aws

Manual do usuário

EventBridge Agendador

EventBridge Agendador: Manual do usuário

Table of Contents

O que é o Amazon EventBridge Scheduler?

O Amazon EventBridge Scheduler é um programador sem servidor que permite criar, executar e gerenciar tarefas a partir de um serviço gerenciado central. Altamente escalável, o EventBridge Scheduler permite que você agende milhões de tarefas que podem invocar mais de 270 AWS serviços e mais de 6.000 operações. 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.

EventBridge O Scheduler entrega suas tarefas de forma confiável, com mecanismos integrados que ajustam seus cronogramas com base na disponibilidade de metas posteriores. Com o EventBridge Scheduler, você pode criar agendas 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

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

- Alvos modelados — O EventBridge Scheduler oferece suporte a alvos modelados para realizar operações comuns API usando Amazon, SQS Amazon, SNS Lambda e. EventBridge Com metas predefinidas, você pode configurar suas agendas rapidamente usando o console do EventBridge Scheduler, o EventBridge Scheduler ou o. SDK AWS CLI
- Metas universais — O EventBridge Scheduler fornece um parâmetro de destino universal (UTP) que você pode usar para criar acionadores personalizados que visam mais de 270 AWS serviços e mais de 6.000 API operações em um cronograma. ComUTP, você pode configurar seus gatilhos personalizados usando o console do EventBridge Scheduler, o EventBridge Scheduler ou o. SDK AWS CLI
- Janelas de tempo flexíveis — O EventBridge Scheduler suporta janelas de tempo 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 a entrega de at-least-once eventos aos alvos, o que significa que pelo menos uma entrega é bem-sucedida com uma resposta do alvo. EventBridge O Agendador permite que você defina o número de novas tentativas para sua agenda para uma tarefa com falha. EventBridge O agendador repete as tarefas que falharam com tentativas atrasadas para melhorar a confiabilidade do seu cronograma e garantir que as metas estejam disponíveis.

Acessando o EventBridge Scheduler

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

Configurando o Amazon EventBridge Scheduler

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

Tópicos

Inscreva-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 a https://portal.aws.amazon.com/billing/inscrição.
2. Siga as instruções online.

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

   Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWSé criado. O usuário raiz tem acesso a todos os Serviços da AWS e atributos 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
No IAM Identity Center (Recomendado)	Use credenciais de curto prazo para acessar a AWS. Isso está de acordo com as práticas recomendadas de segurança. Para obter informações sobre as melhores práticas, consulte as melhores práticas de segurança IAM no Guia IAM do usuário.	Seguindo as instruções em Conceitos básicos no Guia do usuário do AWS IAM Identity Center .	Configure o acesso programático configurando o AWS CLI para uso AWS IAM Identity Center no Guia do AWS Command Line Interface usuário.
Em IAM (Não recomendado)	Use credenciais de curto prazo para acessar a AWS.	Siga as instruções em Como criar seu primeiro usuário IAM administrador e grupo de usuários no Guia IAM do usuário.	Configure o acesso programático gerenciando as chaves de acesso para IAM usuários no Guia do IAM usuário.

Políticas gerenciadas pelo uso

Na etapa anterior, você configura um IAM usuário 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 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 o. API
- the section called “`AmazonEventBridgeSchedulerReadOnlyAccess`”— Concede acesso somente para leitura ao Scheduler. EventBridge

Você pode anexar essas políticas gerenciadas aos seus IAM diretores da mesma forma que anexou a `AdministratorAccess` política na etapa anterior. Para obter mais informações sobre como gerenciar o acesso ao EventBridge Scheduler usando IAM políticas baseadas em identidade, consulte. the section called “Usar políticas baseadas em identidade”

Configurar o perfil de execução

Uma função de execução é uma IAM função 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 destinos modelados”.

Para criar uma função de execução usando o AWS CLI

1. Copie a seguinte JSON política de assumir funções e salve-a localmente como`Scheduler-Execution-Role.json`. Essa política de confiança permite que o EventBridge Scheduler assuma 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 contra representante confuso”.

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 JSON permissão e salve-a localmente como um `.json` arquivo.

   Amazon SQS – SendMessage

   O seguinte permite que o EventBridge Scheduler execute a `sqs:SendMessage` ação em todas as SQS filas da Amazon 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 SNS tópicos da Amazon 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 a políticaARN. Você usa isso ARN na próxima etapa para anexar a política à nossa função de execução.

```
{
    "Policy": {
        "PolicyName": "PolicyName",
        "CreateDate": "2022-03-015T19:31:18.620Z",
        "AttachmentCount": 0,
        "IsAttachable": true,
        "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
        "DefaultVersionId": "v1",
        "Path": "/",
```

```
        "Arn": "arn:aws:iam::123456789012:policy/PolicyName",
        "UpdateDate": "2022-03-015T19:31:18.620Z"
    }
}
```

5. Para associar a política à sua função de execução, execute o comando a seguir. *your-policy-arn*ARNSubstitua pela 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 SQS fila padrão da Amazon com AWS CloudFormation.

Para criar uma nova SQS fila da Amazon

1. Copie o JSON AWS CloudFormation modelo 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 SQS fila da Amazon 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.

Começando com o EventBridge Scheduler

Este tópico descreve a criação de uma nova EventBridge agenda do 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 destino da AmazonSQS. Em seguida, você configurará o log, configurará novas tentativas e definirá um tempo máximo de retenção para tarefas com falha. Depois de criar o agendamento, você verificará se seu agendamento invoca com sucesso o destino e envia uma mensagem para a fila de destino.

> **Note**
>
> Para seguir este guia, recomendamos que você configure IAM usuários com as permissões mínimas exigidas descritas emthe 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 a 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

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

- Conclua as tarefas descritas em Configuração

Crie uma agenda usando o console do EventBridge Scheduler

Para criar uma nova programação usando o console

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

   > **Note**
   >
   > Você pode mudar o seu Região da AWS usando o seletor AWS Management Console de região.

2. Na página Programações, clique em Criar 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. Para adicionar tags a grupos de programação, você usa os grupos de programação.

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 destino 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 rate.

  Escolha a agenda baseada em Cron para configurar uma agenda usando uma cron expressão. Para usar uma expressão de taxa, escolha Programação baseada 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 rate e cron, 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 o destino uma vez a cada hora, a programação será executada em até 15 minutos após o início de cada hora.

5. 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. Um agendamento recorrente sem data de início começará assim que for criada e disponibilizada. Um agendamento recorrente sem data de término continuará a invocar sua meta indefinidamente.
6. Escolha Próximo.
7. Na página Selecionar destino, faça o seguinte:

   a. Selecione alvos modelados e escolha um alvo. API Neste exemplo, escolheremos o alvo SQS `SendMessage` modelado da Amazon.

   b. Na SendMessageseção, para SQSfila, escolha uma SQS fila existente da Amazon, ARN como na `arn:aws:sqs:us-west-2:123456789012:TestQueue` lista suspensa. Para criar uma nova fila, escolha Criar nova SQS fila para navegar até o console da AmazonSQS. Depois de terminar de criar uma fila, retorne ao console do EventBridge Scheduler e atualize o menu suspenso. Sua nova fila ARN aparece e pode ser selecionada.

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 destino: **`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 atributo usando o botão. Por padrão, o EventBridge Agendador ativa sua agenda.

b. Na seção Ação após a conclusão do cronograma, configure a ação que o EventBridge Agendador executa após a conclusão do cronograma:

- Escolha DELETEse você deseja que a agenda seja excluída automaticamente. Para agendamentos únicos, isso ocorre depois que o agendamento invocar o destino uma vez. Para agendamentos recorrentes, isso ocorre após a última invocação planejada do agendamento. Para obter mais informações sobre a exclusão automática, consulte the section called “Exclusão após a conclusão do agendamento”.
- Escolha NONEou não escolha um valor se você não quiser que o EventBridge Agendador execute nenhuma ação após a conclusão do cronograma.

c. Na seção Política de repetição e fila de mensagens mortas (DLQ), em Política de repetição, ative Tentar novamente para configurar uma política de repetição para sua agenda. Com políticas de repetição, se um agendamento falhar em invocar seu destino, o EventBridge Scheduler executará novamente o agendamento. Se configurado, você deve definir o tempo máximo de retenção e as novas tentativas da programação.

d. Em Idade máxima do evento - opcional, insira o (s) máximo (s) de hora (s) e minuto (s) em que o EventBridge Agendador deve manter um evento não processado.

> **Note**
>
> O valor máximo é 24 horas.

e. Em Máximo de tentativas, insira o número máximo de vezes que o EventBridge Scheduler repete o agendamento se o alvo retornar um erro.

> **Note**
>
> O valor máximo é 185 tentativas.

f. Para Dead-letter queue (DLQ), escolha entre as seguintes opções:

- Nenhum — Escolha essa opção se você não quiser configurar umDLQ.
- Selecione uma SQS fila da Amazon em minha AWS conta como DLQ — Escolha essa opção, selecione uma fila ARN na lista suspensa e configure DLQ a Conta da AWS mesma em que você está criando a programação.
- Especifique uma SQS fila da Amazon em outra AWS conta como DLQ — Escolha essa opção e, em seguida, insira a ARN fila configurada como aDLQ, se a fila estiver em outra. Conta da AWS Você deve inserir o exato ARN da fila para usar essa opção.

g. Na seção Criptografia, escolha Personalizar configurações de criptografia (avançadas) para usar uma KMS chave gerenciada pelo cliente para criptografar sua entrada de destino. Se você escolher essa opção, insira uma KMS chave existente ARN ou escolha Criar uma AWS KMS chave para navegar até o AWS KMS console. 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 IAM console para criar uma nova função.

   Se você quiser que o EventBridge Scheduler crie uma nova função de execução para você, escolha Criar nova função para esta agenda. Depois, insira um nome em Nome do perfil. Se você escolher essa opção, o EventBridge Scheduler adicionará à função as permissões necessárias para seu alvo modelado.

10. Escolha Próximo.
11. Na página Revisar e criar programação, revise os detalhes da 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 a lista com as programações novas e existentes na página Programações. Na coluna Status, verifique se a nova programação está Ativada.
13. Para verificar se sua agenda invoca o SQS alvo da Amazon, abra o SQS console da Amazon e faça o seguinte:

    a. Escolha a fila de destino 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 seu agendamento enviou para a fila de destino.

Crie um cronograma usando o AWS CLI

O exemplo a seguir mostra como usar o AWS CLI comando `create-schedule`para criar um cronograma do EventBridge Scheduler com um modelo de destino da AmazonSQS. Substitua os valores do espaço reservado para os seguintes parâmetros por suas informações:

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

```
$ 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"}'
```

Crie uma agenda usando o EventBridge Agendador SDKs

No exemplo a seguir, você usa o EventBridge Scheduler SDKs para criar um cronograma do EventBridge Scheduler com um modelo de destino da AmazonSQS.

Example Python SDK

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

flex_window = { "Mode": "OFF" }

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

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

Example Java SDK

```
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 ou o EventBridge AgendadorSDK, consulteGerenciando um agendamento. AWS CLI
- Para obter mais informações sobre como configurar destinos modelados e aprender a usar o parâmetro de destino universal, consulte Gerenciando destinos.
- Para obter mais informações sobre os tipos de dados e API operações do EventBridge Scheduler, consulte a Referência do EventBridge Scheduler. API

Tipos de EventBridge agendamento no Scheduler

O tópico a seguir descreve os diferentes tipos de EventBridge agendamento que o Amazon Scheduler suporta, bem como a forma como o EventBridge Scheduler lida com o horário de verão e a programação 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 Agendador avalia a expressão.

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

> **Note**
>
> Todos os tipos de EventBridge agendamento no Scheduler invocam seus alvos com precisão de 60 segundos. Isso significa que, se você definir sua programação para ser executada`1:00`, ela invocará a meta API entre `1:00:00` e`1:00:59`, supondo que uma janela de tempo flexível não esteja definida.

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 Scheduler. EventBridge

Tópicos

Agendamentos baseados em taxas

Um agendamento baseado em taxas começa após a data de início que você especificou para seu agendamento e é executado com uma taxa regular que você define até a data de término do agendamento. Você pode configurar os casos de uso mais comuns de agendamento recorrente usando um agendamento baseado em taxas. Por exemplo, se você quiser um agendamento 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 agendamento baseado em taxas para conseguir isso. Você configura um agendamento baseado em taxa usando uma expressão rate.

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 agendamento baseado em taxas, seu agendamento 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 seu agendamento invoque sua meta.

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

Exemplos

O exemplo a seguir mostra como usar expressões de taxa com o AWS CLI `create-schedule` comando para configurar uma programação baseada em taxas. Este exemplo cria uma programação que é executada a cada cinco minutos e entrega uma mensagem para uma SQS fila da Amazon, usando o tipo de `SqsParameters` destino modelado.

Como esse exemplo não define um valor para o parâmetro `--start-date`, o agendamento começa a invocar seu destino 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 uma programação recorrente refinada que é executada em um horário específico de sua escolha. EventBridge O Scheduler suporta a configuração de agendamentos baseados em cron no Tempo Coordenado Universal (UTC) ou no fuso horário que você especifica ao criar sua agenda. Com agendamentos baseados em cron, você tem mais controle sobre quando e com que frequência seu agendamento é executado. Use agendas baseadas em cron quando precisar de uma programação 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 PSTna primeira segunda-feira de cada mês. Você configura um agendamento baseado em cron usando uma expressão cron.

Uma expressão cron consiste em cinco campos obrigatórios separados por espaço em branco: minutos day-of-month, horas day-of-week, mês 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	, - * /
D ay-of-month	1-31	, - * ? / L W
Mês	1-12 ou JAN - DEC	, - * /
D ay-of-week	1-7 ou SUN - SAT	, - * ? L #
Ano	1970-2199	, - * /

Curingas

- A , (vírgula) curinga inclui valores adicionais. No campo Mês,JAN,FEB, 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 ay-of-week campos D ay-of-month e D. 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 11º, 21º e 31º minuto, etc.).
- O curinga ? (interrogação) especifica qualquer um. No ay-of-month campo D, você poderia inserir 7 e, se algum dia da semana fosse aceitável, você poderia inserir? no ay-of-week campo D.
- O curinga L nos ay-of-week campos D ay-of-month ou D especifica o último dia do mês ou da semana.
- O **`W`** caractere curinga no ay-of-month campo D especifica um dia da semana. No ay-of-month campo D, **`3W`** especifica o dia da semana mais próximo do terceiro dia do mês.
- O caractere curinga # no ay-of-week campo D especifica uma determinada instância do dia da semana especificado em 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 somente uma expressão no day-of-week campo. 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 cron com o AWS CLI `create-schedule` comando para configurar uma programação baseada em cron. Este exemplo cria uma programação que funciona às 10h15 UTC +0 na última sexta-feira de cada mês durante os anos de 2022 a 2023 e

entrega uma mensagem para uma SQS fila da Amazon, usando o tipo de destino `SqsParameters` modelado.

```
$ 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 invocará um alvo somente uma vez na data e hora que você especificar usando uma data válida e um carimbo de data/hora. EventBridge O Scheduler oferece suporte ao agendamento no Horário Coordenado Universal (UTC) ou no fuso horário que você especifica ao criar sua agenda.

> **Note**
>
> Um agendamento único ainda conta na cota da sua conta depois de concluir a execução e a invocação de sua meta. Recomendamos excluir seus agendamentos únicos depois que eles concluírem a execução.

Você configura um agendamento único usando uma expressão at. Uma expressão at consiste na data e na 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 EventBridge agendamento único, o Agendador ignora o `StartDate` e `EndDate` você especifica para o agendamento.

Exemplos

O exemplo a seguir mostra como usar expressões at com o AWS CLI `create-schedule` comando para configurar um agendamento único. Este exemplo cria uma programação que é executada uma

vez, das 13h às UTC 8h, em 20 de novembro de 2022, e entrega uma mensagem para uma SQS fila da Amazon, usando o tipo de destino `SqsParameters` modelado.

```
$ 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

EventBridge O Scheduler suporta a configuração de agendamentos únicos e baseados em cron em qualquer fuso horário que você especificar. EventBridge O Scheduler usa o banco de dados de fuso horário mantido pela Autoridade de Números Atribuídos da Internet (IANA).

Com o AWS CLI, você pode definir o fuso horário no qual deseja que o EventBridge Scheduler avalie sua agenda usando o `--schedule-expression-timezone` parâmetro. Por exemplo, o comando a seguir cria uma programação baseada em cron que invoca um SQS `SendMessage` alvo modelo da Amazon na América/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

EventBridge O agendador ajusta automaticamente sua programação 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 agendamento será ignorada. Quando o tempo muda para trás no outono, seu agendamento é executado apenas uma vez e não repete sua invocação. As invocações a seguir ocorrem normalmente na data e hora especificadas.

EventBridge O agendador ajusta sua agenda de acordo com o fuso horário que você especifica ao criar a agenda. 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 da taxa 24 horas após a última invocação da programação.

> **Note**
>
> Alguns fusos horários não observam o horário de verão, de acordo com as regras e regulamentos locais. Se você criar uma programação em um fuso horário que não observe o horário de verão, o EventBridge Scheduler não ajustará sua programação. 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 agendamento 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.

- Avanço — Quando o horário avança na primavera, das 1h59 às 3h, o EventBridge Scheduler pula a invocação da programação naquele dia e retoma a execução da programação normalmente no dia seguinte.
- Retorno — Quando o horário retrocede no outono, das 2h59 às 2h, o EventBridge Scheduler executa a programação apenas uma vez às 2h30 antes da ocorrência do turno, mas não repete a invocação da programação novamente às 2h30 após a mudança de horário.

Gerenciando uma agenda no EventBridge Scheduler

Um cronograma 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. EventBridge O Scheduler oferece suporte a três tipos de agendamentos: tarifa, cron e horários 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 API operação que o EventBridge Scheduler chama em seu nome sempre que sua agenda é executada. EventBridge O Scheduler suporta dois tipos de destinos: alvos modelados chamam API operações comuns em um grupo principal de serviços e o parâmetro de destino 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 destinos, consulte Gerenciando destinos.

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

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

Tópicos

Alterando o estado do EventBridge agendamento no Scheduler

Uma EventBridge agenda do Scheduler tem dois estados: ativada e desativada. O exemplo a seguir usa o `UpdateSchedule` para desativar um agendamento que é acionado a cada cinco minutos e invoca um destino Lambda.

Ao usar`UpdateSchedule`, você deve fornecer todos os parâmetros necessários. EventBridge O Scheduler substitui sua agenda 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 Python SDK e a `UpdateSchedule` operação para desativar um cronograma direcionado à Amazon SQS usando um destino 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 horário flexíveis no EventBridge Scheduler

Quando você configura sua agenda com uma janela de tempo 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 destinos. Definir uma janela de horário flexível melhora a confiabilidade de seu agendamento ao dispersar suas invocações de destino.

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 destino dentro de 15 minutos após o horário agendado. Os SDK exemplos a seguir AWS CLI e o EventBridge Scheduler são usados `UpdateSchedule` para definir uma janela de tempo flexível de 15 minutos para uma programação 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)
```

Configurando a fila de mensagens mortas de uma agenda no Scheduler EventBridge

O Amazon EventBridge Scheduler suporta filas de mensagens mortas () DLQ usando o Amazon Simple Queue Service. Quando um EventBridge agendamento falha em invocar sua meta, o Scheduler entrega uma JSON carga contendo detalhes da invocação e qualquer resposta recebida do destino para uma fila padrão da Amazon SQS que você especificar.

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

Os tópicos a seguir descrevem como você pode configurar uma SQS fila da Amazon DLQ para sua agenda, configurar as permissões que o EventBridge Agendador precisa para entregar mensagens para a Amazon SQS e receber eventos indiretos do. DLQ

Tópicos

- Crie uma SQS fila da Amazon
- Configure as permissões da função de execução

- Especificar uma fila de mensagens não entregues
- Recuperar o evento de mensagens não entregues

Crie uma SQS fila da Amazon

Antes de configurar uma DLQ para sua programação, você deve criar uma SQS fila padrão da Amazon. Para obter instruções sobre como criar uma fila usando o SQS console da Amazon, consulte Criação de uma SQS fila da Amazon no Guia do desenvolvedor do Amazon Simple Queue Service.

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

Use o AWS CLI comando a seguir 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 a filaARN. Você precisará do ARN quando especificar um DLQ para sua EventBridge programação do Scheduler. Você pode encontrar sua fila ARN no SQS console da Amazon ou usando o `get-queue-attributes` AWS CLI comando.

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

Se for bem-sucedido, você verá a fila ARN 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 a Amazon. SQS

Configure as permissões da função de execução

Para permitir que o EventBridge Scheduler entregue eventos com data limite para a AmazonSQS, 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 agendamento, 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 da Amazon SQSAPI.

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

Especificar uma fila de mensagens não entregues

Para especificar umDLQ, use o console do EventBridge Scheduler ou o AWS CLI para atualizar um agendamento existente ou criar um novo.

Console

Para especificar um DLQ usando o console

1. Faça login no e escolha o AWS Management Console link a seguir para abrir a seção EventBridge Agendador do EventBridge console: home https://console.aws.amazon.com/scheduler/
2. No console do EventBridge Agendador, crie uma nova agenda ou escolha uma agenda existente na sua lista de agendas para editar.
3. Na página Configurações, para Dead-letter queue (DLQ), faça o seguinte:
   - Escolha Selecionar uma SQS fila da Amazon em minha AWS conta como uma eDLQ, em seguida, escolha a fila ARN para você na DLQ lista suspensa.
   - Escolha Especificar uma SQS fila da Amazon em outras AWS contas como uma eDLQ, em seguida, entre na fila da ARN sua. DLQ Se você escolher uma fila em outra AWS conta, o console do EventBridge Scheduler não poderá exibir a fila ARNs 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 DLQ os detalhes de uma agenda, escolha o nome da agenda na lista e, em seguida, escolha a guia Fila de cartas mortas na página de detalhes da programação.

AWS CLI

Para atualizar um cronograma existente usando o AWS CLI

- Use o comando `update-schedule` para atualizar sua agenda. Especifique a SQS fila da Amazon que você criou anteriormente como a. DLQ Especifique a IAM função ARN à qual você anexou SQS as permissões necessárias da Amazon 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 um novo agendamento 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 carta morta do. DLQ

Recuperar o evento de mensagens não entregues

Use o `receive-message`comando, 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": "AQEBkNKTdOMrWgHKPoITRBwrPoK3eCSZIcZwVqCY0BZ
+FfTcORFpopJbtCqj36VbBTlHreM8+qM/m5jcwqSlAlGmIJO/hYmMgn/
+dwIty9izE7HnpvRhhEyHxbeTZ5V05RbeasYaBdNyi9WLcnAHviDh6MebLXXNWoFyYNsxdwJuG0f/
w3htX6r3dxpXvvFNPGoQb8ihY37+u0gtsbuIwhLtUSmE8rbldEEwiUfi3IJ1zEZpUS77n/k1GWrMrnYg0Gx/
BuaLzOrFi2F738XI/
Hnh45uv3ca6OYwS1ojPQ1LtX2URg1haV5884FYlaRvY8jRlpCZabTkYRTZKSXG5KNgYZnHpmsspii6JNkjitYVFKPo0H91w
            "MD5OfBody": "07adc3fc889d6107d8bb8fda42fe0573",
            "Body": "{\"MessageBody\":\"Hello, world!\",\"QueueUrl\":\"https://sqs.us-
west-2.amazonaws.com/123456789012/does-not-exist\"}",
            "Attributes": {
                "SenderId": "AROA2DZE3W4CTL5ZR7EIN:ff00212d8c453aaaae644bc6846d4723",
                "ApproximateFirstReceiveTimestamp": "1652499058144",
                "ApproximateReceiveCount": "2",
```

```
                "SentTimestamp": "1652490733042"
            },
            "MD5OfMessageAttributes": "f72c1d78100860e00403d849831d4895",
            "MessageAttributes": {
                "ERROR_CODE": {
                    "StringValue": "AWS.SimpleQueueService.NonExistentQueue",
                    "DataType": "String"
                },
                "ERROR_MESSAGE": {
                    "StringValue": "The specified queue does not exist for this wsdl
 version.",
                    "DataType": "String"
                },
                "EXECUTION_ID": {
                    "StringValue": "ad06616e51cdf74a",
                    "DataType": "String"
                },
                "EXHAUSTED_RETRY_CONDITION": {
                    "StringValue": "MaximumEventAgeInSeconds",
                    "DataType": "String"
                }
                "IS_PAYLOAD_TRUNCATED": {
                    "StringValue": "false",
                    "DataType": "String"
                },
                "RETRY_ATTEMPTS": {
                    "StringValue": "0",
                    "DataType": "String"
                },
                "SCHEDULED_TIME": {
                    "StringValue": "2022-05-14T01:12:00Z",
                    "DataType": "String"
                },
                "SCHEDULE_ARN": {
                    "StringValue": "arn:aws:scheduler:us-west-2:123456789012:schedule/
DLQ-test",
                    "DataType": "String"
                },
                "TARGET_ARN": {
                    "StringValue": "arn:aws:scheduler:::aws-sdk:sqs:sendMessage",
                    "DataType": "String"
                }
            }
        }
```

```
    ]
}
```

Observe os atributos a seguir no evento de mensagens não entregues para ajudá-lo a identificar e solucionar possíveis motivos pelos quais a inovação do destino falhou.

- **`ERROR_CODE`**— Contém o código de erro que o EventBridge Scheduler recebe do serviço API do alvo. No exemplo anterior, o código de erro retornado pela Amazon SQS é`AWS.SimpleQueueService.NonExistentQueue`. Se o agendamento falhar em 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 a mensagem de erro que o EventBridge Scheduler recebe do serviço API do alvo. No exemplo anterior, a mensagem de erro retornada pela 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 alvo que sua agenda invoca, no seguinte ARN formato de serviço:. `arn:aws:scheduler:::aws-sdk:`*`service`*`:`*`apiAction`*
- **`EXHAUSTED_RETRY_CONDITION`**— Indica por que o evento foi entregue aoDLQ. Esse atributo estará presente se o erro do destino API for um erro que pode ser repetido e não permanente. O atributo pode conter os valores `MaximumRetryAttempts` se o EventBridge Agendador o tiver enviado para o DLQ após exceder o máximo de tentativas configurado 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 destino que especificamos para o agendamento não existe.

Excluindo uma agenda no Scheduler EventBridge

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 agendamento

- Exclusão manual

Exclusão após a conclusão do agendamento

Configure a exclusão automática após a conclusão do cronograma se quiser evitar ter que gerenciar individualmente seus recursos de EventBridge agendamento no Scheduler. Em aplicativos em que você cria milhares de agendamentos ao mesmo tempo e precisa de flexibilidade para aumentar a escala verticalmente do 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 EventBridge agendamento, o Scheduler exclui o agendamento após sua última invocação de destino. Para agendamentos únicos, isso ocorre após o agendamento ter invocado seu destino uma vez. Para agendamentos 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 Agendador 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 Configurações.
4. Na seção Ação após a conclusão do cronograma, DELETEselecione na lista suspensa e salve suas alterações.

AWS CLI

1. Abra uma nova janela de prompt.

2. Use o AWS CLI comando update-schedule para atualizar uma agenda existente, conforme mostrado a seguir. O comando define o `--action-after-completion` para `DELETE`. Este exemplo pressupõe que você tenha definido sua configuração de destino localmente em um JSON arquivo. Para atualizar um agendamento, você deve fornecer a meta, bem como quaisquer outros parâmetros de agendamento que você queira configurar para o seu agendamento 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 destinos modelados para as funções Lambda e Step e aprender a usar o parâmetro de destino universal, consulte Gerenciando destinos.
- Para obter mais informações sobre os tipos de dados e API operações do EventBridge Scheduler, consulte a Referência do EventBridge Scheduler. API

Gerenciando um grupo de EventBridge agendamento no Scheduler

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

Você Conta da AWS vem com um grupo de `default` agendadores. Você pode associar um novo agendamento 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) ou outras informações confidenciais ou sigilosas nas tags. As tags podem ser acessadas por vários AWS serviços, incluindo faturamento. As tags não devem ser usadas para dados privados ou confidenciais.

Um grupo de agendamento tem dois estados possíveis: ACTIVEDELETINGe.

Quando você cria um grupo pela primeira vez, é `ACTIVE` por padrão. Você pode adicionar agendamentos a um grupo `ACTIVE`. Quando você exclui um grupo, o estado muda para `DELETING` até que o EventBridge Agendador conclua a exclusão dos agendamentos associados. Depois que o EventBridge Agendador 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 programação ao grupo. Por fim, você excluirá o grupo.

Tópicos

- Recursos relacionados

Criando um grupo de EventBridge agendamento no Scheduler

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 um novo agendamento a esse grupo.

> **Note**
>
> Depois de criar um grupo, você não pode remover um agendamento desse grupo nem associar o agendamento a um grupo diferente. Você só pode associar um agendamento a um grupo ao criar o agendamento pela primeira vez.

Etapa 1: criar um novo grupo de agendamento

Os seguintes tópicos 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 usando o AWS Management Console

1. Faça login no AWS Management Console e abra o EventBridge console da Amazon 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 Adicionar nova tag.
   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 usando o AWS CLI

1. Abra uma nova janela do prompt de comando.
2. No AWS Command Line Interface (AWS CLI), digite o `create-schedule-group`comando a seguir 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 agendamento 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 novos agendamentos ao novo grupo que você criou.

Etapa 2: associar um agendamento ao grupo

Use as etapas a seguir para associar um novo agendamento 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 EventBridge console da Amazon 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 Agendamentos, escolha Criar agendamento para criar um novo agendamento.
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. No AWS Command Line Interface (AWS CLI), insira o seguinte `create-schedule`comando. Isso cria um agendamento e o associa ao grupo da etapa anterior, denominada `sqs-test-schedule`. Esse cronograma usa o tipo de SQS alvo modelo da Amazon para invocar a operação. `SendMessage` Substitua o nome da agenda, o destino 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"}'
```

Seu novo agendamento agora está associado ao grupo de agendamentos do `TestGroup`.

Excluindo um grupo de agendamentos no EventBridge Scheduler

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

> **Note**
>
> Depois de criar um grupo, você não pode remover um agendamento desse grupo nem associar o agendamento a um grupo diferente. Você só pode associar um agendamento a um grupo ao criar o agendamento 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 EventBridge console da Amazon 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 Programar grupos, na lista de grupos existentes no atual Região da AWS, localize o grupo que você deseja excluir. Se você não encontrar o grupo que está procurando, 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 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 Agendador 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. No AWS Command Line Interface (AWS CLI), digite o `delete-schedule-group`comando 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 AWS CLI operação 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"
}
```

EventBridge O agendador exclui o grupo depois de excluir os agendamentos associados ao grupo. Se você executar `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:

- CreateScheduleGroupoperação na APIReferência do EventBridge Scheduler.
- DeleteScheduleGroupoperação na APIReferência do EventBridge Scheduler.

Gerenciando alvos no EventBridge Scheduler

Os tópicos a seguir descrevem como usar alvos modelados e universais com o EventBridge Scheduler e fornecem uma lista dos AWS serviços suportados que você pode configurar usando o parâmetro de destino universal do EventBridge Scheduler.

Os alvos modelados são um conjunto de API operações comuns em um grupo de AWS serviços principais, como AmazonSQS, Lambda e Step Functions. Por exemplo, você pode direcionar a API operação Invoke do Lambda fornecendo a função ARN ou a `SendMessage`operação SQS da Amazon com a fila ARN do destino.

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

Para configurar metas padronizadas ou universais, sua agenda deve ter permissão para chamar a API operação 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 `SendMessage`operação SQS da Amazon, a função de execução recebe permissão para realizar a `sqs:SendMessage` ação. Na maioria dos casos, você pode adicionar as permissões necessárias usando as políticas gerenciadas da AWS aceitas pelo serviço de destino. 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 destino 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

- Usando alvos modelados no EventBridge Scheduler
- Usando alvos universais no EventBridge Scheduler
- Adicionando atributos de contexto no EventBridge Scheduler
- Próximas etapas

Usando alvos modelados no EventBridge Scheduler

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

Para configurar programaticamente um alvo modelado usando o AWS CLI ou um dos EventBridge AgendadoresSDKs, você precisa especificar a função ARN de execução, o recurso ARN para o alvo, uma entrada opcional que você deseja que o EventBridge Agendador entregue ao alvo e, para alguns alvos 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 API operação suportada para esse serviço. Se você quiser que o EventBridge Scheduler direcione uma API operação diferente para o serviço, você deve configurar o destino como um alvo 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 APIReferência do EventBridge Agendador.

- 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 como configurar diferentes alvos modelados e as IAM permissões necessárias para cada alvo descrito.

Amazon SQS **SendMessage**

Example Política de permissão para função de execução

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

Example AWS CLI

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

Example Python SDK

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

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

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

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

Example Java SDK

```
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 Java SDK

```
package com.example;

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


public class MySchedulerApp {

    public static void main(String[] args) {

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

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

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

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

Funções de Etapa **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 Java SDK

```
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 no EventBridge Scheduler

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

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

- RoleArn— ARN Para a função de execução que você deseja usar para o alvo. A função de execução especificada deve ter as permissões para chamar a API operação que você deseja que sua agenda vise.
- Arn — O serviço completoARN, incluindo a API operação que você deseja atingir, no seguinte formato:`arn:aws:scheduler:::aws-sdk:`*`service`*`:`*`apiAction`*.

  Por exemplo, para a AmazonSQS, o nome do serviço que você especifica é`arn:aws:scheduler:::aws-sdk:`**`sqs:sendMessage`**.
- Entrada — Uma entrada bem formada JSON que você especifica com os parâmetros de solicitação que o EventBridge Scheduler envia ao destino. API Os parâmetros e a forma do JSON que você configurou `Input` são determinados pelo serviço que API sua agenda invoca. Para encontrar essas informações, consulte a API referência do serviço que você deseja segmentar.

Ações não compatíveis

EventBridge O Scheduler não suporta API ações somente para leitura, como `GET` operações comuns, 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
invokeModel
```

Por exemplo, o serviço ARN para a `GetQueueUrl`APIação seria o seguinte:`arn:aws:scheduler:::aws-sdk:sqs:`**`getQueueURL`**. Como a API ação começa com o `get` prefixo, o EventBridge Scheduler não suporta esse alvo. Da mesma forma, a `ListBrokers`ação do Amazon MQ não é suportada como destino porque a operação começa com o prefixo. `list`

Exemplos usando o destino universal

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

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

Os exemplos a seguir mostram como usar o alvo universal para invocar API operações comuns com Lambda, SQS Amazon 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 no EventBridge Scheduler

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

- **<aws.scheduler.schedule-arn>**— O ARN do cronograma.
- **<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 cronograma que é acionado a cada cinco minutos e invoca a SQS `SendMessage` operação da Amazon 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 API operações do EventBridge Scheduler, consulte a Referência do EventBridge Scheduler. API

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

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

Tópicos

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, consulteSolução 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. Em seguida, você deve enviar solicitações ao IAM administrador para alterar as permissões dos usuários do serviço. Revise as informações nesta página para entender os conceitos básicos doIAM. Para saber mais sobre como sua empresa pode usar o IAM EventBridge Scheduler, consulteComo o EventBridge Scheduler funciona com IAM.

IAMadministrador — Se você for IAM administrador, 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 emIAM, consulte. Usando políticas baseadas em identidade no Scheduler EventBridge

Autenticando com identidades

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

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Os usuários (do 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 entra como uma identidade federada, seu administrador configurou previamente a federação de identidades usando IAM funções. 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 você mesmo assinar solicitações, consulte Assinar AWS API solicitações no Guia IAM do usuário.

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 Uso da autenticação multifator (MFA) AWS no Guia do IAMusuário.

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 ver a lista completa de tarefas que exigem que você faça login como usuário raiz, consulte Tarefas que exigem credenciais de usuário raiz no Guia do IAM usuário.

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 informações sobre o IAM Identity Center, consulte O que é o IAM Identity Center? no Guia do AWS IAM Identity Center usuário.

Grupos e usuários do IAM

Um IAMusuário é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos confiar em credenciais temporárias em vez de criar IAM usuários que tenham 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 IAM os usuários, recomendamos que você alterne as chaves de acesso. Para obter mais informações, consulte Alterne as chaves de acesso regularmente para casos de uso que exigem credenciais de longo prazo no Guia do IAMusuário.

Um IAMgrupo é uma identidade que especifica uma coleção de IAM usuários. 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 IAMAdminse conceder a esse grupo permissões para administrar IAM recursos.

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 IAM usuário (em vez de uma função) no Guia do IAM usuário.

IAMfunções

Uma IAMfunção é uma identidade dentro da sua Conta da AWS que tem permissões específicas. É semelhante a um IAM usuário, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma IAM função no AWS Management Console trocando de funções. Você pode assumir uma função chamando uma AWS API operação AWS CLI or ou usando uma personalizadaURL. Para obter mais informações sobre métodos de uso de funções, consulte Usando IAM funções no Guia IAM do usuário.

IAMfunções 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 informações sobre funções para federação, consulte Criação de uma função para um provedor de identidade terceirizado no Guia IAM do usuário. Se você usa o IAM Identity Center, configura um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o IAM Identity Center correlaciona o conjunto de permissões a uma função em. IAM Para obter informações sobre conjuntos de permissões, consulte Conjuntos de Permissões no Manual do Usuário do AWS IAM Identity Center .
- Permissões temporárias IAM de IAM usuário — Um usuário ou função pode assumir uma IAM função para assumir temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas — Você pode usar uma IAM função para permitir que alguém (um diretor confiável) em uma conta diferente 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 IAM no Guia do IAM usuário.
- 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 na 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 IAM usuário ou uma função para realizar ações em AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. FASusa as permissões do diretor chamando um Serviço da AWS, combinadas com a solicitação Serviço da AWS para fazer solicitações aos serviços posteriores. FASas solicitações 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 FAS solicitações, consulte Encaminhar sessões de acesso.
- Função de serviço — Uma função de serviço é uma IAMfunção que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamenteIAM. Para obter mais informações, consulte Criação de uma função para delegar permissões a uma Serviço da AWS no Guia do IAM usuário.
- Função vinculada ao serviço — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. 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 IAM administrador pode visualizar, mas não editar, as permissões das funções vinculadas ao serviço.

- Aplicativos em execução na Amazon EC2 — Você pode usar uma IAM função para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma EC2 instância e fazendo AWS CLI AWS API solicitações. Isso é preferível ao armazenamento de chaves de acesso na EC2 instância. Para atribuir uma AWS função a uma EC2 instância e disponibilizá-la para todos os aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém a função e permite que os programas em execução na EC2 instância recebam credenciais temporárias. Para obter mais informações, consulte Como usar uma IAM função para conceder permissões a aplicativos executados em EC2 instâncias da Amazon no Guia IAM do usuário.

Para saber se usar IAM funções ou IAM usuários, consulte Quando criar uma IAM função (em vez de um usuário) no Guia do IAM usuário.

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 JSON documentos. Para obter mais informações sobre a estrutura e o conteúdo dos documentos de JSON política, consulte Visão geral das JSON políticas no Guia IAM do usuário.

Os administradores podem usar AWS JSON políticas 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 permissão aos usuários para realizar ações nos recursos de que precisam, um IAM administrador pode criar IAM políticas. O administrador pode então adicionar as IAM políticas às funções e os usuários podem assumir as funções.

IAMas políticas definem permissões para uma ação, independentemente do método usado para realizar 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 do AWS Management Console AWS CLI, do ou do AWS API.

Políticas baseadas em identidade

Políticas baseadas em identidade são documentos de políticas de JSON permissões que você pode anexar a uma identidade, como um IAM usuário, grupo de usuários ou função. 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 Criação de IAM políticas no Guia do IAMusuário.

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 Escolha entre políticas gerenciadas e políticas em linha no Guia do IAMusuário.

Políticas baseadas no recurso

Políticas baseadas em recursos são documentos JSON de política que você anexa a um recurso. Exemplos de políticas baseadas em recursos são políticas de confiança de IAM funções e 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 de uma política baseada IAM em recursos.

Listas de controle de acesso (ACLs)

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

Amazon S3, AWS WAF, e Amazon VPC são exemplos de serviços que oferecem suporte. ACLs Para saber maisACLs, consulte a 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 recurso avançado no qual você define as permissões máximas que uma política baseada em identidade pode conceder a uma IAM entidade (IAMusuário ou função). 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 IAM entidades no Guia IAM do usuário.
- Políticas de controle de serviço (SCPs) — SCPs são JSON políticas 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 vários Contas da AWS que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as

suas contas. Os SCP limites de permissões para entidades nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre Organizations eSCPs, consulte Políticas de controle de serviços no Guia AWS Organizations do Usuário.

- 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 IAM do usuário.

Vários tipos de política

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

Como o EventBridge Scheduler funciona com IAM

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

IAMrecursos que você pode usar com o Amazon EventBridge Scheduler

IAMrecurso	EventBridge Suporte ao agendador
Políticas baseadas em identidade	Sim
Políticas baseadas em recursos	Não
Ações das políticas	Sim
Atributos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não

IAMrecurso	EventBridge Suporte ao agendador
ABAC(tags nas políticas)	Parcial
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Sim
Perfis vinculados ao serviço	Não

Para obter uma visão de alto nível de como o EventBridge Scheduler e outros AWS serviços funcionam com a maioria dos IAM recursos, consulte AWS os serviços que funcionam com IAM no Guia do IAMUsuário.

Políticas baseadas em identidade para o Scheduler EventBridge

Compatível com políticas baseadas em identidade: Sim

Políticas baseadas em identidade são documentos de políticas de JSON permissões que você pode anexar a uma identidade, como um IAM usuário, grupo de usuários ou função. 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 Criação de IAM políticas no Guia do IAMusuário.

Com políticas IAM baseadas em identidade, você pode especificar ações e recursos permitidos ou negados, bem 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 você pode usar em uma JSON política, consulte a referência IAM JSON de elementos de política no Guia IAM do usuário.

Exemplos de políticas baseadas em identidade para o Scheduler EventBridge

Para ver exemplos de políticas baseadas em identidade do EventBridge Scheduler, consulte. Usando políticas baseadas em identidade no Scheduler EventBridge

Políticas baseadas em recursos no Scheduler EventBridge

Suporte a políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos JSON de política que você anexa a um recurso. Exemplos de políticas baseadas em recursos são políticas de confiança de IAM funções e 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 habilitar o acesso entre contas, você pode especificar uma conta ou IAM entidades inteiras em outra conta como principal em uma política baseada em recursos. 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 IAM administrador 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 IAM no Guia do IAM usuário.

Ações de política para o EventBridge Scheduler

Compatível com ações de políticas: Sim

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

O `Action` elemento de uma JSON política 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 AWS API operação associada. Há algumas exceções, como ações somente com permissão que não têm uma operação correspondente. API 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

Compatível com recursos de políticas: Sim

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

O elemento `Resource` JSON de política especifica o objeto ou objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu Amazon Resource Name (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 seusARNs, consulte Recursos definidos pelo Amazon EventBridge Scheduler na Referência de autorização de serviço. Para

saber com quais ações você pode especificar cada recurso, consulte Ações definidas pelo Amazon EventBridge Scheduler. ARN

Para ver exemplos de políticas baseadas em identidade do EventBridge Scheduler, consulte. Usando políticas baseadas em identidade no Scheduler EventBridge

Chaves de condição de política para o EventBridge Scheduler

Compatível com chaves de condição de política específicas de serviço: Sim

Os administradores podem usar AWS JSON políticas 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, você pode conceder permissão a um IAM usuário para acessar um recurso somente se ele estiver marcado com o nome de IAM usuário. Para obter mais informações, consulte elementos de IAM política: variáveis e tags no Guia IAM do usuário.

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 IAM do usuário.

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. Usando políticas baseadas em identidade no Scheduler EventBridge

ACLsno EventBridge Scheduler

SuportesACLs: Não

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

ABACcom EventBridge Scheduler

Suportes ABAC (tags nas políticas): Parciais

O controle de acesso baseado em atributos (ABAC) é uma estratégia de autorização que define permissões com base em atributos. Em AWS, esses atributos são chamados de tags. Você pode anexar tags a IAM entidades (usuários ou funções) e a muitos AWS recursos. Marcar entidades e recursos é a primeira etapa doABAC. Em seguida, você cria ABAC políticas para permitir operações quando a tag do diretor corresponde à tag do recurso que ele está tentando acessar.

ABACé útil em ambientes que estão crescendo rapidamente e ajuda em situações em que o gerenciamento de políticas se torna complicado.

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 sobreABAC, consulte O que éABAC? no Guia do IAM usuário. Para ver um tutorial com etapas de configuraçãoABAC, consulte Usar controle de acesso baseado em atributos (ABAC) no Guia do IAMusuário.

Usando credenciais temporárias com EventBridge o Scheduler

Compatível 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 esse trabalho IAM no Guia do IAM usuário.

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 a troca de funções, consulte Alternando para uma função (console) no Guia IAM do usuário.

Você pode criar manualmente credenciais temporárias usando o AWS CLI ou AWS API. 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 obter mais informações, consulte Credenciais de segurança temporárias emIAM.

Permissões principais entre serviços para EventBridge o Scheduler

Suporta sessões de acesso direto (FAS): Sim

Quando você usa um IAM usuário ou uma função para realizar ações em AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. FASusa as permissões do diretor chamando um Serviço da AWS, combinadas com a solicitação Serviço da AWS para fazer solicitações aos serviços posteriores. FASas solicitações 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 FAS solicitações, consulte Encaminhar sessões de acesso.

Funções de serviço do EventBridge Scheduler

Compatível com perfis de serviço: Sim

Uma função de serviço é uma IAMfunção que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamenteIAM. Para obter mais informações, consulte Criação de uma função para delegar permissões a uma Serviço da AWS no Guia do IAM usuário.

> **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

Compatível com perfis vinculados ao serviço: Não

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. 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 IAM administrador pode visualizar, mas não editar, as permissões das funções vinculadas ao serviço.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas a serviços, consulte AWS serviços que funcionam com. 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.

Usando políticas baseadas em identidade no Scheduler EventBridge

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

Para saber como criar uma política IAM baseada em identidade usando esses exemplos de documentos de JSON política, consulte Criação de IAM políticas no Guia do IAMusuário.

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

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 seus 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 AWS gerenciadas ou políticas AWS gerenciadas para funções de trabalho no Guia IAM do usuário.
- Aplique permissões com privilégios mínimos — Ao definir permissões com IAM políticas, conceda somente as permissões necessárias para realizar 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 IAM como usar para aplicar permissões, consulte Políticas e permissões IAM no Guia IAM do usuário.
- Use condições nas IAM políticas para restringir ainda mais o acesso — Você pode adicionar uma condição às suas políticas para limitar o acesso a ações e recursos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usandoSSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica Serviço da AWS, como AWS CloudFormation. Para obter mais informações, consulte elementos IAM JSON da política: Condição no Guia IAM do usuário.
- Use o IAM Access Analyzer para validar suas IAM políticas e garantir permissões seguras e funcionais — o IAM Access Analyzer valida políticas novas e existentes para que as políticas sigam a linguagem da IAM política (JSON) e as melhores práticas. IAM IAMO Access Analyzer fornece mais de 100 verificações de políticas e recomendações práticas para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte Validação da política do IAM Access Analyzer no Guia do IAM Usuário.
- Exigir autenticação multifatorial (MFA) — Se você tiver um cenário que exija IAM usuários ou um usuário root Conta da AWS, ative MFA para obter segurança adicional. Para exigir MFA quando API as operações são chamadas, adicione MFA condições às suas políticas. Para obter mais informações, consulte Configurando o API acesso MFA protegido no Guia do IAMusuário.

Para obter mais informações sobre as melhores práticas emIAM, consulte as melhores práticas de segurança IAM no Guia IAM do usuário.

EventBridge Permissões do agendador

Para que um IAM diretor (usuário, grupo ou função) crie agendas no EventBridge Agendador e acesse os recursos do EventBridge Agendador por meio do console ou doAPI, 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 `CreateSchedule` API operação. 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 IAM políticas autônomas 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 AWS gerenciadas no Guia IAM do usuário. 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 o. 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 AWSchave 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 em 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 KMS chave especificada a 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"
            }
        }
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
```

```
            "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 as 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
the section called “AmazonEventBridgeSchedulerFullAccess”: 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
the section called “AmazonEventBridgeSchedulerReadOnlyAccess”: 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 delegada confusa no EventBridge Scheduler

O problema de "confused deputy" é uma questão de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executá-la. 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:SourceArnchaves de contexto de condição aws:SourceAccountglobal 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 confuso problema do deputado é usar a chave de contexto ARN de condição aws:SourceArn global com todo o 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 a totalidade ARN do recurso ou se estiver especificando vários recursos, use a chave de condição de contexto aws:SourceArn global com caracteres curinga (*) para as partes desconhecidas do. ARN Por exemplo: arn:aws:scheduler:*:*123456789012*:schedule-group/*.

O valor de aws:SourceArn deve ser seu grupo de EventBridge agendamento do Scheduler ARN 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 agendamento.

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. 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 exemplo de erro a seguir ocorre quando o `mateojackson` IAM usuário tenta usar o console para ver detalhes sobre um *`my-example-widget`* recurso fictício, mas não tem as permissões fictícias`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 IAM usuário 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 suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o EventBridge Scheduler oferece suporte a esses recursos, consulteComo o EventBridge Scheduler funciona com IAM.

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

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 privacidade de dados, consulte Privacidade de dados FAQ. Para obter informações sobre proteção de dados na Europa, consulte o Modelo de Responsabilidade AWS Compartilhada e GDPR a postagem no blog AWS 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 a autenticação multifator (MFA) com cada conta.
- UseSSL/TLSpara se comunicar com AWS os recursos. Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Configure API e registre as 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 FIPS 140-3 módulos criptográficos validados ao acessar AWS por meio de uma interface de linha de comando ou umaAPI, use um endpoint. FIPS Para obter mais informações sobre os FIPS endpoints disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

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 outro Serviços da AWS usando o console,API, AWS CLI, ou AWS SDKs. 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, é altamente recomendável que você não inclua informações de credenciais no URL para validar sua solicitação para esse servidor.

Criptografia em repouso no EventBridge Scheduler

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 suporta dois tipos de KMS chaves: Chaves pertencentes à AWSe chaves gerenciadas pelo cliente.

> **Note**
>
> EventBridge O Scheduler só suporta o uso de chaves de criptografia KMSsimétricas.

Chaves pertencentes à AWSsão KMS chaves 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 KMS chaves 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 KMS chaves
- 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 KMS chave 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 especific a no parâmetro `TargetInput` do agendamen to ao configurar o agendamento para ser entregue ao destino.	Compatível	Compatível
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	Compatível	Sem compatibilidade

Tipo de dados	Descrição	Chave pertencente à AWS	chave gerenciada pelo cliente
	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.		
Configurações de destino	O Amazon Resource Name (ARN) do alvo 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 KMS chaves” seção a seguir para saber como você deve gerenciar seus IAM recursos e políticas principais para usar uma chave gerenciada pelo cliente com o EventBridge Scheduler.

Gerenciando KMS chaves

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 agendador usa IAM permissões associadas ao diretor 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 o EventBridge Agendador. API 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 chamá-la 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 IAM as permissões necessárias para usar uma chave gerenciada pelo cliente no EventBridge Scheduler.

Tópicos

- Adicionar IAM permissões
- Gerenciar a política de chave

Adicionar IAM permissões

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

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 o EventBridge Agendador API 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 de criptografia KMSsimétrica.
- **`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 seguinte ação à política de permissões da função de execução do seu cronograma para fornecer acesso ao EventBridge Scheduler para chamá-lo 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 API chamadas. Isso inclui API chamadas 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": "ABCDEFGHI1JKLMNOP2Q3",
      "sessionContext": {
          "sessionIssuer": {
              "type": "Role",
              "principalId": "ABCDEABCD1AB12ABABAB0",
              "arn": "arn:aws:iam::123456789012:role/execution-role",
              "accountId": "123456789012",
              "userName": "execution-role"
          },
          "webIdFederationData": {},
          "attributes": {
              "creationDate": "2022-10-31T21:03:15Z",
              "mfaAuthenticated": "false"
          }
      }
    },
    "eventTime": "2022-10-31T21:03:15Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "eu-north-1",
    "sourceIPAddress": "13.50.87.173",
    "userAgent": "aws-sdk-java/2.17.295 Linux/4.14.291-218.527.amzn2.x86_64 OpenJDK_64-
Bit_Server_VM/11.0.17+9-LTS Java/11.0.17 kotlin/1.3.72-release-468 (1.3.72) vendor/
Amazon.com_Inc. md/internal exec-env/AWS_ECS_FARGATE io/sync http/Apache cfg/retry-
mode/standard AwsCrypto/2.4.0",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-west-2:123456789012:key/2321abab-2110-12ab-a123-
a2b34c5abc67",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "encryptionContext": {
            "aws:scheduler:schedule:arn": "arn:aws:scheduler:us-
west-2:123456789012:schedule/default/execution-role"
        }
    },
    "responseElements": null,
```

```
    "requestID": "request-id",
    "eventID": "event-id",
    "readOnly": true,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:123456789012:key/2321abab-2110-12ab-a123-
a2b34c5abc67"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
      "tlsVersion": "TLSv1.3",
      "cipherSuite": "TLS_AES_256_GCM_SHA384",
      "clientProvidedHostHeader": "kms.us-west-2.amazonaws.com"
  }
}
```

Criptografia em trânsito no EventBridge Scheduler

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 API operação do EventBridge Scheduler, bem como quando o EventBridge Scheduler chama qualquer alvo APIs ao invocar sua agenda. Por padrão, o EventBridge Scheduler usa TLS 1.2 ao criptografar seus dados em trânsito. Você não precisa configurar a criptografia em trânsito e não pode escolher uma TLS versão diferente ao usar o EventBridge Scheduler.

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

Usando o alvo APIs — Quando o EventBridge Agendador invoca sua agenda, ele chama a meta API que você especificou quando criou 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 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 HIPAA segurança e conformidade na Amazon Web Services — Este whitepaper descreve como as empresas podem usar AWS para criar HIPAA aplicativos qualificados.

> **Note**
>
> Nem todos Serviços da AWS são HIPAA elegíveis. Para obter mais informações, consulte a Referência de serviços HIPAA elegíveis.

- 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 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 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, por exemplo PCIDSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.
- AWS Audit Manager— Isso 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 mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center 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 API chamadas AWS publicadas para acessar o EventBridge Scheduler pela rede. Os clientes devem oferecer suporte para:

- Segurança da camada de transporte (TLS). Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Suítes de criptografia com sigilo direto perfeito (), como (Ephemeral PFS Diffie-Hellman) ou DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando uma ID de chave de acesso e uma chave de acesso secreta associada a um IAM principal. 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 da manutenção da confiabilidade, disponibilidade e desempenho do Amazon EventBridge Scheduler e de suas outras AWS soluções. AWS fornece as seguintes ferramentas de monitoramento para monitorar o EventBridge Scheduler, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- A Amazon CloudWatch monitora seus AWS recursos e os aplicativos em que você executa AWS em tempo real. É possível 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 CloudWatch do usuário da Amazon.
- AWS CloudTrailcaptura API chamadas e eventos relacionados feitos por ou em nome de sua AWS conta e entrega os arquivos de log para um bucket do Amazon S3 que você especificar. Você pode identificar quais usuários e contas ligaram AWS, o endereço IP de origem a partir do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte o Guia do usuário do AWS CloudTrail.

Tópicos

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 mensagens mortas associada (). DLQ Se você configurar um 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

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 que são 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. `default`

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, InvocationAttemptCountou InvocationDroppedCount, 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 todos os EventBridge agendamentos do Scheduler, bem como métricas adicionais para os agendamentos para os quais você configurou um. 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 alvo retorna uma exceção após o EventBridge Scheduler chamar o alvo. API Use isso para verificar quando a entrega para um destino falha.
AWS/Scheduler	TargetErrorThrottledCount	Contagem	Emitido quando a invocação do alvo falha devido à API limitação do alvo. Use isso para diagnosticar falhas de entrega quando o motivo subjacente são as chamadas de API limitação de destino feitas pelo Scheduler. EventBridge

Namespace	Métrica	Unidade	Descrição
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 a cota limite do acelerador de invocações. Para obter mais informações sobre Service Quotas, consulte Cotas.
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 RetryPolicyna Referência do EventBridge Agendador. API

Métricas para agendas com um DLQ

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

Namespace	Métrica	Unidade	Descrição
AWS/Scheduler	InvocationsFailedToBeSentToDeadLetterCount_<error_code>	Contagem	entregar um evento para o. DLQ Use essas duas métricas para determinar o motivo pelo qual o EventBridge Scheduler não consegue enviar um evento para o DLQ e modifique sua DLQ configuração para resolver o problema. Veja a seguir um exemplo da InvocationsFailedToBeSentToDeadLetterCount_<error_code> métrica quando a

Namespace	Métrica	Unidade	Descrição
			SQS fila da Amazon que você especifica como DLQ a não existe: `InvocationsFailedToBeSentToDeadLetterCount_` **AWS.SimpleQueueService.NonExistentQueue**

Namespace	Métrica	Unidade	Descrição
AWS/Scheduler	InvocationsSentToDeadLetterCount_Truncated_MessageSizeExceeded	Contagem	Emitido quando a carga útil do evento enviado para o DLQ excede o tamanho máximo permitido pela AmazonSQS, e o EventBridge Scheduler trunca a carga que você especifica no atributo de uma programação. Input

EventBridge Métricas de uso do agendador

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 deAWS/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 `Resource` dimensão especifica o nome da API operação que está sendo rastreada.

Por exemplo, a `CallCount` métrica com as dimensões a seguir indica o número de vezes que a Agendador do EventBridge `CreateSchedule` API operação 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		
`CallCount`	O número de operações especific adas executadas em sua conta.		

Dimensões

Dimensão	Descrição		
`Service`	O nome do AWS serviço que contém o recurso. Para métricas de Agendador do EventBridge uso, o valor dessa dimensão é`Scheduler` .		
`Class`	A classe do recurso sob acompanha mento.		

Dimensão	Descrição
	Agendador do EventBridge APIas métricas de uso usam essa dimensão com um valor de`None`.
`Type`	O tipo de recurso que está sendo acompanhado. No momento, quando a dimensão `Service` é `Scheduler` , o único valor válido para `Type` é `API`.
`Resource`	O nome da API operação. Entre os valores válidos estão os seguintes: • `CreateSchedule` • `CreateScheduleGroup` • `DeleteSchedule` • `DeleteScheduleGroup` • `GetSchedule` • `GetScheduleGroup` • `ListScheduleGroups` • `ListSchedules` • `ListTagsForResource` • `TagResource` • `UntagResource` • `UpdateSchedule`

Registrando API chamadas do Amazon EventBridge Scheduler usando AWS CloudTrail

O Amazon EventBridge Scheduler é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no EventBridge Scheduler. CloudTrail captura todas as API chamadas para o EventBridge Scheduler como eventos. As

chamadas capturadas incluem chamadas do console do EventBridge Scheduler e chamadas de código para as operações do EventBridge Scheduler. API Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para EventBridge o Scheduler. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao EventBridge Scheduler, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o Guia AWS CloudTrail do usuário.

EventBridge Informações do agendador em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre no EventBridge Scheduler, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes no seu Conta da AWS. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em seu Conta da AWS, incluindo eventos do EventBridge Scheduler, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para 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 eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para obter mais informações, consulte as informações a seguir.

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando SNS notificações da Amazon para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as API ações EventBridge do Scheduler são registradas CloudTrail e documentadas na Amazon EventBridge API Scheduler Reference. Por exemplo, chamadas para o `CreateSchedule` `UpdateSchedule` e `DeleteSchedule` as ações geram entradas nos arquivos de CloudTrail log.

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 root ou AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte o CloudTrail userIdentityelemento.

Compreendendo as entradas do arquivo EventBridge de log do Scheduler

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log 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 da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das API chamadas públicas, portanto, eles não aparecem em nenhuma 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 a maioria das cotas, mas 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 o aumento da quota, consulte Solicitar um aumento de quota no Guia do usuário do Service Quotas. Se a cota ainda não estiver disponível no Service Quotas, use o formulário de aumento de limite.

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

Nome	Padrão	Ajustá	Descrição
CreateSchedule taxa de solicitação	ca-central-1:250 eu-central-1: 1.000 Cada uma das outras regiões compatíveis: 50	Sim	Máximo de CreateSch edule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
CreateScheduleGroup taxa de solicitaç ão	Cada região compatível: 10	Sim	Máximo de CreateSch eduleGroup solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
DeleteSchedule taxa de solicitação	ca-central-1:250	Sim	Máximo de DeleteSch edule solicitações por segundo. Quando você

Nome	Padrão	Ajustá	Descrição
	eu-central-1: 1.000 Cada uma das outras regiões compatíveis: 50		atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
DeleteScheduleGroup taxa de solicitação	Cada região compatível: 10	Sim	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	ca-central-1:250 eu-central-1: 1.000 Cada uma das outras regiões compatíveis: 50	Sim	Máximo de GetSchedule solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
GetScheduleGroup taxa de solicitação	Cada região compatível: 10	Sim	Máximo de GetScheduleGroup solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.

Nome	Padrão	Ajustá	Descrição
Limite de controle de utilização de invocações em transações por segundo	eu-central-1: 1.000 Cada uma das outras regiões compatíveis: 500	Sim	Uma invocação é uma carga útil do cronograma que está sendo entregue ao destino definido. Depois que o limite for atingido, as invocações serão limitadas; isto é, elas ainda acontecem, mas serão atrasadas.
ListScheduleGroups taxa de solicitação	Cada região compatível: 10	Sim	Máximo de ListSched uleGroups solicitações por segundo. Quando você atinge essa cota, o EventBridge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
ListSchedules taxa de solicitação	Cada região compatível: 50	Sim	Máximo de ListSchedules solicitações por segundo. Quando você atinge essa cota, o EventBrid ge Scheduler rejeita as solicitações dessa operação pelo restante do intervalo.
ListTagsForResource taxa de solicitaç ão	Cada região compatível: 10	Sim	Lista todas as tags associadas ao recurso do Scheduler.
Número de grupos de agendamento	Cada região com suporte: 500	Sim	Número máximo de grupos de agendamento por região.

Nome	Padrão	Ajustá	Descrição
Número de esquemas	ca-central-1:10.000.000 eu-central-1:10.000.000 Cada uma das outras regiões suportadas: 1.000.000	Sim	O número máximo de agendamentos por região. Essa cota inclui programações únicas com execução concluída. Recomendamos configurar suas agendas para serem excluídas automaticamente após a conclusão do uso do ActionAfterCompletion recurso.
TagResource taxa de solicitação	Cada região compatível: 1	Sim	Atribui uma ou mais tags (pares chave-valor) ao recurso especificado do Scheduler.
UntagResource taxa de solicitação	Cada região compatível: 1	Sim	Remove uma ou mais tags do recurso especificado do Scheduler.
UpdateSchedule taxa de solicitação	ca-central-1:250 eu-central-1: 1.000 Cada uma das outras regiões compatíveis: 50	Sim	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

Solução de problemas de cotas no EventBridge Scheduler

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar em relação às cotas do EventBridge Scheduler.

ServiceQuotaExceededException

Estou recebendo erros de limitação na taxa de `UpdateSchedule` solicitação `CreateSchedule` `DeleteScheduleGetSchedule`,, ou, mesmo estando abaixo do limite de taxa padrão.

Causa comum

Em 7 de setembro de 2023, o EventBridge Scheduler começou a oferecer suporte às políticas de confiança da função de execução ScheduleGroup ARN (Amazon Resource Name) ARN em vez da Schedule in execution. Os clientes autorizados a continuar usando o Schedule ARNs em sua política de confiança podem ter limites de 50TPS, em vez dos limites padrão de 250 a 1000 TPS (dependendo da região).

Resolução

Entre em contato com o suporte para solicitar um limite máximo maior.

Prevenção

Modifique suas políticas de confiança existentes de uma das seguintes formas:

- Removendo todo o escopo da função.
- Definindo o escopo da função para que ela possa ser assumida usando o Cronograma ARN ou o. ScheduleGroup ARN

  Por exemplo, suponha que você tenha a seguinte política de confiança existente:

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "scheduler.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
        "StringEquals": {
```

```
            "aws:SourceArn":
 "arn:aws:scheduler:region:account:schedule/schedule_group/schedule"
        }
    }
}
```

Você pode atualizar a política de confiança para o seguinte:

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

Histórico de documentos do Guia do usuário do EventBridge Scheduler

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

Alteração	Descrição	Data
Mudanças na função de execução e prevenção do problema de segurança delegada confusa	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. • the section called “Prevenção contra representante confuso”	7 de setembro de 2023
Exclusão automática de agendamentos após a conclusão	EventBridge O Scheduler suporta a exclusão automática. Quando você configura a exclusão automática, o EventBridge Scheduler exclui sua agenda após a última invocação planejada. • the section called “Exclusão após a conclusão do agendamento”	2 de agosto de 2023
Tópico atualizado sobre o uso de destinos universais	Atualizou a lista de serviços compatíveis que o EventBridge Scheduler pode segmentar	17 de março de 2023

	e integrar. Essa atualizaç ão também inclui uma lista de `GET` API operações não suportadas e inclui melhorias nos exemplos de metas universais, bem como outras pequenas melhorias em todo o guia. • the section called “Usando destinos universais”	
Informações atualizadas sobre agendamentos baseados em taxas que não têm uma data de início	Foram adicionadas informaçõ es sobre como o EventBrid ge Scheduler lida com agendamentos baseados em taxas se você não especificar um. `StartDate` • the section called “Agendamentos baseados em taxas”	17 de março de 2023
Novo tópico sobre gerenciam ento de grupos de agendador es	Foi adicionado um novo capítulo sobre como criar grupos de agendadores com o EventBridge Agendador . Use este capítulo para aprender como criar um grupo, adicionar agendas ao grupo, aplicar tags para gerenciar e monitorar mais facilment e os recursos do EventBrid ge Scheduler e, finalmente, excluir um grupo. • Gerenciando um grupo de agendamento	17 de março de 2023

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. • the section called “Horário de verão” • the section called “Fusos horários”	17 de novembro de 2022
Novo tópico sobre métricas	Foi adicionado um novo tópico que descreve as métricas nas quais o EventBridge Scheduler publica. CloudWatch Você pode usar essas métricas para monitorar falhas de invocação e entender como resolver problemas com seus agendamentos. • the section called “Monitoramento com CloudWatch”	15 de novembro de 2022
Lançamento 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.