



Manual do usuário

# EventBridge Agendador



# EventBridge Agendador: Manual do usuário

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

---

# Table of Contents

|   |    |
|---|----|
| O que é o EventBridge Scheduler? .....                    | 1  |
| Principais características do EventBridge Scheduler ..... | 1  |
| Acessando o EventBridge Scheduler .....                   | 2  |
| Configuração .....  | 3  |
| Cadastre-se para AWS .....                                | 3  |
| Criar um usuário do IAM .....                             | 3  |
| Políticas gerenciadas pelo uso .....                      | 4  |
| Configurar o perfil de execução .....                     | 5  |
| Configurar um destino .....                               | 9  |
| Próximas etapas .....                                     | 12 |
| Conceitos básicos .....                                   | 13 |
| Pré-requisitos .....                                      | 14 |
| Usar o console .....                                      | 14 |
| Usar a AWS CLI .....                                      | 18 |
| Uso de SDKs .....   | 18 |
| Próximas etapas .....                                     | 20 |
| Tipos de agendamento .....                                | 21 |
| Agendamentos baseados em taxas .....                      | 22 |
| Sintaxe .....   | 22 |
| Exemplos .....  | 22 |
| Agendamentos baseados em cron .....                       | 23 |
| Sintaxe .....   | 23 |
| Exemplos .....  | 24 |
| Programações únicas .....                                 | 25 |
| Sintaxe .....   | 25 |
| Exemplos .....  | 25 |
| Fusos horários .....                                      | 26 |
| Horário de verão .....                                    | 26 |
| Gerenciando um agendamento .....                          | 28 |
| Alterando o estado do cronograma .....                    | 29 |
| Configurando janelas de tempo flexíveis .....             | 30 |
| Configurar uma fila de mensagens mortas .....             | 31 |
| Criar uma fila do Amazon SQS .....                        | 32 |
| Configure as permissões da função de execução .....       | 33 |

|   |     |
|---|-----|
| Especificar uma fila de mensagens mortas .....        | 33  |
| Recuperar o evento de mensagens mortas .....          | 35  |
| Excluir um agendamento. ....                          | 37  |
| Exclusão após a conclusão do cronograma .....         | 38  |
| Exclusão manual .....                                 | 39  |
| Próximas etapas .....                                 | 39  |
| Gerenciando um grupo de agendamento .....             | 40  |
| Criando um grupo de agendamento .....                 | 41  |
| Etapa 1: criar um novo grupo de agendamento .....     | 41  |
| Associando um cronograma .....                        | 42  |
| Excluindo um grupo de agendamento .....               | 44  |
| Recursos relacionados .....                           | 46  |
| Gerenciando alvos .....                               | 47  |
| Uso de alvos modelados .....                          | 48  |
| Amazon SQS SendMessage .....                          | 49  |
| Lambda Invoke .....                                   | 51  |
| Step Functions StartExecution .....                   | 53  |
| Usando alvos universais .....                         | 55  |
| Ações não compatíveis .....                           | 55  |
| Exemplos .....  | 56  |
| Adicionando atributos de contexto .....               | 58  |
| Próximas etapas .....                                 | 60  |
| Segurança .....                                       | 61  |
| Gerenciamento de acesso .....                         | 62  |
| Público .....   | 62  |
| Autenticando com identidades .....                    | 63  |
| Gerenciando acesso usando políticas .....             | 67  |
| Como o EventBridge Scheduler funciona com o IAM ..... | 69  |
| Usar políticas baseadas em identidade .....           | 77  |
| Prevenção do problema do “confused deputy” .....      | 88  |
| Solução de problemas .....                            | 90  |
| Proteção de dados .....                               | 92  |
| Criptografia em repouso .....                         | 93  |
| Criptografia em trânsito .....                        | 101 |
| Validação de conformidade .....                       | 102 |
| Resiliência .....                                     | 103 |

---

|  |       |
|--|-------|
| Segurança da infraestrutura .....  | 103   |
| Monitoramento e métricas .....   | 105   |
| Monitoramento com CloudWatch .....   | 105   |
| Termos .....   | 106   |
| Dimensões .....  | 106   |
| Acesso às métricas do .....  | 107   |
| Lista de métricas .....  | 107   |
| Métricas de uso .....  | 113   |
| Monitorar com logs do CloudTrail .....   | 115   |
| Informações do EventBridge Scheduler no CloudTrail .....                           | 116   |
| Noções básicas sobre as entradas de arquivos de log do EventBridge Scheduler ..... | 117   |
| Cotas .....  | 118   |
| Histórico do documento .....   | 123   |
| .....  | cxxvi |

# O que é o Amazon EventBridge Scheduler?

O Agendador do Amazon EventBridge é um agendador sem servidor que permite criar, executar e gerenciar tarefas de um serviço gerenciado central. Altamente escalável, o EventBridge Scheduler permite que você agende milhões de tarefas que podem invocar mais de 270 serviços de AWS e mais de 6.000 operações de API. Sem a necessidade de provisionar e gerenciar a infraestrutura ou integrar-se a vários serviços, o EventBridge Scheduler oferece a capacidade de entregar cronogramas em grande escala e reduzir os custos de manutenção.

O EventBridge Scheduler entrega suas tarefas de forma confiável, com mecanismos integrados que ajustam seus cronogramas com base na disponibilidade de metas posteriores. Com o Agendador do EventBridge, você pode criar programações usando expressões cron e rate para padrões recorrentes ou configurar invocações únicas. É possível configurar janelas de tempo flexíveis para entrega, bem como definir limites de repetição e o tempo máximo de retenção para gatilhos com falha.

## Tópicos

- [Principais características do EventBridge Scheduler](#)
- [Acessando o EventBridge Scheduler](#)

## Principais características do EventBridge Scheduler

O EventBridge Scheduler oferece os seguintes recursos principais que você pode usar para configurar metas e escalar suas agendas.

- Alvos modelados — O EventBridge Scheduler oferece suporte a alvos modelados para realizar operações comuns de API usando Amazon SQS, Amazon SNS, Lambda e EventBridge. Com alvos predefinidos, você pode configurar suas agendas rapidamente usando o console do EventBridge Scheduler, o EventBridge Scheduler SDK ou o AWS CLI.
- Alvos universais — O EventBridge Scheduler fornece um parâmetro de alvo universal (UTP) que você pode usar para criar acionadores personalizados que visam mais de 270 serviços de AWS e mais de 6.000 operações de API em um cronograma. Com o UTP, você pode configurar seus acionadores personalizados usando o console do EventBridge Scheduler, o EventBridge Scheduler SDK ou o AWS CLI.

- **Janelas de horário flexíveis** — O EventBridge Scheduler oferece suporte a janelas de horário flexíveis, permitindo que você disperse suas agendas e melhore a confiabilidade de seus gatilhos para casos de uso que não exigem invocação programada precisa de alvos.
- **Tentativas repetidas** — O EventBridge Scheduler fornece pelo menos uma vez a entrega do evento aos alvos, o que significa que pelo menos uma entrega é bem-sucedida com uma resposta do alvo. O EventBridge Scheduler permite que você defina o número de novas tentativas para sua agenda para uma tarefa com falha. O EventBridge Scheduler repete tarefas que falharam com tentativas atrasadas para melhorar a confiabilidade de sua agenda e garantir que as metas estejam disponíveis.

## Acessando o EventBridge Scheduler

Você pode usar o EventBridge Scheduler por meio do console do EventBridge Scheduler, do EventBridge Scheduler SDK, do AWS CLI, ou usando diretamente a API do EventBridge Scheduler.

# Configurando o Amazon EventBridge Scheduler

Antes de usar o EventBridge Scheduler, você deve concluir as etapas a seguir.

## Tópicos

- [Cadastre-se para AWS](#)
- [Criar um usuário do IAM](#)
- [Políticas gerenciadas pelo uso](#)
- [Configurar o perfil de execução](#)
- [Configurar um destino](#)
- [Próximas etapas](#)

## Cadastre-se para AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções on-line.

Parte do procedimento de inscrição envolve receber uma chamada telefônica e digitar um código de verificação no teclado do telefone.

Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar [tarefas que exigem acesso de usuário-raiz](#).

## Criar um usuário do IAM

Para criar um usuário administrador, selecione uma das opções a seguir.



| Selecionar uma forma de gerenciar o administrador | Para  | Por   | Você também pode   |
|---|---|---|--|
| Centro de Identidade do IAM<br>(Recomendado)      | Use credenciais de curto prazo para acessar a AWS.<br><br>Isso está de acordo com as práticas recomendadas de segurança. Para obter informações sobre as práticas recomendadas, consulte <a href="#">Práticas recomendadas de segurança no IAM</a> no Guia do usuário do IAM. | Seguindo as instruções em <a href="#">Conceitos básicos</a> no Guia do usuário do AWS IAM Identity Center .                                   | Configure o acesso programático <a href="#">configurando o AWS CLI para uso AWS IAM Identity Center</a> no Guia do AWS Command Line Interface usuário. |
| No IAM<br>(Não recomendado)                       | Use credenciais de curto prazo para acessar a AWS.  | Seguindo as instruções em <a href="#">Criar o seu primeiro usuário administrador e um grupo de usuários do IAM</a> no Guia do usuário do IAM. | Para configurar o acesso programático, consulte <a href="#">Gerenciamento de chaves de acesso de usuários do IAM</a> no Guia do usuário do IAM.        |

## Políticas gerenciadas pelo uso

Na etapa anterior, você configurou um usuário do IAM com as credenciais para acessar seus AWS recursos. Na maioria dos casos, para usar o EventBridge Agendador com segurança, recomendamos que você crie usuários, grupos ou funções separados apenas com as permissões

necessárias para usar o Agendador. EventBridge O Scheduler oferece suporte às seguintes políticas gerenciadas para casos de uso comuns.

- [the section called “AmazonEventBridgeSchedulerFullAccess”](#)— Concede acesso total ao EventBridge Scheduler usando o console e a API.
- [the section called “AmazonEventBridgeSchedulerReadOnlyAccess”](#)— Concede acesso somente para leitura ao Scheduler. EventBridge

Você pode anexar essas políticas gerenciadas às suas entidades principais do IAM da mesma forma que anexou a política de `AdministratorAccess` na etapa anterior. Para obter mais informações sobre como gerenciar o acesso ao EventBridge Scheduler usando políticas do IAM baseadas em identidade, consulte [the section called “Usar políticas baseadas em identidade”](#)

## Configurar o perfil de execução

Uma função de execução é uma função do IAM que o EventBridge Scheduler assume para interagir com outras pessoas Serviços da AWS em seu nome. Você anexa políticas de permissão a essa função para conceder ao EventBridge Agendador acesso para invocar alvos.

Você também pode criar uma nova função de execução ao usar o console para [criar uma nova agenda](#). Se você usa o console, o EventBridge Scheduler cria uma função em seu nome com permissões com base no alvo escolhido. Quando o EventBridge Scheduler cria uma função para você, a política de confiança da função inclui [chaves de condição](#) que limitam quais diretores podem assumir a função em seu nome. Isso evita o potencial [problema de segurança delegada confusa](#).

As etapas a seguir descrevem como criar uma nova função de execução e como conceder acesso ao EventBridge Scheduler para invocar um destino. Este tópico descreve as permissões para destinos modelados populares. Para obter informações sobre como adicionar permissões para outros destinos, consulte [the section called “Uso de alvos modelados”](#).

Para criar uma função de execução usando o AWS CLI

1. Copie a seguinte política JSON de assumir a função e salve-a localmente como `Scheduler-Execution-Role.json`. Essa política de confiança permite que o EventBridge Scheduler assumira a função em seu nome.

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

```

    "Statement": [
      {
        "Effect": "Allow",
        "Principal": {
          "Service": "scheduler.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
      }
    ]
  }
}

```

### Important

Para configurar uma função de execução em um ambiente de produção, recomendamos a implementação de salvaguardas adicionais para evitar problemas de segurança delegada confusa. Para obter mais informações e um exemplo de política, consulte [the section called “Prevenção do problema do “confused deputy””](#).

2. No AWS Command Line Interface (AWS CLI), insira o comando a seguir para criar uma nova função. Substitua *SchedulerExecutionRole* pelo nome que você deseja atribuir a essa função.

```

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

```

Se o teste for bem-sucedido, você verá o seguinte resultado:

```

{
  "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. Para criar uma nova política que permita que o EventBridge Scheduler invoque um alvo, escolha um dos seguintes alvos comuns. Copie a política de permissão JSON e salve-a localmente como um arquivo `.json`.

#### Amazon SQS – SendMessage

O seguinte permite que o EventBridge Scheduler chame a `sqs:SendMessage` ação em todas as filas do Amazon SQS em sua conta.

```

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

```

#### Amazon SNS – Publish

O seguinte permite que o EventBridge Scheduler execute a `sns:Publish` ação em todos os tópicos do Amazon SNS em sua conta.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "sns:Publish"
      ],

```

```

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

```

## Lambda – Invoke

O seguinte permite que o EventBridge Scheduler chame a `lambda:InvokeFunction` ação em todas as funções do Lambda em sua conta.

```

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

```

4. Execute o seguinte comando para criar a nova política de permissão: Substitua *PolicyName* pelo nome que você deseja atribuir a essa política.

```

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

```

Se for bem-sucedido, você verá o seguinte resultado: Observe o ARN da política. Você usa esse ARN na próxima etapa para anexar a política à nossa função de execução.

```

{
  "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"
  }
}
```

5. Para associar a política à sua função de execução, execute o comando a seguir. Substitua *your-policy-arn* pelo ARN da política que você criou na etapa anterior. Substitua *SchedulerExecutionRole* pelo nome da sua função de execução.

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

A operação `attach-role-policy` não retorna uma resposta na linha de comando.

## Configurar um destino

Antes de criar uma agenda do EventBridge Scheduler, você precisa de pelo menos uma meta para sua agenda invocar. Você pode usar um AWS recurso existente ou criar um novo. As etapas a seguir mostram como criar uma nova fila padrão do Amazon SQS com AWS CloudFormation

### Criar uma nova fila do Amazon SQS

1. Copie o AWS CloudFormation modelo JSON a seguir e salve-o localmente como `SchedulerTargetSQS.json`.

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

```
        "QueueName"
      ]
    }
  },
  "QueueURL": {
    "Description": "The URL of the queue",
    "Value": {
      "Ref": "MyQueue"
    }
  },
  "QueueARN": {
    "Description": "The ARN of the queue",
    "Value": {
      "Fn::GetAtt": [
        "MyQueue",
        "Arn"
      ]
    }
  }
}
}
```

2. A partir do AWS CLI, execute o comando a seguir para criar uma AWS CloudFormation pilha a partir do Scheduler-Target-SQS.json modelo.

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

Se o teste for bem-sucedido, você verá o seguinte resultado:

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

3. Execute o comando a seguir para ver as informações resumidas da sua AWS CloudFormation pilha. Essas informações incluem o status da pilha e as saídas especificadas no modelo.

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

Se for bem-sucedido, o comando cria a fila Amazon SQS e retorna a seguinte saída:

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

Posteriormente neste guia, você usará o valor de QueueARN para configurar a fila como destino para o EventBridge Scheduler.



## Próximas etapas

Depois de concluir a etapa de configuração, use o guia de [introdução](#) para criar seu primeiro EventBridge agendador do Scheduler e invocar um alvo.

# Introdução ao EventBridge Scheduler

Este tópico descreve a criação de uma nova agenda do EventBridge Scheduler. Você usa o console do EventBridge Scheduler, AWS Command Line Interface (AWS CLI) ou AWS SDKs para criar um cronograma com um modelo de alvo do Amazon SQS. Em seguida, você configurará o registro, configurará novas tentativas e definirá um tempo máximo de retenção para tarefas com falha. Depois de criar a agenda, você verificará se sua agenda invoca com sucesso o alvo e envia uma mensagem para a fila de alvo.

## Note

Para seguir esse guia, recomendamos que você configure os usuários do IAM com as permissões mínimas exigidas descritas em [the section called “Usar políticas baseadas em identidade”](#). Depois de criar e configurar um usuário, execute o comando a seguir para definir suas credenciais de acesso. Você precisa de seu ID de chave de acesso da e de uma chave de acesso secreta para configurar o 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
```

Para obter mais informações sobre as diferentes maneiras de definir suas credenciais, consulte [Configurações e precedência](#) no Guia do usuário da versão 2 do AWS Command Line Interface.

## Tópicos

- [Pré-requisitos](#)
- [Criar uma agenda usando o console do Agendador do EventBridge](#)
- [Crie um cronograma usando o AWS CLI](#)
- [Criar uma agenda usando os SDKs do EventBridge Scheduler](#)
- [Próximas etapas](#)

## Pré-requisitos

Antes de executar as etapas nesta seção, você deverá fazer o seguinte:

- Conclua as tarefas descritas em [Configuração](#)

## Criar uma agenda usando o console do Agendador do EventBridge

Para criar uma nova programação usando o console

1. Faça login no AWS Management Console e, em seguida, escolha o link a seguir para abrir a seção EventBridge Scheduler do console EventBridge: <https://us-west-2.console.aws.amazon.com/scheduler/home?region=us-west-2#home>

### Note

Você pode mudar o seu Região da AWS usando o seletor de região do AWS Management Console.

2. Na página Programações, escolha Criar uma programação.
3. Na página Especificar detalhes da programação, na seção Nome e descrição da programação, faça o seguinte:
  - a. Em Nome da programação, insira um nome para a programação. Por exemplo, **MyTestSchedule**
  - b. Para Descrição - opcional, insira uma descrição para a seu agendamento. Por exemplo, **My first schedule**.
  - c. Para Grupo de agendamento, escolha um grupo de agendamento na lista suspensa. Se você ainda não criou nenhum grupo de agendamento, pode escolher o grupo default para sua agenda. Para criar um novo grupo de agendamento, escolha o link criar seu próprio agendamento na descrição do console. Você usa grupos de programação para adicionar tags a eles.
4. Na seção Schedule details (Detalhes do agendamento), faça o seguinte:
  - a. Em Ocorrência, selecione uma das opções de padrão a seguir. As opções de configuração mudam dependendo do padrão selecionado.

- Agendamento único – Um agendamento único invoca um alvo somente uma vez na data e hora que você especificar.

Em Data e hora, insira uma data válida no formato YYYY/MM/DD. Em seguida, especifique um carimbo de data e hora no formato hh:mm de 24 horas. Por fim, escolha um fuso horário nas opções suspensas.


- Agendamento recorrente – Um agendamento recorrente invoca uma meta em uma taxa que você especifica usando uma expressão cron ou expressão de taxa.

Escolho agendamento baseado em Cron para configurar um agendamento usando uma expressão cron. Para usar uma expressão de taxa, escolho agendamento baseado em Taxa e insira um número positivo para Valor e, em seguida, escolha uma Unidade nas opções suspensas.

Para obter mais informações sobre o uso de expressões cron e rate, consulte [Tipos de agendamento](#).

- b. Para Janela de tempo flexível, escolha Desativado para desativar a opção ou escolher uma das janelas de tempo predefinidas da lista em cascata. Por exemplo, se você escolher 15 minutos e definir uma programação recorrente para invocar sua meta uma vez a cada hora, a programação será executada em até 15 minutos após o início de cada hora.

5.

 Note

O recurso de janela de horário flexível não está disponível com agendas únicas.


Se você escolheu agendamento recorrente na etapa anterior, na seção Prazo, especifique um fuso horário e, opcionalmente, defina uma data e hora de início e uma data e hora de término para o agendamento. Uma agenda recorrente sem data de início começará assim que for criada e disponibilizada. Uma agenda recorrente sem data de término continuará invocando sua meta indefinidamente.

6. Escolha Next (Próximo).

7. Na página Selecionar destino, faça o seguinte:


- a. Selecione alvos modelados e escolha uma API-alvo. Neste exemplo, escolheremos o alvo modelado do **SendMessage** do Amazon SQS.

- b. Na seção `SendMessage`, para a fila SQS, escolha um ARN de fila existente do Amazon SQS, como `arn:aws:sqs:us-west-2:123456789012:TestQueue` na lista suspensa. Para criar uma nova fila, escolha `Create new SQS queue` para navegar até o console do Amazon SQS. Depois de terminar de criar uma fila, retorne ao console do EventBridge Scheduler e atualize o menu suspenso. O novo ARN da fila aparece e pode ser selecionado.
  - c. Em `Target`, insira a carga que você deseja que o EventBridge Scheduler entregue ao alvo. Neste exemplo, enviaremos a seguinte mensagem para a fila de alvo: **Hello, it's EventBridge Scheduler.**
8. Escolha `Avançar` e, na página `Configurações` - opcional, faça o seguinte:
  9.
    - a. Na seção `Estado do agendamento`, em `Ativar agendamento`, ative ou desative o recurso usando o botão. Por padrão, o EventBridge Scheduler habilita seu agendamento.
    - b. Na seção `Ação após a conclusão do cronograma`, configure a ação que o EventBridge Scheduler executará após a conclusão do agendamento:
      - Escolha `EXCLUIR` se quiser que a agenda seja excluída automaticamente. Para agendamentos únicos, isso ocorre depois que o cronograma invoca o alvo uma vez. Para agendas recorrentes, isso ocorre após a última invocação planejada da agenda. Para obter mais informações sobre a exclusão automática, consulte [the section called "Exclusão após a conclusão do cronograma"](#).
      - Escolha `NENHUM`, ou não escolha um valor, se você não quiser que o EventBridge Scheduler execute nenhuma ação após a conclusão do agendamento.
    - c. Na seção `Política de novas tentativas e fila de mensagens não entregues (DLQ)`, em `Política de novas tentativas`, ative `novas tentativas` para configurar uma política de novas tentativas para a seu agendamento. Com políticas de novas tentativas, se uma programação falhar em invocar seu destino, o Agendador do EventBridge executará novamente a programação. Se configurado, você deve definir o tempo máximo de retenção e as novas tentativas para a programação.
    - d. Para `Idade máxima do evento` - opcional, insira as horas e os minutos máximos que o EventBridge Scheduler deve manter um evento não processado.

 Note

O valor máximo é 24 horas.

- e. Para Máximo de tentativas, insira o número máximo de vezes que o Agendador do EventBridge tentará novamente a programação se o destino retornar um erro.

 Note

O valor máximo é 185 novas tentativas.

- f. Para Filas de mensagens não entregues (DLQ), selecione uma das seguintes opções:
- Nenhum — Escolha essa opção se você não quiser configurar uma DLQ.
  - Selecione uma fila do Amazon SQS em minha conta de AWS como DLQ — Escolha essa opção e, em seguida, selecione um ARN de fila na lista suspensa e configure um DLQ com Conta da AWS igual àquela onde você está criando o agendamento.
  - Especifique uma fila do Amazon SQS em outra conta do AWS como DLQ — Escolha essa opção e, em seguida, insira o ARN da fila configurada como DLQ, se a fila estiver em outra Conta da AWS. Você deve inserir o ARN exato da fila para usar essa opção.
- g. Na seção de Criptografia, selecione Personalizar configurações de criptografia (avanzado) para utilizar uma chave de KMS gerenciada pelo cliente para criptografar sua entrada de alvo. Se você escolher essa opção, insira uma chave ARN do KMS existente ou escolha Criar uma chave KMS do AWS para navegar para o console do AWS KMS. Para obter mais informações sobre como o EventBridge Scheduler criptografa seus dados em repouso, consulte [the section called “Criptografia em repouso”](#).
- h. Em Permissões, escolha Usar função existente e selecione a função que você criou durante o procedimento de [configuração](#) na lista suspensa. Você também pode escolher Ir para o console do IAM para criar uma nova função.

Para que o EventBridge Scheduler crie um novo perfil de execução para você, escolha Criar um novo perfil para este agendamento. Depois, insira um nome para Nome do perfil. Se você escolher essa opção, o EventBridge Scheduler adicionará as permissões necessárias para seu alvo de exemplo ao perfil.

10. Escolha Next (Próximo).
11. Na página Revisar e criar uma programação, revise os detalhes da sua programação. Em cada seção, escolha Editar para voltar a essa etapa e editar seus detalhes.
12. Escolha Criar agendamento para concluir a criação da nova agenda. Você pode ver uma lista das suas programações novas e existentes na página Programações. Na coluna Status, verifique se sua nova programação está Ativada.

13. Para verificar se seu agendamento invoca o alvo do Amazon SQS, abra o console do Amazon SQS e faça o seguinte:
  - a. Escolha a fila de alvo na lista Filas.
  - b. Escolha Send and receive messages (Enviar e receber mensagens).
  - c. Na página Enviar e receber mensagens, em Receber mensagens, escolha Sondar mensagens para recuperar as mensagens de teste que sua agenda enviou para a fila de alvo.

## Crie um cronograma usando o AWS CLI

O exemplo a seguir mostra como usar o comando de AWS CLI [create-schedule](#) para criar um agendamento do EventBridge Scheduler com um alvo modelo do Amazon SQS. Substitua os valores do espaço reservado para os seguintes parâmetros por suas informações:

- `--name` – Insira um nome para o agendamento.
- `RoleArn` — Insira o ARN da função de execução que você deseja associar ao cronograma.
- `Arn` — Insira o ARN do alvo. Nesse caso, o alvo é uma fila do Amazon SQS.
- `Entrada` — Insira uma mensagem que o EventBridge Scheduler entrega à fila de alvo.

```
$ 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" }'
```

## Criar uma agenda usando os SDKs do EventBridge Scheduler

No exemplo a seguir, você usa os SDKs do EventBridge Scheduler para criar um agendamento do EventBridge Scheduler com um alvo modelo do Amazon SQS.

### 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 SDK do Java

```
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");
    }
}
```

## Próximas etapas

- Para obter mais informações sobre como gerenciar sua agenda usando o console, AWS CLI, ou o SDK do EventBridge Scheduler, consulte [Gerenciando um agendamento](#).
- Para obter mais informações sobre como configurar alvos modelados e aprender a usar o parâmetro de alvo universal, consulte [Gerenciando alvos](#).
- Para obter mais informações sobre os tipos de dados e as operações da API do EventBridge Scheduler, consulte a [Referência da API do EventBridge Scheduler](#).

# Tipos de agendamento no EventBridge Scheduler

O tópico a seguir descreve os diferentes tipos de agendamento que o Amazon EventBridge Scheduler suporta, bem como a forma como o EventBridge Scheduler lida com o horário de verão e o agendamento em diferentes fusos horários. Você pode escolher entre três tipos de agendamento ao configurar seu agendamento: agendamentos baseados em taxas, em cron e horários únicos.

Tanto os agendamentos baseados em taxas quanto os baseados em cron são recorrentes. Você configura cada tipo de agendamento recorrente usando uma expressão de agendamento para o tipo de agendamento que você deseja configurar e especificando um fuso horário no qual o EventBridge Scheduler avalia a expressão.

Um agendamento único é um agendamento que invoca um alvo somente uma vez. Você configura um agendamento único ao especificar a hora, a data e o fuso horário em que o EventBridge Scheduler avalia o agendamento.

## Note

Todos os tipos de agendamento no EventBridge Scheduler invocam seus alvos com precisão de 60 segundos. Isso significa que se você definir seu agendamento para ser executado em 1:00, ela invocará a API alvo entre 1:00:00 e 1:00:59.

Use as seções a seguir para aprender sobre como configurar expressões de agendamento para cada tipo de agendamento recorrente e como configurar um agendamento único no EventBridge Scheduler.

## Tópicos

- [Agendamentos baseados em taxas](#)
- [Agendamentos baseados em cron](#)
- [Programações únicas](#)
- [Fusos horários no EventBridge Scheduler](#)
- [Horário de verão no EventBridge Scheduler](#)

## Agendamentos baseados em taxas

Um cronograma baseado em taxas começa após a data de início que você especificou para seu cronograma e é executado com uma taxa regular que você define até a data de término do cronograma. Você pode configurar os casos de uso mais comuns de agendamento recorrente usando um cronograma baseado em taxas. Por exemplo, se você quiser um cronograma que invoque sua meta a cada 15 minutos, uma vez a cada duas horas ou uma vez a cada cinco dias, você pode usar um cronograma baseado em taxas para conseguir isso. Você configura um cronograma baseado em taxa usando uma expressão de taxa.

Com agendamentos baseados em taxas, você usa a propriedade [StartDate](#) para definir a primeira ocorrência do agendamento. Se você não fornecer uma `StartDate` para um cronograma baseado em taxas, seu cronograma começará a invocar a meta imediatamente.

As expressões de taxa têm dois campos obrigatórios separados por um espaço em branco, conforme mostrado a seguir.

### Sintaxe

```
rate(value unit)
```

valor

Um número positivo.

unidade

A unidade de tempo em que você deseja que sua agenda invoque sua meta.

Entradas válidas: `minutes` | `hours` | `days`

### Exemplos

O exemplo a seguir mostra como usar expressões de taxa com o comando AWS CLI `create-schedule` para configurar um agendamento baseado em taxas. Este exemplo cria um agendamento que é executado a cada cinco minutos e entrega uma mensagem para uma fila do Amazon SQS, usando o tipo de alvo de `SqsParameters` modelado.

Como esse exemplo não define um valor para o parâmetro `--start-date`, o cronograma começa a invocar seu alvo imediatamente após você criá-lo e ativá-lo.

```
$ 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" }'
```

## Agendamentos baseados em cron

Uma expressão cron cria um agendamento recorrente e refinado que é executado em um horário específico de sua escolha. O EventBridge Scheduler suporta a configuração de horários baseados em cron no Horário Coordenado Universal (UTC) ou no fuso horário que você especifica ao criar seu agendamento. Com agendamentos baseados em cron, você tem mais controle sobre quando e com que frequência sua agenda é executada. Use agendamentos baseados em cron quando precisar de um agendamento de recorrência personalizada que não seja suportada por uma das expressões de taxa do EventBridge Scheduler. Por exemplo, é possível criar um agendamentos baseados em cron que seja executada às 8h00 PST na primeira segunda-feira de cada mês. Você configura um cronograma baseado em cron usando uma expressão cron.

Uma expressão cron consiste em cinco campos obrigatórios separados por espaço em branco: minutos, horas, dia do mês, mês, dia da semana e um campo opcional, ano, conforme mostrado a seguir.

## Sintaxe

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

| Campo         | Valores         | Curingas      |
|---------------|-----------------|---------------|
| minutos       | 0-59            | , - * /       |
| Horas         | 0-23            | , - * /       |
| Dia do mês    | 1-31            | , - * ? / L W |
| Mês           | 1-12 ou JAN-DEZ | , - * /       |
| Dia da semana | 1-7 ou DOM-SÁB  | , - * ? L #   |
| Ano           | 1970-2199       | , - * /       |

## Curingas

- A , (vírgula) curinga inclui valores adicionais. No campo Mês, JAN, FEV, MAR inclui janeiro, fevereiro e março.
- O - (traço) curinga especifica intervalos. No campo Dia, 1-15 inclui os dias 1 a 15 do mês especificado.
- O \* (asterisco) curinga inclui todos os valores no campo. No campo Hours (Horas), \* inclui todas as horas. Você não pode usar \* nos campos Day-of-month (Dia do mês) e Day-of-week (Dia da semana). Se você usá-lo em um deles, utilize ? no outro.
- A / (barra) curinga especifica incrementos. No campo Minutos, você pode inserir 1/10 para especificar cada décimo minuto a partir do primeiro minuto da hora (por exemplo, o 11<sup>o</sup>, 21<sup>o</sup> e 31<sup>o</sup> minuto, etc.).
- O curinga ? (interrogação) especifica qualquer um. No campo Dia do mês, você pode inserir 7 e qualquer dia da semana for aceitável, pode inserir ? no campo Dia da semana.
- O curinga L nos campos Dia do mês ou Dia da semana especifica o último dia do mês ou da semana.
- O curinga W no campo Dia do mês especifica um dia da semana. No campo Dia do mês, **3W** especifica o dia mais próximo do terceiro dia da semana do mês.
- O curinga # no campo Dia da semana especifica uma determinada instância do dia da semana definido dentro de um mês. Por exemplo, 3#2 seria a segunda terça-feira do mês: o 3 refere-se a terça-feira, porque é o terceiro dia de cada semana, e o 2 refere-se ao segundo dia desse tipo dentro do mês.

### Note

Se você usar um caractere "#", poderá definir apenas uma expressão no campo do dia da semana. Por exemplo, o valor "3#1,6#3" não é válido porque é interpretado como duas expressões.

## Exemplos

O exemplo a seguir mostra como usar expressões de taxa com o comando AWS CLI `create-schedule` para configurar uma programação baseada em taxas. Este exemplo cria um agendamento que é executado às 10h15 UTC+0 na última sexta-feira de cada mês durante os anos

de 2022 a 2023 e entrega uma mensagem para uma fila do Amazon SQS, usando o tipo de alvo modelado `SqsParameters`.

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

## Programações únicas

Um agendamento único invoca um alvo somente uma vez na data e hora que você especificar. O EventBridge Scheduler suporta agendamentos baseados em cron no Horário Coordenado Universal (UTC) ou no fuso horário que você especifica ao criar seu agendamento.

### Note

Uma agenda única ainda conta na cota da sua conta depois de concluir a execução e a invocação de sua meta. Recomendamos excluir suas agendas únicas depois que elas concluírem a execução.

Você configura um agendamento único usando uma expressão `at`. Uma expressão `at` consiste na data e hora em que você deseja que o EventBridge Scheduler invoque sua agenda, conforme mostrado a seguir.

## Sintaxe

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

Quando você configura um agendamento único, o EventBridge Scheduler ignora a `StartDate` e a `EndDate` e você especifica para o agendamento.

## Exemplos

O exemplo a seguir mostra como usar expressões `at` com o comando AWS CLI `create-schedule` para configurar um agendamento de uma ocasião. Este exemplo cria um agendamento que é executado uma vez às 13:00 no fuso horário UTC-8 em 20 de novembro de 2022 e entrega uma mensagem para uma fila do Amazon SQS, usando o tipo de alvo modelado de `SqsParameters`.

```
$ 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" }'
```

## Fusos horários no EventBridge Scheduler

O EventBridge Scheduler suporta a configuração de agendamentos baseados únicos e baseados em cron em qualquer fuso horário que você especificar. O EventBridge Scheduler usa o [banco de dados de fuso horário](#) mantido pela Internet Assigned Numbers Authority (IANA).

Com o AWS CLI, você pode definir o fuso horário no qual deseja que o EventBridge Scheduler avalie seu agendamento usando o parâmetro `--schedule-expression-timezone`. Por exemplo, o comando a seguir cria um agendamento baseado em cron que invoca um alvo modelo do Amazon SQS de SendMessage em America/New\_York todos os dias às 8h30.

```
$ 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" }'
```

## Horário de verão no EventBridge Scheduler

O EventBridge Scheduler ajusta automaticamente seu agendamento para o horário de verão. Quando o tempo avança na primavera, se uma expressão cron cair em uma data e hora inexistentes, a invocação do cronograma será ignorada. Quando o tempo muda para trás no outono, sua agenda é executada apenas uma vez e não repete sua invocação. As invocações a seguir ocorrem normalmente na data e hora especificadas.

O EventBridge Scheduler ajusta seu agendamento de acordo com o fuso horário especificado ao criar o agendamento. Se você configurar um agendamento em America/New\_York, seu agendamento será ajustado quando a hora mudar nesse fuso horário, enquanto um agendamento em America/Los\_Angeles será ajustado três horas depois, quando a hora mudar na costa oeste.

Para horários baseados em taxas que usam days como unidade, por exemplo `rate(1 days)`, days representa uma duração de 24 horas no relógio. Isso significa que quando o horário de verão

faz com que um dia diminua para 23 horas ou se estenda para 25 horas, o EventBridge Scheduler ainda avalia a expressão de taxa 24 horas após a última invocação do agendamento.

### Note

Alguns fusos horários não observam o horário de verão, de acordo com as regras e regulamentos locais. Se você criar um agendamento em um fuso horário que não observe o horário de verão, o EventBridge Scheduler não ajustará seu agendamento. Os ajustes do horário de verão não se aplicam aos horários no horário coordenado universal (UTC).

## Exemplo

Considere um cenário em que você cria um cronograma usando a seguinte expressão cron em America/Los\_Angeles: `cron(30 2 * * ? *)`. Esse agendamento é executado todos os dias às 2h30 no fuso horário especificado.

- **Adiantamento na primavera** — Quando o horário é adiantado na primavera, das 1h59 às 3h, o EventBridge Scheduler ignora a invocação do agendamento naquele dia e retoma a execução do agendamento normalmente no dia seguinte.
- **Atraso no outono** — Quando o horário é atrasado no outono, das 2h59 às 2h, o EventBridge Scheduler executa o agendamento apenas uma vez às 2h30 antes da ocorrência do turno, mas não repete a invocação do agendamento novamente às 2h30 após a mudança de horário.



# Gerenciando um agendamento

Um agendamento é o principal recurso que você cria, configura e gerencia usando o Amazon EventBridge Scheduler.

Cada agendamento tem uma expressão de cronograma que determina quando e com que frequência o cronograma é executado. O EventBridge Scheduler suporta três tipos de agendamentos: taxa, cron e agendamentos únicos. Para obter mais informações sobre os diferentes tipos de agendamento, consulte [Tipos de agendamento](#).

Ao criar um agendamento, você configura uma meta para o agendamento a ser invocado. Um alvo é uma operação de API que o EventBridge Scheduler chama em seu nome sempre que seu agendamento é executado. O EventBridge Scheduler suporta dois tipos de alvos: alvos modelados chamam operações de API comuns em um grupo principal de serviços e o parâmetro de alvo universal (UTP) que você pode usar para chamar mais de 6.000 operações em mais de 270 serviços. Para obter mais informações sobre a configuração de alvos, consulte [Gerenciando alvos](#).

Você configura como seu agendamento lida com as falhas, quando o EventBridge Scheduler não consegue entregar um evento com êxito a um alvo, usando dois mecanismos principais: uma política de repetição e uma fila de mensagens não entregues (DLQ). Uma política de repetição determina o número de vezes que o EventBridge Scheduler deve repetir um evento com falha e por quanto tempo manter um evento não processado. Uma DLQ é uma fila padrão do Amazon SQS que o EventBridge Scheduler usa para entregar eventos com falha, após o esgotamento da política de repetição. Você pode usar um DLQ para solucionar problemas com seu agendamento ou sua meta posterior. Para ter mais informações sobre, consulte [the section called “Configurar uma fila de mensagens mortas”](#).

Nesta seção, você pode encontrar exemplos para gerenciar seus agendamentos do EventBridge Scheduler usando o console, o AWS CLI e os SDKs do EventBridge Scheduler.

## Tópicos

- [Alterando o estado do cronograma](#)
- [Configurando janelas de tempo flexíveis](#)
- [Configurar uma fila de mensagens mortas para um agendamento](#)
- [Excluir um agendamento](#)
- [Próximas etapas](#)

## Alterando o estado do cronograma

Um agendamento do EventBridge Scheduler tem dois estados: ativado e desativado. O exemplo a seguir é usa o `UpdateSchedule` para desativar um agendamento que é acionado a cada cinco minutos e invoca um alvo Lambda.

Ao usar o `UpdateSchedule`, você deve fornecer todos os parâmetros necessários. O EventBridge Scheduler substitui seu agendamento pelas informações que você fornece. Se você não especificar um parâmetro definido anteriormente, definirá `null` como padrão.

### 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"
}
```

O exemplo a seguir usa o SDK do Python e a operação de `UpdateSchedule` para desativar um cronograma direcionado ao Amazon SQS usando um alvo modelado.

### 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')
```

## Configurando janelas de tempo flexíveis

Quando você configura seu agendamento com uma janela de horário flexível, o EventBridge Scheduler invoca a meta dentro da janela de tempo que você definiu. Isso é útil em casos que não exigem invocação programada precisa de alvos. Definir uma janela de horário flexível melhora a confiabilidade de seu agendamento ao dispersar suas invocações de alvo.

Por exemplo, se você configurar uma janela de horário flexível de 15 minutos para um agendamento que é executado a cada hora, ela invoca o alvo dentro de 15 minutos após o horário agendado. O seguinte AWS CLI, e os exemplos do EventBridge Scheduler SDK, usam o `UpdateSchedule` para definir uma janela de tempo flexível de 15 minutos para um agendamento que é executada uma vez a cada hora.

### Note

Você deve especificar se deseja definir uma janela de horário flexível ou não. Se você não quiser definir essa opção, especifique `OFF`. Se você definir o valor como `FLEXIBLE`, deverá especificar uma janela máxima de tempo durante a qual seu agendamento será executada.

### 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)
```

## Configurar uma fila de mensagens mortas para um agendamento

O Amazon EventBridge Scheduler oferece suporte a filas de mensagens não entregues (DLQ) usando o Amazon Simple Queue Service. Quando um agendamento falha em invocar seu alvo, o EventBridge Scheduler entrega uma carga JSON contendo detalhes da invocação e qualquer resposta recebida do alvo para uma fila padrão do Amazon SQS que você especificar.

O tópico a seguir se refere a esse JSON como um evento de letra morta. Um evento sem saída permite que você solucione problemas com sua agenda ou metas. Se você configurar uma política de repetição para sua agenda, o EventBridge Scheduler entregará o evento de letra morta que ele tem, esgotando o número máximo de novas tentativas que você definiu.

Os tópicos a seguir descrevem como você pode configurar uma fila do Amazon SQS como DLQ para seu agendamento, configurar as permissões que o EventBridge Scheduler precisa para entregar mensagens ao Amazon SQS e receber eventos de carta morta do DLQ.

### Tópicos

- [Criar uma fila do Amazon SQS](#)
- [Configure as permissões da função de execução](#)
- [Especificar uma fila de mensagens mortas](#)
- [Recuperar o evento de mensagens mortas](#)

## Criar uma fila do Amazon SQS

Antes de configurar uma DLQ para seu agendamento, você deve criar uma fila padrão do Amazon SQS. Para obter instruções sobre criar uma fila utilizando o Amazon SQS, consulte o tópico sobre como [Criar uma fila do Amazon SQS](#), no Guia do desenvolvedor do Amazon Simple Queue Service.

### Note

O EventBridge Scheduler não suporta o uso de uma fila FIFO como DLQ da sua agenda.

Use o comando AWS CLI a seguir da CLI do para criar uma fila padrão.

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

Se o for bem-sucedido, você verá QueueURL no resultado.

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

Depois de criar a fila, anote o ARN da fila. Você precisará do ARN ao especificar um DLQ para seu agendamento do EventBridge Scheduler. Você pode encontrar o ARN da fila no console do Amazon SQS ou usando o comando [get-queue-attributes](#) AWS CLI.

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

Se for bem-sucedido, você verá o ARN da fila na saída.

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

Na próxima seção, você adicionará as permissões necessárias à sua função de execução do cronograma para permitir que o EventBridge Scheduler entregue eventos sem saída para o Amazon SQS.

## Configure as permissões da função de execução

Para permitir que o EventBridge Scheduler entregue eventos com data limite para o Amazon SQS, sua função de execução do cronograma precisa da seguinte política de permissão. Para obter mais informações sobre como anexar uma nova política de permissão à sua função de execução do cronograma, consulte [Configurando a função de execução](#).

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

### Note

Sua função de execução do cronograma pode já ter as permissões necessárias anexadas se você usar o EventBridge Scheduler para invocar um alvo de API do Amazon SQS.

Na próxima seção, você usará o console do EventBridge Scheduler e especificará uma DLQ para sua agenda.

## Especificar uma fila de mensagens mortas

Para especificar uma DLQ, use o console do EventBridge Scheduler ou o AWS CLI para atualizar uma agenda existente ou criar uma nova.

## Console

Para especificar uma DLQ usando o console

1. Faça login no AWS Management Console e, em seguida, escolha o link a seguir para abrir a seção EventBridge Scheduler do console EventBridge: <https://console.aws.amazon.com/scheduler/home>
2. No console do EventBridge Scheduler, crie um novo agendamento ou escolha um agendamento existente na sua lista de agendamentos para edição.
3. Na página Configurações, para a fila de mensagens não entregues (DLQ), faça o seguinte:
  - Escolha Seleccionar uma fila do Amazon SQS em minha conta de AWS como DLQ e, em seguida, escolha o ARN da fila para seu DLQ na lista suspensa.
  - Escolha Especificar uma fila do Amazon SQS em outras contas de AWS como DLQ e, em seguida, insira o ARN da fila para sua DLQ. Se você escolher uma fila em outra conta de AWS, o console do EventBridge Scheduler não poderá exibir os ARNs da fila em uma lista suspensa.
4. Revise suas seleções e escolha Criar agenda ou Salvar agenda para concluir a configuração de uma DLQ.
5. (Opcional) Para ver os detalhes do DLQ de um cronograma, escolha o nome do agendamento na lista e, em seguida, escolha a guia fila de mensagens não entregues na página de detalhes do cronograma.

## AWS CLI

Para atualizar um agendamento existente, utilizando o AWS CLI

- Use o comando [update-schedule](#) para atualizar sua agenda. Especifique a fila do Amazon SQS que você criou anteriormente como DLQ. Especifique o ARN da função do IAM ao qual você anexou as permissões necessárias do Amazon SQS como função de execução. Substitua todos os outros valores de espaço reservado por suas informações.

```
$ 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" }'
```

Para criar uma novo agendamento com um DLQ usando o AWS CLI

- Para criar um agendamento, use o comando [create-schedule](#). Substitua todos os valores de espaço reservado por suas informações.

```
$ 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"}
```

Na próxima seção, você usará o AWS CLI para receber um evento de mensagens não entregues do DLQ.

## Recuperar o evento de mensagens mortas

Use o comando [receive-message](#), conforme mostrado a seguir, para recuperar um evento de letra morta do DLQ. Você pode definir o número de mensagens a serem recuperadas usando o atributo `--max-number-of-messages`.

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

Se for bem-sucedido, você verá uma saída semelhante à seguinte.

```
{
  "Messages": [
    {
      "MessageId": "2aeg3510-fe3a-4f5a-ab6a-6906560eaf7e",
      "ReceiptHandle": "AQEBkNKTd0MrWgHKPoITRBwrPoK3eCSZICzWvqCY0BZ
+FfTcORFpopJbtCqj36VbBT1HreM8+qM/m5jcwqS1A1GmIJ0/hYmMgn/
+dwIty9izE7HnpvRhhEyHxbEzT5V05RbeasYaBdNyi9WLcnAHviDh6MebLXXNWoFyYnsxdwJuG0f/
w3htX6r3dpxXvvFNPGoQb8ihY37+u0gtsbuIwhLtUSmE8rbldEEwiUfi3IJ1zEZpUS77n/k1GWrMrnYg0Gx/
BuaLz0rFi2F738XI/
Hnh45uv3ca60YwS1ojPQ1LtX2URg1haV5884FY1aRvY8jR1pCZabTkYRTZKSXG5KNGYZnHpmsspii6JNkjitYVFKPo0H91w
      "MD5OfBody": "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"
  }
}

```

```
    }  
  ]  
}
```

Observe os atributos a seguir no evento de carta morta para ajudá-lo a identificar e solucionar possíveis motivos pelos quais a inovação do alvo falhou.

- **ERROR\_CODE** — Contém o código de erro que o EventBridge Scheduler recebe da API de serviço do alvo. No exemplo anterior, o código de erro retornado pelo Amazon SQS é `AWS.SimpleQueueService.NonExistentQueue`. Se o agendamento falhar ao invocar um alvo devido a um problema com o EventBridge Scheduler, você verá o seguinte código de erro em vez disso: `AWS.Scheduler.InternalServerError`.
- **ERROR\_MESSAGE** — Contém o código de erro que o EventBridge Scheduler recebe da API de serviço do alvo. No exemplo anterior, a mensagem de erro retornada pelo Amazon SQS é `The specified queue does not exist for this wsdl version`. Se o agendamento falhar devido a um problema com o EventBridge Scheduler, você verá a seguinte mensagem de erro em vez disso: `Unexpected error occurred while processing the request`.
- **TARGET\_ARN** — O ARN do alvo que sua agenda invoca, no seguinte formato de ARN de serviço: `arn:aws:scheduler:::aws-sdk:service:apiAction`.
- **EXHAUSTED\_RETRY\_CONDITION** — Indica por que o evento foi entregue ao DLQ. Esse atributo estará presente se o erro da API-alvo for um erro que pode ser repetido e não permanente. O atributo pode conter os valores `MaximumRetryAttempts` se o EventBridge Scheduler o tiver enviado para a DLQ depois de exceder o máximo de tentativas que você configurou para o agendamento ou `MaximumEventAgeInSeconds`, se o evento for maior que a idade máxima que você configurou no agendamento e ainda não está sendo entregue.

No exemplo anterior, podemos determinar, com base no código de erro e na mensagem de erro, que a fila de alvo que especificamos para o agendamento não existe.

## Excluir um agendamento.

Você pode excluir um agendamento configurando a exclusão automática ou excluindo manualmente um agendamento individual. Use os tópicos a seguir para saber como excluir um agendamento usando os dois métodos e por que você pode escolher um método em vez do outro.

### Tópicos

- [Exclusão após a conclusão do cronograma](#)

- [Exclusão manual](#)

## Exclusão após a conclusão do cronograma

Configure a exclusão automática após a conclusão do agendamento se quiser evitar ter que gerenciar individualmente seus recursos de agendamento no EventBridge Scheduler. Em aplicativos em que você cria milhares de agendamentos ao mesmo tempo e precisa de flexibilidade para aumentar o número de seus agendamentos sob demanda, a exclusão automática pode garantir que você não atinja a cota da sua conta para o [número de agendamentos](#) em uma região específica.

Quando você configura a exclusão automática de um agendamento, o EventBridge Scheduler exclui o agendamento após sua última invocação de alvo. Para agendamentos únicos, isso ocorre após o agendamento ter invocado seu alvo uma vez. Para agendas recorrentes que você configura com expressões `rate` ou `cron`, seu agendamento é excluído após a última invocação. A última invocação de um agendamento recorrente é a invocação que ocorre mais próxima da [EndDate](#) que você especificou. Se você configurar um agendamento com exclusão automática, mas não especificar um valor para `EndDate`, o EventBridge Scheduler não excluirá automaticamente o agendamento.

Você pode configurar a exclusão automática ao criar um agendamento pela primeira vez ou atualizar as preferências de um agendamento existente. As etapas a seguir descrevem como configurar exclusão automática para um agendamento existente.

### AWS Management Console

1. Abra o console do EventBridge Scheduler em <https://console.aws.amazon.com/scheduler/>.
2. Na lista de agendamentos, selecione o agendamento que você deseja editar e escolha Editar.
3. No painel de navegação à esquerda, selecione Settings (Configurações).
4. Na seção Ação após a conclusão do cronograma, selecione EXCLUIR na lista suspensa e salve suas alterações.

### AWS CLI

1. Abra uma nova janela de prompt.
2. Use o comando `update-schedule` AWS CLI para atualizar um agendamento existente, conforme mostrado a seguir. O comando define o `--action-after-completion` para DELETE. Este exemplo pressupõe que você definiu sua configuração de alvo localmente

em um arquivo JSON. Para atualizar um agendamento, você deve fornecer a meta, bem como quaisquer outros parâmetros de agendamento que você queira configurar para o seu cronograma existente.

Essa é um agendamento recorrente com uma taxa de uma invocação por hora. Portanto, você especifica uma data de término ao definir o parâmetro `--action-after-completion`.

```
$ 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" }' \
```

## Exclusão manual

Quando não precisar mais de um agendamento, você pode excluí-lo usando a operação [DeleteSchedule](#).

### 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")
```

## Próximas etapas

- Para obter mais informações sobre como configurar alvos modelados para as funções Lambda e Step e aprender a usar o parâmetro de alvo universal, consulte [Gerenciando alvos](#).
- Para obter mais informações sobre os tipos de dados e as operações da API do EventBridge Scheduler, consulte a [Referência da API do EventBridge Scheduler](#).

# Gerenciando um grupo de agendamento

Um grupo de agendamento é um recurso do Amazon EventBridge Scheduler que você usa para organizar suas agendas.

Sua Conta da AWS vem com um grupo de agendadores default. Você pode associar uma nova agenda ao grupo default ou aos grupos de agendamentos que você cria e gerencia. Você pode criar até [500 grupos de agendamento](#) no seu Conta da AWS. Com o EventBridge Scheduler, você organiza grupos de agendamentos, em vez de agendas individuais, aplicando [tags](#).

Uma tag é um rótulo composto por uma chave com distinção entre maiúsculas e minúsculas e um valor que você define. Você pode criar tags para categorizar agendamentos por finalidade, proprietário ou ambiente. Por exemplo, você pode identificar o ambiente ao qual suas agendamentos pertencem com a seguinte tag: `environment:production`.

## Important

Não adicione informações de identificação pessoal (PII) nem outras informações confidenciais ou sigilosas em tags. As tags são acessíveis a muitos serviços da AWS, incluindo faturamento. As tags não devem ser usadas para dados privados ou confidenciais.

Um grupo de agendamentos tem dois [estados](#) possíveis: ATIVO e EXCLUINDO.

Quando você cria um grupo pela primeira vez, é ACTIVE por padrão. Você pode adicionar agendas a um grupo ACTIVE. Quando você exclui um grupo, o estado muda para DELETING até que o EventBridge Scheduler conclua a exclusão das agendas associadas. Depois que o EventBridge Scheduler excluir as agendas do grupo, o grupo não estará mais disponível em sua conta.

Use os tópicos a seguir para criar um grupo de agendamentos e aplicar uma tag a ele. Você também associará uma agenda ao grupo e, finalmente, excluirá o grupo.

## Tópicos

- [Criando um grupo de agendamento](#)
- [Excluindo um grupo de agendamento](#)
- [Recursos relacionados](#)

# Criando um grupo de agendamento

Use grupos de agendamentos e marcações para organizar agendamentos que compartilhem um propósito comum ou pertençam ao mesmo ambiente. Nas etapas a seguir, você cria um novo grupo de agendamentos e o rotula usando uma tag. Em seguida, você associa uma nova agenda a esse grupo.

## Note

Depois de criar um grupo, você não pode remover uma agenda desse grupo nem associar a agenda a um grupo diferente. Você só pode associar uma agenda a um grupo ao criar a agenda pela primeira vez.

## Etapa 1: criar um novo grupo de agendamento

Os tópicos a seguir descrevem como criar um novo grupo de agendamento e rotulá-lo com a seguinte tag: `environment:development`.

### AWS Management Console

Para criar um novo grupo utilizando o AWS Management Console

1. Faça login no AWS Management Console e abra o console do Amazon EventBridge em <https://console.aws.amazon.com/events/>.
2. No painel de navegação esquerdo, escolha Grupos de agendamento.
3. Na página agendamentos, escolha Criar um agendamento.
4. Na seção Detalhes do grupo de agendamento, em Nome, insira um nome para o grupo. Por exemplo, **TestGroup**.
5. Na seção Tags, faça o seguinte:
  - a. Selecione Add new tag (Adicionar nova etiqueta).
  - b. Em Chave, insira o nome que você deseja atribuir a essa chave. Neste tutorial, para rotular o ambiente ao qual esse grupo de agendamentos pertence, insira **environment**.
  - c. Em Valor - opcional, insira o valor que você deseja atribuir a essa chave. Para este tutorial, insira o valor **development** da sua chave de ambiente.

**Note**

Você pode adicionar outras tags ao seu grupo depois de criá-lo.

6. Para terminar, escolha Criar grupo de agendamento. Seu novo grupo aparece na lista de grupos de agendamento.
7. (Opcional) Para editar um grupo ou gerenciar suas tags, marque a caixa de seleção do novo grupo e escolha Editar.

**Note**

Não é possível editar o grupo de agendamento default.

## AWS CLI

Para criar um novo grupo utilizando o AWS CLI

1. Abra uma nova janela do prompt de comando.
2. Do AWS Command Line Interface (AWS CLI), insira o seguinte comando de [create-schedule-group](#) para criar um novo grupo. Esse comando cria um grupo com uma tag: `environment:development`. Você pode usar essa tag ou um sistema de marcação similar para rotular seus grupos de agendamento de acordo com o ambiente ao qual eles pertencem.

Substitua o nome do cronograma e a chave e o valor da tag por suas informações.

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

Por padrão, seu novo grupo está no estado ACTIVE. Agora você pode associar novas agendas ao novo grupo que você criou.

## Etapa 2: associar uma agenda ao grupo

Use as etapas a seguir para associar uma nova agenda ao grupo que você criou na [etapa anterior](#).

## AWS Management Console

Para associar uma agenda a um grupo usando o AWS Management Console

1. Faça login no AWS Management Console e abra o console do Amazon EventBridge em <https://console.aws.amazon.com/events/>.
2. No painel de navegação à esquerda, escolha agendamentos no painel de navegação à esquerda.
3. Na tabela Agendas, escolha Criar agenda para criar uma nova agenda.
4. Na página Especificar detalhes da agenda, em Grupo da agenda, selecione o nome do seu novo grupo na lista suspensa. Por exemplo, consulte TestGroup.
5. Especifique um padrão de agendamento, meta e configurações e, em seguida, revise sua seleção na página Revisar e salvar o agendamento. Para obter mais informações sobre como configurar um novo agendamento, consulte [Conceitos básicos](#).
6. Para finalizar e salvar sua agenda, escolha Salvar agenda.

## AWS CLI

Para associar uma agenda a um grupo usando o AWS CLI

1. Abra uma nova janela do prompt de comando.
2. Do AWS Command Line Interface (AWS CLI), insira o seguinte comando de [create-schedule](#). Isso cria uma agenda e a associa ao grupo da [etapa anterior](#), denominada sqs-test-schedule. Esse cronograma usa o tipo de alvo modelo do [Amazon](#) SQS para invocar a operação de SendMessage. Substitua o nome da agenda, o alvo e o nome do grupo por suas informações.

```
$ 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" }'
```

Sua nova agenda agora está associada ao grupo de agendamentos do TestGroup.



## Excluindo um grupo de agendamento

A seguir, você pode aprender como excluir um grupo de agendamentos usando o AWS Management Console e o AWS Command Line Interface. Quando você exclui um grupo, ele permanece no estado DELETING até que o EventBridge Scheduler exclua todas as agendas do grupo. Depois que o EventBridge Scheduler excluir as agendas do grupo, o grupo não estará mais disponível em sua conta.

### Note

Depois de criar um grupo, você não pode remover uma agenda desse grupo nem associar a agenda a um grupo diferente. Você só pode associar uma agenda a um grupo ao criar a agenda pela primeira vez.

### AWS Management Console

Para excluir um grupo usando o AWS Management Console

1. Faça login no AWS Management Console e abra o console do Amazon EventBridge em <https://console.aws.amazon.com/events/>.
2. No painel de navegação à esquerda, escolha agendamentos no painel de navegação à esquerda.
3. Na página Grupos de agendamento, na lista de grupos existentes no atual Região da AWS, localize o grupo que você deseja excluir. Se não encontrar o grupo que estava buscando, escolha outro Região da AWS.

### Note

Você não pode excluir, ou editar, o grupo padrão.

4. Marque a caixa de seleção para o grupo que deseja excluir.
5. Escolha Delete (Excluir).
6. Na caixa de diálogo Excluir grupo de agendamento, insira o nome do grupo para confirmar sua escolha e escolha Excluir.

7. Na lista Grupos de agendamento, a coluna Status muda para indicar que seu grupo agora está excluindo. O grupo permanece nesse estado até que o EventBridge Scheduler exclua todas as agendas associadas ao grupo.
8. Para atualizar a lista e confirmar que o grupo foi excluído, escolha o ícone Atualizar.

## AWS CLI

Para excluir um grupo usando o AWS CLI

1. Abra uma nova janela do prompt de comando.
2. Do AWS Command Line Interface (AWS CLI), digite o comando de [delete-schedule-group](#) a seguir para excluir o grupo de agendamentos. Substitua o valor de `--name` por suas informações.

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

Se for bem-sucedida, essa operação de AWS CLI não retornará uma resposta.

3. Para verificar se o grupo está no estado DELETING, execute o comando a seguir [get-schedule-group](#).

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

Se for executado com êxito, você receberá um resultado semelhante a este.

```
{
  "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"
}
```

O EventBridge Scheduler exclui o grupo depois de excluir as agendas associadas ao grupo. Se você executar o `get-schedule-group` novamente, receberá a seguinte resposta de `ResourceNotFoundException`:

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

## Recursos relacionados

Para obter mais informações sobre grupos de agendamento, consulte os seguintes recursos:

- Operação [CreateScheduleGroup](#) na Referência da API do EventBridge Scheduler.
- Operação [DeleteScheduleGroup](#) na Referência da API do EventBridge Scheduler.

# Gerenciando alvos

Os tópicos seguintes descrevem como usar alvos padronizados e universais com o EventBridge Scheduler e fornece uma lista dos serviços do AWS suportados que você pode configurar usando o parâmetro de alvo universal do EventBridge Scheduler.

Os alvos modelados são um conjunto de operações de API comuns em um grupo de serviços principais do AWS, como Amazon SQS, Lambda e Step Functions. Por exemplo, você pode direcionar a operação de [Invocar](#) de API do Lambda fornecendo o ARN da função ou a operação de [SendMessage](#) do Amazon SQS usando o ARN do alvo.

O alvo universal é um conjunto personalizável de parâmetros que permite invocar um conjunto mais amplo de operações de API para muitos serviços do AWS. Por exemplo, você pode usar o parâmetro de alvo do EventBridge Scheduler (UTP) para criar uma nova fila do Amazon SQS usando a operação [CreateQueue](#).

Para configurar metas padronizadas ou universais, seu agendamento deve ter permissão para chamar a operação de API que você configura como sua meta. Para fazer isso, anexe as permissões necessárias ao perfil de execução de seu agendamento. Por exemplo, para direcionar a operação de [SendMessage](#) do Amazon SQS, a função de execução recebe permissão para realizar a ação de `sqs:SendMessage`. Na maioria dos casos, você pode adicionar as permissões necessárias usando as [políticas gerenciadas de AWS](#) suportadas pelo serviço de alvo. No entanto, você também pode criar suas próprias [políticas gerenciadas pelo cliente](#) ou adicionar [permissões embutidas](#) a uma política existente anexada à função de execução. Os tópicos a seguir demonstram exemplos de adição de permissões para tipos de alvo modelados e universais.

Para obter mais informações sobre como configurar uma função de execução para um agendamento, consulte [the section called “Configurar o perfil de execução”](#).

## Tópicos

- [Uso de alvos modelados](#)
- [Usando alvos universais](#)
- [Adicionando atributos de contexto](#)
- [Próximas etapas](#)

## Uso de alvos modelados

Os alvos modelados são um conjunto de operações de API comuns em um grupo de AWS serviços principais, como Amazon SQS, Lambda e Step Functions. Por exemplo, você pode direcionar a operação do Lambda de [Invoke](#) fornecendo o ARN da função ou a operação do Amazon SQS de [SendMessage](#) usando o ARN da fila. Para configurar um alvo modelado, você também deve conceder permissões à função de execução do cronograma para realizar a operação de API direcionada.

Para configurar um alvo modelado de forma programática usando o AWS CLI ou um dos SDKs do EventBridge Scheduler, você precisa especificar o ARN da função de execução, o ARN do recurso de destino, uma entrada opcional que você deseja que o EventBridge Scheduler entregue ao destino e, para alguns destinos modelados, um conjunto exclusivo de parâmetros com opções de configuração adicionais para esse alvo. Quando você especifica o ARN para um recurso de destino modelado, o EventBridge Scheduler assume automaticamente que você deseja chamar a operação de API compatível para esse serviço. Se você quiser que o EventBridge Scheduler direcione uma operação de API diferente para o serviço, você deve configurar o destino como um [destino universal](#).

A seguir está uma lista completa de todos os alvos modelados que o EventBridge Scheduler suporta e, se aplicável, o conjunto exclusivo de parâmetros associados de cada alvo. Escolha o link para cada conjunto de parâmetros para ver os campos obrigatórios e opcionais na Referência da API EventBridge Scheduler.

- CodeBuild – [StartBuild](#)
- CodePipeline – [StartPipelineExecution](#)
- Amazon ECS – [RunTask](#)
  - Parâmetros: [EcsParameters](#)
- EventBridge – [PutEvents](#)
  - Parâmetros: [EventBridgeParameters](#)
- Amazon Inspector – [StartAssessmentRun](#)
- Kinesis – [PutRecord](#)
  - Parâmetros: [KinesisParameters](#)
- Firehose — [PutRecord](#)
- Lambda: [Invoke](#)
- SageMaker – [StartPipelineExecution](#)

- Parâmetros: [SageMakerPipelineParameters](#)
- Amazon SNS – [Publish](#)
- Amazon SQS – [SendMessage](#)
  - Parâmetros: [SqsParameters](#)
- Step Functions – [StartExecution](#)

Use os exemplos a seguir para aprender a configurar diferentes alvos modelados e as permissões necessárias do IAM para cada alvo descrito.

## Amazon SQS `SendMessage`

Exemplo Política de permissão para função de execução

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

Exemplo 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"}
```

Exemplo 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 SDK do Java

```
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 Política de permissão para função de execução

```

{
  "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 SDK do Java

```
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())
            .build();
```

```

        .clientToken("<Token GUID>")
        .build();

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

```

## Step Functions **StartExecution**

Example Política de permissão para função de execução

```

{
  "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 SDK do Java

```
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");
}
}
```

## Usando alvos universais

Um alvo universal é um conjunto personalizável de parâmetros que permite invocar um conjunto mais amplo de operações de API para muitos serviços do AWS. Por exemplo, você pode usar um parâmetro de alvo universal (UTP) para criar uma nova fila do Amazon SQS usando a operação [CreateQueue](#).

Para configurar uma meta universal para sua agenda usando o AWS CLI ou um dos SDKs do EventBridge Scheduler, você precisa especificar as seguintes informações:

- **RoleArn** — O ARN da função de execução que você deseja usar para o alvo. A função de execução que você especificar deve ter as permissões para chamar a operação de API que você deseja que seu cronograma vise.
- **Arn** — O ARN completo do serviço, incluindo a operação de API que você deseja atingir, no seguinte formato: `arn:aws:scheduler::aws-sdk:service:apiAction`.

Por exemplo, para o Amazon SQS, o nome do serviço que você especifica é `arn:aws:scheduler::aws-sdk:sqs:sendMessage`.

- **Entrada** — Um JSON bem formado que você especifica com os parâmetros de solicitação que o EventBridge Scheduler envia para a API-alvo. Os parâmetros e a forma do JSON que você configura em `Input` são determinados pela API de serviço que sua agenda invoca. Para obter essas informações, consulte a referência da API do serviço da que você deseja segmentar.

## Ações não compatíveis

O EventBridge Scheduler não suporta ações de API somente para leitura, como operações comuns de GET, que começam com a seguinte lista de prefixos:

```
get
describe
list
```

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

Por exemplo, o ARN do serviço para a ação da API de [GetQueueUrl](#) seria o seguinte: `arn:aws:scheduler::aws-sdk:sqs:getQueueURL`. Como a ação da API começa com o prefixo `get`, o EventBridge Scheduler não é compatível com esse alvo. Da mesma forma, a ação [ListBrokers](#) do Amazon MQ não é suportada como alvo porque a operação começa com o prefixo `list`.

## Exemplos usando o alvo universal

Os parâmetros transmitidos no campo de agendamento de Input dependem dos parâmetros de solicitação que a API de serviço que você deseja invocar aceita. Por exemplo, para visar o [Invoke Lambda](#), você pode definir os parâmetros listados na [Referência API de AWS Lambda](#). Isso inclui a [carga](#) JSON opcional que você pode passar para uma função Lambda.

Para determinar os parâmetros que você pode definir para diferentes APIs, consulte a referência da API para esse serviço. Semelhante ao Invoke Lambda, algumas APIs aceitam parâmetros de URI, bem como uma carga útil do corpo da solicitação. Nesses casos, você especifica os parâmetros do caminho do URI, bem como a carga útil do JSON na Input de sua agenda.

Os exemplos a seguir mostram como usar a meta universal para invocar operações comuns de API com Lambda, Amazon SQS e Step Functions.

### Example Lambda

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

### Example Amazon SQS

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

flex_window = { "Mode": "OFF" }

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

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

### Example Step Functions

```
package com.example;

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

public class MySchedulerApp {
```

```
public static void main(String[] args) {

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

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

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

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

## Adicionando atributos de contexto

Use as seguintes palavras-chave na carga que você passa para o alvo para coletar metadados sobre o cronograma. O EventBridge Scheduler substitui cada palavra-chave por seu respectivo valor quando sua agenda invoca o alvo.

- **<aws.scheduler.schedule-arn>** — O ARN do agendamento.
- **<aws.scheduler.scheduled-time>** — O horário que você especificou para o agendamento invocar sua meta, por exemplo, 2022-03-22T18:59:43Z.
- **<aws.scheduler.execution-id>** — O ID exclusivo que o EventBridge Scheduler atribui para cada tentativa de invocação de um alvo, por exemplo, d32c5kddcf5bb8c3.

- **<aws.scheduler.attempt-number>** — Um contador que identifica o número da tentativa para a invocação atual, por exemplo, 1.

Este exemplo mostra a criação de um agendamento que é acionada a cada cinco minutos e invoca a operação de SendMessage do Amazon SQS como uma meta universal. O corpo da mensagem inclui o valor para 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)
```



## Próximas etapas

Para obter mais informações sobre os tipos de dados e as operações da API do EventBridge Scheduler, consulte a [Referência da API do EventBridge Scheduler](#).

# Segurança no Amazon EventBridge Scheduler

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- **Segurança da nuvem** — AWS é responsável por proteger a infraestrutura que executa AWS os serviços no Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de [AWS](#) de . Para saber mais sobre os programas de conformidade que se aplicam ao Amazon EventBridge Scheduler, consulte [AWS Serviços no escopo do programa de conformidade AWS](#) .
- **Segurança na nuvem** — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o EventBridge Scheduler. Os tópicos a seguir mostram como configurar o EventBridge Scheduler para atender aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos do EventBridge Scheduler.

## Tópicos

- [Gerenciando o acesso ao Amazon EventBridge Scheduler](#)
- [Proteção de dados no Amazon EventBridge Scheduler](#)
- [Validação de conformidade para o Amazon EventBridge Scheduler](#)
- [Resiliência no Amazon Scheduler EventBridge](#)
- [Segurança da infraestrutura no Amazon EventBridge Scheduler](#)

# Gerenciando o acesso ao Amazon EventBridge Scheduler

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar os recursos do EventBridge Scheduler. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

## Tópicos

- [Público](#)
- [Autenticando com identidades](#)
- [Gerenciando acesso usando políticas](#)
- [Como o EventBridge Scheduler funciona com o IAM](#)
- [Usar políticas baseadas em identidade](#)
- [Prevenção do problema do “confused deputy”](#)
- [Solução de problemas de identidade e acesso ao Amazon EventBridge Scheduler](#)

## Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz no EventBridge Scheduler.

**Usuário do serviço** — Se você usar o serviço EventBridge Scheduler para fazer seu trabalho, seu administrador fornecerá as credenciais e as permissões de que você precisa. À medida que você usa mais recursos EventBridge do Scheduler para fazer seu trabalho, talvez precise de permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu administrador. Se você não conseguir acessar um recurso no EventBridge Scheduler, consulte [Solução de problemas de identidade e acesso ao Amazon EventBridge Scheduler](#).

**Administrador de serviços** — Se você é responsável pelos recursos do EventBridge Scheduler em sua empresa, provavelmente tem acesso total ao EventBridge Scheduler. É seu trabalho determinar quais recursos e recursos do EventBridge Scheduler seus usuários do serviço devem acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender os Introdução ao

IAM. Para saber mais sobre como sua empresa pode usar o IAM com o EventBridge Scheduler, consulte [Como o EventBridge Scheduler funciona com o IAM](#).

Administrador do IAM — Se você for administrador do IAM, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso ao EventBridge Scheduler. Para ver exemplos de políticas baseadas em identidade EventBridge do Scheduler que você pode usar no IAM, consulte. [Usar políticas baseadas em identidade](#)

## Autenticando com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como o Usuário raiz da conta da AWS, como usuário do IAM ou assumindo uma função do IAM.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Usuários (IAM Identity Center), a autenticação de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou no portal de AWS acesso. Para obter mais informações sobre como fazer login em AWS, consulte [Como fazer login Conta da AWS](#) no Guia do Início de Sessão da AWS usuário.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [Assinatura de solicitações de AWS API](#) no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte [Autenticação multifator](#) no Guia AWS IAM Identity Center do usuário e [Utilizar a autenticação multifator \(MFA\) na AWS](#) no Guia do usuário do IAM.

## Conta da AWS usuário root

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é chamada de usuário Conta da AWS raiz e é acessada fazendo login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte [Tarefas que exigem credenciais de usuário raiz](#) no Guia do Usuário do IAM.

## Identidade federada

Como prática recomendada, exija que usuários humanos, incluindo usuários que precisam de acesso de administrador, usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório de usuários corporativo, de um provedor de identidade da web AWS Directory Service, do diretório do Identity Center ou de qualquer usuário que acesse usando credenciais fornecidas Serviços da AWS por meio de uma fonte de identidade. Quando as identidades federadas são acessadas Contas da AWS, elas assumem funções, e as funções fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o AWS IAM Identity Center. Você pode criar usuários e grupos no IAM Identity Center ou pode se conectar e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todos os seus Contas da AWS aplicativos. Para obter mais informações sobre o Centro de Identidade do IAM, consulte [“O que é o Centro de Identidade do IAM?”](#) no Guia do usuário AWS IAM Identity Center .

## Usuários e grupos do IAM

Um [usuário do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte [Altere as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo](#) no Guia do Usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários

de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de um perfil\)](#) no Guia do usuário do IAM.

## Perfis do IAM

Uma [função do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma função do IAM no AWS Management Console [trocando de funções](#). Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos para o uso de perfis, consulte [Utilizar perfis do IAM](#) no Guia do usuário do IAM.

Funções do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter mais informações sobre perfis para federação, consulte [Criar um perfil para um provedor de identidades de terceiros](#) no Guia do Usuário do IAM. Se você usar o Centro de identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de permissões](#) no Guia do usuário AWS IAM Identity Center .
- **Permissões temporárias para usuários do IAM** — um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- **Acesso entre contas** — é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para saber a diferença entre funções e políticas baseadas em recursos para acesso entre contas, consulte [Acesso a recursos entre contas no IAM no Guia do](#) usuário do IAM.

- **Acesso entre serviços** — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado a serviço.
- **Sessões de acesso direto (FAS)** — Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado o principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).
- **Função de serviço**: um perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.
- **Função vinculada ao serviço** — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para funções vinculadas ao serviço.
- **Aplicativos em execução no Amazon EC2** — Você pode usar uma função do IAM para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma instância do EC2 e fazendo AWS CLI solicitações de API. É preferível fazer isso e armazenar chaves de acesso na instância do EC2. Para atribuir uma AWS função a uma instância do EC2 e disponibilizá-la para todos os seus aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém o perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Utilizar um perfil do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Guia do usuário do IAM.

Para saber se deseja usar perfis do IAM, consulte [Quando criar um perfil do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

## Gerenciando acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do Usuário do IAM.

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e funções não têm permissões. Para conceder aos usuários permissões para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de função da AWS Management Console AWS CLI, da ou da AWS API.

### Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criando políticas do IAM](#) no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do Usuário do IAM.



## Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

## Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

O Amazon S3 e o Amazon VPC são exemplos de serviços que oferecem suporte a ACLs. AWS WAF Para saber mais sobre ACLs, consulte [Visão geral da lista de controle de acesso \(ACL\)](#) no Guia do Desenvolvedor do Amazon Simple Storage Service.

## Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do Usuário do IAM.

- Políticas de controle de serviço (SCPs) — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em AWS Organizations. AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS que sua empresa possui. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre as Organizações e SCPs, consulte [How SCPs work](#) (Como os SCPs funcionam) no Guia do usuário do AWS Organizations .
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do Usuário do IAM.

## Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determina se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

## Como o EventBridge Scheduler funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao EventBridge Scheduler, saiba quais recursos do IAM estão disponíveis para uso com o EventBridge Scheduler.

Recursos do IAM que você pode usar com o Amazon EventBridge Scheduler

| Atributo do IAM                                  | EventBridge Suporte ao agendador |
|--|----------------------------------|
| <a href="#">Políticas baseadas em identidade</a> | Sim                              |
| <a href="#">Políticas baseadas em recursos</a>   | Não                              |
| <a href="#">Ações das políticas</a>              | Sim                              |
| <a href="#">Atributos de políticas</a>           | Sim                              |

| Atributo do IAM   | EventBridge Suporte ao agendador |
|---|----------------------------------|
| <a href="#">Chaves de condição de política (específicas do serviço)</a> | Sim                              |
| <a href="#">ACLs</a>  | Não                              |
| <a href="#">ABAC (tags em políticas)</a>                                | Parcial                          |
| <a href="#">Credenciais temporárias</a>                                 | Sim                              |
| <a href="#">Permissões de entidade principal</a>                        | Sim                              |
| <a href="#">Perfis de serviço</a>                                       | Sim                              |
| <a href="#">Perfis vinculados ao serviço</a>                            | Não                              |

Para ter uma visão de alto nível de como o EventBridge Scheduler e outros AWS serviços funcionam com a maioria dos recursos do IAM, consulte [AWS os serviços que funcionam com o IAM no Guia do usuário do IAM](#).

## Políticas baseadas em identidade para o Scheduler EventBridge

|  |     |
|--|-----|
| Suporta políticas baseadas em identidade | Sim |
|--|-----|

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criando políticas do IAM](#) no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte [Referência de elementos da política JSON do IAM](#) no Guia do Usuário do IAM.

## Exemplos de políticas baseadas em identidade para o Scheduler EventBridge

Para ver exemplos de políticas baseadas em identidade do EventBridge Scheduler, consulte [Usar políticas baseadas em identidade](#)

## Políticas baseadas em recursos no Scheduler EventBridge

|  |     |
|--|-----|
| Oferece compatibilidade com políticas baseadas em recursos | Não |
|--|-----|

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em atributo. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso são diferentes Contas da AWS, um administrador do IAM na conta confiável também deve conceder permissão à entidade principal (usuário ou função) para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Para obter mais informações, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

## Ações de política para o EventBridge Scheduler

|  |     |
|--|-----|
| Oferece compatibilidade com ações de políticas | Sim |
|--|-----|

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de ações do EventBridge Agendador, consulte [Ações definidas pelo Amazon EventBridge Scheduler na Referência](#) de Autorização de Serviço.

As ações de política no EventBridge Scheduler usam o seguinte prefixo antes da ação:

```
scheduler
```

Para especificar várias ações em uma única instrução, separe-as com vírgulas.

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

Você também pode especificar várias ações usando caracteres-curinga (\*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

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

## Recursos de política para o EventBridge Scheduler

|   |     |
|---|-----|
| Oferece compatibilidade com recursos de políticas | Sim |
|---|-----|

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu [nome do recurso da Amazon \(ARN\)](#). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (\*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Para ver uma lista dos tipos de recursos do EventBridge Scheduler e seus ARNs, consulte [Recursos definidos pelo Amazon EventBridge Scheduler na Referência](#) de autorização de serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte [Ações definidas pelo Amazon EventBridge Scheduler](#).

Para ver exemplos de políticas baseadas em identidade do EventBridge Scheduler, consulte. [Usar políticas baseadas em identidade](#)

## Chaves de condição de política para o EventBridge Scheduler

|   |     |
|---|-----|
| Suporta chaves de condição de política específicas de serviço | Sim |
|---|-----|

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Condition` (ou bloco `Condition`) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem [agentes de condição](#), como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos `Condition` em uma instrução ou várias chaves em um único `Condition` elemento, a AWS os avaliará usando uma operação lógica AND. Se você especificar vários valores para uma única chave de condição, AWS avalia a condição usando uma OR operação

lógica. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um atributo somente se ele estiver marcado com seu nome de usuário do IAM. Para obter mais informações, consulte [Elementos da política do IAM: variáveis e tags](#) no Guia do usuário do IAM.

AWS suporta chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição AWS globais, consulte as [chaves de contexto de condição AWS global](#) no Guia do usuário do IAM.

Para ver uma lista das chaves de condição do EventBridge Scheduler, consulte [Chaves de condição do Amazon EventBridge Scheduler](#) na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte [Ações definidas pelo Amazon EventBridge Scheduler](#).

Para ver exemplos de políticas baseadas em identidade do EventBridge Scheduler, consulte. [Usar políticas baseadas em identidade](#)

## ACLs no Scheduler EventBridge

|                                  |     |
|----------------------------------|-----|
| Oferece compatibilidade com ACLs | Não |
|----------------------------------|-----|

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

## ABAC com Scheduler EventBridge

|  |         |
|--|---------|
| Oferece compatibilidade com ABAC (tags em políticas) | Parcial |
|--|---------|

O controle de acesso baseado em recurso (ABAC) é uma estratégia de autorização que define permissões com base em recursos. Em AWS, esses atributos são chamados de tags. Você pode anexar tags a entidades do IAM (usuários ou funções) e a vários AWS recursos. A marcação de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para

permitir operações quando a tag da entidade principal corresponder à tag do recurso que ela estiver tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações onde o gerenciamento de políticas se torna um problema.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou chaves de condição `aws:TagKeys`.

Se um serviço oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobre o ABAC, consulte [O que é ABAC?](#) no Guia do Usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte [Utilizar controle de acesso baseado em atributos \(ABAC\)](#) no Guia do usuário do IAM.

## Usando credenciais temporárias com EventBridge o Scheduler

|   |     |
|---|-----|
| Oferece compatibilidade com credenciais temporárias | Sim |
|---|-----|

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, incluindo quais Serviços da AWS funcionam com credenciais temporárias, consulte Serviços da AWS [“Trabalhe com o IAM”](#) no Guia do usuário do IAM.

Você está usando credenciais temporárias se fizer login AWS Management Console usando qualquer método, exceto um nome de usuário e senha. Por exemplo, quando você acessa AWS usando o link de login único (SSO) da sua empresa, esse processo cria automaticamente credenciais temporárias. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar funções, consulte [Alternar para um perfil \(console\)](#) no Guia do usuário do IAM.

Você pode criar manualmente credenciais temporárias usando a AWS API AWS CLI ou. Em seguida, você pode usar essas credenciais temporárias para acessar AWS. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para mais informações, consulte [Credenciais de segurança temporárias no IAM](#).



## Permissões principais entre serviços para EventBridge o Scheduler

|  |     |
|--|-----|
| Suporte para o recurso Encaminhamento de sessões de acesso (FAS) | Sim |
|--|-----|

Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).

## Funções de serviço do EventBridge Scheduler

|  |     |
|--|-----|
| Oferece compatibilidade com funções de serviço | Sim |
|--|-----|

O perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.

### Warning

Alterar as permissões de uma função de serviço pode interromper a funcionalidade do EventBridge Agendador. Edite as funções de serviço somente quando o EventBridge Agendador fornecer orientação para fazer isso.

## Funções vinculadas ao serviço para o Scheduler EventBridge

|  |     |
|--|-----|
| Oferece suporte a perfis vinculados ao serviço | Não |
|--|-----|

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode presumir a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar ou gerenciar perfis vinculados a serviços, consulte [Serviços da AWS que funcionam com o IAM](#). Encontre um serviço na tabela que inclua um Yes na coluna Função vinculada ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a serviço desse serviço.

## Usar políticas baseadas em identidade

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos do EventBridge Agendador. Eles também não podem realizar tarefas usando a AWS API AWS Management Console, AWS Command Line Interface (AWS CLI) ou. Para conceder aos usuários permissão para executar ações nos recursos de que precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documento de política JSON, consulte [Criação de políticas do IAM](#) no Guia do Usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos pelo EventBridge Scheduler, incluindo o formato dos ARNs para cada um dos tipos de recursos, consulte [Ações, recursos e chaves de condição do Amazon EventBridge Scheduler](#) na Referência de Autorização de Serviço.

### Tópicos

- [Melhores práticas de política](#)
- [EventBridge Permissões do agendador](#)
- [AWS políticas gerenciadas para o EventBridge Scheduler](#)
- [Políticas gerenciadas pelo cliente para o EventBridge Scheduler](#)
- [AWS atualizações de políticas gerenciadas](#)

## Melhores práticas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do EventBridge Scheduler em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do Usuário do IAM.
- Aplique permissões de privilégio mínimo — ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em atributos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do Usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso — você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode gravar uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como AWS CloudFormation. Para obter mais informações, consulte [Elementos da política JSON do IAM: Condição](#) no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais — o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter

mais informações, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do Usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas Recomendadas de Segurança no IAM](#) no Guia do Usuário do IAM.

## EventBridge Permissões do agendador

Para que um diretor do IAM (usuário, grupo ou função) crie EventBridge agendamentos no Scheduler e acesse os recursos do EventBridge Scheduler por meio do console ou da API, o diretor deve ter um conjunto de permissões adicionado à sua política de permissões. Você pode configurar essas permissões dependendo da função de trabalho da entidade principal. Por exemplo, um usuário ou função que usa apenas o console do EventBridge Scheduler para visualizar uma lista de agendamentos existentes não precisa ter as permissões necessárias para chamar a operação da `CreateSchedule` API. Recomendamos personalizar suas permissões baseadas em identidade para fornecer somente o acesso com privilégio mínimo.

A lista a seguir mostra os recursos do EventBridge Scheduler e suas ações suportadas correspondentes.

- Schedule (Programação)
  - `scheduler:ListSchedules`
  - `scheduler:GetSchedule`
  - `scheduler>CreateSchedule`
  - `scheduler:UpdateSchedule`
  - `scheduler>DeleteSchedule`
- Grupo de agendamento
  - `scheduler:ListScheduleGroups`
  - `scheduler:GetScheduleGroup`
  - `scheduler>CreateScheduleGroup`
  - `scheduler>DeleteScheduleGroup`
  - `scheduler:ListTagsForResource`
  - `scheduler:TagResource`
  - `scheduler:UntagResource`

Você pode usar as permissões do EventBridge Scheduler para criar suas próprias políticas gerenciadas pelo cliente para usar com o EventBridge Scheduler. Você também pode usar as políticas AWS gerenciadas descritas na seção a seguir para conceder as permissões necessárias para casos de uso comuns sem precisar gerenciar suas próprias políticas.

## AWS políticas gerenciadas para o EventBridge Scheduler

AWS aborda muitos casos de uso comuns fornecendo políticas autônomas do IAM que AWS criam e administram. Políticas gerenciadas, ou predefinidas, concedem as permissões necessárias para casos de uso comuns para que você não precise investigar quais permissões são necessárias. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) no Guia do usuário do IAM. As seguintes políticas AWS gerenciadas que você pode anexar aos usuários em sua conta são específicas do EventBridge Scheduler:

- [the section called “AmazonEventBridgeSchedulerFullAccess”](#)— Concede acesso total ao EventBridge Scheduler usando o console e a API.
- [the section called “AmazonEventBridgeSchedulerReadOnlyAccess”](#)— Concede acesso somente para leitura ao Scheduler. EventBridge

### AmazonEventBridgeSchedulerFullAccess

A política AmazonEventBridgeSchedulerFullAccess gerenciada concede permissões para usar todas as ações do EventBridge Agendador para agendas e grupos de agendas.

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

```
    }
  }
]
}
```

## AmazonEventBridgeSchedulerReadOnlyAccess

A política gerenciada AmazonEventBridgeSchedulerReadOnlyAccess concede permissões somente leitura para visualizar detalhes sobre seus agendamentos e grupos de agendamentos.

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

## Políticas gerenciadas pelo cliente para o EventBridge Scheduler

Use os exemplos a seguir para criar suas próprias políticas gerenciadas pelo cliente para o EventBridge Scheduler. [As políticas gerenciadas pelo cliente](#) permitem que você conceda permissões somente para as ações e recursos necessários para aplicativos e usuários em sua equipe, de acordo com a função de trabalho da entidade principal.

### Tópicos

- [Exemplo: CreateSchedule](#)
- [Exemplo: GetSchedule](#)
- [Exemplo: UpdateSchedule](#)
- [Exemplo: DeleteScheduleGroup](#)

## Exemplo: **CreateSchedule**

Ao criar um novo agendamento, você escolhe se deseja criptografar seus dados no EventBridge Scheduler usando uma chave gerenciada pelo cliente ou uma [Chave pertencente à AWS](#) chave gerenciada pelo cliente.

A política a seguir permite que uma entidade principal crie um agendamento e aplique criptografia usando uma Chave pertencente à AWS. Com um Chave pertencente à AWS, AWS gerencia recursos ativados AWS Key Management Service (AWS KMS) para você, para que você não precise de permissões adicionais para interagir AWS KMS.

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

Use a política a seguir para permitir que um diretor crie uma agenda e use uma chave gerenciada pelo AWS KMS cliente para criptografia. Para usar uma chave gerenciada pelo cliente, o diretor

deve ter permissão para acessar os AWS KMS recursos em sua conta. Essa política concede acesso a uma única chave KMS especificada para ser usada para criptografar dados no EventBridge Scheduler. Como alternativa, você pode usar um caractere curinga (\*) para conceder acesso a todas as chaves em uma conta ou a um subconjunto que corresponda a um determinado padrão de nome.

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



```

    "Resource": "arn:aws:iam::123456789012:role/*",
    "Condition": {
      "StringLike": {
        "iam:PassedToService": "scheduler.amazonaws.com"
      }
    }
  }
]
}

```

### Exemplo: **GetSchedule**

Use a política a seguir para permitir que uma entidade principal obtenha informações sobre um agendamento.

```

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

```

### Exemplo: **UpdateSchedule**

Use as políticas a seguir para permitir que uma entidade principal atualize um agendamento chamando a ação `scheduler:UpdateSchedule`. Da mesma forma `CreateSchedule`, a política depende se o cronograma usa uma chave gerenciada pelo cliente AWS KMS Chave pertencente à AWS ou uma chave gerenciada pelo cliente para criptografia. Para um agendamento configurado com um Chave pertencente à AWS, use a seguinte política:

```

{
  "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"
        }
      }
    }
  ]
}

```

Para um agendamento configurado com uma chave gerenciada pelo cliente, use a política a seguir. Essa política inclui permissões adicionais que permitem que um diretor acesse AWS KMS recursos em sua conta:

```

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

```

```

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

```

### Exemplo: **DeleteScheduleGroup**

Use a política a seguir para permitir que uma entidade principal exclua um grupo de agendamentos. Ao excluir um grupo, você também exclui os agendamentos associados a esse grupo. A entidade principal que exclui o grupo deve ter permissão para também excluir os agendamentos associados

a esse grupo. Essa política concede uma permissão da entidade principal para chamar a ação `scheduler:DeleteScheduleGroup` nos grupos de agendamentos especificados, bem como em todos os agendamentos do grupo:

### Note

EventBridge O Scheduler não suporta a especificação de permissões em nível de recurso para agendas individuais. Por exemplo, a declaração a seguir é inválida e não deve ser incluída em sua política.

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

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

## AWS atualizações de políticas gerenciadas

| Alteração  | Descrição   | Data                   |
|--|---|------------------------|
| <a href="#">the section called “AmazonEventBridgeSchedulerFullAccess”</a> :<br>Nova política gerenciada  | EventBridge O Scheduler adiciona suporte a uma nova política gerenciada que concede aos usuários acesso total a todos os recursos, incluindo agendas e grupos de agendamentos.              | 10 de novembro de 2022 |
| <a href="#">the section called “AmazonEventBridgeSchedulerReadOnlyAccess”</a> : Nova política gerenciada | EventBridge O Scheduler adiciona suporte a uma nova política gerenciada que concede aos usuários acesso somente de leitura a todos os recursos, incluindo agendas e grupos de agendamentos. | 10 de novembro de 2022 |
| EventBridge O agendador começou a rastrear as alterações   | EventBridge O Scheduler começou a rastrear as alterações em suas políticas AWS gerenciadas.   | 10 de novembro de 2022 |

## Prevenção do problema do “confused deputy”

O problema do “confused deputy” é um problema de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executar a ação. Em AWS, a falsificação de identidade entre serviços pode resultar no problema confuso do deputado. A personificação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Recomendamos usar as [aws:SourceArn](#) chaves de contexto de condição [aws:SourceAccount](#) global em sua função de execução do cronograma para limitar as permissões que o EventBridge Agendador concede a outro serviço para acessar o recurso. Use `aws:SourceArn` se quiser apenas um recurso associado a acessibilidade de serviço. Use `aws:SourceAccount` se quiser permitir que qualquer recurso nessa conta seja associado ao uso entre serviços.

A maneira mais eficaz de se proteger contra o problema do substituto confuso é usar a chave de contexto de condição global `aws:SourceArn` com o ARN completo do recurso. A seguinte condição tem como escopo um grupo de agendamento individual: `arn:aws:scheduler:*:123456789012:schedule-group/your-schedule-group`

Se você não souber o ARN completo do recurso ou especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com caracteres curinga (\*) para as partes desconhecidas do ARN. Por exemplo: `arn:aws:scheduler:*:123456789012:schedule-group/*`.

O valor de `aws:SourceArn` deve ser o ARN do grupo de EventBridge agendamentos do Scheduler para o qual você deseja definir o escopo dessa condição.

#### Important

Não defina o escopo da instrução `aws:SourceArn` para um agendamento específico ou um prefixo de nome de agendamento. O ARN que você especificar deve ser um grupo de agendamentos.

O exemplo a seguir mostra como é possível usar as chaves de contexto de condição globais `aws:SourceArn` e `aws:SourceAccount` na sua função de política de confiança de execução para evitar o problema de segurança delegada confusa.

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

## Solução de problemas de identidade e acesso ao Amazon EventBridge Scheduler

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o EventBridge Scheduler e o IAM.

### Tópicos

- [Não estou autorizado a realizar uma ação no EventBridge Scheduler](#)
- [Não estou autorizado a realizar iam: PassRole](#)
- [Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos do EventBridge Scheduler](#)

### Não estou autorizado a realizar uma ação no EventBridge Scheduler

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, é preciso atualizar suas políticas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM mateojackson tenta usar o console para exibir detalhes sobre um recurso do *my-example-widget* fictício, mas não tem as permissões fictícias do scheduler: *GetWidget*.

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

Nesse caso, a política de Mateo deve ser atualizada para permitir que ele tenha acesso ao recurso *my-example-widget* usando a ação scheduler: *GetWidget*.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

## Não estou autorizado a realizar iam: PassRole

Se você receber um erro informando que não está autorizado a realizar a `iam:PassRole` ação, suas políticas devem ser atualizadas para permitir que você passe uma função para o EventBridge Agendador.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando um usuário do IAM chamado `marymajor` tenta usar o console para realizar uma ação no EventBridge Scheduler. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

## Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos do EventBridge Scheduler

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o EventBridge Scheduler oferece suporte a esses recursos, consulte [Como o EventBridge Scheduler funciona com o IAM](#).



- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como [fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui](#) no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como [fornecer acesso Contas da AWS a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para saber a diferença entre usar funções e políticas baseadas em recursos para acesso entre contas, consulte Acesso a [recursos entre contas no IAM no Guia do](#) usuário do IAM.

## Proteção de dados no Amazon EventBridge Scheduler

O modelo de [responsabilidade AWS compartilhada O modelo](#) se aplica à proteção de dados no Amazon EventBridge Scheduler. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as [Perguntas Frequentes sobre Privacidade de Dados](#). Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog [AWS LGPD e Modelo de Responsabilidade Compartilhada](#) no AWS Blog de Segurança.

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos. AWS Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.

- Se você precisar de módulos criptográficos validados pelo FIPS 140-2 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para ter mais informações sobre endpoints do FIPS disponíveis, consulte [Federal Information Processing Standard \(FIPS\) 140-2](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o EventBridge Scheduler ou outros Serviços da AWS usando o console, a API ou os AWS SDKs. AWS CLI Quaisquer dados inseridos em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

## Tópicos

- [Criptografia em repouso](#)
- [Criptografia em trânsito](#)

## Criptografia em repouso

Esta seção descreve como o Amazon EventBridge Scheduler criptografa e descriptografa seus dados em repouso. Os dados em repouso são dados armazenados no EventBridge Scheduler e nos componentes subjacentes do serviço. EventBridge O Scheduler se integra com AWS Key Management Service (AWS KMS) para criptografar e descriptografar seus dados usando um. [AWS KMS key](#) EventBridge O Scheduler oferece suporte a dois tipos de chaves KMS: [Chaves pertencentes à AWS](#) e chaves [gerenciadas pelo cliente](#).

### Note

EventBridge O Scheduler só suporta o uso de chaves KMS de criptografia [simétrica](#).

Chaves pertencentes à AWS são chaves KMS que um AWS serviço possui e gerencia para uso em várias AWS contas. Embora os usos do Chaves pertencentes à AWS EventBridge Agendador não estejam armazenados em sua AWS conta, o EventBridge Agendador os usa para proteger seus dados e recursos. Por padrão, o EventBridge Scheduler criptografa e descriptografa todos os seus dados usando uma chave própria. AWS Você não precisa gerenciar sua Chave pertencente à AWS

ou a política de acesso dela. Você não incorre em nenhuma taxa quando o EventBridge Scheduler usa Chaves pertencentes à AWS para proteger seus dados, e o uso deles não conta como parte de suas AWS KMS cotas em sua conta.

As chaves gerenciadas pelo cliente são chaves KMS armazenadas em sua AWS conta que você cria, possui e gerencia. Se seu caso de uso específico exigir que você controle e audite as chaves de criptografia que protegem seus dados no EventBridge Scheduler, você pode usar uma chave gerenciada pelo cliente. Se você escolher uma chave gerenciada pelo cliente, será necessário gerenciar sua política de chave. Chaves gerenciadas pelo cliente geram uma taxa mensal e uma taxa para uso que excede o nível gratuito. Usar uma chave gerenciada pelo cliente também conta como parte da sua [cota do AWS KMS](#). Para obter mais informações sobre a definição de preço, consulte [Definição de preço do AWS Key Management Service](#).

## Tópicos

- [Artefatos de criptografia](#)
- [Gerenciando chaves KMS](#)
- [CloudTrail exemplo de evento](#)

## Artefatos de criptografia

A tabela a seguir descreve os diferentes tipos de dados que o EventBridge Scheduler criptografa em repouso e que tipo de chave KMS ele suporta para cada categoria.

| Tipo de dados           | Descrição   | Chave pertencente à AWS | chave gerenciada pelo cliente |
|-------------------------|---|-------------------------|-------------------------------|
| Carga útil (até 256 KB) | Os dados que você especifica no parâmetro <code>TargetInput</code> do agendamento ao configurar o agendamento para ser entregue ao destino. | Compatível              | Compatível                    |

| Tipo de dados                                      | Descrição   | Chave pertencente à AWS | chave gerenciada pelo cliente |
|--|---|-------------------------|-------------------------------|
| Identificador e estado                             | O nome exclusivo e o estado (ativar, desativar) da agenda.  | Compatível              | Sem compatibilidade           |
| Configuração de agendamento.                       | A expressão de agendamento, como a expressão cron ou rate para agendamentos recorrentes, e o carimbo de data/hora para invocações únicas, bem como a data de início, a data de término e o fuso horário do agendamento. | Compatível              | Sem compatibilidade           |
| Configurações de destino                           | O nome do recurso da Amazon (ARN) e outros detalhes de configuração relacionados ao destino.  | Compatível              | Sem compatibilidade           |
| Configuração de invocação e comportamento de falha | Configuração de janela de tempo flexível, a política de repetição do agendamento e os detalhes da fila de mensagens não entregues usados para entregas malsucedidas.  | Compatível              | Sem compatibilidade           |

EventBridge O Scheduler usa suas chaves gerenciadas pelo cliente somente ao criptografar e descriptografar a carga de destino, conforme descrito na tabela anterior. Se você optar por usar uma chave gerenciada pelo cliente, o EventBridge Scheduler criptografará e descriptografará a carga duas vezes: uma usando o padrão Chave pertencente à AWS e outra usando a chave gerenciada pelo cliente que você especificar. Para todos os outros tipos de dados, o EventBridge Scheduler usa apenas o padrão Chave pertencente à AWS para proteger seus dados em repouso.

Use a [the section called “Gerenciando chaves KMS”](#) seção a seguir para saber como você deve gerenciar seus recursos e políticas de chaves do IAM para usar uma chave gerenciada pelo cliente com o EventBridge Scheduler.

## Gerenciando chaves KMS

Opcionalmente, você pode fornecer uma chave gerenciada pelo cliente para criptografar e descriptografar a carga útil que sua agenda entrega ao destino. EventBridge O Scheduler criptografa e descriptografa sua carga útil de até 256 KB de dados. Utilizar uma chave gerenciada pelo cliente gera uma taxa mensal e uma taxa que excede o nível gratuito. Usar uma chave gerenciada pelo cliente conta como parte da sua [cota do AWS KMS](#). Para obter mais informações sobre a definição de preço, consulte [Definição de preço do AWS Key Management Service](#).

EventBridge O Scheduler usa permissões do IAM associadas ao principal que cria um cronograma para criptografar seus dados. Isso significa que você deve anexar as permissões AWS KMS relacionadas necessárias ao usuário ou função que chama a API EventBridge Scheduler. Além disso, o EventBridge Scheduler usa políticas baseadas em recursos para descriptografar seus dados. Isso significa que a função de execução associada à sua agenda também deve ter as permissões AWS KMS relacionadas necessárias para chamar a AWS KMS API ao descriptografar dados.

### Note

EventBridge O Scheduler não suporta o uso de [concessões](#) para permissões temporárias.

Use a seção a seguir para saber como você pode gerenciar sua [política de AWS KMS chaves](#) e as permissões necessárias do IAM para usar uma chave gerenciada pelo cliente no EventBridge Scheduler.

## Tópicos

- [Adicionar permissões do IAM](#)

- [Gerenciar a política de chave](#)

## Adicionar permissões do IAM

Para usar uma chave gerenciada pelo cliente, você deve adicionar as seguintes permissões à entidade principal do IAM baseado em identidade que cria um agendamento, bem como à função de execução que você associa ao agendamento:

### Permissões baseadas em identidade para chaves gerenciadas pelo cliente

Você deve adicionar as seguintes AWS KMS ações à política de permissão associada a qualquer principal (usuários, grupos ou funções) que chame a API EventBridge Scheduler ao criar um agendamento.

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

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

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

- **kms:DescribeKey**: necessário para validar se a chave fornecida é uma chave do KMS de criptografia [simétrica](#).
- **kms:GenerateDataKey**— Necessário para gerar a chave de dados que o EventBridge Scheduler usa para realizar a criptografia do lado do cliente.
- **kms:Decrypt**— É necessário descriptografar a chave de dados criptografada que o EventBridge Scheduler armazena junto com seus dados criptografados.

## Permissões da função de execução para chaves gerenciadas pelo cliente

Você deve adicionar a ação a seguir à política de permissões da função de execução do seu cronograma para fornecer acesso ao EventBridge Scheduler para chamar a AWS KMS API ao descriptografar seus dados.

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Sid" : "Allow EventBridge Scheduler to decrypt data using a customer managed
key",
      "Effect" : "Allow",
      "Action" : [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:your-region:123456789012:key/your-key-id"
    }
  ]
}
```

- **kms:Decrypt**— É necessário descriptografar a chave de dados criptografada que o EventBridge Scheduler armazena junto com seus dados criptografados.

Se você usar o console do EventBridge Scheduler para criar uma nova função de execução ao criar um novo EventBridge agendamento, o Scheduler anexará automaticamente a permissão necessária à sua função de execução. No entanto, se você escolher uma função de execução existente, deverá adicionar as permissões necessárias à função para poder usar suas chaves gerenciadas pelo cliente.

### Gerenciar a política de chave

Quando você cria uma chave gerenciada pelo cliente usando AWS KMS, por padrão, sua chave tem a seguinte política de chaves para fornecer acesso às funções de execução de seus cronogramas.

```
{
  "Id": "key-policy-1",
  "Version": "2012-10-17",
  "Statement": [
    {
```

```

    "Sid": "Provide required IAM Permissions",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::123456789012:root"
    },
    "Action": "kms:*",
    "Resource": "*"
  }
]
}

```

Opcionalmente, você pode limitar o escopo da sua política de chave para fornecer acesso somente à função de execução. Você pode fazer isso se quiser usar sua chave gerenciada pelo cliente somente com os recursos do EventBridge Scheduler. Use o exemplo de [política de chaves](#) a seguir para limitar quais recursos do EventBridge Scheduler podem usar sua chave.

```

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

```



## CloudTrail exemplo de evento

AWS CloudTrail captura todos os eventos de chamadas de API. Isso inclui chamadas de API sempre que o EventBridge Scheduler usa sua chave gerenciada pelo cliente para descriptografar seus dados. O exemplo a seguir mostra uma entrada de CloudTrail evento que demonstra o EventBridge Scheduler usando a `kms:Decrypt` ação usando uma chave gerenciada pelo cliente.

```
{
  "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"
  }
}
```

## Criptografia em trânsito

EventBridge O Scheduler criptografa seus dados em trânsito enquanto eles viajam pela rede. O Transport Layer Security (TLS) criptografa seus dados quando você chama qualquer operação da API do EventBridge Scheduler, bem como quando o EventBridge Scheduler chama qualquer API de destino ao invocar sua agenda. Por padrão, o EventBridge Scheduler usa o TLS 1.2 ao criptografar seus dados em trânsito. Você não precisa configurar a criptografia em trânsito e não pode escolher uma versão diferente do TLS ao usar o EventBridge Scheduler.

Usando a API EventBridge Scheduler — Quando você executa uma operação de API, como `CreateSchedule`, por exemplo, o EventBridge Scheduler criptografa toda a solicitação HTTP, incluindo o corpo e os cabeçalhos da solicitação. EventBridge O Scheduler também criptografa todo o objeto de resposta que você recebe de nossas APIs.

Usando APIs de destino — Quando o EventBridge Scheduler invoca sua agenda, ele chama a API de destino que você especificou ao criar a agenda. Ao entregar um evento a um destino, o EventBridge Scheduler criptografa toda a solicitação, incluindo o corpo da solicitação e todos os cabeçalhos, bem como a resposta que recebe do destino.

## Validação de conformidade para o Amazon EventBridge Scheduler

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte [Serviços da AWS Escopo por Programa de Conformidade](#) [Serviços da AWS](#) e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de [AWS conformidade Programas AWS](#) de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte [Baixar relatórios em AWS Artifact](#) .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. AWS fornece os seguintes recursos para ajudar na conformidade:

- [Guias de início rápido sobre segurança e conformidade](#) — Esses guias de implantação discutem considerações arquitetônicas e fornecem etapas para a implantação de ambientes básicos AWS focados em segurança e conformidade.
- [Arquitetura para segurança e conformidade com a HIPAA na Amazon Web Services](#) — Este whitepaper descreve como as empresas podem usar AWS para criar aplicativos qualificados para a HIPAA.

### Note

Nem todos Serviços da AWS são elegíveis para a HIPAA. Para mais informações, consulte a [Referência dos serviços qualificados pela HIPAA](#).

- AWS Recursos de <https://aws.amazon.com/compliance/resources/> de conformidade — Essa coleção de pastas de trabalho e guias pode ser aplicada ao seu setor e local.
- [AWS Guias de conformidade do cliente](#) — Entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as melhores práticas de proteção Serviços da AWS e mapeiam as diretrizes para controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).

- [Avaliação de recursos com regras](#) no Guia do AWS Config desenvolvedor — O AWS Config serviço avalia o quão bem suas configurações de recursos estão em conformidade com as práticas internas, as diretrizes e os regulamentos do setor.
- [AWS Security Hub](#)— Isso AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança interno AWS. O Security Hub usa controles de segurança para avaliar os recursos da AWS e verificar a conformidade com os padrões e as práticas recomendadas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a [Referência de controles do Security Hub](#).
- [Amazon GuardDuty](#) — Isso AWS service (Serviço da AWS) detecta possíveis ameaças às suas cargas de trabalho Contas da AWS, contêineres e dados monitorando seu ambiente em busca de atividades suspeitas e maliciosas. GuardDuty pode ajudá-lo a atender a vários requisitos de conformidade, como o PCI DSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.
- [AWS Audit Manager](#)— Isso AWS service (Serviço da AWS) ajuda você a auditar continuamente seu AWS uso para simplificar a forma como você gerencia o risco e a conformidade com as regulamentações e os padrões do setor.

## Resiliência no Amazon Scheduler EventBridge

A infraestrutura AWS global é construída em torno Regiões da AWS de zonas de disponibilidade. Regiões da AWS fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicativos e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de datacenters tradicionais.

Para obter mais informações sobre zonas de disponibilidade Regiões da AWS e zonas de disponibilidade, consulte [Infraestrutura AWS global](#).

Além da infraestrutura AWS global, o EventBridge Scheduler oferece vários recursos para ajudar a suportar suas necessidades de resiliência e backup de dados.

## Segurança da infraestrutura no Amazon EventBridge Scheduler

Como um serviço gerenciado, o Amazon EventBridge Scheduler é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a

infraestrutura, consulte [AWS Cloud Security](#). Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte [Proteção](#) de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar o EventBridge Scheduler pela rede. Os clientes devem oferecer suporte para:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

# Monitoramento e métricas do Amazon EventBridge Scheduler

O monitoramento é uma parte importante para manter a confiabilidade, a disponibilidade e a performance do Amazon EventBridge e das outras soluções do AWS. O AWS fornece as ferramentas de monitoramento a seguir para observar o Amazon EventBridge, informar quando algo está errado e realizar ações automaticamente quando apropriado:

- O Amazon CloudWatch monitora os recursos do AWS e os aplicativos que você executa no AWS em tempo real. Você pode coletar e rastrear métricas, criar painéis personalizados e definir alarmes que o notificam ou que realizam ações quando uma métrica especificada atinge um limite definido. Para obter mais informações, consulte o [Guia do usuário do Amazon CloudWatch](#).
- O AWS CloudTrail captura chamadas de API e eventos relacionados feitos por ou em nome de sua conta da AWS e entrega os arquivos de log a um bucket do Amazon S3 que você especificar. Você pode identificar quais usuários e contas chamaram a AWS, o endereço IP de origem do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte o [Guia do usuário da AWS CloudTrail](#).

## Tópicos

- [Monitorando o Amazon EventBridge Scheduler com a Amazon CloudWatch](#)
- [Registro de chamadas da API Amazon EventBridge Scheduler usando o AWS CloudTrail](#)

## Monitorando o Amazon EventBridge Scheduler com a Amazon CloudWatch

Você pode monitorar o Amazon EventBridge Scheduler usando CloudWatch, que coleta dados brutos e os processa em métricas legíveis, quase em tempo real. EventBridge O Scheduler emite um conjunto de métricas para todos os agendamentos e um conjunto adicional de métricas para agendamentos que têm uma fila de cartas mortas (DLQ) associada. Se você [configurar uma DLQ](#) para sua agenda, o EventBridge Scheduler publicará métricas adicionais quando sua agenda esgotar sua política de repetição.

Essas estatísticas são mantidas por 15 meses, para que você possa acessar informações históricas e obter uma perspectiva melhor sobre por que um agendamento está falhando e solucionar

problemas subjacentes. Você também pode definir alarmes que observam determinados limites e enviam notificações ou realizam ações quando esses limites são atingidos. Para obter mais informações, consulte o [Guia CloudWatch do usuário da Amazon](#).

## Tópicos

- [Termos](#)
- [Dimensões](#)
- [Acesso às métricas do](#)
- [Lista de métricas](#)
- [Agendador do EventBridge métricas de uso](#)

## Termos

### Namespace

Um namespace é um contêiner para as CloudWatch métricas de um AWS serviço. Para o EventBridge Scheduler, o namespace é `AWS/Scheduler`

### CloudWatch métricas

Uma CloudWatch métrica representa um conjunto ordenado por tempo de pontos de dados específicos de CloudWatch.

### Dimensão

Uma dimensão é um par de nome/valor que faz parte da identidade de uma métrica.

### Unidade

Uma estatística tem uma unidade de medida. Para o EventBridge Scheduler, as unidades incluem Count.

## Dimensões

Esta seção descreve o agrupamento de CloudWatch dimensões para métricas do EventBridge Scheduler em CloudWatch

| Dimensão      | Descrição   |
|---------------|---|
| ScheduleGroup | O grupo de agendamentos para o qual você deseja visualizar as métricas usando CloudWatch. Se você ainda não criou nenhum grupo, o EventBridge Scheduler associa suas agendas ao grupo. <code>default</code> |

## Acesso às métricas do

Esta seção descreve como acessar as métricas de desempenho de um cronograma específico do EventBridge Scheduler. CloudWatch

Para visualizar as métricas de desempenho para uma dimensão

1. Abra a [página Métricas](#) no CloudWatch console.
2. Use o seletor de AWS região para escolher a região para sua programação
3. Escolha o Agendador do namespace.
4. Na guia Todas as métricas, escolha uma dimensão, por exemplo, Agendar métricas do grupo. Para ver as métricas de todos os agendamentos que você criou na região selecionada, escolha Métricas da conta.
5. Escolha uma CloudWatch métrica para uma dimensão. Por exemplo, InvocationAttemptContar ou InvocationDroppedContar e, em seguida, escolha Pesquisa gráfica.
6. Escolha a guia Métricas representadas graficamente para visualizar as estatísticas de desempenho das métricas do EventBridge Scheduler.

## Lista de métricas

As tabelas a seguir listam as métricas de todas as EventBridge programações do Scheduler, bem como métricas adicionais das programações para as quais você configurou uma DLQ.



## Métricas para todos os agendamentos

| Namespace     | Métrica                   | Unidade  | Descrição  |
|---------------|---------------------------|----------|--|
| AWS/Scheduler | InvocationAttemptCount    | Contagem | Emitido para cada tentativa de invocação. Use essa métrica para verificar se o EventBridge Scheduler está tentando invocar suas agendas e para ver quando as invocações se aproximam das cotas de sua conta.   |
| AWS/Scheduler | TargetErrorCount          | Contagem | Emitido quando o destino retorna uma exceção após o EventBridge Scheduler chamar a API de destino. Use isso para verificar quando a entrega para um destino falha.   |
| AWS/Scheduler | TargetErrorThrottledCount | Contagem | Emitido quando a invocação do destino falha devido ao controle de utilização da API pelo destino. Use isso para diagnosticar falhas de entrega quando o motivo subjacente são as chamadas de limitação da API. |

| Namespace     | Métrica                 | Unidade  | Descrição   |
|---------------|-------------------------|----------|---|
|               |                         |          | de destino feitas pelo Scheduler. EventBridge   |
| AWS/Scheduler | InvocationThrottleCount | Contagem | Emitido quando o EventBridge Scheduler limita uma invocação de destino porque excede suas cotas de serviço definidas pelo Scheduler. EventBridge Use isso para determinar quando você excedeu as cotas do EventBridge Scheduler. Para obter mais informações sobre Service Quotas, consulte <a href="#">Cotas</a> . |
| AWS/Scheduler | InvocationDroppedCount  | Contagem | Emitido quando o EventBridge Scheduler para de tentar invocar o alvo após o esgotamento da política de repetição de um agendamento. Para obter mais informações sobre políticas de repetição, consulte a Referência <a href="#">RetryPolicy</a> da API EventBridge Scheduler.                                       |

## Métricas para agendamentos com uma DLQ

| Namespace     | Métrica                                    | Unidade  | Descrição  |
|---------------|--|----------|--|
| AWS/Scheduler | InvocationsSentToDeadLetterCount           | Contagem | Emitido para cada entrega bem-sucedida na DLQ de um agendamento. Use isso para determinar quando os eventos são enviados para uma DLQ e, em seguida, verifique o evento entregue à DLQ do agendamento para obter detalhes adicionais que o ajudem a determinar a causa da falha. |
| AWS/Scheduler | InvocationsFailedToBeSentToDeadLetterCount | Contagem | Emitido quando o EventBridge Scheduler não   |

| Namespace     | Métrica   | Unidade  | Descrição   |
|---------------|---|----------|---|
| AWS/Scheduler | InvocationsFailedToBeSentToDeadLetterCount_<error_code> | Contagem | <p>consegue entregar um evento ao DLQ. Use essas duas métricas para determinar o motivo pelo qual o EventBridge Scheduler não consegue enviar um evento para a DLQ e modifique sua configuração de DLQ para resolver o problema.</p> <p>Veja a seguir um exemplo da métrica <code>InvocationsFailedToBeSentToDeadLetterCount_&lt;error_code&gt;</code> quando a fila do</p> |

| Namespace | Métrica | Unidade | Descrição  |
|-----------|---------|---------|--|
|           |         |         | Amazon SQS que você especifica como DLQ não existe: <code>InvocationsFailedToBeSentToDeadLetterQueueCount_</code> <b>AWS.Simp</b><br><b>eQueueService.NonExistentQueue</b> |

| Namespace     | Métrica   | Unidade  | Descrição  |
|---------------|---|----------|--|
| AWS/Scheduler | InvocationsSentToDeadLetterCount_Truncated_MessageSize Exceeded | Contagem | Emitido quando a carga útil do evento enviada para a DLQ excede o tamanho máximo permitido pelo Amazon SQS, e o EventBridge Scheduler trunca a carga que você especifica no atributo de uma programação. Input |

## Agendador do EventBridge métricas de uso

CloudWatch coleta métricas que rastreiam o uso de alguns AWS recursos. Essas métricas correspondem às cotas AWS de serviço. O rastreamento dessas métricas pode ajudar a gerenciar as cotas proativamente. Use as métricas a seguir para determinar quando você excedeu suas cotas do EventBridge Scheduler. Para obter mais informações sobre Service Quotas, consulte [Cotas](#).

Essas métricas estão contidas no AWS/Usage namespace, em vez de AWS/Scheduler, e são coletadas a cada minuto.

Atualmente, o único nome de métrica CloudWatch publicado nesse namespace é `CallCount`. Essa métrica é publicada com as dimensões `Resource`, `Service` e `Type`. A dimensão `Resource` especifica o nome da operação da API que está sendo acompanhada.

Por exemplo, a `CallCount` métrica com as dimensões a seguir indica o número de vezes que a operação da Agendador do EventBridge `CreateSchedule` API foi chamada em sua conta:

- “Serviço”: “Agendador”
- “Tipo”: “API”
- “Recurso”: “CreateSchedule”

A métrica `CallCount` não tem uma unidade especificada. A estatística mais útil para a métrica é `SUM`, que representa a contagem total de operações para o período de 1 minuto.

## Metrics

| Métrica                | Descrição  |  |  |
|------------------------|--|--|--|
| <code>CallCount</code> | O número de operações especificadas executadas em sua conta. |  |  |

## Dimensões

| Dimensão             | Descrição   |  |  |
|----------------------|---|--|--|
| <code>Service</code> | O nome do AWS serviço que contém o recurso.<br><br>Para métricas de Agendador do EventBridge uso, o valor dessa dimensão é <code>Scheduler</code> . |  |  |
| <code>Class</code>   | A classe do recurso sob acompanhamento.   |  |  |

| Dimensão | Descrição  |  |  |
|----------|--|--|--|
| Type     | <p>Agendador do EventBridge As métricas de uso da API usam essa dimensão com um valor deNone.</p> <p>O tipo de recurso que está sendo acompanhado.</p> <p>No momento, quando a dimensão Service é Scheduler , o único valor válido para Type é API.</p>  |  |  |
| Resource | <p>O nome da operação da API. Entre os valores válidos estão os seguintes :</p> <ul style="list-style-type: none"> <li>• CreateSchedule</li> <li>• CreateScheduleGroup</li> <li>• DeleteSchedule</li> <li>• DeleteScheduleGroup</li> <li>• GetSchedule</li> <li>• GetScheduleGroup</li> <li>• ListScheduleGroups</li> <li>• ListSchedulesCallCount</li> <li>• ListTagsForResource</li> <li>• TagResource</li> <li>• UntagResource</li> <li>• UpdateSchedule</li> </ul> |  |  |

## Registro de chamadas da API Amazon EventBridge Scheduler usando o AWS CloudTrail

O Amazon EventBridge Scheduler é integrado ao AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, uma função ou um serviço do AWS no EventBridge



Scheduler. O CloudTrail captura as chamadas de API do EventBridge Scheduler como eventos. As chamadas capturadas incluem as chamadas do console do EventBridge Scheduler e as chamadas de código para as operações de API do EventBridge Scheduler. Se você criar uma trilha, poderá habilitar a entrega contínua de eventos do CloudTrail para um bucket do Amazon S3, incluindo eventos para o EventBridge Scheduler. Se você não configurar uma trilha, ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history (Histórico de eventos). Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação feita para o EventBridge Scheduler, o endereço IP no qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, além de detalhes adicionais.

Para saber mais sobre o CloudTrail, consulte o [Guia do usuário do AWS CloudTrail](#).

## Informações do EventBridge Scheduler no CloudTrail

O CloudTrail é habilitado em sua Conta da AWS quando ela é criada. Quando a atividade ocorrer no EventBridge Scheduler, ela será registrada em um evento do CloudTrail juntamente com outros eventos de serviços do AWS no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua Conta da AWS. Para obter mais informações, consulte [Viewing events with CloudTrail Event history](#) (Como visualizar eventos com o histórico de eventos do CloudTrail).

Para obter um registro contínuo de eventos em sua Conta da AWS, incluindo eventos do EventBridge Scheduler, crie uma trilha. Uma trilha permite que o CloudTrail entregue arquivos de log a um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as Regiões da AWS. A trilha registra em log eventos de todas as regiões na partição da AWS e entrega os arquivos de log para o bucket do Amazon S3 especificado por você. Além disso, é possível configurar outros serviços da AWS para analisar mais ainda mais e agir com base nos dados de eventos coletados nos logs do CloudTrail. Para obter mais informações, consulte:

- [Visão geral da criação de uma trilha](#)
- [Serviços e integrações compatíveis com o CloudTrail](#)
- [Configurar notificações do Amazon SNS para o CloudTrail](#)
- [Receber arquivos de log do CloudTrail de várias regiões](#) e [Receber arquivos de log do CloudTrail de várias contas](#)

Todas as ações do EventBridge Scheduler são registradas pelo CloudTrail e são documentadas na [Referência de API do Amazon EventBridge Scheduler](#). Por exemplo, as chamadas para as ações

`CreateSchedule`, `UpdateSchedule` e `DeleteSchedule` geram entradas nos arquivos de log do CloudTrail.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário raiz ou do AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro serviço da AWS.

Para obter mais informações, consulte [Elemento userIdentity do CloudTrail](#).

## Noções básicas sobre as entradas de arquivos de log do EventBridge Scheduler

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log a um bucket do Amazon S3 especificado. Os arquivos de log do CloudTrail contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros de solicitação e assim por diante. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada de chamadas de API pública. Dessa forma, eles não são exibidos em uma ordem específica.

## Cotas para o Amazon EventBridge Scheduler

Sua AWS conta tem cotas padrão, anteriormente chamadas de limites, para cada AWS serviço. A menos que especificado de outra forma, cada cota é específica da região. Você pode solicitar aumentos para algumas cotas e algumas não podem ser aumentadas.

Para ver as cotas do EventBridge Scheduler, abra o console [Service Quotas](#). No painel de navegação, escolha AWS serviços e, em seguida, selecione EventBridge Agendador.

Para solicitar um aumento da cota, consulte [Requesting a quota increase](#) no Guia do usuário do Service Quotas. Se a cota ainda não estiver disponível no Service Quotas, use o [formulário de aumento de limite](#).

### Note

As cotas de `CreateSchedule`, `UpdateSchedule`, `GetSchedule`, e `DeleteSchedule` transações por segundo (TPS) do EventBridge Scheduler são ajustáveis até milhares de TPS. A cota do controle de invocações é ajustável até a medida de dezenas de milhares de TPS.

Sua AWS conta tem as seguintes cotas relacionadas ao EventBridge Scheduler.

| Nome                                    | Padrão                      | Ajusté              | Descrição   |
|---|-----------------------------|---------------------|---|
| CreateSchedule taxa de solicitação      | Cada região compatível: 50  | <a href="#">Sim</a> | Máximo de CreateSchedule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo. |
| CreateScheduleGroup taxa de solicitação | Cada região com suporte: 10 | <a href="#">Sim</a> | Máximo de CreateScheduleGroup solicitações por segundo. Quando  |

| Nome                                    | Padrão                      | Ajuste              | Descrição  |
|---|-----------------------------|---------------------|--|
|   |                             |                     | você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.  |
| DeleteSchedule taxa de solicitação      | Cada região compatível: 50  | <a href="#">Sim</a> | Máximo de DeleteSchedule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.      |
| DeleteScheduleGroup taxa de solicitação | Cada região com suporte: 10 | <a href="#">Sim</a> | Máximo de DeleteScheduleGroup solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo. |
| GetSchedule taxa de solicitação         | Cada região compatível: 50  | <a href="#">Sim</a> | Máximo de GetSchedule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.         |

| Nome  | Padrão                       | Ajusté              | Descrição   |
|---|------------------------------|---------------------|---|
| GetScheduleGroup taxa de solicitação                        | Cada região com suporte: 10  | <a href="#">Sim</a> | Máximo de GetScheduleGroup solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.                 |
| As invocações controlam o limite de transações por segundo. | Cada região com suporte: 500 | <a href="#">Sim</a> | Uma invocação é uma carga útil do cronograma que está sendo entregue ao alvo definido. Após o limite ser atingido, as invocações são limitadas ; ou seja, elas ainda ocorrem, mas com atraso. |
| ListScheduleGroups taxa de solicitação                      | Cada região com suporte: 10  | <a href="#">Sim</a> | Máximo de ListScheduleGroups solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.               |

| Nome                                    | Padrão                            | Ajuste              | Descrição   |
|---|-----------------------------------|---------------------|---|
| ListSchedules taxa de solicitação       | Cada região compatível: 50        | <a href="#">Sim</a> | Máximo de ListSchedules solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.  |
| ListTagsForResource taxa de solicitação | Cada região com suporte: 10       | <a href="#">Sim</a> | Lista as tags associadas ao recurso Scheduler.  |
| Número de grupos de agendamento         | Cada região com suporte: 500      | <a href="#">Sim</a> | Número máximo de grupos de agendamento por região.  |
| Número de agendamentos                  | Cada região compatível: 1.000.000 | <a href="#">Sim</a> | O número máximo de agendamentos por região. Essa cota inclui agendamentos únicos que foram concluídas em execução. Recomendamos excluir suas programações únicas depois que elas concluírem a execução e invocarem um alvo. |
| TagResource taxa de solicitação         | Cada região compatível: 1         | <a href="#">Sim</a> | Atribui uma ou mais tags (pares chave-valor) ao recurso do Agendador especificado.  |
| UntagResource taxa de solicitação       | Cada região compatível: 1         | <a href="#">Sim</a> | Remove uma ou mais tags do recurso especificado do Scheduler.   |

| Nome                               | Padrão                     | Ajuste              | Descrição   |
|------------------------------------|----------------------------|---------------------|---|
| UpdateSchedule taxa de solicitação | Cada região compatível: 50 | <a href="#">Sim</a> | Máximo de UpdateSchedule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo. |

Para obter mais informações sobre cotas e endpoints de serviço para o EventBridge Scheduler, consulte os [endpoints e cotas do Amazon EventBridge Scheduler](#) no guia de referência geral. AWS

# Histórico do documento do Guia do usuário do EventBridge Scheduler

A tabela a seguir descreve as versões da documentação do EventBridge Scheduler.

| Alteração  | Descrição  | Data                  |
|--|--|-----------------------|
| <a href="#">Mudanças na função de execução e prevenção do problema de segurança delegada confusa</a> | <p>Esta atualização descreve as alterações na forma como a função de execução é aplicada a um recurso de grupo de agendamento quando você implementa a prevenção do problema de segurança delegada confusa na política de permissão da função.</p> <ul style="list-style-type: none"><li>• <a href="#">the section called “Prevenção o do problema do “confused deputy””</a></li></ul> | 7 de setembro de 2023 |
| <a href="#">Exclusão automática de agendamentos após a conclusão</a>                                 | <p>O EventBridge Scheduler suporta a exclusão automática. Quando você configura a exclusão automática, o EventBridge Scheduler exclui sua agenda após a última invocação planejada.</p> <ul style="list-style-type: none"><li>• <a href="#">the section called “Exclusão após a conclusão do cronograma”</a></li></ul>   | 2 de agosto de 2023   |
| <a href="#">Tópico atualizado sobre o uso de alvos universais</a>                                    | <p>Atualizou a lista de serviços compatíveis que o EventBridge Scheduler pode segmentar</p>  | 17 de março de 2023   |



e integrar. Essa atualização também inclui uma lista de operações de API de GET não suportadas e inclui melhorias nos exemplos de metas universais, bem como outras pequenas melhorias em todo o guia.

- [the section called “Usando alvos universais”](#)

### [Informações atualizadas sobre cronogramas baseados em taxas que não têm uma data de início](#)

Foram adicionadas informações sobre como o EventBridge Scheduler lida com agendamentos baseados em taxas se você não especificar um [StartDate](#) .

17 de março de 2023

- [the section called “Agendamentos baseados em taxas”](#)

### [Novo tópico sobre gerenciamento de grupos de agendadores](#)

Foi adicionado um novo capítulo sobre como criar grupos de agendadores com o EventBridge Scheduler . Use este capítulo para aprender como criar um grupo, adicionar agendas ao grupo, aplicar tags para gerenciar e monitorar mais facilmente seus recursos do EventBridge Scheduler e, finalmente, excluir um grupo.

17 de março de 2023

- [Gerenciando um grupo de agendamento](#)

## [Novos tópicos sobre horário de verão e fusos horários](#)

Foram adicionadas novas seções que descrevem como o EventBridge Scheduler lida com o horário de verão e como você pode criar horários em diferentes fusos horários.

17 de novembro de 2022

- [the section called “Horário de verão”](#)
- [the section called “Fusos horários”](#)

## [Novo tópico sobre métricas](#)

Foi adicionado um novo tópico que descreve as métricas que o EventBridge Scheduler publica no CloudWatch. Você pode usar essas métricas para monitorar falhas de invocação e entender como resolver problemas com suas agendas.

15 de novembro de 2022

- [the section called “Monitoramento com CloudWatch”](#)

## [Versão inicial](#)

Versão inicial do Guia do usuário do EventBridge Scheduler.

10 de novembro de 2022

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.