#)

## 在 EventBridge Scheduler 中建立排程群組

使用排程群組和標記來組織具有共同目的或屬於相同環境的排程。在下列步驟中，您會建立新的排程群組，並使用標籤來標記它。然後，您將新的排程與該群組建立關聯。

**Note**

建立群組後，您無法從該群組中移除排程，或將排程與不同的群組建立關聯。您只能在第一次建立排程時將排程與群組建立關聯。

## 步驟一：建立新的排程群組

下列主題說明如何建立新的排程群組，並使用下列標籤加以標記：`environment:development`。

### AWS 管理主控台

使用 建立新群組 AWS 管理主控台

1. 登入 AWS 管理主控台，並在 <https://console.aws.amazon.com/events/>：// 開啟 Amazon EventBridge 主控台。
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，針對名稱輸入群組的名稱。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. 針對金鑰，輸入您要指派給此金鑰的名稱。在本教學課程中，若要標記此排程群組所屬的環境，請輸入 **environment**。
  - c. 針對值 - 選用，輸入您要指派給此金鑰的值。在此教學課程中，輸入 **development** 環境金鑰的值。

**Note**

您可以在建立其他標籤之後，將標籤新增至您的群組。

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. ( 選用 ) 若要編輯群組或管理其標籤，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

使用 建立新群組 AWS CLI

1. 開啟新的命令提示視窗。
2. 從 AWS Command Line Interface (AWS CLI) 輸入下列 `create-schedule-group` 命令來建立新的群組。此命令會建立具有一個標籤的群組：environment:development。您可以使用此標籤或類似的標記系統，根據排程群組所屬的環境來標記排程群組。

將排程名稱和標籤索引鍵和值取代為您的資訊。

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

根據預設，您的新群組處於 ACTIVE 狀態。您現在可以將新排程與您建立的新群組建立關聯。

## 步驟 2：將排程與群組建立關聯

使用下列步驟，將新的排程與您在 [上一個步驟](#) 中建立的群組建立關聯。

### AWS 管理主控台

使用 將排程與群組建立關聯 AWS 管理主控台

1. 登入 AWS 管理主控台，並在 <https://console.aws.amazon.com/events/> 開啟 Amazon EventBridge 主控台。
2. 在左側導覽窗格中，選擇左側導覽窗格中的排程。
3. 從排程表中，選擇建立排程以建立新的排程。
4. 在指定排程詳細資訊頁面上，針對排程群組，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，然後在檢閱和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 [開始使用](#)。
6. 若要完成並儲存您的排程，請選擇儲存排程。

## AWS CLI

使用 將排程與群組建立關聯 AWS CLI

1. 開啟新的命令提示視窗。
2. 從 AWS Command Line Interface (AWS CLI) 輸入下列 `create-schedule` 命令。這會建立排程，並將其與 [上一個步驟](#) 中的群組建立關聯，該群組名為 `sqs-test-schedule`。此排程使用範本的 [Amazon SQS](#) 目標類型來叫用 `SendMessage` 操作。將排程名稱、目標和群組名稱取代為您的資訊。

```
$ 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" }'
```

您的新排程現在已與 `TestGroup` 排程群組建立關聯。

## 在 EventBridge 排程器中刪除排程群組

在以下，您可以了解如何使用 AWS 管理主控台 和 刪除排程群組 AWS Command Line Interface。當您刪除群組時，會處於 `DELETING` 狀態，直到 EventBridge Scheduler 刪除群組中的所有排程為止。EventBridge Scheduler 刪除群組中的排程後，您的帳戶就不再提供該群組。

### Note


建立群組後，您無法從該群組中移除排程，或將排程與不同的群組建立關聯。您只能在第一次建立排程時將排程與群組建立關聯。

## AWS 管理主控台

使用 刪除群組 AWS 管理主控台

1. 登入 AWS 管理主控台，並在 <https://console.aws.amazon.com/events/> : // 開啟 Amazon EventBridge 主控台。
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，從目前中的現有群組清單中 AWS 區域，找到您要刪除的群組。如果您沒有看到要尋找的群組，請選擇另一個群組 AWS 區域。

 Note

您無法刪除或編輯預設群組。

4. 選取您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，輸入群組的名稱以確認您的選擇，然後選擇刪除。
7. 在排程群組清單中，狀態資料欄會變更，指出您的群組現在正在刪除。群組會保持此狀態，直到 EventBridge Scheduler 刪除與該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

### 使用 刪除群組 AWS CLI

1. 開啟新的命令提示視窗。
2. 從 AWS Command Line Interface (AWS CLI) 輸入下列 [delete-schedule-group](#) 命令來刪除排程群組。將 的值取代 `--name` 為您的資訊。

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

如果成功，AWS CLI 此操作不會傳回回應。

3. 若要驗證群組是否處於 DELETING 狀態，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，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：

- EventBridge Scheduler API 參考中的 [CreateScheduleGroup](#) 操作。
- EventBridge Scheduler API 參考中的 [DeleteScheduleGroup](#) 操作。

## 在 EventBridge 排程器中管理目標

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

範本化目標是一組核心 AWS 服務的常見 API 操作，例如 Amazon SQS、Lambda 和 Step Functions。例如，您可以使用目標的佇列 ARN 提供函數 ARN 或 Amazon SQS 的操作，以 Lambda 的調用 API [SendMessage](#) 操作為目標。

通用目標是一組可自訂的參數，可讓您為許多 AWS 服務調用更廣泛的 API 操作。例如，您可以使用 EventBridge 排程器的通用目標參數 (UTP)，使用 [CreateQueue](#) 操作建立新的 Amazon SQS 佇列。

若要設定範本或通用目標，您的排程必須具有呼叫您設定為目標之 API 操作的許可。您可以透過將必要的許可連接到排程的執行角色來執行此操作。例如，若要以 Amazon SQS [SendMessage](#) 的操作為目標，會授予執行角色執行 `sqs:SendMessage` 動作的許可。在大多數情況下，您可以使用目標服務支援的 [AWS 受管政策](#) 來新增必要的許可。不過，您也可以建立自己的 [客戶受管政策](#)，或將 [內嵌許可](#) 新增至連接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許可的範例。

如需設定排程執行角色的詳細資訊，請參閱 [the section called “設定執行角色”](#)。

### 主題

- [在 EventBridge 排程器中使用範本目標](#)
- [在 EventBridge 排程器中使用通用目標](#)
- [在 EventBridge 排程器中新增內容屬性](#)
- [後續步驟？](#)

## 在 EventBridge 排程器中使用範本目標

範本化目標是一組核心 AWS 服務的一組常見 API 操作，例如 Amazon SQS、Lambda 和 Step Functions。例如，您可以透過提供函數 ARN 來鎖定 Lambda [Invoke](#) 的操作，或使用佇列 ARN 來鎖定 Amazon SQS [SendMessage](#) 的操作。若要設定範本目標，您還必須將許可授予排程的執行角色，以執行目標 API 操作。

若要使用 AWS CLI 或其中一個 EventBridge 排程器 SDKs 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 排程器交付至目標的選用輸入，以及針對某些範本目標，指定一組具有該目標額外組態選項的唯一參數。當您為範本化目標資源指定 ARN

時，EventBridge 排程器會自動假設您想要呼叫該服務的支援 API 操作。如果您希望 EventBridge 排程器以服務的不同 API 操作為目標，您必須將目標設定為[通用目標](#)。

以下是 EventBridge 排程器支援的所有範本目標的完整清單，如果適用，則為每個目標的唯一關聯參數集。選擇每個參數集的連結，以查看 EventBridge 排程器 API 參考中的必要和選用欄位。

- CodeBuild – [StartBuild](#)
- CodePipeline – [StartPipelineExecution](#)
- Amazon ECS – [RunTask](#)
  - Parameters: [EcsParameters](#)
- EventBridge – [PutEvents](#)
  - Parameters: [EventBridgeParameters](#)

**⚠ Important**

使用 EventBridge PutEvents 作為範本目標時 EventBridge 排程器不支援跨區域交付。AWS 帳戶 只有在事件匯流排與 AWS 區域 排程相同時，您才可以使用此目標將事件傳送至不同 中的事件匯流排。若要將事件傳送至不同區域中的事件匯流排，請在目標事件匯流排所在的區域中建立排程。

- Amazon Inspector – [StartAssessmentRun](#)
- kinesis : [PutRecord](#)
  - Parameters: [KinesisParameters](#)
- Firehose – [PutRecord](#)
- Lambda – [Invoke](#)
- SageMaker AI – [StartPipelineExecution](#)
  - Parameters: [SageMakerPipelineParameters](#)
- Amazon SNS – [Publish](#)
- Amazon SQS : [SendMessage](#)
  - Parameters: [SqsParameters](#)
- 步驟函數 – [StartExecution](#)

使用下列範例來了解如何設定不同的範本目標，以及每個所述目標所需的 IAM 許可。

# Amazon SQS SendMessage

Example執行角色的許可政策

JSON

```
{
  "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 軟體開發套件

```
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執行角色的許可政策

JSON

```
{  
  "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 軟體開發套件

```
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");
    }
}
```

## 步驟函數 StartExecution

Example執行角色的許可政策

JSON

```
{
  "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 軟體開發套件

```
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");
}
}
```

## 在 EventBridge 排程器中使用通用目標

通用目標是一組可自訂的參數，可讓您為許多 AWS 服務調用更廣泛的 API 操作。例如，您可以使用通用目標參數 (UTP)，使用 [CreateQueue](#) 操作建立新的 Amazon SQS 佇列。

若要使用 AWS CLI 或其中一個 EventBridge 排程器 SDKs 來設定排程的通用目標，您需要指定下列資訊：

- 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希望排程設為目標之 API 操作的許可。
- Arn – 完整的服務 ARN，包括您想要鎖定的 API 操作，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`。

### Note

通用目標 ARN 中的 `##` 值必須符合目標服務的 AWS SDK 服務識別符。此識別符可能與服務的端點字首不同。例如，對於 Amazon Cognito 身分提供者，請使用 `cognitoidentityprovider` (而非 `cognito-idp`)。若要尋找正確的服務識別符，請參閱您要鎖定之服務的 AWS SDK 文件。

- 輸入 – 您使用 EventBridge 排程器傳送至目標 API 的請求參數指定的格式良好的 JSON。您在中設定的 JSON 參數和形狀 Input 取決於您的排程調用的服務 API。若要尋找此資訊，請參閱您要鎖定之服務的 API 參考。

## 不支援的動作

EventBridge 排程器不支援唯讀 API 動作，例如以下列字首清單開頭的常見 GET 操作：

```
get
```

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

例如，[GetQueueUrl](#) API 動作的服務 ARN 如下：`arn:aws:scheduler:::aws-sdk:sqs:getQueueURL`。由於 API 動作以 `get` 字首開頭，EventBridge 排程器不支援此目標。同樣，[ListBrokers](#) 不支援 Amazon MQ 動作做為目標，因為操作以字首開頭 `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 接受的請求參數。例如，若要以 Lambda 為目標 [Invoke](#)，您可以設定 [AWS Lambda API 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](#)。

若要判斷您可以為不同 APIs 設定的參數，請參閱該服務的 API 參考。與 Lambda 類似 `Invoke`，某些 APIs 接受 URI 參數，以及請求內文承載。在這種情況下，您可以在排程中指定 URI 路徑參數和 JSON 承載 Input。

下列範例示範如何使用通用目標，透過 Lambda、Amazon SQS 和 Step Functions 叫用常見的 API 操作。

## 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 步驟函數

```
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");
    }
}
```

## 在 EventBridge 排程器中新增內容屬性

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 排程器會在您的排程調用目標時，將每個關鍵字取代為其各自的值。

- **<aws.scheduler.schedule-arn>** – 排程的 ARN。
- **<aws.scheduler.scheduled-time>** – 您為排程指定調用其目標的時間，例如 2022-03-22T18:59:43Z。
- **<aws.scheduler.execution-id>** – EventBridge Scheduler 為每個嘗試調用目標指派的唯一 ID，例如 d32c5kddcf5bb8c3。
- **<aws.scheduler.attempt-number>** – 識別目前調用嘗試次數的計數器，例如 1。

此範例顯示建立每五分鐘觸發一次的排程，並呼叫 Amazon SQS SendMessage 操作做為通用目標。訊息內文包含的值 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 EventBridge 排程器資料類型和 API 操作的詳細資訊，請參閱 [EventBridge 排程器 API 參考](#)。

# 使用介面端點存取 Amazon EventBridge 排程器 (AWS PrivateLink)

您可以使用在 VPC 和 Amazon EventBridge 排程器之間 AWS PrivateLink 建立私有連線。您可以像在 VPC 中一樣存取 EventBridge 排程器，無需使用網際網路閘道、NAT 裝置、VPN 連接或 Direct Connect 連線。VPC 中的執行個體不需要公有 IP 地址即可存取 EventBridge 排程器。

您可以建立由 AWS PrivateLink 提供支援的介面端點來建立此私有連線。我們會在您為介面端點啟用的每個子網中建立端點網路介面。這些是請求者管理的網路介面，可做為目的地為 EventBridge 排程器之流量的進入點。

如需詳細資訊，請參閱《AWS PrivateLink 指南》中的[AWS 服務 透過 存取 AWS PrivateLink](#)。

## EventBridge 排程器的考量事項

在您設定 EventBridge 排程器的介面端點之前，請檢閱《AWS PrivateLink 指南》中的[考量事項](#)。

EventBridge 排程器支援透過介面端點呼叫其所有 API 動作。

## 建立 EventBridge 排程器的介面端點

您可以使用 Amazon VPC 主控台或 AWS Command Line Interface () 為 EventBridge 排程器建立介面端點 AWS CLI。如需詳細資訊，請參閱《AWS PrivateLink 指南》中的[建立介面端點](#)。

使用下列服務名稱建立 EventBridge 排程器的介面端點：

```
com.amazonaws.region.scheduler
```

如果您為介面端點啟用私有 DNS，您可以使用其預設的區域 DNS 名稱向 EventBridge 排程器提出 API 請求。例如 scheduler.us-east-1.amazonaws.com。

## 為您的介面端點建立端點政策

端點政策為 IAM 資源，您可將其連接至介面端點。預設端點政策允許透過介面端點完整存取 EventBridge 排程器。若要控制允許從 VPC 存取 EventBridge 排程器，請將自訂端點政策連接至介面端點。

端點政策會指定以下資訊：

- 可執行動作 (AWS 帳戶、IAM 使用者和 IAM 角色) 的主體。
- 可執行的動作。
- 可供執行動作的資源。

如需詳細資訊，請參閱「AWS PrivateLink 指南」中的[使用端點政策控制對服務的存取](#)。

#### 範例：EventBridge 排程器動作的 VPC 端點政策

以下是自訂端點政策的範例。當您將此政策連接到介面端點時，它會授予所有資源上所有主體所列出的 EventBridge 排程器動作的存取權。

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "scheduler:GetSchedule",
        "scheduler:ListSchedules",
        "scheduler:GetScheduleGroup",
        "scheduler:ListScheduleGroups"
      ],
      "Resource": "*"
    }
  ]
}
```

# Amazon EventBridge 排程器中的安全性

的雲端安全性 AWS 是最高優先順序。身為 AWS 客戶，您可以受益於資料中心和網路架構，這些架構是為了符合最安全敏感組織的需求而建置。

安全性是 AWS 與您之間共同責任。[共同責任模式](#)將其描述為雲端的安全性，和雲端中的安全性：

- 雲端的安全性 – AWS 負責保護在 中執行 AWS 服務的基礎設施 AWS 雲端。AWS 也為您提供可安全使用的服務。在[AWS 合規計畫](#)中，第三方稽核人員會定期測試和驗證我們安全的有效性。若要了解適用於 Amazon EventBridge 排程器的合規計畫，請參閱[AWS 合規計畫的服務範圍](#)。
- 雲端的安全性 – 您的責任取決於您使用 AWS 的服務。您也必須對其他因素負責，包括資料的機密性、您的公司的要求和適用法律和法規。

本文件可協助您了解如何在使用 EventBridge 排程器時套用共同責任模型。下列主題說明如何設定 EventBridge 排程器以符合您的安全與合規目標。您也會了解如何使用其他 AWS 服務來協助您監控和保護 EventBridge 排程器資源。

## 主題

- [管理對 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 排程器中的彈性](#)
- [Amazon EventBridge 排程器中的基礎設施安全](#)

## 管理對 Amazon EventBridge 排程器的存取

AWS Identity and Access Management (IAM) 是 AWS 服務，可協助管理員安全地控制對 AWS 資源的存取。IAM 管理員可控制誰可以進行身分驗證（登入）和授權（具有許可），以使用 EventBridge 排程器資源。IAM 是 AWS 服務 您可以免費使用的。

## 主題

- [目標對象](#)
- [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](#)

- [在 EventBridge 排程器中使用身分型政策](#)
- [EventBridge 排程器中的混淆代理人預防](#)
- [對 Amazon EventBridge 排程器身分和存取進行故障診斷](#)

## 目標對象

使用方式 AWS Identity and Access Management (IAM) 會根據您的角色而有所不同：

- 服務使用者 — 若無法存取某些功能，請向管理員申請所需許可 (請參閱 [對 Amazon EventBridge 排程器身分和存取進行故障診斷](#))
- 服務管理員 — 負責設定使用者存取權並提交相關許可請求 (請參閱 [EventBridge 排程器如何與 IAM 搭配使用](#))
- IAM 管理員 — 撰寫政策以管理存取控制 (請參閱 [在 EventBridge 排程器中使用身分型政策](#))

## 使用身分驗證

身分驗證是您 AWS 使用身分憑證登入的方式。您必須以 AWS 帳戶根使用者、IAM 使用者或擔任 IAM 角色身分進行身分驗證。

您可以使用身分來源的登入資料，例如 AWS IAM Identity Center (IAM Identity Center)、單一登入身分驗證或 Google/Facebook 登入資料，以聯合身分的形式登入。如需有關登入的詳細資訊，請參閱《AWS 登入 使用者指南》中的[如何登入您的 AWS 帳戶](#)。

對於程式設計存取，AWS 提供 SDK 和 CLI 以密碼編譯方式簽署請求。如需詳細資訊，請參閱《IAM 使用者指南》中的[API 請求的AWS 第 4 版簽署程序](#)。

## AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個名為 AWS 帳戶 theroot 使用者的登入身分開始，該身分具有對所有 AWS 服務和資源的完整存取權。強烈建議不要使用根使用者來執行日常任務。有關需要根使用者憑證的任務，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

最佳實務是要求人類使用者使用聯合身分提供者，以 AWS 服務使用臨時憑證存取。

聯合身分是來自您企業目錄、Web 身分提供者的使用者，或使用來自身分來源的 AWS 服務登入資料存取 Directory Service 的使用者。聯合身分會擔任角色，而該角色會提供臨時憑證。

若需集中化管理存取權限，建議使用 AWS IAM Identity Center。如需詳細資訊，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[https://docs.aws.amazon.com/IAM/latest/UserGuide/id\\_users.html](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users.html)是一種身分具備單人或應用程式的特定許可權。建議以臨時憑證取代具備長期憑證的 IAM 使用者。如需詳細資訊，請參閱《IAM 使用者指南》中的[要求人類使用者使用聯合身分提供者來 AWS 使用臨時憑證存取](#)。

[IAM 群組](#)會指定 IAM 使用者集合，使管理大量使用者的許可權更加輕鬆。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 使用者的使用案例](#)。

## IAM 角色

IAM 角色[https://docs.aws.amazon.com/IAM/latest/UserGuide/id\\_roles.html](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html)的身分具有特定許可權，其可以提供臨時憑證。您可以透過[從使用者切換到 IAM 角色（主控台）](#)或呼叫 AWS CLI 或 AWS API 操作來擔任角色。如需詳細資訊，請參閱《IAM 使用者指南》中的[擔任角色的方法](#)。

IAM 角色適用於聯合身分使用者存取、臨時 IAM 使用者許可、跨帳戶存取權與跨服務存取，以及在 Amazon EC2 執行的應用程式。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 中的快帳戶資源存取](#)。

## 使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策定義與身分或資源相關聯的許可。當委託人提出請求時 AWS，會評估這些政策。大多數政策會以 JSON 文件 AWS 形式存放在中。如需進一步了解 JSON 政策文件，請參閱《IAM 使用者指南》中的[JSON 政策概觀](#)。

管理員會使用政策，透過定義哪些主體可在哪些條件下對哪些資源執行動作，以指定可存取的範圍。

預設情況下，使用者和角色沒有許可。IAM 管理員會建立 IAM 政策並將其新增至角色，供使用者後續擔任。IAM 政策定義動作的許可，無論採用何種方式執行。

## 身分型政策

身分型政策是附加至身分 (使用者、使用者群組或角色) 的 JSON 許可政策文件。這類政策控制身分可對哪些資源執行哪些動作，以及適用的條件。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的[透過客戶管理政策定義自訂 IAM 許可](#)。

身分型政策可分為內嵌政策 (直接內嵌於單一身分) 與受管政策 (可附加至多個身分的獨立政策)。如需了解如何在受管政策及內嵌政策之間做選擇，請參閱《IAM 使用者指南》中的[在受管政策與內嵌政策之間選擇](#)。

## 資源型政策

資源型政策是附加到資源的 JSON 政策文件。範例包括 IAM 角色信任政策與 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。您必須在資源型政策中[指定主體](#)。

資源型政策是位於該服務中的內嵌政策。您無法在資源型政策中使用來自 IAM 的 AWS 受管政策。

## 其他政策類型

AWS 支援其他政策類型，可設定更多常見政策類型授予的最大許可：

- 許可界限 — 設定身分型政策可授與 IAM 實體的最大許可。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 實體許可界限](#)。
- 服務控制政策 (SCP) — 為 AWS Organizations 中的組織或組織單位指定最大許可。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的[服務控制政策](#)。
- 資源控制政策 (RCP) — 設定您帳戶中資源可用許可的上限。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的[資源控制政策 \(RCP\)](#)。
- 工作階段政策 — 在以程式設計方式為角色或聯合身分使用者建立臨時工作階段時，以參數形式傳遞的進階政策。如需詳細資訊，請參《IAM 使用者指南》中的[工作階段政策](#)。

## 多種政策類型

當多種類型的政策套用到請求時，產生的許可會更複雜而無法理解。若要了解如何 AWS 在涉及多種政策類型時決定是否允許請求，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 排程器的存取權之前，請先了解哪些 IAM 功能可與 EventBridge 排程器搭配使用。

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
<a href="#">身分型政策</a>	是
<a href="#">資源型政策</a>	否
<a href="#">政策動作</a>	是

IAM 功能	EventBridge 排程器支援
<a href="#">政策資源</a>	是
<a href="#">政策條件索引鍵 (服務特定)</a>	是
<a href="#">ACL</a>	否
<a href="#">ABAC(政策中的標籤)</a>	部分
<a href="#">臨時憑證</a>	是
<a href="#">主體許可</a>	是
<a href="#">服務角色</a>	是
<a href="#">服務連結角色</a>	否

若要全面了解 EventBridge 排程器和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱 [《AWS IAM 使用者指南》](#) 中的 [與 IAM 搭配使用的 服務](#)。

## EventBridge 排程器的身分型政策

支援身分型政策：是

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。如需了解如何建立身分型政策，請參閱 [《IAM 使用者指南》](#) 中的 [透過客戶管理政策定義自訂 IAM 許可](#)。

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。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](#)。

### EventBridge 排程器的身分型政策範例

若要檢視 EventBridge 排程器身分型政策的範例，請參閱 [在 EventBridge 排程器中使用身分型政策](#)。

## EventBridge 排程器中的資源型政策

支援資源型政策：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](#)。委託人可以包含帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

如需啟用跨帳戶存取權，您可以在其他帳戶內指定所有帳戶或 IAM 實體作為資源型政策的主體。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 中的快帳戶資源存取](#)。

## EventBridge 排程器的政策動作

支援政策動作：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON 政策的 Action 元素描述您可以用來允許或拒絕政策中存取的動作。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 排程器動作的清單，請參閱《服務授權參考》中的[Amazon EventBridge 排程器定義的動作](#)。

EventBridge 排程器中的政策動作在動作之前使用下列字首：

```
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 排程器的政策資源

支援政策資源：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。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。最佳實務是使用其 [Amazon Resource Name \(ARN\)](#) 來指定資源。若動作不支援資源層級許可，使用萬用字元 (\*) 表示該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 ARNs 的清單，請參閱《服務授權參考》中的 [Amazon EventBridge 排程器定義的資源](#)。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 [Amazon EventBridge 排程器定義的動作](#)。

若要檢視 EventBridge 排程器身分型政策的範例，請參閱 [在 EventBridge 排程器中使用身分型政策](#)。

## EventBridge 排程器的政策條件索引鍵

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 元素會根據定義的條件，指定陳述式的執行時機。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。若要查看所有 AWS 全域條件索引鍵，請參閱《IAM 使用者指南》中的 [AWS 全域條件內容索引鍵](#)。

若要查看 EventBridge 排程器條件索引鍵的清單，請參閱《服務授權參考》中的 [Amazon EventBridge 排程器的條件索引鍵](#)。若要了解您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 EventBridge 排程器身分型政策的範例，請參閱 [在 EventBridge 排程器中使用身分型政策](#)。

## EventBridge 排程器中的 ACLs

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。

## ABAC 搭配 EventBridge 排程器

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，根據稱為標籤的屬性定義許可權。您可以將標籤連接至 IAM 實體 AWS 和資源，然後設計 ABAC 政策，以便在委託人的標籤符合資源上的標籤時允許操作。

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 [使用 ABAC 授權定義許可](#)。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 排程器使用臨時憑證

支援臨時憑證：是

暫時登入資料提供 AWS 資源的短期存取權，並在您使用聯合或切換角色時自動建立。AWS 建議您動態產生暫時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 中的臨時安全憑證與可與 IAM 搭配運作的 AWS 服務](#)。

## EventBridge 排程器的跨服務主體許可

支援轉寄存取工作階段 (FAS)：是

轉送存取工作階段 (FAS) 使用呼叫的委託人許可 AWS 服務，並結合請求 AWS 服務向下游服務提出請求。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](#)。

### Warning

變更服務角色的許可可能會中斷 EventBridge 排程器功能。只有在 EventBridge 排程器提供指引時，才能編輯服務角色。

## EventBridge 排程器的服務連結角色

支援服務連結角色：否

服務連結角色是連結至的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶，並由服務擁有。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](#)。在資料表中尋找服務，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。

## 在 EventBridge 排程器中使用身分型政策

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。

如需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 \(主控台\)](#)。

如需 EventBridge 排程器定義之動作和資源類型的詳細資訊，包括每種資源類型的 ARNs 格式，請參閱《服務授權參考》中的 [Amazon EventBridge 排程器的動作、資源和條件索引鍵](#)。

### 主題

- [政策最佳實務](#)
- [EventBridge 排程器許可](#)
- [AWS EventBridge 排程器的 受管政策](#)
- [EventBridge 排程器的客戶受管政策](#)
- [AWS 受管政策更新](#)

## 政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 EventBridge 排程器資源。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 受管政策並邁向最低權限許可 – 若要開始將許可授予您的使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 中使用 AWS 帳戶。我們建議您定義特定於使用案例 AWS 的客戶受管政策，進一步減少許可。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。如果透過特定等使用服務動作 AWS 服務，您也可以使用條件來授予其存取權 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，可協助您撰寫安全且實用的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [使用 IAM Access Analyzer 驗證政策](#)。
- 需要多重要素驗證 (MFA) – 如果您的案例需要 IAM 使用者或中的根使用者 AWS 帳戶，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [透過 MFA 的安全 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 排程器許可

為了讓 IAM 委託人（使用者、群組或角色）在 EventBridge 排程器中建立排程，並透過主控台或 API 存取 EventBridge 排程器資源，委託人必須擁有一組新增至其許可政策的許可。您可以根據委託人的任務函數來設定這些許可。例如，僅使用 EventBridge 排程器主控台來檢視現有排程清單的使用者或角色，不需要具有呼叫 CreateSchedule API 操作所需的許可。我們建議您量身打造身分型許可，只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 排程器許可來建立自己的客戶受管政策，以搭配 EventBridge 排程器使用。您也可以使用下一節所述的 AWS 受管政策，授予常見使用案例的必要許可，而不必管理您自己的政策。

## AWS EventBridge 排程器的 受管政策

AWS 提供 AWS 建立和管理的獨立 IAM 政策，以解決許多常見的使用案例。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](#)。下列可連接到您帳戶中使用者的 AWS 受管政策專屬於 EventBridge 排程器：

- AmazonEventBridgeSchedulerFullAccess

准許針對排程和排程群組使用所有 EventBridge 排程器動作。

若要檢視此政策的許可，請參閱《AWS 受管政策參考》中的 [AmazonEventBridgeSchedulerFullAccess](#)。

- AmazonEventBridgeSchedulerReadOnlyAccess

授予唯讀許可，以檢視排程和排程群組的詳細資訊。

若要檢視此政策的許可，請參閱《AWS 受管政策參考》中的 [AmazonEventBridgeSchedulerReadOnlyAccess](#)。

## EventBridge 排程器的客戶受管政策

使用下列範例為 EventBridge 排程器建立自己的客戶受管政策。[客戶受管政策](#)可讓您根據委託人的工作職能，僅針對團隊中的應用程式和使用者所需的動作和資源授予許可。

### 主題

- [範例：CreateSchedule](#)
- [範例：GetSchedule](#)

- [範例：UpdateSchedule](#)
- [範例：DeleteScheduleGroup](#)

### 範例：CreateSchedule

當您建立新的排程時，您可以選擇要使用 [AWS 擁有的金鑰](#) 或 [客戶受管金鑰](#) 來加密 EventBridge 排程器上的資料。

下列政策允許主體建立排程並使用 套用加密 AWS 擁有的金鑰。透過 AWS 擁有的金鑰，會為您 AWS 管理 AWS Key Management Service (AWS KMS) 上的資源，因此您不需要額外的許可來與 互動 AWS KMS。

JSON

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

```
}
```

使用下列政策允許委託人建立排程，並使用 AWS KMS 客戶受管金鑰進行加密。若要使用客戶受管金鑰，委託人必須具有存取您帳戶中 AWS KMS 資源的許可。此政策授予單一指定 KMS 金鑰的存取權，用於加密 EventBridge 排程器上的資料。或者，您可以使用萬用字元 (\*) 字元來授予帳戶中所有金鑰的存取權，或是符合指定名稱模式的子集。

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "scheduler:CreateSchedule"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:scheduler:us-east-1:123456789012: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"
      ],
      "Condition": {
        "StringLike": {
          "kms:ViaService": "scheduler.us-east-1.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

使用下列政策允許委託人取得排程的相關資訊。

JSON

```

{
  "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

使用下列政策，以允許委託人透過呼叫 `scheduler:UpdateSchedule` 動作來更新排程。與類似 `CreateSchedule`，政策取決於排程是否使用 AWS KMS AWS 擁有的金鑰 或客戶受管金鑰進行加密。對於使用設定的排程 AWS 擁有的金鑰，請使用下列政策：

JSON

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

對於使用客戶受管金鑰設定的排程，請使用下列政策。此政策包含其他許可，允許委託人存取您帳戶中 AWS KMS 的資源：

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "scheduler:UpdateSchedule"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:scheduler:us-east-1:123456789012: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"
      ],
      "Condition": {
        "StringLike": {
          "kms:ViaService": "scheduler.us-east-1.amazonaws.com",
          "kms:EncryptionContext:aws:scheduler:schedule:arn":
            "arn:aws:scheduler:us-east-1:123456789012: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

使用下列政策允許委託人刪除排程群組。當您刪除群組時，也會刪除與該群組相關聯的排程。刪除群組的委託人也必須具有許可，才能刪除與該群組相關聯的排程。此政策授予委託人許可，以呼叫指定排程群組上的scheduler:DeleteScheduleGroup動作，以及群組中的所有排程：

#### Note

EventBridge 排程器不支援指定個別排程的資源層級許可。例如，下列陳述式無效，不應包含在政策中：

```

"Resource": "arn:aws:scheduler:us-west-2:123456789012:schedule/my-group/my-schedule-name"

```

## JSON

```

{
  "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 受管政策更新

變更	描述	Date
<a href="#">AmazonEventBridgeSchedulerFullAccess</a> – 新受管政策	EventBridge 排程器新增對新受管政策的支援，該政策會授予使用者所有資源的完整存取權，包括排程和排程群組。	2022 年 11 月 10 日
<a href="#">AmazonEventBridgeSchedulerReadOnlyAccess</a> – 新的受管政策	EventBridge 排程器新增對新受管政策的支援，該政策會授予使用者所有資源的唯讀存取權，包括排程和排程群組。	2022 年 11 月 10 日
<a href="#">AmazonEventBridgeSchedulerReadOnlyAccess</a> – 更新受管政策	EventBridge 排程器更新 AmazonEventBridgeSchedulerReadOnlyAccess 了受管政策，以使用萬用字元動作 (scheduler:List* 和 scheduler:Get* )，以確保在新增新的	2026 年 3 月 25 日

變更	描述	Date
	唯讀動作時，政策保持最新狀態。	
EventBridge 排程器已開始追蹤變更	EventBridge 排程器開始追蹤其 AWS 受管政策的變更。	2022 年 11 月 10 日

## EventBridge 排程器中的混淆代理人預防

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代理人問題。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 排程器提供其他服務存取資源的許可。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。下列條件的範圍限定為個別排程群組：

```
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` 必須是您要限制此條件範圍的 EventBridge 排程器排程群組 ARN。

### Important

請勿將 `aws:SourceArn` 陳述式範圍限定為特定排程或排程名稱字首。您指定的 ARN 必須是排程群組。

下列範例示範如何在執行角色信任政策中使用 `aws:SourceArn` 和 `aws:SourceAccount` 全域條件內容金鑰，以防止混淆代理人問題：

## JSON

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

如果您在 的值中使用萬用字元 `aws:SourceArn`，則必須使用 `ArnLike` 而非 `StringEquals` 作為條件運算子。例如，下列信任政策會使用 `ArnLike` 來比對帳戶中的任何排程群組：

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

```
    }  
  }  
}  
]  
}
```

## 對 Amazon EventBridge 排程器身分和存取進行故障診斷

使用以下資訊來協助您診斷和修正使用 EventBridge 排程器和 IAM 時可能遇到的常見問題。

### 主題

- [我無權在 EventBridge 排程器中執行動作](#)
- [我未獲得執行 iam:PassRole 的授權](#)
- [我想要允許以外的人員 AWS 帳戶 存取我的 EventBridge 排程器資源](#)

### 我無權在 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 管理員。您的管理員提供您的簽署憑證。

### 我未獲得執行 iam:PassRole 的授權

如果您收到錯誤，告知您無權執行 iam:PassRole 動作，您的政策必須更新，以允許您將角色傳遞給 EventBridge 排程器。

有些 AWS 服務可讓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 marymajor 使用者嘗試使用主控台在 EventBridge 排程器中執行動作時，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 沒有將角色傳遞給服務的許可。

```
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 管理員。您的管理員提供您的簽署憑證。

## 我想要允許以外的人員 AWS 帳戶 存取我的 EventBridge 排程器資源

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：

- 若要了解 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 AWS 帳戶 在您擁有的 資源之間提供存取權，請參閱 [《IAM 使用者指南》中的在您擁有 AWS 帳戶 的另一個 中為 IAM 使用者提供存取權](#)。
- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱 [《IAM 使用者指南》中的將存取權提供給第三方 AWS 帳戶 擁有](#)。
- 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 \(聯合身分\)](#)。
- 如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，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

[共同責任模型](#)適用於 AWS Amazon EventBridge Scheduler 中的資料保護。如此模型所述，AWS 負責保護執行所有 的全域基礎設施 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](#)。如需歐洲資料保護的資訊，請參閱 [「一般資料保護法規 \(GDPR\) 中心」](#)。

基於資料保護目的，我們建議您保護 AWS 帳戶 登入資料，並使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 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。如需有關使用 CloudTrail 追蹤擷取 AWS 活動的資訊，請參閱AWS CloudTrail 《使用者指南》中的[使用 CloudTrail 追蹤](#)。
- 使用 AWS 加密解決方案，以及其中的所有預設安全控制 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 AWS 透過命令列界面或 API 存取 時需要 FIPS 140-3 驗證的密碼編譯模組，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 \(FIPS\) 140-3](#)。

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 EventBridge 排程器或使用 AWS 服務 主控台、API AWS CLI 或其他 AWS SDKs 時。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。

## EventBridge 排程器中的靜態加密

本節說明 Amazon EventBridge 排程器如何加密和解密靜態資料。靜態資料是存放在 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，以使用 加密和解密您的資料[AWS KMS key](#)。EventBridge 排程器支援兩種 KMS 金鑰類型：[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 服務擁有和管理用於多個 AWS 帳戶的 KMS 金鑰。雖然 AWS 擁有的金鑰 EventBridge 排程器使用的 不會存放在 AWS 您的帳戶中，但 EventBridge 排程器會使用它們來保護您的資料和資源。根據預設，EventBridge 排程器會使用 AWS 擁有的金鑰來加密和解密所有資料。您不需要管理 AWS 擁有的金鑰 或其存取政策。當 EventBridge 排程器使用 AWS 擁有的金鑰 來保護您的資料時，您不需要支付任何費用，而且其用量不會計入您帳戶中的 AWS KMS 配額中。

**客戶受管金鑰**是您建立、擁有和管理之 AWS 帳戶中存放的 KMS 金鑰。如果您的特定使用案例要求您控制和稽核可在 EventBridge 排程器上保護資料的加密金鑰，您可以使用客戶受管金鑰。如果您選擇客戶受管金鑰，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。使用客戶受管金鑰也算作[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 排程器靜態加密的不同資料類型，以及每個類別支援的 KMS 金鑰類型。

資料類型	說明	AWS 擁有的金鑰	客戶受管金鑰
承載 (最多 256KB)	當您設定要交付至目標的排程時，在排程的 TargetInput 參數中指定的資料。	支援	支援
識別符和狀態	排程的唯一名稱和狀態 (啟用、停用)。	支援	不支援
Scheduling configuration (排程組態)	排程表達式，例如週期性排程的速率或 Cron 表達式，以及一次性調用的時間戳記，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 Resource Name (ARN) 和其他目標相關組態詳細資訊。	支援	不支援
叫用和失敗行為組態	彈性的時段組態、排程的重試政策，以及用於失敗交付的無效字母佇列詳細資訊。	支援	不支援

EventBridge 排程器只會在加密和解密目標承載時使用客戶受管金鑰，如上表所述。如果您選擇使用客戶受管金鑰，EventBridge 排程器會加密和解密承載兩次：一次使用預設值 AWS 擁有的金鑰，另一次使用您指定的客戶受管金鑰。對於所有其他資料類型，EventBridge 排程器只會使用預設值 AWS 擁有的金鑰來保護靜態資料。

使用下[the section called “管理 KMS 金鑰”](#)一節來了解如何管理您的 IAM 資源和金鑰政策，以便將客戶受管金鑰與 EventBridge 排程器搭配使用。

## 管理 KMS 金鑰

您可以選擇性地提供客戶受管金鑰，以加密和解密排程交付給其目標的承載。EventBridge 排程器會加密和解密您最多 256KB 的資料承載。使用客戶受管金鑰會產生月費，以及超過免費方案的費用。使用客戶受管金鑰計為[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 排程器使用與建立排程以加密資料的主體相關聯的 IAM 許可。這表示您必須將必要的 AWS KMS 相關許可連接到呼叫 EventBridge 排程器 API 的使用者或角色。此外，EventBridge 排程器會使用資源型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備必要的 AWS KMS 相關許可，才能在解密資料時呼叫 AWS KMS API。

### Note

EventBridge 排程器不支援使用暫時許可的[授予](#)。

使用下一節了解如何管理 AWS KMS [金鑰政策和](#)所需的 IAM 許可，以在 EventBridge 排程器上使用客戶受管金鑰。

### 主題

- [新增 IAM 許可](#)
- [管理金鑰政策](#)

### 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，以及與排程建立關聯的執行角色。

## 客戶受管金鑰的身分型許可

您必須將下列 AWS KMS 動作新增至與建立排程時呼叫 EventBridge 排程器 API 的任何委託人（使用者、群組或角色）相關聯的許可政策。

- **kms:DescribeKey** – 驗證您提供的金鑰是[對稱](#)加密 KMS 金鑰時需要。
- **kms:GenerateDataKey** – 為了產生 EventBridge 排程器用來執行用戶端加密的資料金鑰，此為必要項目。
- **kms:Decrypt** – 需要解密 EventBridge Scheduler 與您的加密資料一起存放的加密資料金鑰。

這些是下列動作的補充：

- **scheduler:\***
- **iam:PassRole** – 傳遞執行角色時需要。

## 客戶受管金鑰的執行角色許可

您必須將下列動作新增至排程的執行角色許可政策，以提供 EventBridge 排程器在解密資料時呼叫 AWS KMS API 的存取權。

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowEventBridgeSchedulerToDecryptDataUsingCMKMS",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-east-1:123456789012:key/your-key-id"
    }
  ]
}
```

- **kms:Decrypt** – 需要解密 EventBridge 排程器與您的加密資料一起存放的加密資料金鑰。

如果您在建立新排程時使用 EventBridge 排程器主控台來建立新的執行角色，EventBridge 排程器會自動將必要的許可連接到您的執行角色。不過，如果您選擇現有的執行角色，則必須將必要的許可新增至角色，才能使用您的客戶受管金鑰。

## 管理金鑰政策

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

或者，您可以將金鑰政策的範圍限制為僅提供對執行角色的存取。如果您只想將客戶受管金鑰與 EventBridge 排程器資源搭配使用，則可以這樣做。使用下列[金鑰政策](#)範例來限制哪些 EventBridge 排程器資源可以使用您的金鑰。

## CloudTrail 事件範例

AWS CloudTrail 會擷取所有 API 呼叫事件。這包括每當 EventBridge 排程器使用您的客戶受管金鑰解密資料時，API 呼叫。下列範例顯示 CloudTrail 事件項目，該項目示範 EventBridge 排程器使用客戶受管金鑰的 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": "ABCDEFGH11JKLMNOP2Q3",
    "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"
}
}
```

## EventBridge 排程器中的傳輸中加密

EventBridge 排程器會在傳輸網路時加密傳輸中的資料。Transport Layer Security (TLS) 會在您呼叫任何 EventBridge 排程器 API 操作，以及在 EventBridge 排程器調用您的排程時呼叫任何目標 APIs 時加密您的資料。根據預設，EventBridge 排程器會在加密傳輸中的資料時使用 TLS 1.2。您不需要設定傳輸中的加密，而且在使用 EventBridge 排程器時，您無法選擇不同的 TLS 版本。

使用 EventBridge 排程器 API – 當您執行 API 操作時，例如 `CreateSchedule`，EventBridge 排程器會加密整個 HTTP 請求，包括請求內文和標頭。EventBridge 排程器也會加密您從 APIs 收到的整個回應物件。

使用目標 APIs – EventBridge 排程器調用排程時，它會呼叫您在建立排程時指定的目標 API。將事件交付至目標時，EventBridge 排程器會加密整個請求，包括請求內文和所有標頭，以及從目標收到的回應。

## Amazon EventBridge 排程器的合規驗證

若要了解 是否 AWS 服務 在特定合規計劃的範圍內，請參閱[AWS 服務 合規計劃範圍內](#)然後選擇您感興趣的合規計劃。如需一般資訊，請參閱[AWS 合規計劃](#)。

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[在中下載報告 AWS Artifact](#)。

您使用 時的合規責任 AWS 服務 取決於資料的機密性、您公司的合規目標，以及適用的法律和法規。如需使用 時合規責任的詳細資訊 AWS 服務，請參閱 [AWS 安全文件](#)。

## Amazon EventBridge 排程器中的彈性

AWS 全球基礎設施是以 AWS 區域 和 可用區域 為基礎建置。AWS 區域 提供多個實體分隔和隔離的可用區域，這些可用區域與低延遲、高輸送量和高備援聯網連接。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 全球基礎設施](#)。

除了 AWS 全球基礎設施之外，EventBridge 排程器還提供數種功能，以協助支援您的資料彈性和備份需求。

## Amazon EventBridge 排程器中的基礎設施安全

Amazon EventBridge 排程器是受管服務，受到 AWS 全球網路安全的保護。如需 AWS 安全服務以及如何 AWS 保護基礎設施的相關資訊，請參閱[AWS 雲端安全](#)。若要使用基礎設施安全的最佳實務設計您的 AWS 環境，請參閱安全支柱 AWS Well-Architected Framework 中的[基礎設施保護](#)。

您可以使用 AWS 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。

# Amazon EventBridge Scheduler 的監控和指標

監控是維護 Amazon EventBridge Scheduler 和您其他 AWS 解決方案可靠性、可用性和效能的重要部分。AWS 提供下列監控工具來監看 EventBridge Scheduler、在發生錯誤時報告，並適時採取自動動作：

- Amazon CloudWatch AWS 會即時監控您的 AWS 資源和您在 上執行的應用程式。您可以收集和追蹤指標、建立自訂儀板表，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 CloudWatch 使用者指南](#)。
- AWS CloudTrail 會擷取由您的帳戶或代表 AWS 您的帳戶發出的 API 呼叫和相關事件，並將日誌檔案交付至您指定的 Amazon S3 儲存貯體。您可以識別呼叫的使用者和帳戶 AWS、進行呼叫的來源 IP 地址，以及呼叫的時間。如需詳細資訊，請參閱 [《AWS CloudTrail 使用者指南》](#)。

## 主題

- [使用 Amazon CloudWatch 監控 Amazon EventBridge 排程器](#)
- [使用 記錄 Amazon EventBridge Scheduler API 呼叫 AWS CloudTrail](#)

## 使用 Amazon CloudWatch 監控 Amazon EventBridge 排程器

您可以使用 CloudWatch 監控 Amazon EventBridge 排程器，這會收集原始資料並將其處理為可讀且近乎即時的指標。EventBridge 排程器會為所有排程發出一組指標，並為具有關聯無效字串佇列 (DLQ) 的排程發出一組額外的指標。如果您為排程 [設定 DLQ](#)，EventBridge 排程器會在排程耗盡其重試政策時發佈其他指標。

這些統計資料會保留 15 個月，讓您可以存取歷史資訊，並更清楚排程失敗的原因，以及對基礎問題進行疑難排解。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](#)。

### CloudWatch 指標盡力交付

CloudWatch 指標是盡力交付。大多數 EventBridge 排程器操作都會將資料點傳送至 CloudWatch。不保證指標的完成程度與時間先後順序。指定分鐘的資料點可能會在透過 CloudWatch 提供之前延遲，或者完全無法交付。CloudWatch 指標可讓您近乎即時地了解活動的性質。它們並非所有操作的完整會計。

## 主題

- [條款](#)
- [維度](#)
- [存取 指標](#)
- [指標清單](#)
- [EventBridge 排程器用量指標](#)

## 條款

### 命名空間

命名空間是 AWS 服務的 CloudWatch 指標的容器。對於 EventBridge 排程器，命名空間為 AWS/Scheduler。

### CloudWatch 指標

CloudWatch 指標代表 CloudWatch 特有的一組按時間順序排列的資料點。

### 維度

維度是一組名稱值對，是指標身分的一部分。

### 單位

統計資料具有度量單位。對於 EventBridge 排程器，單位包含計數。

## 維度

本節說明 CloudWatch 中 EventBridge 排程器指標的 CloudWatch 維度分組。

維度	說明
ScheduleGroup	您要使用 CloudWatch 檢視指標的排程群組。如果您尚未建立任何群組，EventBridge 排程器會將您的排程與該default群組建立關聯。

## 存取 指標

本節說明如何存取 CloudWatch 中特定 EventBridge 排程器排程的效能指標。

## 檢視維度的效能指標

1. 在 CloudWatch 主控台上開啟[指標頁面](#)。
2. 使用 AWS 區域選擇器為您的排程選擇區域
3. 選擇排程器命名空間。
4. 在所有指標索引標籤中，選擇維度，例如排程群組指標。若要查看您在所選區域中建立之所有排程的指標，請選擇帳戶指標。
5. 選擇維度的 CloudWatch 指標。例如，InvocationAttemptCount 或 InvocationDroppedCount，然後選擇圖形搜尋。
6. 選擇圖形化指標索引標籤，以檢視 EventBridge 排程器指標的效能統計資料。

## 指標清單

下表列出所有 EventBridge 排程器排程的指標，以及您已設定 DLQ 之排程的其他指標。

### 所有排程的指標

命名空間	指標	單位	說明
AWS/Scheduler	InvocationAttemptCount	計數	每次呼叫嘗試都會發出。使用此指標來檢查 EventBridge 排程器是否嘗試調用您的排程，並查看調用何時接近您的帳戶配額。
AWS/Scheduler	TargetErrorCount	計數	在 EventBridge 排程器呼叫目標 API 之後，目標傳回例外狀況時發出。使用此項目來檢查交付至目標失敗的時間。
AWS/Scheduler	TargetErrorThrottledCount	計數	當目標調用因目標的 API 限流而失敗時發出。當基礎原因為

命名空間	指標	單位	說明
			EventBridge 排程器發出的目標 API 限流呼叫時，請使用此選項來診斷交付失敗
AWS/Scheduler	InvocationThrottleCount	計數	當 EventBridge 排程器調節目標調用時發出，因為它超過 EventBridge 排程器設定的服務配額。使用此值來判斷您何時超過調用調節限制配額。如需服務配額的詳細資訊，請參閱 <a href="#">配額</a> 。
AWS/Scheduler	InvocationDroppedCount	計數	在排程的重試政策用盡後，EventBridge 排程器停止嘗試調用目標時發出。如需重試政策的詳細資訊，請參閱 EventBridge 排程器 API 參考中的 <a href="#">RetryPolicy</a> 。

## 使用 DLQ 排程的指標

命名空間	指標	單位	說明
AWS/Scheduler	InvocationsSentToDeadLetterCount	計數	每次成功交付至排程的 DLQ 時發出。使用此選項來判斷事件何時傳送至

命名空間	指標	單位	說明
			DLQ，然後檢查傳送至排程 DLQ 的事件，以取得可協助您判斷失敗原因的其他詳細資訊。

命名空間	指標	單位	說明
AWS/Scheduler	InvocationsFailedToBeSentToDeadLetterCount	計數	當 EventBridge 排程器無法將事件交付至 DLQ 時發出。使用這兩個指標來判斷 EventBridge 排程器無法將事件傳送至 DLQ 的原因，並修改您的 DLQ 組態以解決問題。
AWS/Scheduler	InvocationsFailedToBeSentToDeadLetterCount_<error_code>	計數	<p>以下是您指定為 <code>InvocationsFailedToBeSentToDeadLetterCount_&lt;error_code&gt;</code> DLQ 的 Amazon SQS 佇列不存在時的指標範例：</p> <p><code>InvocationsFailedToBeSentToDeadLetterCount_ AWS.SimpleQueueService.NonE</code></p>

命名空間	指標	單位	說明
			<b>xistentQueue</b>
AWS/Scheduler	InvocationsSentToDeadLetterCount_Truncated_MessageSize Exceeded	計數	當傳送至 DLQ 的事件承載超過 Amazon SQS 允許的大小上限時發出，且 EventBridge 排程器會截斷您在排程 Input 屬性中指定的承載。

## EventBridge 排程器用量指標

CloudWatch 會收集追蹤某些 AWS 資源使用情況的指標。這些指標對應至 AWS 服務配額。追蹤這些指標可協助您主動管理配額。如需服務配額的詳細資訊，請參閱 [配額](#)。

這些指標包含在 AWS/Usage 命名空間中，而不是 AWS/Scheduler，並且每分鐘收集一次。會在此命名空間中 CloudWatch 發佈具有維度 Service、Class、Type 和 的指標 Resource。

### 用量指標的常見維度

維度	說明
Service	包含資源 AWS 的服務名稱。對於 EventBridge 排程器 用量指標，值為 Scheduler 。
Class	正在追蹤的資源類別。對於 EventBridge 排程器 用量指標，值為 None。
Type	正在追蹤的資源類型。對於 API 用量指標，值為 API。對於資源計數指標，值為 Resource。

維度	說明
Resource	要追蹤的特定資源。對於 API 用量指標，這是 API 操作名稱。對於資源計數指標，這是要計數的資源類型。

## API 用量指標

API 用量指標會追蹤您帳戶中執行的 API 操作數目。使用這些指標來監控 API 呼叫量並管理您的 API 速率配額。

指標名稱為 CallCount。此指標最有用的統計資料是 SUM，代表 1 分鐘期間的總操作計數。

### API 用量指標

指標	資源	說明
CallCount	CreateSchedule	您帳戶中執行的 CreateSchedule API 操作數目。
CallCount	CreateScheduleGroup	您帳戶中執行的 CreateScheduleGroup API 操作數目。
CallCount	DeleteSchedule	您帳戶中執行的 DeleteSchedule API 操作數目。
CallCount	DeleteScheduleGroup	您帳戶中執行的 DeleteScheduleGroup API 操作數目。
CallCount	GetSchedule	您帳戶中執行的 GetSchedule API 操作數目。
CallCount	GetScheduleGroup	您帳戶中執行的 GetScheduleGroup API 操作數目。
CallCount	ListScheduleGroups	您帳戶中執行的 ListScheduleGroups API 操作數目。
CallCount	ListSchedules	您帳戶中執行的 ListSchedules API 操作數目。

指標	資源	說明
CallCount	ListTagsForResource	您帳戶中執行的 ListTagsForResource API 操作數目。
CallCount	TagResource	您帳戶中執行的 TagResource API 操作數目。
CallCount	UntagResource	您帳戶中執行的 UntagResource API 操作數目。
CallCount	UpdateSchedule	您帳戶中執行的 UpdateSchedule API 操作數目。

例如，具有下列維度的 CallCount 指標表示您的帳戶中呼叫 CreateSchedule API 操作的次數：

- "Service" : "Scheduler"
- "類別" : "無"
- "Type" : "API"
- "Resource" : "CreateSchedule"

## 資源計數指標

資源計數指標會追蹤您帳戶中的大約資源數量。使用這些指標來監控何時接近您的服務配額限制，可讓您在容量不足之前請求增加配額。

指標名稱為 ResourceCount。此指標最實用的統計資料是 Maximum。

## 資源計數指標

指標	資源	說明
ResourceCount	ApproximateSchedule	您帳戶中的排程大約數量。使用此指標來監控您何時接近排程配額限制，以協助您避免在呼叫 ServiceQuotaExceededException 時發生錯誤 CreateSchedule 。

指標	資源	說明
		當排程數量少於 100 萬個時，此指標可能會顯示零。若要在接近排程配額時通知警示，建議您使用閾值為 500 萬或更高的 Maximum 統計資料，因為預設排程配額為 1,000 萬。
ResourceCount	ApproximateScheduleGroup	您帳戶中的排程群組大約數量。使用此指標來監控您何時接近排程群組配額限制，以協助您避免在呼叫 ServiceQuotaExceededException 時發生錯誤 CreateScheduleGroup。

例如，具有下列維度的 ResourceCount 指標和 Maximum 統計資料指出您帳戶中的排程大致數量：

- "Service" : "Scheduler"
- "類別" : "無"
- "Type" : "Resource"
- "Resource" : "ApproximateSchedule"

## 使用記錄 Amazon EventBridge Scheduler API 呼叫 AWS CloudTrail

Amazon EventBridge Scheduler 已與整合 AWS CloudTrail，此服務提供 EventBridge Scheduler AWS 中使用者、角色或服務所採取動作的記錄。CloudTrail 會將 EventBridge Scheduler 的所有 API 呼叫擷取為事件。擷取的呼叫包括從 EventBridge Scheduler 主控台進行的呼叫，以及對 EventBridge Scheduler API 操作的程式碼呼叫。如果您建立追蹤，則可以啟用 CloudTrail 事件持續交付至 Amazon S3 儲存貯體，包括 EventBridge Scheduler 的事件。即使您未設定追蹤，依然可以透過 CloudTrail 主控台的事件歷史記錄檢視最新事件。您可以使用 CloudTrail 收集的資訊，判斷對 EventBridge Scheduler 提出的請求、提出請求的 IP 地址、提出請求的人員、提出請求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱 [「AWS CloudTrail 使用者指南」](#)。

## CloudTrail 中的 EventBridge 排程器資訊

建立帳戶 AWS 帳戶 時，您的上會啟用 CloudTrail。當活動在 EventBridge Scheduler 中發生時，該活動會記錄於 CloudTrail 事件，以及事件歷史記錄中的其他服務 AWS 事件。您可以在 [中檢視、搜尋和下載最近的事件 AWS 帳戶](#)。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 中的事件 AWS 帳戶，包括 EventBridge Scheduler 的事件，請建立追蹤。線索能讓 CloudTrail 將日誌檔案交付至 Amazon 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。追蹤會記錄 AWS 分割區中所有 區域的事件，並將日誌檔案交付至您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析 CloudTrail 日誌中收集的事件資料並對其採取行動。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務和整合](#)
- [設定 CloudTrail 的 Amazon SNS 通知](#)
- [接收多個區域的 CloudTrail 日誌檔案](#)和[接收多個帳戶的 CloudTrail 日誌檔案](#)

CloudTrail 會記錄所有 EventBridge Scheduler API 動作，並記錄在 [Amazon EventBridge Scheduler API 參考](#)中。例如，對 CreateSchedule、UpdateSchedule 和 DeleteSchedule 動作發出的呼叫會在 CloudTrail 記錄檔案中產生專案。

每一筆事件或日誌專案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：

- 請求是否使用根或 AWS Identity and Access Management (IAM) 使用者登入資料提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱 [CloudTrail userIdentity 元素](#)。

## 了解 EventBridge Scheduler 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到您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌專案。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 日誌檔並非依公有 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

AWS 您的帳戶具有每個 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的。您可以為大多數配額請求增加，但有些無法增加。

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 排程器最常參考的配額是排程數（每個區域預設 10,000,000 個，可調整至數十億個）、CreateSchedule 請求率（主要區域中預設 5,000 TPS，可調整至數萬個）和調用調節限制（主要區域中預設 1,000 TPS，可調整至數萬個）。這三個都是可調整的 — 如需詳細資訊，請參閱 Service Quotas [Service Quotas 使用者指南中的請求提高配額](#)。

AWS 您的帳戶具有與 EventBridge 排程器相關的下列配額。

名稱	預設	可調整	說明
CreateSchedule 請求率	us-east-1 :	<a href="#">是</a>	每秒的 CreateSchedule 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。這可以調整為每秒數萬個請求。
	5,000		
	us-east-2 :		
	5,000		
	us-west-2 :		
5,000			
	ap-northeast-1 :		
	5,000		
	ap-south-1 :		
	5,000		

名稱	預設	可調整	說明
	ap-southeast-1 : 5 , 000  ap-southeast-2 : 5 , 000  eu-central-1 : 5 , 000  eu-west-1 : 5 , 000  eu-west-2 : 5 , 000  sa-east-1 : 5 , 000  每個其他支援的區域 : 250		
CreateScheduleGroup 請求率	每個受支援的區域 : 10	<u>是</u>	每秒的 CreateScheduleGroup 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。

名稱	預設	可調整	說明
DeleteSchedule 請求率	us-east-1 : 1,000  us-east-2 : 1,000  us-west-2 : 1,000  ap-northeast-1 : 1,000  ap-south-1 : 1,000  ap-southeast-1 : 1,000  ap-southeast-2 : 1,000  eu-central-1 : 1,000  eu-west-1 : 1,000  eu-west-2 : 1,000  sa-east-1 : 1,000  每個其他支援的區域 : 250	<u>是</u>	每秒 DeleteSchedule 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。這可以調整為每秒數萬個請求。

名稱	預設	可調整	說明
DeleteScheduleGroup 請求率	每個受支援的區域：10	<u>是</u>	每秒 DeleteScheduleGroup 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。

名稱	預設	可調整	說明
GetSchedule 請求率	us-east-1 : 1,000  us-east-2 : 1,000  us-west-2 : 1,000  ap-northeast-1 : 1,000  ap-south-1 : 1,000  ap-southeast-1 : 1,000  ap-southeast-2 : 1,000  eu-central-1 : 1,000  eu-west-1 : 1,000  eu-west-2 : 1,000  sa-east-1 : 1,000  每個其他支援的區域 : 250	<u>是</u>	每秒的 GetSchedule 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。這可以調整為每秒數萬個請求。

名稱	預設	可調整	說明
GetScheduleGroup 請求率	每個受支援的區域：10	<u>是</u>	每秒的 GetScheduleGroup 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。

名稱	預設	可調整	說明
調用限流每秒交易的限制	us-east-1 : 1,000  us-east-2 : 1,000  us-west-2 : 1,000  ap-northeast-1 : 1,000  ap-south-1 : 1,000  ap-southeast-1 : 1,000  ap-southeast-2 : 1,000  eu-central-1 : 1,000  eu-west-1 : 1,000  eu-west-2 : 1,000  sa-east-1 : 1,000  每個其他受支援的 區域 : 500	<u>是</u>	調用是交付至已定義目標的排程承載。達到上限之後，就會限流調用，也就是說，雖然仍會繼續呼叫，但是會延遲一些。這可以調整為每秒數萬筆交易。

名稱	預設	可調整	說明
ListScheduleGroups 請求率	每個受支援的區域：10	<a href="#">是</a>	每秒 ListScheduleGroups 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。
ListSchedules 請求率	每個受支援的區域：50	<a href="#">是</a>	每秒的 ListSchedules 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。
ListTagsForResource 請求率	每個受支援的區域：10	<a href="#">是</a>	列出與排程器資源相關聯的標籤。
排程群組的數量	每個受支援的區域：500	<a href="#">是</a>	每個區域的排程群組數目上限。
排程數量	每個支援的區域：10,000,000	<a href="#">是</a>	每個區域的排程數目上限。此配額包括已完成執行的一次性排程。我們建議您使用 ActionAfterCompletion 功能，將排程設定為完成後自動刪除。這可以調整成數十億個排程。
TagResource 請求率	每個受支援的區域：1	<a href="#">是</a>	將一或多個標籤（索引鍵/值對）指派給指定的排程器資源。
UntagResource 請求率	每個受支援的區域：1	<a href="#">是</a>	從指定的排程器資源移除一或多個標籤。

名稱	預設	可調整	說明
UpdateSchedule 請求率	us-east-1 : 1,000  us-east-2 : 1,000  us-west-2 : 1,000  ap-northeast-1 : 1,000  ap-south-1 : 1,000  ap-southeast-1 : 1,000  ap-southeast-2 : 1,000  eu-central-1 : 1,000  eu-west-1 : 1,000  eu-west-2 : 1,000  sa-east-1 : 1,000  每個其他支援的區域 : 250	<u>是</u>	每秒的 UpdateSchedule 請求上限。當您達到此配額時，EventBridge 排程器會在剩餘的間隔內拒絕此操作的請求。這可以調整為每秒數萬個請求。

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](#)。

## EventBridge 排程器中的配額疑難排解

使用下列資訊來協助您診斷和修正有關 EventBridge Scheduler 配額的常見問題。

### ServiceQuotaExceededException

我在 CreateSchedule、GetSchedule、DeleteSchedule 或 UpdateSchedule 請求率上收到限流錯誤，即使我低於預設費率限制。

#### 常見原因

2023 年 9 月 7 日，EventBridge 排程器開始支援 ScheduleGroup ARN (Amazon Resource Name)，而不是執行角色信任政策中的排程 ARN。允許在其信任政策中繼續使用排程 ARNs 的客戶可能有 50 TPS 的限制，而不是預設限制 250 到 1000 TPS（取決於區域）。

#### Resolution

請聯絡 [支援](#) 以請求更高的上限。

#### 預防

使用下列其中一種方式修改現有的信任政策：

- 從角色移除所有範圍。
- 調整角色的規模，使其可以使用排程 ARN 或 ScheduleGroup ARN 擔任。

例如，假設您有下列現有的信任政策：

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "scheduler.amazonaws.com"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
    "StringEquals": {
      "aws:SourceArn":
        "arn:aws:scheduler:region:account:schedule/schedule_group/schedule"
    }
  }
}
```

```
    }  
  }  
}
```

您可以將信任政策更新為下列項目：

```
{  
  "Effect": "Allow",  
  "Principal": {  
    "Service": "scheduler.amazonaws.com"  
  },  
  "Action": "sts:AssumeRole",  
  "Condition": {  
    "ForAnyValue:StringEquals": {  
      "aws:SourceArn": [  
        "arn:aws:scheduler:region:account:schedule/schedule_group/schedule",  
        "arn:aws:scheduler:region:account:schedule-group/schedule_group"  
      ]  
    }  
  }  
}
```

# Amazon EventBridge 排程器故障診斷

您可以使用本節中的主題來疑難排解常見的 Amazon EventBridge 排程器問題。

## 主題

- [我的排程失敗，並出現目標錯誤](#)
- [排程執行角色許可問題](#)
- [了解和管理服務配額](#)
- [排程模式和觸發計時問題](#)
- [建立排程模式和 Cron 表達式](#)
- [我的目標是否被觸發？](#)
- [範本化與通用目標](#)
- [無效的通用目標輸入組態](#)
- [觸發非預期調用的排程更新](#)
- [停用或啟用一次性排程](#)

## 我的排程失敗，並出現目標錯誤

目標呼叫失敗是 EventBridge 排程器最常見的問題之一。這些失敗可能有多種原因：

### 常見原因：

- 遺失或不正確的目標參數。
- 網路連線問題。
- API 限流。
- 目標組態不正確。

## 疑難排解步驟

1. 設定無效字母佇列 (DLQ)
  - DLQ 可協助您擷取和分析失敗的調用。
  - 失敗的調用會傳送至 DLQ，其中包含詳細的錯誤訊息。

- 若要設定 DLQ，請將其新增至排程組態：

```
{
  "DeadLetterConfig": {
    "Arn": "arn:aws:sqs:region:account-id:MyDLQ"
  }
}
```

注意：如果您的 DLQ 使用 KMS 金鑰加密，請確定金鑰政策允許 EventBridge 排程器使用它：

```
{
  "Sid": "Allow EventBridge Scheduler to use the key",
  "Effect": "Allow",
  "Principal": {
    "Service": "scheduler.amazonaws.com"
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*"
}
```

## 2. 驗證 API 參數

- 確保目標 API 呼叫的所有必要參數都存在且格式正確。
- 檢查參數值是否在允許的範圍內。
- 如果使用 VPC 端點，請確認可從 VPC 存取 API 端點。

## 3. 檢閱網路組態

- 如果呼叫因暫時性網路問題而失敗，請實作[重試](#)邏輯。
- 重試政策範例：

```
{
  "RetryPolicy": {
    "MaximumRetryAttempts": 3,
    "MaximumEventAgeInSeconds": 3600
  }
}
```

```
}  
}
```

#### 4. 檢查目標特定的組態

- 對於範本目標（例如 ECS 任務），請確保您透過排程建立 API 的 `Target.Input` 參數提供覆寫。
- 確認您的目標服務受到[支援](#)且設定正確。

## 排程執行角色許可問題

IAM 角色許可問題是排程執行失敗的常見原因。以下是疑難排解和解決這些問題的方法：

### 常見原因

- 缺少目標服務所需的許可
- 排程中的角色組態不正確
- 缺少與 EventBridge 排程器服務的信任關係
- 存取加密資源的許可不足

### 徵狀

- 在 CloudWatch `TargetErrorCount` 中增加指標
- 排程無法在排程組態中沒有明顯問題的情況下執行

### 疑難排解步驟

#### 1. 監控 CloudWatch 指標

- 檢查 CloudWatch 中的 `TargetErrorCount` 指標。

#### 2. 使用無效字母佇列 (DLQ) 確認許可問題

- 為您的排程設定 DLQ。
- 如果您的目標有許可問題，且 DLQ 已正確設定，則您會在 DLQ 中看到失敗的調用，其中包含許可相關的錯誤訊息。
- 如果即使 CloudWatch 指標中顯示的執行失敗，DLQ 仍然空白，這可能表示許可問題，導致 EventBridge 排程器無法寫入 DLQ 本身。

**Note**

確保 DLQ 本身具有正確的許可。如果加密，請確定 EventBridge 排程器具有使用 KMS 金鑰的許可。

### 3. 驗證信任關係

- 確保您的 IAM 角色與 EventBridge 排程器具有正確的信任關係：

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

### 4. 檢查排程執行角色許可

- 排程的執行角色需要特定許可，才能叫用不同的目標類型。
- 包含在排程執行角色政策中的許可範例：

```
// For Lambda function targets - add to schedule execution role
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "lambda:InvokeFunction"
    ],
    "Resource": "arn:aws:lambda:region:account-id:function:function-name"
  }]
}

// For SQS queue targets - add to schedule execution role
{
```

```
"Version": "2012-10-17",
"Statement": [{
  "Effect": "Allow",
  "Action": [
    "sqs:SendMessage"
  ],
  "Resource": "arn:aws:sqs:region:account-id:queue-name"
}]
}
```

## 5. 檢查加密的資源存取

- 如果您的目標使用加密的資源（例如 KMS 加密的 SQS 佇列），請確定您的角色具有使用 KMS 金鑰的許可：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "arn:aws:kms:region:account-id:key/key-id"
    }
  ]
}
```

## 6. 驗證角色 ARN 組態

- 確保排程組態中的角色 ARN 正確無誤。
- 確認角色與您的排程位於相同的 AWS 帳戶和區域中。

# 了解和管理服務配額

如果您在建立排程或查看節制調用時遇到問題，您可能會達到服務配額限制。EventBridge 排程器具有排程數量、排程群組和調用率的配額，可能因區域而異。

## 識別配額問題

若要判斷您是否達到配額限制：

### 1. 監控 CloudWatch 指標

- 檢查 `InvocationThrottleCount` 指標。此指標的增加表示您超過了調用率限制。
- 檢閱 `InvocationAttemptCount` 指標以了解您目前的用量。

### 2. 注意特定錯誤訊息

- 建立或修改排程時，`LimitExceededException` 表示您已達到排程或排程群組的數量上限。
- API 呼叫傳回調節錯誤，表示您超過 API 請求配額。

## 解決配額問題

如果您確定要達到配額限制：

1. 檢閱並最佳化您目前的排程。請考慮合併類似的排程或移除未使用的排程。
2. 針對 API 限流，請在 API 呼叫中實作 [具有退避的重試](#)。
3. 如果您需要更高的配額，請透過 Service Quotas 主控台請求提高配額。選取 EventBridge 排程器，選擇您需要增加的配額，然後提交具有業務理由的請求。

## 排程模式和觸發計時問題

使用者有時會遇到排程未在預期時間觸發的問題。這最常見的原因是對排程模式、日光節約時間變更或彈性時段的誤解。

### 常見原因

- cron 表達式的錯誤解譯。
- 在日光節約時間變更期間的非預期行為。
- 關於彈性時段的混淆。
- 對速率表達式的誤解。

## 疑難排解步驟

1. 驗證 cron 表達式

- 確保您的 Cron 表達式格式正確。
- 請注意，您無法在 Cron 表達式中同時指定 day-of-month 和 day-of-week 欄位。

## 2. 時區考量

- 在建立排程時選取您偏好的時區。
- 了解日光節約時間如何影響您的排程，因為此調整是以 UTC 為基礎。

日光節約影響的範例：如果您設定排程在 GMT 上午 7:00 執行：

- 冬天：排程在 GMT 上午 7:00 執行（因為 GMT = UTC）
- 在夏天：排程仍然在 UTC 上午 7:00 執行，現在是 GMT/BST 上午 6:00

如果您需要在全年的相同當地時間執行排程，請務必在建立排程時選取適當的時區，以及日光節約如何影響該時區。

## 3. 了解彈性時段

- [彈性時段](#) 允許 EventBridge 排程器最佳化調用。
- 排程可能不會在視窗開始時完全觸發。
- 監控實際調用時間，以了解行為。

## 4. 檢閱速率和 Cron 表達式

- 確保速率表達式格式正確（例如 `rate(5 minutes)`、`rate(1 hour)`）。
- 對於速率和 Cron 表達式，請注意，排程調用不會緊固到一分鐘的第 0 秒。
- 排程可能會在指定的分鐘內觸發，但不一定會在分鐘的確切開始時觸發。

例如：

- 使用的排程 `rate(1 hour)` 可能會在下午 2:00:45、下午 3:00:32、下午 4:00:18 等時間執行。
- 為設定的 Cron 排程 `0 * * * ? *`（每小時）可能會在下午 2:00:15、下午 3:00:07、下午 4:00:52 等時間執行。

## 5. 監控 CloudWatch 指標

- 使用 `InvocationAttemptCount` 指標來驗證您的排程是否正在觸發。
- 檢查調用 `TargetErrorCount` 是否失敗。
- 如果您已設定無效字母佇列，請監控 `InvocationsSentToDeadLetterCount` 以追蹤失敗的呼叫。

## 建立排程模式和 Cron 表達式

使用者經常在建立排程模式時遇到問題，特別是使用 Cron 表達式時。以下是一些常見問題以及如何解決這些問題：

### 常見問題

- 不正確的 cron 語法
- 嘗試使用不支援的 cron 功能
- 混淆哪些欄位可以一起使用

### 疑難排解步驟

1. 檢閱 cron 表達式語法
  - 確保您的 Cron 表達式遵循正確的[格式](#)：Minutes Hours Day-of-month Month Day-of-week Year。
  - 請記住，EventBridge 排程器使用 cron 標準搭配額外的年份欄位。
2. 了解限制
  - 您無法同時指定day-of-month和day-of-week欄位，如[此處](#)所述。
  - 不支援頻率多於 1 分鐘的 Cron 表達式。
3. 使用排程預覽功能
  - 建立或編輯排程時，EventBridge 排程器會提供接下來 10 個執行時間的預覽。
  - 使用此預覽來驗證您的排程將在預定時間執行。
  - 如果預覽不符合您的期望，請檢閱和調整您的 Cron 表達式。

## 我的目標是否被觸發？

若要確認您的目標是否正在觸發：

1. 檢查 CloudWatch 指標：
  - InvocationAttemptCount 顯示嘗試的調用次數
  - TargetErrorCount 指出是否有任何調用失敗
  - TargetErrorThrottledCount 顯示您的目標是否受到調節

- `InvocationDroppedCount` 指出是否已捨棄任何調用
2. [設定無效字母佇列 \(DLQ\)](#) 以擷取和分析任何失敗的調用。

## 範本化與通用目標

如果您收到「提供的請求無效：【服務】不是目標的支援服務」之類的錯誤，您可能嘗試使用不支援的服務做為範本化目標。

若要解決此問題：

1. 檢查所需的服務是否支援做為[範本化目標](#)。
2. 如果不支援，請改用[通用目標](#)，並將其設定為對服務進行適當的 API 呼叫。

## 無效的通用目標輸入組態

當您使用[通用目標](#)建立排程時，EventBridge 排程器會驗證目標 ARN 格式，但不會針對下游服務的 API 驗證 Input 欄位的內容。這表示即使 Input 包含目標服務在調用時將拒絕的值，也可以成功建立排程。

具有無效目標輸入組態的排程會在其設定的表達式上觸發，但每次調用都會失敗。在調用排程之前，您可能不會發現組態錯誤，這可能是建立後的數小時或數天。

## 徵狀

- 排程建立時不會發生錯誤，但 `TargetErrorCount` CloudWatch 指標會在每次調用時增加。
- DLQ 訊息包含來自目標服務的錯誤代碼（例如，`InvalidParameterValueException` 或 `ValidationException`），而非 `AWS.Scheduler.InternalServerError`。
- DLQ 訊息 `ERROR_MESSAGE` 中的 參考特定的輸入參數驗證失敗。

## 範例

下列範例顯示通用目標的常見無效輸入組態 AWS Lambda (`arn:aws:scheduler:::aws-sdk:lambda:invoke`)。

不符的限定詞

具有下列輸入的排程會在 2 中指定版本，FunctionName 並在 1 Qualifier 欄位中指定版本：

```
{
  "FunctionName": "MyFunction:2",
  "Qualifier": "1"
}
```

此排程已成功建立，但每次調用都會失敗。DLQ 訊息包含：

- ERROR\_CODE: InvalidParameterValueException
- ERROR\_MESSAGE: The derived qualifier from the function name does not match the specified qualifier.

### 無效的函數名稱

具有下列輸入的排程會指定 的空格限定值FunctionName：

```
{
  "FunctionName": " "
}
```

DLQ 訊息包含：

- ERROR\_CODE: ValidationException
- ERROR\_MESSAGE：表示函數名稱與所需模式不相符的驗證錯誤。

## 如何解決

1. 設定 DLQ。一律為使用通用目標的排程[設定無效字母佇列](#)。DLQ 訊息屬性 (ERROR\_CODE 和 ERROR\_MESSAGE) 包含目標服務傳回的特定錯誤，可識別無效的輸入參數。
2. 根據目標服務 API 驗證輸入參數。建立排程之前，請直接呼叫目標 API，確認 Input 欄位中的 JSON 是否包含有效值。例如，使用 AWS Lambda Invoke API 以相同的參數叫用 AWS Lambda 函數，以確認請求成功。
3. 使用一次性排程進行測試。建立一次性排程，以在設定週期性排程之前驗證目標調用是否成功。
4. 檢閱目標服務 API 參考。檢查您目標服務的 API 參考，以確認所需的參數、有效值範圍和限制條件。如需 AWS Lambda Invoke，請參閱《AWS Lambda 開發人員指南》中的[叫用](#)。

## 觸發非預期調用的排程更新

當您變更排程時，調用可能不會立即反映更新的排程。允許一小段時間來讓變生效。例如，如果您更新接近其原始觸發時間的排程，您可能會根據原始排程組態看到 呼叫。

## 停用或啟用一次性排程

在超過原始排程時間後重新啟用一次性排程時，排程可能會立即叫用其目標。即使排程在原始執行時間之前已停用，也會發生這種情況。

例如：

- 目前時間：13：15 UTC
- 為 建立一次性排程：13：30 UTC
- 在 UTC 13：30 之前停用排程
- 排程在 UTC 14：00 重新啟用
- 結果：重新啟用時可立即叫用目標

# EventBridge 排程器使用者指南的文件歷史記錄

下表說明 EventBridge 排程器的文件版本。

變更	描述	日期
<a href="#">已更新 AmazonEventBridgeSchedulerReadOnlyAccess 受管政策</a>	更新 AmazonEventBridgeSchedulerReadOnlyAccess 受管政策以使用萬用字元動作 (scheduler:List* 和 scheduler:Get* )。 <ul style="list-style-type: none"> <li>• <a href="#">the section called “AWS 受管政策更新”</a></li> </ul>	2026 年 3 月 25 日
<a href="#">變更執行角色和預防混淆代理人</a>	此更新說明當您在角色的許可政策中實作預防混淆代理人時，如何將執行角色套用至排程群組資源的變更。 <ul style="list-style-type: none"> <li>• <a href="#">the section called “預防混淆代理人”</a></li> </ul>	2023 年 9 月 7 日
<a href="#">完成後自動刪除排程</a>	EventBridge 排程器支援自動刪除。當您設定自動刪除時，EventBridge 排程器會在最後一次計劃調用後刪除您的排程。 <ul style="list-style-type: none"> <li>• <a href="#">the section called “排程完成後刪除”</a></li> </ul>	2023 年 8 月 2 日
<a href="#">更新使用通用目標的主題</a>	更新 EventBridge 排程器可以鎖定目標並與之整合的支援服務清單。此更新也包含不支援的 GET API 操作清單，包括對通用目標範例的改進，以及指南中對的其他次要改進。	2023 年 3 月 17 日

<a href="#">更新了沒有開始日期的以速率為基礎的排程資訊</a>	<ul style="list-style-type: none"><li>• <a href="#">the section called “使用通用目標”</a></li></ul> <p>新增如果您未指定 <code>StartDate</code>，EventBridge 排程器如何處理速率型排程的相關資訊 <code>StartDate</code>。</p>	2023 年 3 月 17 日
<a href="#">管理排程器群組的新主題</a>	<ul style="list-style-type: none"><li>• <a href="#">the section called “以速率為基礎的排程”</a></li></ul> <p>新增有關如何使用 EventBridge 排程器建立排程器群組的新章節。使用本章了解如何建立群組、將排程新增至群組、套用標籤以更輕鬆地管理和監控 EventBridge 排程器資源，最後刪除群組。</p>	2023 年 3 月 17 日
<a href="#">有關日光節約時間和時區的新主題</a>	<ul style="list-style-type: none"><li>• <a href="#">管理排程群組</a></li></ul> <p>新增章節，說明 EventBridge 排程器如何處理日光節約時間，以及如何在不同時區建立排程。</p>	2022 年 11 月 17 日
<a href="#">指標的新主題</a>	<ul style="list-style-type: none"><li>• <a href="#">the section called “日光節約時間”</a></li><li>• <a href="#">the section called “時區”</a></li></ul> <p>新增了新主題，說明 EventBridge 排程器發佈至 CloudWatch 的指標。您可以使用這些指標來監控調用失敗，並了解如何解決排程的問題。</p>	2022 年 11 月 15 日

## [初始版本](#)

EventBridge 排程器使用者指南 2022 年 11 月 10 日的初始版本。

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。