



Guia do Desenvolvedor

AWS Serverless Application Repository



AWS Serverless Application Repository: Guia do Desenvolvedor

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

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

Table of Contents

O que é o AWS Serverless Application Repository?	1
Próximas etapas	1
Quick Start: Publishing Applications	3
Visão geral	3
Aplicativo Hello World	3
Antes de começar	4
Etapa 1: Inicializar o aplicativo	4
Etapa 2: Testar o aplicativo localmente	5
Etapa 3: Empacotar o aplicativo	6
Etapa 4: publicar o aplicativo	8
Próximas etapas	8
Mais informações	9
Publicar aplicativos	10
Como usar o AWS SAM com a AWS Serverless Application Repository	11
CompatívelAWSRecursos noAWS Serverless Application Repository	11
Modelos de política	12
Lista de compatíveisAWSRecursos	12
Como publicar aplicativos	19
Publicar um aplicativo (AWS CLI)	19
Publicar um novo aplicativo (console)	20
Compartilhar um aplicativo	25
Cancelar o compartilhamento de um aplicativo	28
Excluir um aplicativo	30
Publicar novas versões do aplicativo	30
Selo de autor verificado	32
Solicitar um selo de autor verificado	32
Compartilhar camadas do Lambda	33
Como funcionam	33
Exemplo	33
Como implantar aplicativos	35
Permissões de implantação do aplicativo	35
Recursos do aplicativo	36
Encontrar e confirmar recursos do aplicativo (console)	37
Visualizar recursos do aplicativo (AWS CLI)	37

Como implantar aplicativos	38
Implantar um novo aplicativo (console)	38
Implantar um novo aplicativo (AWS CLI)	39
Exclusão de pilhas do aplicativo	41
Atualizar aplicativos	41
Segurança	43
Proteção de dados	44
Criptografia em trânsito	45
Criptografia em repouso	45
Identity and Access Management	45
Público	46
Autenticação com identidades	46
Gerenciamento do acesso usando políticas	50
Como AWS Serverless Application Repository funciona com IAM	52
Exemplos de políticas baseadas em identidade	58
Exemplos de política de usuário	67
Referência de permissões da API do AWS Serverless Application Repository	73
Solução de problemas	76
Registro em log e monitoramento	79
Registro em log de chamadas à API do AWS Serverless Application Repository com o AWS CloudTrail	79
Validação de conformidade	83
Resiliência	84
Segurança da infraestrutura	84
Cotas	86
Solução de problemas	87
Não é possível tornar um aplicativo público	87
Uma cota foi excedida	88
Um arquivo Readme atualizado não aparece imediatamente	88
Você não pode implantar um aplicativo devido a permissões do IAM insuficientes	88
Não é possível implantar o mesmo aplicativo duas vezes	88
Por que meu aplicativo não está publicamente disponível?	89
Como entrar em contato com o Support	89
Operações	90
Recursos	92
Applications	92

URI	92
HTTPmétodos	92
Esquemas	94
Propriedades	98
Consulte também	116
Aplicações applicationId	117
URI	117
HTTPmétodos	117
Esquemas	121
Propriedades	124
Consulte também	138
Conjuntos de applicationId alterações de aplicativos	139
URI	139
HTTPmétodos	139
Esquemas	140
Propriedades	142
Consulte também	150
Applications applicationId Dependencies	151
URI	151
HTTPmétodos	151
Esquemas	153
Propriedades	155
Consulte também	158
applicationId Política de aplicativos	158
URI	158
HTTPmétodos	159
Esquemas	161
Propriedades	163
Consulte também	167
Applications applicationId Templates	168
URI	168
HTTPmétodos	168
Esquemas	169
Propriedades	171
Consulte também	175
Applications applicationId Templates templateId	176

URI	176
HTTPmétodos	176
Esquemas	178
Propriedades	179
Consulte também	183
Applications applicationId Unshare	184
URI	184
HTTPmétodos	184
Esquemas	185
Propriedades	187
Consulte também	189
applicationId Versões de aplicativos	190
URI	190
HTTPmétodos	190
Esquemas	192
Propriedades	193
Consulte também	197
applicationId Versões de aplicativos semanticVersion	198
URI	198
HTTPmétodos	198
Esquemas	199
Propriedades	202
Consulte também	211
Histórico do documento	212
Glossário da AWS	216
.....	ccxvii

O que é o AWS Serverless Application Repository?

O AWS Serverless Application Repository Com o, os desenvolvedores e as empresas podem encontrar, implantar e publicar de forma mais fácil e rápida aplicativos sem servidor no AWS Cloud. Para obter mais informações sobre aplicativos sem servidor, consulte [Computação e aplicativos sem servidor](#) no AWS Website.

É possível publicar aplicativos facilmente, compartilhá-los com a comunidade em geral ou de maneira particular entre membros da sua equipe ou em toda a organização. Para publicar um aplicativo sem servidor (ou aplicativo), é possível usar o AWS Management Console, o AWS SAM Interface da linha de comando do (AWS SAM ou AWS SDKs para carregar seu código. Juntamente com seu código, você pode fazer upload de um arquivo manifesto simples, também conhecido como modelo AWS Serverless Application Model (AWS SAM). Para obter mais informações sobre o AWS SAM, consulte o [Guia do desenvolvedor do AWS Serverless Application Model](#).

O AWS Serverless Application Repository está profundamente integrado com o console do AWS Lambda. A integração significa que os desenvolvedores de todos os níveis podem começar a usar a computação sem servidor sem necessidade de aprender coisas novas. Você pode usar palavras-chave de categoria para procurar aplicativos como back-ends da web e de dispositivos móveis, aplicativos de processamento de dados ou chatbots. Você também pode pesquisar aplicativos por nome, editor ou fonte de evento. Para usar um aplicativo, basta selecioná-lo, configurar os campos necessários e implantá-lo com alguns cliques.

Neste guia, mostraremos as duas maneiras de trabalhar com o AWS Serverless Application Repository:

- [Publicar aplicativos](#)— Configure e carregue aplicativos para disponibilizá-los a outros desenvolvedores e publique novas versões de aplicativos.
- [Como implantar aplicativos](#)— Procure aplicativos e visualize informações sobre eles, incluindo código-fonte e arquivos “readme”. Além disso, instale, configure e implante os aplicativos que você preferir.

Próximas etapas

- Para um tutorial sobre a publicação de um aplicativo de amostra no AWS Serverless Application Repository, consulte [Quick Start: Publishing Applications](#).

- Para obter instruções sobre como implantar aplicativos a partir do AWS Serverless Application Repository, consulte [Como implantar aplicativos](#).

Quick Start: Publishing Applications

Este guia orienta você pelas etapas de download, criação, teste e publicação de um aplicativo de exemplo sem servidor no AWS Serverless Application Repository usando a CLI do AWS SAM. É possível usar esse aplicativo de exemplo como ponto inicial para desenvolver e publicar o seu próprio aplicativo sem servidor.

Visão geral

As etapas a seguir definem como baixar, criar e publicar um exemplo de aplicativo sem servidor:

1. Inicializar. Faça download de um exemplo de aplicativo a partir do modelo usando `sam init`.
2. Testar localmente. Teste o aplicativo localmente usando `sam local invoke` e/ou `sam local start-api`. Observe que, com esses comandos, mesmo que sua função Lambda seja invocada localmente, ela lê e grava em AWS recursos na AWS nuvem.
3. Pacote. Quando estiver satisfeito com sua função do Lambda, agrupe a função, o AWS SAM modelo e quaisquer dependências do Lambda em um pacote de AWS CloudFormation implantação usando `sam package`. Nesta etapa você também incluirá as informações sobre o aplicativo que será carregado no AWS Serverless Application Repository.
4. Publicar. Publique o aplicativo no AWS Serverless Application Repository usando `sam publish`. Ao final desta etapa, você poderá visualizar seu aplicativo AWS Serverless Application Repository e implantá-lo na AWS nuvem usando AWS Serverless Application Repository.

O exemplo [Aplicativo Hello World](#) na seção a seguir guia você por essas etapas na criação e publicação de um aplicativo sem servidor.

Aplicativo Hello World

Neste exercício, baixe e teste um aplicativo Hello World sem servidor que representa um simples back-end da API. Consiste em um endpoint do Amazon API Gateway que suporta uma operação GET e uma função Lambda. Quando uma solicitação GET é enviada ao endpoint, o API Gateway invoca a função do Lambda. Depois, o AWS Lambda executará a função, que basicamente retornará uma mensagem `hello world`.

O aplicativo tem os seguintes componentes:

- Um AWS SAM modelo que define dois AWS recursos para o aplicativo Hello World: um serviço de API Gateway com uma operação GET e uma função Lambda. O modelo também define o mapeamento entre a operação GET do API Gateway e a função Lambda.
- Código do aplicativo gravado em Python.

Antes de começar

Certifique-se de que tenha a configuração necessária para este exercício:

- Você deve ter uma AWS conta com um usuário do IAM que tenha permissões administrativas. Consulte [Configurar uma AWS conta](#).
- É necessário que a CLI (interface da linha de comando) do AWS SAM tenha sido instalada. Consulte [Instalar a CLI do AWS SAM](#).
- É necessário ter a versão 1.16.77 ou posterior da AWS CLI instalada. Consulte [Instalar a AWS Command Line Interface](#).

Etapa 1: Inicializar o aplicativo

Nesta seção, baixe o exemplo de aplicativo, que consiste em um modelo do AWS SAM e no código do aplicativo.

Como inicializar o aplicativo

1. Execute o seguinte comando em um prompt de comando da CLI do AWS SAM.

```
sam init --runtime python3.6
```

2. Revise o conteúdo do diretório criado pelo comando (`sam-app/`):
 - `template.yaml`— Define dois AWS recursos que o aplicativo Hello World precisa: uma função Lambda e um endpoint do API Gateway que suporte uma operação GET. O modelo também define o mapeamento entre os dois recursos.
 - Conteúdo relacionado ao código do aplicativo Hello World:
 - `hello_world/diretório` — Contém o código do aplicativo, que retorna `hello world` quando você o executa.

Note

Para este exercício, o código do aplicativo está gravado em Python e você especifica o tempo de execução no comando `init`. O AWS Lambda oferece suporte a idiomas adicionais para criar o código do aplicativo. Se você especificar outro tempo de execução compatível, o comando `init` oferecerá o código Hello World no idioma especificado e um arquivo `README.md` que você poderá seguir no idioma correspondente. Para obter informações sobre tempos de execução compatíveis, consulte [Ambiente de execução do Lambda e bibliotecas disponíveis](#).

Etapa 2: Testar o aplicativo localmente

Agora que você tem o aplicativo do AWS SAM na máquina local, siga as etapas abaixo para testá-la localmente.

Como testar o aplicativo localmente

1. Inicie o endpoint do gateway da API localmente. É necessário executar o comando a seguir a partir do diretório que contém o arquivo `template.yaml`.

```
sam-app> sam local start-api --region us-east-1
```

O comando retorna um endpoint do API Gateway, para o qual você pode enviar solicitações para testes locais.

2. Testar o aplicativo Copie o URL do endpoint do API Gateway, cole-o no navegador e escolha Enter. Um exemplo de URL de endpoint do API Gateway é `http://127.0.0.1:3000/hello`.

O API Gateway invoca localmente a função Lambda para a qual o endpoint está mapeado. A função Lambda é executada no contêiner local do Docker e retorna `hello world`. O API Gateway retorna uma resposta ao navegador que contém o texto.

Exercício: alterar a string da mensagem.

Depois de testar o exemplo de aplicativo com êxito, é possível testar fazendo uma simples modificação: alterar a string da mensagem retornada.

1. Edite o arquivo `/hello_world/app.py` para alterar a string da mensagem de `'hello world'` para `'Hello World!'`.
2. Recarregue o URL de teste no navegador e observe a nova string.

Você notará que o novo código será carregado de forma dinâmica, sem precisar reiniciar o processo do `sam local`.

Etapa 3: Empacotar o aplicativo

Depois de testar o aplicativo localmente, use a CLI do AWS SAM para criar um pacote de implantação e um modelo do AWS SAM empacotado.

Note

Nas etapas a seguir, crie um arquivo `.zip` para o conteúdo do diretório `hello_world/`, que contém o código do aplicativo. Este arquivo `.zip` é o pacote de implantação para o seu aplicativo sem servidor. Para obter mais informações, consulte [Criando um Package de implantação \(Python\)](#) no Guia do AWS Lambda desenvolvedor.

Como Como

1. Adicione uma seção de Metadata ao arquivo do modelo do AWS SAM fornecendo as informações necessárias do aplicativo. Para obter mais informações sobre a seção Metadata dos modelos do AWS SAM, consulte [Propriedades da seção dos metadados do modelo do AWS SAM](#) no Guia do desenvolvedor do AWS Serverless Application Model.

Veja um exemplo da seção Metadata:

```
Metadata:
  AWS::ServerlessRepo::Application:
    Name: my-app
    Description: hello world
    Author: user1
    SpdxLicenseId: Apache-2.0
```

```
LicenseUrl: LICENSE.txt
ReadmeUrl: README.md
Labels: ['tests']
HomePageUrl: https://github.com/user1/my-app-project
SemanticVersion: 0.0.1
SourceCodeUrl: https://github.com/user1/my-app-project
```

`ReadmeUrl`As propriedades `LicenseUrl` and podem ser referências a arquivos locais (como no exemplo acima) ou podem ser links para buckets do Amazon S3 que já hospedam esses artefatos.

2. Crie um bucket do S3 no local onde deseja salvar o código empacotado. Se você quiser usar um bucket do S3 existente, ignore esta etapa.

```
sam-app> aws s3 mb s3://bucketname
```

3. Crie o pacote de implantação da função Lambda executando o seguinte comando `dapackageAWS SAM CLI`.

```
sam-app> sam package \
  --template-file template.yaml \
  --output-template-file packaged.yaml \
  --s3-bucket bucketname
```

O comando faz o seguinte:

- Compacta o conteúdo do `aws-sam/hello_world/` diretório e faz o upload para o Amazon S3.
- Carrega o pacote de implantação, o arquivo `README` e o arquivo `LICENSE` para o bucket do Amazon S3 especificado pela `--s3-bucket` opção.
- Gera um novo arquivo de modelo, chamado `packaged.yaml`, que será usado na próxima etapa para publicar o aplicativo no AWS Serverless Application Repository. O arquivo `depackaged.yaml` modelo é semelhante ao arquivo de modelo original (`template.yaml`), mas tem uma diferença fundamental: as `ReadmeUrl` propriedades `CodeUri` `LicenseUrl`, e apontam para o bucket do Amazon S3 e para os objetos que contêm os respectivos artefatos. O seguinte trecho de um arquivo de modelo de exemplo `packaged.yaml` mostra a propriedade `CodeUri`:

```
HelloWorldFunction:
```

```
Type: AWS::Serverless::Function # For more information about function
resources, see https://github.com/awslabs/serverless-application-model/blob/
master/versions/2016-10-31.md#awsserverlessfunction
```

```
Properties:
```

```
  CodeUri: s3://bucketname/fb77a3647a4f47a352fc0bjectGUID
```

```
...
```

Etapa 4: publicar o aplicativo

Agora que você criou o pacote de implantação, use-o para publicar o aplicativo no AWS Serverless Application Repository.

Como publicar o aplicativo sem servidor no AWS Serverless Application Repository

- Execute o comando a seguir para publicar o novo aplicativo no AWS Serverless Application Repository com a primeira versão criada como 0.0.1.

```
sam-app> sam publish \  
  --template packaged.yaml \  
  --region us-east-1
```

Note

O aplicativo será criado como privado por padrão. Você deve compartilhar o aplicativo antes que outras AWS contas possam visualizar e implantar seu aplicativo. Consulte Next Steps (Próximas etapas) abaixo para obter mais detalhes sobre como compartilhar o aplicativo.

Próximas etapas

Agora que você publicou o exemplo de aplicativo, veja algumas coisas que poderá fazer com ele.

- Exibir seu aplicativo em AWS Serverless Application Repository — A saída do `sam publish` comando incluirá um link AWS Serverless Application Repository diretamente para a página de detalhes do seu aplicativo. Também é possível ir para a página inicial do AWS Serverless Application Repository e pesquisar o aplicativo.

- **Compartilhe seu aplicativo** — Como seu aplicativo está definido como privado por padrão, ele não é visível para outras AWS contas. Para compartilhar sua inscrição com outras pessoas, você deve torná-la pública ou conceder permissão a uma lista específica de AWS contas. Para obter informações sobre como compartilhar o aplicativo usando a AWS CLI, consulte [AWS Serverless Application Repository Exemplos de políticas de aplicativos](#). Para obter informações sobre como compartilhar o aplicativo usando o console, consulte [Compartilhar um aplicativo](#).

Mais informações

Para obter mais informações sobre a seção Metadata dos modelos do AWS SAM, sobre os comandos `sam package` e `sam publish` da CLI do AWS SAM, consulte [Publicar aplicativos usando a CLI do AWS SAM](#) no Guia do desenvolvedor do AWS Serverless Application Model.

Publicar aplicativos

Quando você publica um aplicativo sem servidor no AWS Serverless Application Repository, ele será disponibilizado para que outros o encontrem e o implantem.

Primeiro você define o aplicativo com um modelo do AWS Serverless Application Model (AWS SAM). Ao definir o aplicativo, é necessário considerar se os consumidores do aplicativo precisarão confirmar os recursos dele. Para obter mais informações sobre como usar o AWS SAM e confirmar os recursos, consulte [Como usar o AWS SAM com a AWS Serverless Application Repository](#).

Você pode publicar aplicativos sem servidor usando o AWS Management Console, o AWS SAM Interface da linha de comando (AWS SAM CLI), ou um AWS SDK. Para saber mais sobre os procedimentos para publicar aplicativos no AWS Serverless Application Repository, consulte [Como publicar aplicativos](#).

Quando você publica seu aplicativo, ele é inicialmente definido como privado, o que significa que só está disponível para a sua conta AWS. Para compartilhar seu aplicativo com outras pessoas, você deve defini-lo como Compartilhado de modo privado (Compartilhado apenas com um conjunto específico de contas AWS), ou Publicly shared (compartilhado com todos).

Quando você publica um aplicativo no AWS Serverless Application Repository e o define como público, o serviço disponibilizará o aplicativo para consumidores de todas as regiões. Quando um consumidor implanta um aplicativo em uma região diferente daquela na qual o aplicativo foi publicado inicialmente, o AWS Serverless Application Repository copia os artefatos de implantação do aplicativo para um bucket do Amazon S3 na região de destino. Ele atualiza todos os recursos no AWS SAM Modelo que usa esses artefatos para referenciar os arquivos no bucket do Amazon S3 para a região de destino. Os artefatos de implantação podem incluir o código da função do Lambda, os arquivos de definição da API, entre outros.

Note

Private e Compartilhado de modo privado Os aplicativos só estão disponíveis na região AWS em que eles são criados. Publicly shared Os aplicativos estão disponíveis em todas as regiões AWS. Para saber mais sobre como compartilhar aplicativos, consulte [AWS Serverless Application Repository Exemplos de políticas de aplicativos](#).

Tópicos

- [Como usar o AWS SAM com a AWS Serverless Application Repository](#)
- [Como publicar aplicativos](#)
- [Selo de autor verificado](#)
- [Compartilhar camadas do Lambda](#)

Como usar o AWS SAM com a AWS Serverless Application Repository

O AWS Serverless Application Model (AWS SAM) é uma estrutura de código aberto que você pode usar para criar [aplicativos sem servidor](#) em AWS. Para obter mais informações sobre como usar AWS SAM para criar seu aplicativo sem servidor, consulte o Guia do desenvolvedor [AWS Serverless Application Model](#).

Ao criar aplicativos que serão publicados no AWS Serverless Application Repository, você deve considerar o conjunto de recursos suportados e modelos de políticas disponíveis para uso. As seções abaixo descrevem esses tópicos em mais detalhes.

Compatível com recursos no AWS Serverless Application Repository

O AWS Serverless Application Repository oferece suporte a aplicativos sem servidor compostos por vários recursos do AWS SAM e do AWS CloudFormation. Para ver a lista completa de recursos que são suportados por AWS Serverless Application Repository, consulte [Lista de recursos compatíveis](#).

Se você quiser solicitar suporte para um recurso adicional, contate [o suporte para AWS](#).

Important

Se o modelo de aplicativo contém uma das seguintes funções do IAM ou políticas de recursos personalizadas, o aplicativo não será exibido nos resultados da pesquisa por padrão. Além disso, os clientes devem reconhecer as funções do IAM e as políticas de recursos antes de implantá-lo. Para mais informações, consulte [Reconhecer recursos do aplicativo](#).

A lista de recursos ao que ele se aplica são:

- Funções do IAM: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), e [AWS::IAM::Role](#).

- Políticas de recursos: [AWS::Lambda::LayerVersionPermissão](#), [AWS::Lambda::Permission](#), [AWS::Events::EventBusPolítica](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoEscalabilidade::ScalingPolicy](#), [AWS::S3::BucketPolicy](#), [AWS::SQS::QueuePolicy](#), e [AWS::SNS::TopicPolicy](#).

Se o aplicativo contém o recurso [AWS::Serverless::Application](#), os clientes precisam confirmar que o aplicativo contém um aplicativo aninhado antes de poderem implantá-lo. Para mais informações sobre aplicativos aninhados, consulte [Aplicativos aninhados](#) no AWS Serverless Application Model Guia do desenvolvedor. Para mais informações sobre reconhecer recursos, consulte [Reconhecer recursos do aplicativo](#).

Modelos de política

AWS SAMO fornece uma lista de modelos de política para definir o escopo das permissões de funções do Lambda para os recursos que são usados pelo aplicativo. O uso de modelos de políticas não exige confirmações adicionais do cliente para pesquisar, navegar ou implantar o aplicativo.

Para ver a lista de padrão AWS SAM modelos de política, consulte [AWS SAM Modelos de política](#) no [AWS Serverless Application Model Guia do desenvolvedor](#).

Lista de compatíveis AWS Recursos

Esta é a lista completa de AWS recursos que são suportados pelo AWS Serverless Application Repository.

- `AWS::AccessAnalyzer::Analyzer`
- `AWS::AmazonMQ::Broker`
- `AWS::AmazonMQ::Configuration`
- `AWS::AmazonMQ::ConfigurationAssociation`
- `AWS::ApiGateway::Account`
- `AWS::ApiGateway::ApiKey`
- `AWS::ApiGateway::Authorizer`
- `AWS::ApiGateway::BasePathMapping`

- `AWS::ApiGateway::ClientCertificate`
- `AWS::ApiGateway::Deployment`
- `AWS::ApiGateway::DocumentationPart`
- `AWS::ApiGateway::DocumentationVersion`
- `AWS::ApiGateway::DomainName`
- `AWS::ApiGateway::GatewayResponse`
- `AWS::ApiGateway::Method`
- `AWS::ApiGateway::Model`
- `AWS::ApiGateway::RequestValidator`
- `AWS::ApiGateway::Resource`
- `AWS::ApiGateway::RestApi`
- `AWS::ApiGateway::Stage`
- `AWS::ApiGateway::UsagePlan`
- `AWS::ApiGateway::UsagePlanKey`
- `AWS::ApiGateway::VpcLink`
- `AWS::ApiGatewayV2::Api`
- `AWS::ApiGatewayV2::ApiMapping`
- `AWS::ApiGatewayV2::Authorizer`
- `AWS::ApiGatewayV2::DomainName`
- `AWS::ApiGatewayV2::Deployment`
- `AWS::ApiGatewayV2::Integration`
- `AWS::ApiGatewayV2::IntegrationResponse`
- `AWS::ApiGatewayV2::Model`
- `AWS::ApiGatewayV2::Route`
- `AWS::ApiGatewayV2::RouteResponse`
- `AWS::ApiGatewayV2::Stage`
- `AWS::AppSync::ApiKey`
- `AWS::AppSync::DataSource`
- `AWS::AppSync::GraphQLApi`
- `AWS::AppSync::GraphQLSchema`

- `AWS::AppSync::Resolver`
- `AWS::ApplicationAutoScaling::AutoScalingGroup`
- `AWS::ApplicationAutoScaling::LaunchConfiguration`
- `AWS::ApplicationAutoScaling::ScalableTarget`
- `AWS::ApplicationAutoScaling::ScalingPolicy`
- `AWS::Athena::NamedQuery`
- `AWS::Athena::WorkGroup`
- `AWS::CertificateManager::Certificate`
- `AWS::Chatbot::SlackChannelConfiguration`
- `AWS::CloudFormation::CustomResource`
- `AWS::CloudFormation::Interface`
- `AWS::CloudFormation::Macro`
- `AWS::CloudFormation::WaitConditionHandle`
- `AWS::CloudFront::CachePolicy`
- `AWS::CloudFront::CloudFrontOriginAccessIdentity`
- `AWS::CloudFront::Distribution`
- `AWS::CloudFront::Function`
- `AWS::CloudFront::OriginRequestPolicy`
- `AWS::CloudFront::ResponseHeadersPolicy`
- `AWS::CloudFront::StreamingDistribution`
- `AWS::CloudTrail::Trail`
- `AWS::CloudWatch::Alarm`
- `AWS::CloudWatch::AnomalyDetector`
- `AWS::CloudWatch::Dashboard`
- `AWS::CloudWatch::InsightRule`
- `AWS::CodeBuild::Project`
- `AWS::CodeCommit::Repository`
- `AWS::CodePipeline::CustomActionType`
- `AWS::CodePipeline::Pipeline`
- `AWS::CodePipeline::Webhook`

- `AWS::CodeStar::GitHubRepository`
- `AWS::CodeStarNotifications::NotificationRule`
- `AWS::Cognito::IdentityPool`
- `AWS::Cognito::IdentityPoolRoleAttachment`
- `AWS::Cognito::UserPool`
- `AWS::Cognito::UserPoolClient`
- `AWS::Cognito::UserPoolDomain`
- `AWS::Cognito::UserPoolGroup`
- `AWS::Cognito::UserPoolResourceServer`
- `AWS::Cognito::UserPoolUser`
- `AWS::Cognito::UserPoolUserToGroupAttachment`
- `AWS::Config::AggregationAuthorization`
- `AWS::Config::ConfigRule`
- `AWS::Config::ConfigurationAggregator`
- `AWS::Config::ConfigurationRecorder`
- `AWS::Config::DeliveryChannel`
- `AWS::Config::RemediationConfiguration`
- `AWS::DataPipeline::Pipeline`
- `AWS::DynamoDB::Table`
- `AWS::EC2::EIP`
- `AWS::EC2::InternetGateway`
- `AWS::EC2::NatGateway`
- `AWS::EC2::Route`
- `AWS::EC2::RouteTable`
- `AWS::EC2::SecurityGroup`
- `AWS::EC2::SecurityGroupEgress`
- `AWS::EC2::SecurityGroupIngress`
- `AWS::EC2::Subnet`
- `AWS::EC2::SubnetRouteTableAssociation`
- `AWS::EC2::VPC`

- AWS::EC2::VPCGatewayAttachment
- AWS::EC2::VPCPeeringConnection
- AWS::ECR::Repository
- AWS::Elasticsearch::Domain
- AWS::Events::EventBus
- AWS::Events::EventBusPolicy
- AWS::Events::Rule
- AWS::EventSchemas::Discoverer
- AWS::EventSchemas::Registry
- AWS::EventSchemas::Schema
- AWS::Glue::Classifier
- AWS::Glue::Connection
- AWS::Glue::Crawler
- AWS::Glue::Database
- AWS::Glue::DevEndpoint
- AWS::Glue::Job
- AWS::Glue::Partition
- AWS::Glue::SecurityConfiguration
- AWS::Glue::Table
- AWS::Glue::Trigger
- AWS::Glue::Workflow
- AWS::IAM::Group
- AWS::IAM::InstanceProfile
- AWS::IAM::ManagedPolicy
- AWS::IAM::OIDCProvider
- AWS::IAM::Policy
- AWS::IAM::Role
- AWS::IAM::ServiceLinkedRole
- AWS::IoT::Certificate
- AWS::IoT::Policy

- `AWS::IoT::PolicyPrincipalAttachment`
- `AWS::IoT::Thing`
- `AWS::IoT::ThingPrincipalAttachment`
- `AWS::IoT::TopicRule`
- `AWS::KMS::Alias`
- `AWS::KMS::Key`
- `AWS::Kinesis::Stream`
- `AWS::Kinesis::StreamConsumer`
- `AWS::Kinesis::Streams`
- `AWS::KinesisAnalytics::Application`
- `AWS::KinesisAnalytics::ApplicationOutput`
- `AWS::KinesisFirehose::DeliveryStream`
- `AWS::Lambda::Alias`
- `AWS::Lambda::EventInvokeConfig`
- `AWS::Lambda::EventSourceMapping`
- `AWS::Lambda::Function`
- `AWS::Lambda::LayerVersion`
- `AWS::Lambda::LayerVersionPermission`
- `AWS::Lambda::Permission`
- `AWS::Lambda::Version`
- `AWS::Location::GeofenceCollection`
- `AWS::Location::Map`
- `AWS::Location::PlaceIndex`
- `AWS::Location::RouteCalculator`
- `AWS::Location::Tracker`
- `AWS::Location::TrackerConsumer`
- `AWS::Logs::Destination`
- `AWS::Logs::LogGroup`
- `AWS::Logs::LogStream`
- `AWS::Logs::MetricFilter`

- `AWS::Logs::SubscriptionFilter`
- `AWS::Route53::HealthCheck`
- `AWS::Route53::HostedZone`
- `AWS::Route53::RecordSet`
- `AWS::Route53::RecordSetGroup`
- `AWS::S3::Bucket`
- `AWS::S3::BucketPolicy`
- `AWS::SNS::Subscription`
- `AWS::SNS::Topic`
- `AWS::SNS::TopicPolicy`
- `AWS::SQS::Queue`
- `AWS::SQS::QueuePolicy`
- `AWS::SSM::Association`
- `AWS::SSM::Document`
- `AWS::SSM::MaintenanceWindowTask`
- `AWS::SSM::Parameter`
- `AWS::SSM::PatchBaseline`
- `AWS::SSM::ResourceDataSync`
- `AWS::SecretsManager::ResourcePolicy`
- `AWS::SecretsManager::RotationSchedule`
- `AWS::SecretsManager::Secret`
- `AWS::SecretsManager::SecretTargetAttachment`
- `AWS::Serverless::Api`
- `AWS::Serverless::Application`
- `AWS::Serverless::Function`
- `AWS::Serverless::HttpApi`
- `AWS::Serverless::LayerVersion`
- `AWS::Serverless::SimpleTable`
- `AWS::Serverless::StateMachine`
- `AWS::ServiceDiscovery::HttpNamespace`

- `AWS::ServiceCatalog::CloudFormationProvisionedProduct`
- `AWS::ServiceDiscovery::Instance`
- `AWS::ServiceDiscovery::PrivateDnsNamespace`
- `AWS::ServiceDiscovery::PublicDnsNamespace`
- `AWS::ServiceDiscovery::Service`
- `AWS::SES::ReceiptRule`
- `AWS::SES::ReceiptRuleSet`
- `AWS::StepFunctions::Activity`
- `AWS::StepFunctions::StateMachine`
- `AWS::Wisdom::Assistant`
- `AWS::Wisdom::AssistantAssociation`
- `AWS::Wisdom::KnowledgeBase`

Como publicar aplicativos

Esta seção oferece procedimentos para publicar o aplicativo sem servidor no AWS Serverless Application Repository usando a CLI do AWS SAM ou o AWS Management Console. Ela também mostra como compartilhar seu aplicativo para permitir que outras pessoas o implantem e como excluir seu aplicativo do AWS Serverless Application Repository.

Important

As informações que você insere ao publicar um aplicativo não são criptografadas. Essas informações incluem dados como o nome do autor. Se você tiver informações de identificação pessoal que você não deseja que sejam armazenadas ou tornadas públicas, recomendamos que você não insira essas informações ao publicar o aplicativo.

Publicar um aplicativo (AWS CLI)

A maneira mais fácil de publicar um aplicativo no AWS Serverless Application Repository é usar um conjunto de comandos da CLI do AWS SAM. Para obter mais informações, consulte [Publicar um aplicativo usando a CLI do AWS SAM](#) no Guia do desenvolvedor do AWS Serverless Application Model (AWS SAM).

Publicar um novo aplicativo (console)

Esta seção mostra como usar o AWS Management Console para publicar um novo aplicativo no AWS Serverless Application Repository. Para obter instruções sobre como publicar uma nova versão de um aplicativo existente, consulte [Publicar a nova versão de um aplicativo existente](#).

Pré-requisitos

Antes de publicar um aplicativo no AWS Serverless Application Repository, você precisa do seguinte:

- Um válidoAWSconta.
- Um válidoAWS Serverless Application Model(AWS SAM) modelo que define oAWSRecursos que são usados. Para obter mais informações sobre os modelos do AWS SAM, consulte [Noções básicas sobre modelos do AWS SAM](#).
- Um pacote para o aplicativo que você criou usando o comando AWS CloudFormation do package para a AWS CLI. Este comando cria um pacote dos artefatos locais (caminhos locais) que são referências do seu modelo AWS SAM. Para obter mais detalhes, consulte [package \(pacote\)](#) na documentação do AWS CloudFormation.
- Um URL que aponta para o código-fonte do aplicativo, caso você queira torná-lo público.
- Um arquivo "readme.txt". Esse arquivo descreve como os clientes podem usar seu aplicativo e como configurá-lo antes de implantá-lo por conta própriaAWScontas.
- Um arquivo license.txt ou um identificador de licença válido do [site SPDX](#). Observe que uma licença é necessária somente se você quiser compartilhar o aplicativo publicamente. Se você quiser manter o aplicativo como privado ou apenas compartilhá-lo de forma privada, não será necessário especificar uma licença.
- Uma política de buckets do Amazon S3 válida que concede as permissões de leitura de serviço para os artefatos que foram carregados no Amazon S3 quando você criou o pacote do seu aplicativo. Para definir essa política, siga essas etapas:
 1. Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
 2. Escolha o bucket do Amazon S3 que você usou para empacotar seu aplicativo.
 3. Escolha a guia Permissions (Permissões).
 4. Clique no botão de Política de bucket.
 5. Cole a declaração da política a seguir no Bucket policy editor (Editor de política do bucket). Certifique-se de substituir o nome do bucket noResourceelemento e seuAWSUm ID da conta daConditionElemento. A expressão noConditionElementos garantemAWS Serverless

Application Repository só tem permissão para acessar aplicativos do especificado AWS conta. Para obter mais informações sobre declarações de política, consulte [Referência de elementos de política JSON do IAM](#) no Manual do usuário do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "serverlessrepo.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::bucketname/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

6. Clique no botão Salvar.

Procedimento

Crie um novo aplicativo no AWS Serverless Application Repository usando o seguinte procedimento.

Para criar um novo aplicativo no AWS Serverless Application Repository

1. Abra o console [AWS Serverless Application Repository](#) e escolha Publicar aplicativos.
2. Na página Publish an application (Publicar um aplicativo), insira as seguintes informações do aplicativo e escolha Publish application (Publicar aplicativo):

Propriedade	Obrigatório	Descrição
Application name (Nome do aplicativo)	TRUE	O nome do aplicativo.

Propriedade	Obrigatório	Descrição
		<p>Comprimento mínimo = 1. Tamanho máximo = 140.</p> <p>Padrão: "[a-zA-Z0-9\-\+]"</p>
Author (Autor)	TRUE	<p>O nome do autor que publica o aplicativo.</p> <p>Comprimento mínimo = 1. Tamanho máximo = 127.</p> <p>Padrão: "^[a-z0-9]([a-z0-9]-(?!-))*[a-z0-9]?\$";</p>
Home page (Página inicial)	FALSE	<p>Um URL com mais informações sobre o aplicativo — por exemplo, o local do GitHub repositório para o aplicativo.</p>
Descrição	TRUE	<p>A descrição do aplicativo.</p> <p>Comprimento mínimo = 1. Tamanho máximo = 256.</p>
Rótulos	FALSE	<p>Os rótulos que melhoram a descoberta de aplicativos em resultados de pesquisa.</p> <p>Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de rótulos: 10.</p> <p>Padrão: "^[a-zA-Z0-9+\-_:\V@]+\$";</p>

Propriedade	Obrigatório	Descrição
Spdx license (drop-down list) (Licença Spdx (lista suspensa))	FALSE	Escolha um identificador de licença válido no menu suspenso que contenha as licenças disponíveis no site SPDX . Escolher um item no menu suspenso preencher á a caixa de texto License (Licença) abaixo. Observação: Escolher uma licença no menu suspenso substituirá o conteúdo doLicense (Licença)Caixa de texto e descartará todas as edições manuais feitas.

Propriedade	Obrigatório	Descrição
License (Licença)	FALSE	<p>Carregue um arquivo de licença .txt ou escolha uma licença no menu suspenso Spdx license (Licença Spdx) descrito na linha anterior. Escolher uma licença do menu Spdx license (Licença Spdx) automaticamente preencherá a caixa de texto License (Licença). É possível editar o conteúdo dessa caixa de texto manualmente após carregar um arquivo de licença ou escolher um no menu suspenso Spdx license (Licença Spdx). No entanto, se outra Spdx license (Licença Spdx) for escolhida no menu suspenso, todas as edições manuais feitas serão descartadas.</p> <p>Esse é um campo opcional, mas é necessário fornecer uma licença para compartilhar o aplicativo publicamente.</p>

Propriedade	Obrigatório	Descrição
Readme (Leiamme)	FALSE	Carregue o conteúdo do arquivo Leiamme, que pode estar no formato de texto ou markdown. Esses conteúdos são exibidos na página de detalhes do aplicativo no AWS Serverless Application Repository. É possível editar manualmente o conteúdo dessa caixa de texto após carregar um arquivo.
Semantic version	FALSE	A versão semântica do aplicativo. Para obter mais informações, consulte o site do Versionamento Semântico . Forneça um valor para esta propriedade para tornar o aplicativo público.
Source code URL (URL do código-fonte)	FALSE	Um link para um repositório público para o código-fonte do aplicativo.
SAM template (Modelo do SAM)	TRUE	Um válidoAWS Serverless Application Model(AWS SAM) modelo que define oAWSRecursos que são usados.

Compartilhar um aplicativo

Aplicativos publicados podem ter permissões definidas em uma das três categorias a seguir:

- Privado (padrão)— aplicativos que foram criados com a mesma conta e não foram compartilhados com outras AWS conta. Somente consumidores que compartilham seu AWS Conta tem permissão para implantar aplicativos privados.
- Compartilhado de forma privada— Aplicativos que o editor compartilhou explicitamente com um conjunto específico de AWS contas ou com AWS contas em um AWS Organização. Os consumidores têm permissão para implantar aplicativos que foram compartilhados com seus AWS conta ou AWS Organização. Para obter mais informações sobre o AWS Organizations, consulte o Guia do usuário do [AWS Organizations](#).
- Compartilhado publicamente— Aplicativos que o editor compartilhou com todos. Todos os consumidores têm permissão para implantar um aplicativo compartilhado publicamente.

Depois que você publica um aplicativo no AWS Serverless Application Repository, por padrão, ele é definido como privado. Esta seção mostra como compartilhar um aplicativo de modo privado com o específico AWS contas ou um AWS Organize ou compartilhe publicamente com todos.

Compartilhar um aplicativo por meio do console

Você tem duas opções para compartilhar seu aplicativo com outras pessoas: 1) Compartilhe com específico AWS contas ou o AWS contas dentro do seu AWS organização, ou 2) Compartilhe publicamente com todos. Para obter mais informações sobre o AWS Organizations, consulte o Guia do usuário do [AWS Organizations](#).

Opção 1: Para compartilhar seu aplicativo com específico AWS conta (s) ou contas dentro da sua AWS organização

1. Abra o [console do AWS Serverless Application Repository](#).
2. No painel de navegação, escolha Published Applications (Aplicativos publicados) para ver a lista de aplicativos que você criou.
3. Escolha o aplicativo que você quer compartilhar.
4. Escolha a guia Sharing (Compartilhamento).
5. Na seção Application policy statements (Declarações de política de aplicativo), escolha o botão Create Statement (Criar declaração).
6. Na janela Statement Configuration (Configuração da declaração) preencha os campos com base em como você deseja compartilhar seu aplicativo.

Note

Se você estiver compartilhando com uma organização, só será possível especificar a organização que seu AWS conta é membro de. Se você tentar especificar uma AWS organização da qual você não é membro, ocorrerá um erro.

Para compartilhar seu aplicativo com sua AWS organização, você deve reconhecer que o `UnshareApplication` ação será adicionada à declaração de política, caso o compartilhamento precise ser revogado no futuro.

7. Clique no botão Salvar.

Opção 2: Para compartilhar seu aplicativo publicamente com todos

1. Abra o [console do AWS Serverless Application Repository](#).
2. No painel de navegação, escolha Published Applications (Aplicativos publicados) para ver a lista de aplicativos que você criou.
3. Escolha o aplicativo que você quer compartilhar.
4. Escolha a guia Sharing (Compartilhamento).
5. Na seção Public Sharing (Compartilhamento público) escolha o botão Edit (Editar).
6. Em Public sharing (Compartilhamento público) escolha o botão de opção Enabled (Habilitado).
7. Na caixa de texto digite o nome do aplicativo e escolha o botão Save (Salvar).

Note

Para compartilhar um aplicativo publicamente, ele deve ter as propriedades `LicenseUrl` e `SemanticVersion` definidas.

Compartilhar um aplicativo por meio da AWS CLI

Para compartilhar uma aplicação usando a AWS CLI você concede permissões usando o `put-application-policy` comando para especificar o AWS Conta (s) com as quais você deseja compartilhar como diretores.

Para obter mais informações sobre como compartilhar seu aplicativo usando a AWS CLI, consulte [AWS Serverless Application Repository Exemplos de políticas de aplicativos](#).

Cancelar o compartilhamento de um aplicativo

Há duas opções para cancelar o compartilhamento de um aplicativo de uma AWS Organização:

1. O editor do aplicativo pode remover permissões usando o comando [put-application-policy](#).
2. Um usuário de uma conta de gerenciamento de uma AWS organização pode executar um [cancelar compartilhamento de aplicativo](#) Operation em qualquer aplicativo compartilhado com a organização, mesmo que o aplicativo tenha sido publicado por um usuário de uma conta diferente.

Note

Quando um aplicativo não é compartilhado de uma AWS Organização com a operação “cancelar compartilhamento de aplicativo”, ela não pode ser compartilhada com uma AWS Organização novamente.

Para obter mais informações sobre o AWS Organizations, consulte o Guia do usuário do [AWS Organizations](#).

Editor remove permissões

Editor remove permissões pelo console

Para cancelar o compartilhamento de um aplicativo por meio do AWS Management Console, remova a declaração de política que a compartilha com outras AWS contas. Para isso, siga estas etapas:

1. Abra o [console do AWS Serverless Application Repository](#).
2. Escolha Available Applications (Aplicativos disponíveis) no painel de navegação esquerdo.
3. Escolha o aplicativo do qual deseja cancelar o compartilhamento.
4. Escolha a guia Sharing (Compartilhamento).
5. Na seção Application policy statements (Declarações de política de aplicativo) selecione a declaração de política que está compartilhando o aplicativo com as contas das quais você deseja cancelar o compartilhamento.
6. Escolha Delete (Excluir).

7. Uma mensagem de confirmação será exibida. Escolha Delete (Excluir) novamente.

Editor remove permissões pela AWS CLI

Para cancelar o compartilhamento de um aplicativo por meio do AWS CLI, o editor pode remover ou alterar permissões usando o `put-application-policy` comando para tornar o aplicativo privado ou compartilhar com um conjunto diferente de AWS contas.

Para obter mais informações sobre como alterar permissões usando a AWS CLI, consulte [AWS Serverless Application Repository Exemplos de políticas de aplicativos](#).

Conta de gerenciamento cancelando o compartilhamento de um aplicativo

Conta de gerenciamento cancelando o compartilhamento de um aplicativo de uma AWS Organização pelo console do

Para cancelar o compartilhamento de um aplicativo de uma AWS Organização através do AWS Management Console, um usuário da conta de gerenciamento O pode fazer o seguinte:

1. Abra o [console do AWS Serverless Application Repository](#).
2. Escolha Available Applications (Aplicativos disponíveis) no painel de navegação esquerdo.
3. No bloco do aplicativo, escolha Unshare (Cancelar compartilhamento).
4. Na caixa de mensagem de cancelamento de compartilhamento, confirme que deseja cancelar o compartilhamento do aplicativo inserindo o ID da organização e o nome do aplicativo e escolhendo Save (Salvar).

Conta de gerenciamento cancelando o compartilhamento de um aplicativo de uma AWS Organização através do AWS CLI

Para cancelar o compartilhamento de um aplicativo de uma AWS Organização, um usuário da conta de gerenciamento O pode executar o `aws serverlessrepo unshare-application` comando.

O comando a seguir descompartilha um aplicativo de uma AWS Organização, onde *Application-id* É o nome de recurso da Amazon (ARN) do aplicativo e *id da organização* é o AWS ID da organização:

```
aws serverlessrepo unshare-application --application-id application-id --organization-id organization-id
```

Excluir um aplicativo

É possível excluir aplicativos do AWS Serverless Application Repository usando o AWS Management Console ou a CLI do AWS SAM.

Excluir um aplicativo (console)

Para excluir um aplicativo publicado por meio do AWS Management Console, faça o seguinte.

1. Abra o [console do AWS Serverless Application Repository](#).
2. Em My Applications (Meus aplicativos), escolha o aplicativo a ser excluído.
3. Na página de detalhes do aplicativo, escolha Delete application (Excluir aplicativo).
4. Escolha Delete application (Excluir aplicativo) para concluir a exclusão.

Excluir um aplicativo (AWS CLI)

Para excluir um aplicativo publicado usando a AWS CLI, execute o comando [aws serverlessrepo delete-application](#).

O comando a seguir exclui um aplicativo, onde *application-id* é o nome de recurso da Amazon (ARN) do aplicativo:

```
aws serverlessrepo delete-application --application-id application-id
```

Publicar a nova versão de um aplicativo existente

Esta seção mostra como publicar uma nova versão de um aplicativo existente no AWS Serverless Application Repository usando a CLI do AWS SAM ou o AWS Management Console. Para obter instruções sobre como publicar um novo aplicativo, consulte [Como publicar aplicativos](#).

Publicar a nova versão de um aplicativo existente (AWS CLI)

A maneira mais fácil de publicar uma nova versão de um aplicativo existente é usar um conjunto de comandos da CLI do AWS SAM. Para obter mais informações, consulte [Publicar um aplicativo usando a CLI do AWS SAM](#) no Guia do desenvolvedor do AWS Serverless Application Model (AWS SAM).

Publicar a nova versão de um aplicativo existente (console)

Para publicar uma nova versão de um aplicativo que você publicou anteriormente, siga estas etapas:

1. Abra o [console do AWS Serverless Application Repository](#).
2. No painel de navegação, escolha My Applications (Meus aplicativos) para ver a lista de aplicativos que você criou.
3. Escolha o aplicativo para o qual você quer publicar uma nova versão.
4. Escolha Publish new version (Publicar nova versão).
5. Em Versions (Versões), insira as informações do aplicativo:

Propriedade	Obrigatório	Descrição
Semantic version	TRUE	A versão semântica do aplicativo. Para obter mais informações, consulte o site do Versionamento Semântico . Forneça um valor para esta propriedade para tornar o aplicativo público.
Source code URL (URL do código-fonte)	FALSE	Um link para um repositório público para o código-fonte do aplicativo.
SAM template (Modelo do SAM)	TRUE	Um válidoAWS Serverless Application Model(AWS SAM) modelo que define oAWSRecursos que são usados.

6. Escolha Publish version (Publicar versão).

Selo de autor verificado

Autores verificados no AWS Serverless Application Repository são aqueles para os quais a AWS é uma análise de boa-fé, como provedor de serviços coerente e prudente, das informações fornecidas pelo solicitante e confirma que a identidade do solicitante é verdadeira.

Os aplicativos de autores verificados exibem um selo de autor verificado, junto com um link para o perfil público do autor. O selo de autor verificado é exibido tanto nos resultados de pesquisa quanto na página de detalhes do aplicativo.

Solicitar um selo de autor verificado

É possível solicitar a aprovação como autor verificado no AWS Serverless Application Repository enviando um e-mail para serverlessrepo-verified-author@amazon.com. Você precisará fornecer as seguintes informações:

- Nome do autor
- AWSID da conta da
- Link do perfil publicamente acessível, como o GitHub ou LinkedIn perfil

Depois de enviar uma solicitação para obter um selo de autor verificado, a resposta da AWS dentro de alguns dias. É possível ser solicitado a enviar informações adicionais antes da solicitação ser aprovada.

Após a solicitação ser aprovada, o selo de autor verificado será exibido para seus aplicativos em um dia.

Note

O selo de autor verificado será exibido para todos os aplicativos que correspondem aos dois AWS nome da conta e do autor. Como as contas da AWS podem ter vários autores, os selos não são exibidos em aplicativos com um nome de autor diferente. Para que os selos de autores sejam exibidos nos aplicativos com diferentes nomes de autor, é necessário enviar outra solicitação para esse autor.

Compartilhar camadas do Lambda

Se você implementou a funcionalidade em uma camada do Lambda, talvez queira compartilhar a camada sem hospedar uma instância global dela. Compartilhar camadas dessa forma permite que outros implantem uma instância da camada na própria conta. Isso impede que os aplicativos cliente dependam de uma instância global da camada. O AWS Serverless Application Repository permite compartilhar camadas do Lambda facilmente dessa forma.

Para obter mais informações sobre camadas do Lambda, consulte [AWS Lambda Camadas](#) no AWS Lambda Guia do desenvolvedor.

Como funcionam

Veja a seguir as etapas para compartilhar a camada usando o AWS Serverless Application Repository. Isso permite que uma cópia da camada seja criada na sua conta.

1. Defina um aplicativo sem servidor com um AWS SAM Modelo do que inclua a camada como recurso — ou seja, um [AWS::Serverless::LayerVersion](#) ou um [AWS::Lambda::LayerVersion](#) recurso.
2. Publique o aplicativo no AWS Serverless Application Repository e compartilhe-o (pública ou privadamente).
3. Um cliente implanta o aplicativo, o que cria uma cópia da camada por conta própria na sua conta. O cliente pode referenciar o nome de recurso da Amazon (ARN) da camada na sua conta no aplicativo cliente.

Exemplo

Veja a seguir um exemplo de um AWS SAM Modelo para um aplicativo que contém a camada do Lambda que deseja compartilhar:

```
Resources:
  SharedLayer:
    Type: AWS::Serverless::LayerVersion
    Properties:
      LayerName: shared-layer
      ContentUri: source/layer-code/
      CompatibleRuntimes:
        - python3.7
```

Outputs:**LayerArn:**

Value: !Ref SharedLayer

Quando um cliente implanta o aplicativo a partir do AWS Serverless Application Repository, uma camada é criada em sua conta AWS. O ARN da camada se parece com o seguinte:

```
arn:aws:lambda:us-east-1:012345678901:layer:shared-layer:1
```

O cliente pode referenciar esse ARN para o próprio aplicativo cliente, como neste exemplo:

Resources:**MyFunction:**

Type: AWS::Serverless::Function

Properties:

Handler: index.handler

Runtime: python3.7

CodeUrl: source/app-code/

Layers:

- arn:aws:lambda:us-east-1:012345678901:layer:shared-layer:1

Como implantar aplicativos

Esta seção ajuda você a aprender a localizar e implantar aplicativos sem servidor que foram publicados no AWS Serverless Application Repository. É possível procurar aplicativos que estão disponíveis publicamente sem ter uma AWS conta visitando o [site público](#). Se preferir, você pode pesquisar aplicativos no console do AWS Lambda.

Alguns aplicativos têm um selo de autor verificado, com um link para o perfil do autor. Um autor é considerado um autor verificado quando o AWS fez uma análise de boa-fé, como provedor de serviços coerente e prudente, das informações fornecidas pelo solicitante e confirma que a identidade do solicitante é verdadeira.

Antes de implantar aplicativos do AWS Serverless Application Repository, consulte os seguintes tópicos para saber mais sobre as permissões de implantação e os recursos do aplicativo.

Tópicos

- [Permissões de implantação do aplicativo](#)
- [Recursos do aplicativo: funções do IAM, políticas de recursos e aplicativos aninhados](#)
- [Como implantar aplicativos](#)

Permissões de implantação do aplicativo

Para implantar um aplicativo no AWS Serverless Application Repository, é necessário ter uma permissão. Existem três categorias de aplicativos para os quais você tem permissão para implantar:

- **Private**— Os aplicativos que foram criados com a mesma conta e não foram compartilhados com outras contas. Você tem permissão para implantar aplicativos que foram criados usando a AWS conta.
- **Compartilhado de forma privada**— Compartilhado explicitamente aplicativos que o editor compartilhou com um conjunto específico de AWS contas. Você tem permissão para implantar aplicativos que foram compartilhados com a sua AWS conta.
- **Compartilhado publicamente**— Compartilhado pelo editor aplicativos que o editor compartilhou com todos. Você tem permissão para implantar aplicativos compartilhados publicamente.

É possível apenas pesquisar e explorar os aplicativos para os quais tem permissão. Isso inclui aplicativos que foram criados usando o AWS conta compartilhada em particular com sua AWS conta e compartilhada publicamente. Nenhum dos outros aplicativos será exibido para você.

Important

Os aplicativos que contêm aplicativos aninhados herdam as restrições de compartilhamento dos aplicativos aninhados. Por exemplo, suponha que um aplicativo seja compartilhado publicamente, mas contém um aplicativo aninhado que é compartilhado apenas de forma privada com o aplicativo AWS conta que criou o aplicativo pai. Nesse caso, se o seu AWS conta não tem permissão para implantar o aplicativo aninhado e não é possível implantar o aplicativo pai. Para mais informações sobre aplicativos aninhados, consulte [Aplicativos aninhados](#) no AWS Serverless Application Model Guia do desenvolvedor.

Recursos do aplicativo: funções do IAM, políticas de recursos e aplicativos aninhados

Antes de implantar um aplicativo, o AWS Serverless Application Repository verifica o modelo do aplicativo em busca de funções do IAM, políticas de recursos e aplicativos aninhados que o modelo especifica que ele deve criar. Os recursos do IAM, como uma função do IAM com acesso completo, podem modificar qualquer recurso na sua AWS conta. Portanto, recomendamos que você revise as permissões associadas ao aplicativo antes de prosseguir, para não criar recursos acidentalmente com permissões escalonadas. Para garantir que você fez isso, confirme que o aplicativo possui os recursos antes que o AWS Serverless Application Repository implante o aplicativo por você.

Os aplicativos podem conter qualquer um dos quatro seguintes recursos: `CAPABILITY_IAM`, `CAPABILITY_NAMED_IAM`, `CAPABILITY_RESOURCE_POLICY` e `CAPABILITY_AUTO_EXPAND`.

Os seguintes recursos exigem que você especifique `CAPABILITY_IAM` ou `CAPABILITY_NAMED_IAM`: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), [AWS::IAM::Role](#) e. Se o aplicativo contém recursos do IAM com nomes personalizados, você deverá especificar `CAPABILITY_NAMED_IAM`. Para obter um exemplo de como especificar recursos, consulte [Encontrar e confirmar recursos do aplicativo \(AWS CLI\)](#).

Os seguintes recursos exigem que você especifique `CAPABILITY_RESOURCE_POLICY`: [AWS::Lambda::LayerVersion](#), [AWS::Lambda::Permission](#), [AWS::Events::EventBus](#), [Política](#),

[AWS: :iam:Policy](#) [AWS::ApplicationAutoScaling::ScalingPolicy](#) [AWS::S3::BucketPolicy](#), [AWS::SQS::QueuePolicy](#), [AWS::SNS::TopicPolicy](#) e.

Os aplicativos que contêm um ou mais aplicativos aninhados exigem que você especifique `CAPABILITY_AUTO_EXPAND`. Para mais informações sobre aplicativos aninhados, consulte [Aplicativos aninhados](#) no AWS Serverless Application Model Guia do desenvolvedor.

Encontrar e confirmar recursos do aplicativo (console)

Você pode encontrar aplicativos disponíveis no AWS Serverless Application Repository no [AWS Serverless Application Repository site](#) ou por meio do [console do Lambda \(na página Criar função, abaixo da AWS Serverless Application Repository guia\)](#).

Os aplicativos que exigem a confirmação de recursos para criar funções do IAM ou políticas de recursos personalizadas não aparecem em resultados de pesquisa por padrão. Para pesquisar aplicativos que contenham esses recursos, marque a caixa de seleção `Show apps that create custom IAM roles or resource policies` (Mostrar aplicativos que criam funções do IAM ou políticas de recursos personalizadas).

Você pode revisar os recursos de um aplicativo na guia `Permissões` ao selecionar o aplicativo. Para implantar o aplicativo, você precisa selecionar `I acknowledge this application creates custom IAM roles or resource policies` (Entendo que esse aplicativo cria funções do IAM ou políticas de recursos personalizadas). Se você não reconhecer esses recursos, verá esta mensagem de erro: `Confirmação necessária`. Para implantar, marque a caixa na seção `Configurar parâmetros do aplicativo`.

Visualizar recursos do aplicativo (AWS CLI)

Para visualizar os recursos de um aplicativo usando a AWS CLI, primeiro você precisa do nome de recurso da Amazon (ARN). Depois, poderá executar o seguinte comando:

```
aws serverlessrepo get-application \  
--application-id application-arn
```

A resposta [requiredCapabilities](#) contém uma lista de recursos do aplicativo que você precisará reconhecer antes de implantar o aplicativo. Observe que se a propriedade [requiredCapabilities](#) estiver vazia, o aplicativo não possui os recursos necessários.

Como implantar aplicativos

Esta seção oferece procedimentos para implantar aplicativos sem servidor no AWS Serverless Application Repository usando o AWS Management Console ou a AWS CLI.

Implantar um novo aplicativo (console)

Esta seção mostra como implantar um novo aplicativo do AWS Serverless Application Repository usando o AWS Management Console. Para obter instruções sobre como implantar uma nova versão de um aplicativo existente, consulte [Atualizar aplicativos](#).

Navegar, pesquisar e implantar aplicativos

Encontre, configure e implemente um aplicativo no AWS Serverless Application Repository usando o seguinte procedimento.

Para encontrar e configurar um aplicativo no AWS Serverless Application Repository

1. Abra a [página inicial pública do AWS Serverless Application Repository](#) ou abra o [AWS Lambda console](#). Escolha Create function (Criar função) e selecione Browse serverless app repository (Navegar repositório de aplicativos sem servidor).
2. Pesquise um navegue até um aplicativo.

Note

Para mostrar os aplicativos que contêm funções do IAM ou políticas de recursos personalizadas, selecione o Mostrar aplicativos que criam funções do IAM ou políticas de recursos personalizadas na caixa de seleção . Para obter mais informações sobre funções do IAM e políticas de recursos personalizadas, consulte [Reconhecer recursos do aplicativo](#).

3. Escolha um aplicativo para ver detalhes como suas permissões, recursos e o número de vezes que ele foi implantado pelosAWS clientes.

As contagens de implantação são mostradas para aAWS região em que você está tentando implantar o aplicativo.

4. Na página de detalhes do aplicativo, visualize as permissões e os recursos do aplicativo visualizando o modeloAWS SAM, a licença e o arquivo Leia-me. Nesta página, você também pode encontrar o link Source code URL (URL do código-fonte) de aplicativos que são

compartilhados publicamente. Se o aplicativo inclui algum aplicativo aninhado, você também pode visualizar os detalhes desses aplicativos nesta página.

5. Configure o aplicativo na seção Configurações do aplicativo. Para obter orientações sobre como configurar um aplicativo específico, consulte o arquivo "readme" do aplicativo em questão.

Por exemplo, os requisitos de configuração podem incluir a especificação do nome de um recurso ao qual você deseja que o aplicativo tenha acesso. Esse recurso pode ser uma tabela do Amazon DynamoDB, um bucket do Amazon S3 ou uma API do Amazon API Gateway.

6. Escolha Implantar. Isso direcionará você para a página Deployment status.

Note

Se o aplicativo tiver recursos que exigem confirmação, marque a caixa de seleção I acknowledge this application creates custom IAM roles or resource policies (Eu entendo que este aplicativo cria funções do IAM ou políticas de recursos personalizadas) antes de implantar o aplicativo. Caso contrário, ocorrerá um erro. Para obter mais informações sobre funções do IAM e políticas de recursos personalizadas, consulte [Reconhecer recursos do aplicativo](#).

7. Na página Deployment status (Status da implantação) você pode visualizar o progresso da implantação. Enquanto aguarda a conclusão da implantação, você pode pesquisar e procurar outros aplicativos e retornar a esta página por meio do console do Lambda.

Depois que seu aplicativo for implantado com êxito, você poderá revisar e gerenciar os recursos que foram criados usando AWS as ferramentas existentes.

Implantar um novo aplicativo (AWS CLI)

Esta seção mostra como implantar um novo aplicativo do AWS Serverless Application Repository usando a AWS CLI. Para obter instruções sobre como implantar uma nova versão de um aplicativo existente, consulte [Atualizar aplicativos](#).

Encontrar e confirmar recursos do aplicativo (AWS CLI)

Para confirmar os recursos de um aplicativo usando o AWS CLI, siga estas etapas:

1. Analise os recursos do aplicativo. Use o AWS CLI comando a seguir para revisar os recursos de um aplicativo:

```
aws serverlessrepo get-application \  
--application-id application-arn
```

A resposta [requiredCapabilities](#) contém uma lista de recursos do aplicativo que você precisará reconhecer antes de implantar o aplicativo. Você também pode usar a [GetApplication API](#) nos AWS SDKs para obter esses dados.

2. Crie o conjunto de alterações. Você deve fornecer o conjunto de [recursos](#) necessários ao criar o AWS CloudFormation conjunto de alterações. Por exemplo, use o comando a seguir AWS CLI para implantar um aplicativo ao reconhecer os recursos:

```
aws serverlessrepo create-cloud-formation-change-set \  
--application-id application-arn \  
--stack-name unique-name-for-cloud-formation-stack \  
--capabilities list-of-capabilities
```

O ID do conjunto de alterações é retornado quando esse comando for executado com êxito. Você precisa do ID do conjunto de alterações para ir para a próxima etapa. Você também pode usar a [CreateCloudFormationChangeSet API](#) nos AWS SDKs para criar o conjunto de alterações.

Por exemplo, o AWS CLI comando a seguir reconhece um aplicativo que contém um [AWS::IAM::Role](#) recurso com um nome personalizado e um ou mais aplicativos aninhados:

```
aws serverlessrepo create-cloud-formation-change-set \  
--application-id application-arn \  
--stack-name unique-name-for-cloud-formation-stack \  
--capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND
```

3. Execute o conjunto de alterações. Executar o conjunto de alterações realiza a implantação. Forneça o ID do conjunto de alterações que foi retornado ao criar o conjunto de alterações na etapa anterior.

O comando da AWS CLI do exemplo a seguir executa o conjunto de alterações do aplicativo para implantar o aplicativo:

```
aws cloudformation execute-change-set \  
--change-set-name changeset-id-arn
```

Você também pode usar a [ExecuteChangeSet API](#) nos AWS SDKs para executar o conjunto de alterações.

Exclusão de pilhas do aplicativo

Para excluir um aplicativo implantado anteriormente usando o AWS Serverless Application Repository, siga o mesmo procedimento para excluir uma pilha do AWS CloudFormation:

- **AWS Management Console:** Para excluir um aplicativo usando o AWS Management Console, consulte [Excluindo uma pilha no AWS CloudFormation console](#) no Guia do AWS CloudFormation usuário.
- **AWS CLI:** Para excluir um aplicativo usando o AWS CLI, consulte [Excluindo uma pilha](#) no Guia do AWS CloudFormation usuário.

Atualizar aplicativos

Após implantar um aplicativo do AWS Serverless Application Repository, talvez seja necessário atualizá-lo. Por exemplo, talvez queira alterar a configuração de um aplicativo, ou atualizar o aplicativo para a versão mais recente publicada.

As seções a seguir descrevem como implantar uma nova versão de um aplicativo ao inserir o AWS Management Console ou a AWS CLI.

Atualizar aplicativos (console)

Para atualizar um aplicativo implantado previamente, use o mesmo procedimento usado para implantar um novo aplicativo e forneça o mesmo nome de aplicativo que foi usado originalmente para implantá-lo. Em particular, o AWS Serverless Application Repository anexa o `serverlessrepo-` ao nome do aplicativo. No entanto, para implantar uma nova versão do aplicativo, forneça o nome do aplicativo original sem o `serverlessrepo-` anexado.

Por exemplo, se você implantou um aplicativo com o nome `MyApplication`, o nome da pilha será `serverlessrepo-MyApplication`. Para atualizar esse aplicativo, você forneceria o nome `MyApplication` novamente — não especifique o nome completo da pilha `serverlessrepo-MyApplication`.

Para todas as demais configurações de aplicativo, é possível manter os valores iguais aos da implantação anterior ou fornecer novos valores.

Atualizar aplicativos (AWS CLI)

Para atualizar um aplicativo implantado previamente, use o mesmo procedimento usado para implantar um novo aplicativo e forneça o mesmo `--stack-name` que foi usado originalmente para implantá-lo. Em particular, o AWS Serverless Application Repository anexa o `serverlessrepo-` ao nome da pilha. No entanto, para implantar uma nova versão do aplicativo, forneça o nome da pilha original sem o `serverlessrepo-` anexado.

Por exemplo, se você implantou um aplicativo com o nome da pilha `MyApplication`, o nome da pilha criada será `serverlessrepo-MyApplication`. Para atualizar esse aplicativo, você forneceria o nome `MyApplication` novamente — não especifique o nome completo da pilha `deserverlessrepo-MyApplication`.

Segurança no AWS Serverless Application Repository

A segurança da nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se contará com um datacenter e uma arquitetura de rede criados para atender aos requisitos das organizações com as maiores exigências de segurança.

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

- Segurança da nuvem: a AWS é responsável pela proteção da infraestrutura que executa produtos da AWS na Nuvem AWS. A AWS também fornece serviços que podem ser usados com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos [programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam ao AWS Serverless Application Repository, consulte [Serviços da AWS em escopo por programa de conformidade](#).
- Segurança da nuvem: sua responsabilidade é determinada pelo serviço da AWS que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o AWS Serverless Application Repository. Os tópicos a seguir mostram como configurar a AWS Serverless Application Repository para atender aos seus objetivos de segurança e conformidade. Saiba também como usar outros produtos da AWS que ajudam a monitorar e proteger os recursos do AWS Serverless Application Repository.

Tópicos

- [Proteção de dados no AWS Serverless Application Repository](#)
- [Identity and Access Management para o AWS Serverless Application Repository](#)
- [Registro em log e monitoramento no AWS Serverless Application Repository](#)
- [Validação de conformidade do AWS Serverless Application Repository](#)
- [Resiliência no AWS Serverless Application Repository](#)
- [Segurança de infraestrutura no AWS Serverless Application Repository](#)

Proteção de dados no AWS Serverless Application Repository

A ferramenta AWS modelo de [responsabilidade compartilhada modelo](#) se aplica à proteção de dados em AWS Serverless Application Repository. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todas as Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que você 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 [AWS Modelo de responsabilidade compartilhada e postagem no GDPR](#) blog sobre o AWS Blog de segurança.

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS 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.
- Use SSL/TLS para se comunicar com AWS recursos. Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Configure API e registre as atividades do usuário com AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar AWS atividades, consulte [Trabalhando com CloudTrail trilhas](#) no AWS CloudTrail Guia do usuário.
- Use AWS soluções de criptografia, junto com todos os controles de segurança padrão dentro 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 uma API, use um FIPS endpoint. 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 AWS Serverless Application Repository 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 trânsito

AWS Serverless Application Repository APIs endpoints oferecem suporte apenas a conexões seguras. HTTPS Quando você gerencia AWS Serverless Application Repository recursos com o AWS Management Console, AWS SDK, ou o AWS Serverless Application Repository API, toda comunicação é criptografada com Transport Layer Security (TLS).

Para obter uma lista completa de API endpoints, consulte [AWS Regiões e endpoints](#) no Referência geral da AWS.

Criptografia em repouso

A ferramenta AWS Serverless Application Repository criptografa os arquivos que você envia para o AWS Serverless Application Repository, incluindo pacotes de implantação e arquivos de camadas.

Identity and Access Management para o AWS Serverless Application Repository

AWS Identity and Access Management (IAM) é um AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso ao AWS recursos. IAMos administradores controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar AWS Serverless Application Repository recursos. IAMé um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Para obter uma visão geral de como IAM funciona, consulte [Entendendo como IAM funciona](#) no Guia IAM do usuário.

Tópicos

- [Público](#)
- [Autenticação com identidades](#)
- [Gerenciamento do acesso usando políticas](#)
- [Como AWS Serverless Application Repository funciona com IAM](#)
- [Exemplos de políticas baseadas em identidade do AWS Serverless Application Repository](#)

- [AWS Serverless Application Repository Exemplos de políticas de aplicativos](#)
- [AWS Serverless Application Repository Permissões da API do: Referência de ações e recursos](#)
- [Solução de problemas AWS Serverless Application Repository de identidade e acesso](#)

Público

Como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz em AWS Serverless Application Repository.

Usuário do serviço — Se você usar o AWS Serverless Application Repository serviço para fazer seu trabalho, então seu administrador fornece as credenciais e permissões de que você precisa. À medida que você usa mais AWS Serverless Application Repository recursos para fazer seu trabalho, talvez você 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 AWS Serverless Application Repository, consulte [Solução de problemas AWS Serverless Application Repository de identidade e acesso](#).

Administrador de serviços — Se você é responsável por AWS Serverless Application Repository recursos em sua empresa, você provavelmente tem acesso total aos AWS Serverless Application Repository. É seu trabalho determinar quais AWS Serverless Application Repository recursos e recursos que 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 do IAM. Para saber mais sobre como sua empresa pode usar IAM com AWS Serverless Application Repository, consulte [Como AWS Serverless Application Repository funciona com IAM](#).

IAM administrador — Se você for IAM administrador, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso a AWS Serverless Application Repository. Para ver um exemplo AWS Serverless Application Repository políticas baseadas em identidade que você pode usar em IAM, consulte [Exemplos de políticas baseadas em identidade do AWS Serverless Application Repository](#)

Autenticação com identidades

A autenticação é como você faz login no AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado em AWS) como o Usuário raiz da conta da AWS, como IAM usuário ou assumindo uma IAM função.

Você pode entrar em 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. Quando você acessa AWS ao usar a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou o AWS portal de acesso. Para obter mais informações sobre como fazer login no AWS, veja [Como fazer login no seu Conta da AWS](#) no Início de Sessão da AWS Guia do 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 [Assinatura AWS APIsolicitações](#) no Guia do IAM 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 AWS IAM Identity Center Guia do usuário e [uso da autenticação multifatorial \(MFA\) em AWS](#) no IAM Guia do usuário.

Conta da AWS usuário raiz

Quando você cria um Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS e recursos na conta. Essa identidade é chamada de Conta da AWS usuário root e é acessado 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.

IAMGrupos e usuários

Um [IAMusuário](#) é uma identidade dentro do seu 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 [Altere as chaves de acesso regularmente para casos de uso que exigem credenciais de longo prazo](#) no Guia do IAM usuário.

Um [IAM grupo](#) 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.

IAM Funções

Um [IAM papel](#) é uma identidade dentro de você 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 papéis](#). Você pode assumir uma função chamando um AWS CLI ou AWS API operação ou usando um personalizado URL. Para obter mais informações sobre métodos de uso de funções, consulte [Usando IAM funções](#) no Guia IAM do usuário.

IAM funções 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 AWS IAM Identity Center Guia do usuário.
- **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 use 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 função para realizar ações no AWS, você é considerado diretor. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinado com a solicitação AWS service (Serviço da AWS) para fazer solicitações para serviços posteriores. FAS solicitações são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou recursos para concluir. 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 [IAM função](#) que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamente IAM. Para obter mais informações, consulte [Criação de uma função para delegar permissões a um AWS service \(Serviço da AWS\)](#) no IAM Guia do usuário.
- Função vinculada a serviços — Uma função vinculada a serviços é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode assumir a função de realizar uma ação em seu nome. As funções vinculadas ao serviço aparecem em seu 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 ou AWS API solicitações. Isso é preferível a armazenar chaves de acesso na EC2 instância. Para atribuir um AWS Ao atribuir a uma EC2 instância e disponibilizá-la para todos os seus aplicativos, você cria um perfil de instância que é 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.

Gerenciamento do acesso usando políticas

Você controla o acesso em AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto em AWS que, quando associados a uma identidade ou recurso, definem 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 em 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 JSONpolíticas 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, o AWS CLI, ou o 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 gerenciadas incluem AWS 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 em 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 AWS políticas gerenciadas a partir IAM de uma política baseada 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. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento JSON de política.

Amazon S3, AWS WAF, e a Amazon VPC são exemplos de serviços que oferecem suporte ACLs. Para saber mais ACLs, 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 em contas de membros, incluindo cada Usuário raiz da conta da AWS. Para obter mais informações sobre Organizations e SCPs, consulte [Políticas de controle de serviços](#) no AWS Organizations Guia 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 determina 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 AWS Serverless Application Repository funciona com IAM

Antes de usar IAM para gerenciar o acesso ao AWS Serverless Application Repository, você deve entender quais IAM recursos estão disponíveis para uso com AWS Serverless Application Repository o.

Para obter uma visão geral de como IAM funciona, consulte [Entendendo como IAM funciona](#) no Guia IAM do usuário. Para obter uma visão geral de como o AWS Serverless Application Repository e outros AWS serviços funcionam com IAM, consulte [AWS Serviços que funcionam com IAM](#) no Guia do IAM usuário.

Tópicos

- [AWS Serverless Application Repository Políticas baseadas em identidade](#)
- [AWS Serverless Application Repository Políticas de aplicativos](#)
- [Autorização baseada em tags do AWS Serverless Application Repository](#)
- [AWS Serverless Application Repository IAMFunções](#)

AWS Serverless Application Repository Políticas baseadas em identidade

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. O AWS Serverless Application Repository oferece suporte a ações, recursos e chaves de condição específicos. Para saber mais sobre todos os elementos que você usa em uma JSON política, consulte [Referência IAM JSON de elementos de política](#) no Guia do IAM usuário.

A seguir, um exemplo de uma política de permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:CreateApplication"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CreateApplicationVersion",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:CreateApplicationVersion"
      ],
      "Resource": "arn:partition:serverlessrepo:region:account-
id:applications/application-name"
    }
  ]
}
```

A política tem duas instruções:

- A primeira instrução concede permissões para a AWS Serverless Application Repository ação `serverlessrepo:CreateApplication` em todos os AWS Serverless Application Repository recursos, conforme especificado pelo caractere curinga (*) como Resource valor.
- A segunda declaração concede permissão para a AWS Serverless Application Repository ação `serverlessrepo:CreateApplicationVersion` em um AWS recurso usando o Amazon Resource Name (ARN) para um AWS Serverless Application Repository aplicativo. O aplicativo é especificado pelo valor Resource.

A política não especifica o elemento `Principal` porque, em uma política baseada em identidade, não se especifica o principal que obtém as permissões. Quando você anexar uma política um usuário, o usuário será a entidade principal implícita. Quando você anexa uma política de permissão a uma IAM função, o principal identificado na política de confiança da função obtém as permissões.

Para obter uma tabela mostrando todas as AWS Serverless Application Repository API operações e os AWS recursos aos quais elas se aplicam, consulte [AWS Serverless Application Repository Permissões da API do: Referência de ações e recursos](#).

Ações

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.

As ações políticas no AWS Serverless Application Repository usam o seguinte prefixo antes da ação: `serverlessrepo:`. Por exemplo, para conceder permissão a alguém para executar uma AWS Serverless Application Repository instância com a AWS Serverless Application Repository `SearchApplications` API operação, você inclui a `serverlessrepo:SearchApplications` ação na política dessa pessoa. As instruções de política devem incluir um elemento `Action` ou

NotAction. O AWS Serverless Application Repository define seu próprio conjunto de ações que descrevem as tarefas que você pode executar com esse serviço.

Para especificar várias ações em uma única instrução, separe-as com vírgulas, como segue:

```
"Action": [  
    "serverlessrepo:action1",  
    "serverlessrepo: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": "serverlessrepo:List*"
```

Para ver uma lista de AWS Serverless Application Repository ações, consulte [Ações definidas por AWS Serverless Application Repository](#) no Guia do IAM usuário.

Recursos

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": "*"
```

No AWS Serverless Application Repository, o AWS recurso principal é um AWS Serverless Application Repository aplicativo. AWS Serverless Application Repository os aplicativos têm nomes de recursos exclusivos da Amazon (ARNs) associados a eles, conforme mostrado na tabela a seguir.

AWS Tipo de recurso	Formato do nome do recurso da Amazon (ARN)
Aplicativo	celeiro: <i>partition</i> : repositório sem servidor: <i>region</i> : <i>account-id</i> :aplicativos/ <i>application-name</i>

Para obter mais informações sobre o formato de ARNs, consulte [Amazon Resource Names \(ARNs\) e AWS Service Namespaces](#).

Veja a seguir um exemplo de política que concede permissões para a `serverlessrepo:ListApplications` ação em todos os AWS recursos. Na implementação atual, o AWS Serverless Application Repository não suporta a identificação de AWS recursos específicos usando o AWS recurso ARNs (também conhecido como permissões em nível de recurso) para algumas das ações. API Nesses casos, você deve especificar um caractere curinga (*).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListExistingApplications",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:ListApplications"
      ],
      "Resource": "*"
    }
  ]
}
```

Para ver uma tabela mostrando todas as AWS Serverless Application Repository API ações e os AWS recursos aos quais elas se aplicam, consulte [AWS Serverless Application Repository Permissões da API do: Referência de ações e recursos](#).

Chaves de condição

O AWS Serverless Application Repository não fornece nenhuma chave de condição específica do serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte [Chaves de contexto de condição AWS global](#) no Guia IAM do usuário.

Exemplos

Para ver exemplos de políticas AWS Serverless Application Repository baseadas em identidade, consulte [Exemplos de políticas baseadas em identidade do AWS Serverless Application Repository](#)

AWS Serverless Application Repository Políticas de aplicativos

As políticas de aplicativo determinam as ações que um diretor específico ou principalOrg pode executar em um AWS Serverless Application Repository aplicativo.

Você pode adicionar permissões à política associada a um AWS Serverless Application Repository aplicativo. As políticas de permissões anexadas aos AWS Serverless Application Repository aplicativos são chamadas de políticas de aplicativos. [As políticas de aplicativos](#) são extensões das políticas [IAMbaseadas em recursos](#). O recurso principal é o AWS Serverless Application Repository aplicativo. Você pode usar políticas de AWS Serverless Application Repository aplicativos para gerenciar as permissões de implantação de aplicativos.

AWS Serverless Application Repository as políticas de aplicativos são usadas principalmente pelos editores para conceder permissão aos consumidores para implantar seus aplicativos e operações relacionadas, como pesquisar e visualizar detalhes desses aplicativos. Os editores podem definir permissões de aplicativos às três seguintes categorias:

- Privado — Aplicativos que foram criados com a mesma conta e não foram compartilhados com nenhuma outra conta. Você tem permissão para implantar aplicativos que foram criados usando sua AWS conta.
- Compartilhado de forma privada — Aplicativos que o editor compartilhou explicitamente com um conjunto específico de AWS contas ou Organizations AWS . Você tem permissão para implantar aplicativos que foram compartilhados com sua AWS conta ou AWS organização.
- Compartilhado publicamente — aplicativos que o editor compartilhou com todos. Você tem permissão para implantar aplicativos compartilhados publicamente.

Você pode conceder permissões usando o AWS CLI AWS SDKs, o ou AWS Management Console o.

Exemplos

Para ver exemplos de gerenciamento de políticas de AWS Serverless Application Repository aplicativos, consulte [AWS Serverless Application RepositoryExemplos de políticas de aplicativos](#).

Autorização baseada em tags do AWS Serverless Application Repository

O AWS Serverless Application Repository não oferece suporte ao controle do acesso a recursos ou ações com base em tags.

AWS Serverless Application Repository IAM Funções

Uma [IAM função](#) é uma entidade dentro da sua AWS conta que tem permissões específicas.

Usando credenciais temporárias com o AWS Serverless Application Repository

Você pode usar credenciais temporárias para entrar com a federação, assumir uma IAM função ou assumir uma função entre contas. Você obtém credenciais de segurança temporárias ligando para AWS STS API operações como [AssumeRole](#) ou [GetFederationToken](#).

O AWS Serverless Application Repository suporte usa credenciais temporárias.

Funções vinculadas ao serviço

O AWS Serverless Application Repository não oferece suporte a funções vinculadas a serviços.

Perfis de serviço

AWS Serverless Application Repository Não oferece suporte a funções de serviço.

Exemplos de políticas baseadas em identidade do AWS Serverless Application Repository

Por padrão, os usuários e as funções do IAM não têm permissão para criar ou modificar recursos do AWS Serverless Application Repository. Eles também não podem executar tarefas usando o AWS Management Console, a AWS CLI ou uma API da AWS. Um administrador do IAM deve criar políticas do IAM que concedam aos usuários e perfis permissão para executarem operações de API específicas nos recursos especificados de que precisam. O administrador deve anexar essas políticas aos usuários ou grupos do IAM que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte [Criar políticas na guia JSON](#) no Guia do usuário do IAM.

Tópicos

- [Melhores práticas de políticas](#)
- [Usar o console AWS Serverless Application Repository](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)
- [Exemplos de política gerenciada pelo cliente](#)

Melhores práticas de políticas

As políticas baseadas em identidade são muito eficientes. Elas determinam se alguém pode criar, acessar ou excluir recursos do AWS Serverless Application Repository em sua conta. Essas ações podem incorrer em custos para sua conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Conceder privilégio mínimo: ao criar políticas personalizadas, conceda apenas as permissões necessárias para executar uma tarefa. Comece com um conjunto mínimo de permissões e conceda permissões adicionais conforme necessário. Fazer isso é mais seguro do que começar com permissões que são muito lenientes e tentar restringi-las superiormente. Para obter mais informações, consulte [Conceder privilégio mínimo](#) no Guia do usuário do IAM.
- Habilitar MFA para operações confidenciais: para aumentar a segurança, exija que os usuários do IAM usem Multi-Factor Authentication (MFA) para acessar recursos ou operações de API confidenciais. Para obter mais informações, consulte [Uso da autenticação multifator \(MFA\) na AWS](#) no Manual do usuário do IAM.
- Usar condições de política para segurança adicional: na medida do possível, defina as condições sob as quais suas políticas baseadas em identidade permitem o acesso a um recurso. Por exemplo, você pode gravar condições para especificar um intervalo de endereços IP permitidos do qual a solicitação deve partir. Você também pode escrever condições para permitir somente solicitações em uma data especificada ou período ou para exigir o uso de SSL ou MFA. Para obter mais informações, consulte [Elementos de política JSON do IAM: condição](#) no Guia do usuário do IAM.

Usar o console AWS Serverless Application Repository

O console do AWS Serverless Application Repository fornece um ambiente integrado para você descobrir e gerenciar aplicativos do AWS Serverless Application Repository. O console fornece recursos e fluxos de trabalho que, muitas vezes, exigem permissões para gerenciar um aplicativo do AWS Serverless Application Repository, além das permissões específicas à API documentadas em [AWS Serverless Application Repository Permissões da API do: Referência de ações e recursos](#)

Para obter mais informações sobre permissões necessárias para usar o console do AWS Serverless Application Repository, consulte [Exemplos de política gerenciada pelo cliente](#).

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como você pode criar uma política que permite que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a API da AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Exemplos de política gerenciada pelo cliente

Os exemplos desta seção apresentam um grupo de políticas de amostra que você pode anexar a um usuário. Se você não estiver familiarizado com a criação de políticas, recomendamos primeiro criar um usuário do IAM na conta e anexar as políticas ao usuário sequencialmente. Você também pode usar esses exemplos para criar uma única política personalizada que inclui permissões para executar várias ações e depois anexá-la ao usuário.

Para obter mais informações sobre como anexar políticas aos usuários, consulte [Adicionar permissões a um usuário](#) no Guia do usuário do IAM.

Exemplos

- [Exemplo 1 do editor: Permitir que um editor liste os aplicativos](#)
- [Exemplo 2 do editor: Permitir que um editor visualize os detalhes de um aplicativo ou de uma versão do aplicativo](#)
- [Exemplo 3 do editor: Permitir que um editor crie um aplicativo ou uma versão do aplicativo](#)
- [Exemplo 4 do editor: Permitir que um editor crie uma política de aplicativo para compartilhar aplicativos com outros usuários](#)
- [Exemplo 1 de consumidor: Permitir que um consumidor pesquise por aplicativos](#)
- [Exemplo 2 de consumidor: Permitir que um consumidor visualize detalhes de um aplicativo](#)
- [Exemplo 3 de consumidor: Permitir que um consumidor implante um aplicativo](#)
- [Exemplo 4 do consumidor: negar acesso aos ativos de implantação](#)
- [Exemplo 5 do consumidor: impedir que um consumidor pesquise e implante aplicativos públicos](#)

Exemplo 1 do editor: Permitir que um editor liste os aplicativos

Um usuário do IAM em sua conta deve ter permissões para a operação `serverlessrepo:ListApplications` antes que o usuário possa ver qualquer coisa no console. Quando você concede essas permissões, o console pode mostrar a lista de AWS Serverless Application Repository aplicativos na AWS conta criada na AWS região específica à qual o usuário pertence.

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

```
{
  "Sid": "ListExistingApplications",
  "Effect": "Allow",
  "Action": [
    "serverlessrepo:ListApplications"
  ],
  "Resource": "*"
}
]
```

Exemplo 2 do editor: Permitir que um editor visualize os detalhes de um aplicativo ou de uma versão do aplicativo

Um usuário pode selecionar um aplicativo do AWS Serverless Application Repository e visualizar os detalhes do aplicativo. Esses detalhes incluem autor, descrição, versões e outras informações de configuração. Para isso, o usuário precisa de permissões para as operações `serverlessrepo:ListApplicationVersions` e `serverlessrepo:GetApplication` da API para o AWS Serverless Application Repository.

No exemplo a seguir, essas permissões são concedidas ao aplicativo específico cujo Amazon Resource Name (ARN - Nome de recurso da Amazon) é especificado como o valor de `Resource`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:GetApplication",
        "serverlessrepo:ListApplicationVersions"
      ],
      "Resource": "arn:aws:serverlessrepo:region:account-id:applications/application-name"
    }
  ]
}
```

Exemplo 3 do editor: Permitir que um editor crie um aplicativo ou uma versão do aplicativo

Para conceder permissões a um usuário para criar aplicativos do AWS Serverless Application Repository, é necessário conceder permissões para as operações `serverlessrepo:CreateApplication` e `serverlessrepo:CreateApplicationVersions`, conforme exibido na política a seguir.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:CreateApplication",
        "serverlessrepo:CreateApplicationVersion",
      ],
      "Resource": "*"
    }
  ]
}
```

Exemplo 4 do editor: Permitir que um editor crie uma política de aplicativo para compartilhar aplicativos com outros usuários

Para que os usuários compartilhem aplicativos com outros usuários, você deve conceder a eles permissões para criar políticas de aplicativo, conforme exibido na política a seguir.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ShareApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:PutApplicationPolicy",
        "serverlessrepo:GetApplicationPolicy",
      ],
      "Resource": "*"
    }
  ]
}
```

```
]
}
```

Exemplo 1 de consumidor: Permitir que um consumidor pesquise por aplicativos

Para que os consumidores pesquem por aplicativos, você deve conceder as seguintes permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SearchApplications",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:SearchApplications"
      ],
      "Resource": "*"
    }
  ]
}
```

Exemplo 2 de consumidor: Permitir que um consumidor visualize detalhes de um aplicativo

Um usuário pode selecionar um aplicativo do AWS Serverless Application Repository e visualizar detalhes do aplicativo, como autor, descrição, versões e outras informações de configuração. Para isso, o usuário deve ter permissões para as seguintes operações do AWS Serverless Application Repository.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:GetApplication",
        "serverlessrepo:ListApplicationVersions"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "*"
  }
]
}

```

Exemplo 3 de consumidor: Permitir que um consumidor implante um aplicativo

Para que os clientes implantem aplicativos, você deve conceder a eles permissões para executar várias operações. A política a seguir fornece as permissões necessárias aos clientes.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeployApplication",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:CreateCloudFormationChangeSet",
        "cloudformation:CreateChangeSet",
        "cloudformation:ExecuteChangeSet",
        "cloudformation:DescribeStacks"
      ],
      "Resource": "*"
    }
  ]
}

```

Note

A implantação de um aplicativo pode exigir permissões para usar AWS recursos adicionais. Como o AWS Serverless Application Repository usa o mesmo mecanismo de implantação subjacente que AWS CloudFormation, consulte [Controlando o acesso com AWS Identity and Access Management](#) para obter mais informações. Para obter ajuda com problemas de implantação relacionados a permissões, consulte [Solução de problemas: permissões do IAM insuficientes](#).

Exemplo 4 do consumidor: negar acesso aos ativos de implantação

Quando um aplicativo é compartilhado de forma privada com uma AWS conta, por padrão, todos os usuários dessa conta podem acessar os ativos de implantação de todos os outros usuários na mesma conta. A política a seguir impede que os usuários de uma conta acessem os ativos de implantação, que são armazenados no bucket do Amazon S3 para AWS Serverless Application Repository.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyDeploymentAssetAccess",
      "Effect": "Deny",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::awsserverlessrepo-changesets*/*"
      ]
    }
  ]
}
```

Exemplo 5 do consumidor: impedir que um consumidor pesquise e implante aplicativos públicos

É possível impedir que os usuários executem determinadas ações em aplicativos.

A política a seguir se aplica a aplicativos públicos especificando `serverlessrepo:applicationType` como `public`. Ela impede que os usuários realizem uma série de ações especificando `Effect` como `Deny`. Para obter mais informações sobre as chaves de condição disponíveis para o AWS Serverless Application Repository, consulte [Ações, recursos e chaves de condição do AWS Serverless Application Repository](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Condition": {
        "StringEquals": {
          "serverlessrepo:applicationType": "public"
        }
      }
    }
  ]
}
```

```
    }
  },
  "Action": [
    "serverlessrepo:SearchApplications",
    "serverlessrepo:GetApplication",
    "serverlessrepo:CreateCloudFormationTemplate",
    "serverlessrepo:CreateCloudFormationChangeSet",
    "serverlessrepo:ListApplicationVersions",
    "serverlessrepo:ListApplicationDependencies"
  ],
  "Resource": "*",
  "Effect": "Deny"
}
]
```

Note

Essa declaração de política também pode ser usada como uma Política de Controle de Serviços e aplicada a uma AWS organização. Para obter mais informações sobre políticas de controle de serviço, consulte [Políticas de controle de serviço](#) no Guia AWS Organizations do usuário.

AWS Serverless Application Repository Exemplos de políticas de aplicativos

As políticas de permissões anexadas aos AWS Serverless Application Repository aplicativos são chamadas de políticas de aplicativos. As políticas do aplicativo determinam as ações que um diretor específico ou PrincipalOrg pode executar em um AWS Serverless Application Repository aplicativo.

Um AWS Serverless Application Repository aplicativo é o principal AWS recurso no AWS Serverless Application Repository. As políticas de aplicativos são usadas principalmente pelos editores para conceder permissão aos consumidores para implantar seus aplicativos e operações relacionadas, como pesquisar e visualizar detalhes desses aplicativos.

Os editores podem definir permissões de aplicativos às três seguintes categorias:

- Privado — Aplicativos que foram criados com a mesma conta e não foram compartilhados com nenhuma outra conta. Somente consumidores que compartilham sua AWS conta têm permissão para implantar aplicativos privados.

- **Compartilhado de forma privada** — Aplicativos que o editor compartilhou explicitamente com um conjunto específico de AWS contas ou com AWS contas em uma AWS organização. Os consumidores têm permissão para implantar aplicativos que foram compartilhados com sua AWS conta ou AWS organização. Para obter mais informações sobre AWS organizações, consulte o [Guia AWS Organizations do usuário](#).
- **Compartilhado publicamente** — Aplicativos que o editor compartilhou com todos. Todos os consumidores têm permissão para implantar um aplicativo compartilhado publicamente.

Note

Para aplicativos compartilhados de forma privada, o AWS Serverless Application Repository único compatível com AWS contas como diretores. Os editores podem conceder ou negar todos os usuários de uma AWS conta como um único grupo para um AWS Serverless Application Repository aplicativo. Os editores não podem conceder ou negar usuários individuais de uma AWS conta a um AWS Serverless Application Repository aplicativo.

Para obter instruções sobre como configurar permissões para aplicativos usando o AWS Management Console, consulte [Compartilhar um aplicativo](#).

Para obter instruções sobre como configurar permissões de aplicativos usando a AWS CLI e exemplos, consulte as seguintes seções.

Permissões de aplicativos (AWS CLI e AWS SDKs)

Ao usar o AWS CLI ou os AWS SDKs para definir permissões para um AWS Serverless Application Repository aplicativo, você pode especificar as seguintes ações:

Ação	Descrição
GetApplication	Concede permissão para visualizar informações sobre o aplicativo.
CreateCloudFormationChangeSet	Concede permissão para o aplicativo ser implantado. Observação: essa ação não concede nenhuma outra permissão além da permissão de implantação.

Ação	Descrição
CreateCloudFormationTemplate	Concede permissão para criar um modelo AWS CloudFormation para o aplicativo.
ListApplicationVersions	Concede permissão para listar as versões do aplicativo.
ListApplicationDependencies	Concede permissão para relacionar os aplicativos que estão aninhados dentro do aplicativo.
SearchApplications	Concede permissão para o aplicativo ser pesquisado.
Implante	Essa ação habilita todas as ações listadas anteriormente na tabela. Ou seja, concede permissão para o aplicativo ser visualizado, implantado, as versões serem listadas e o aplicativo ser pesquisado.

Exemplos de política de usuário

Os exemplos a seguir mostram como conceder permissões usando a AWS CLI. Para obter informações sobre como conceder permissões usando o AWS Management Console, consulte [Compartilhar um aplicativo](#).

Todos os exemplos nesta seção usam esses comandos da AWS CLI para gerenciar políticas de permissões associadas aos aplicativos do AWS Serverless Application Repository:

- [put-application-policy](#)
- [get-application-policy](#)

Tópicos

- [Exemplo 1: compartilhar um aplicativo com outra conta](#)
- [Exemplo 2: Compartilhar um aplicativo publicamente](#)
- [Exemplo 3: Tornar um aplicativo privado](#)
- [Exemplo 4: Especificação de várias contas e permissões](#)
- [Exemplo 5: compartilhar um aplicativo com todas as contas em uma AWS organização](#)
- [Exemplo 6: Compartilhando um aplicativo com algumas contas em uma AWS organização](#)
- [Exemplo 7: recuperar uma política de aplicativo](#)

- [Exemplo 8: permitir que o aplicativo seja aninhado por contas específicas](#)

Exemplo 1: compartilhar um aplicativo com outra conta

Para compartilhar um aplicativo com outra conta específica, mas evitar que ele seja compartilhado com outras pessoas, você especifica o ID da AWS conta com a qual deseja compartilhar como principal. Isso também é conhecido como configurar o aplicativo como compartilhado de forma privada. Para fazer isso, execute o seguinte comando da AWS CLI.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=account-id,Actions=Deploy
```

Note

Aplicativos compartilhados de forma privada só podem ser usados na mesma AWS região em que o aplicativo foi criado.

Exemplo 2: Compartilhar um aplicativo publicamente

Para tornar um aplicativo público, você o compartilha com todos os usuários especificando "" como a entidade principal, como no exemplo a seguir. Os aplicativos compartilhados publicamente ficam disponíveis em todas as regiões.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=*,Actions=Deploy
```

Note

Para compartilhar um aplicativo publicamente, ele deve ter as propriedades `LicenseUrl` e `SemanticVersion` definidas.

Exemplo 3: Tornar um aplicativo privado

Você pode tornar um aplicativo privado, para que ele não seja compartilhado com ninguém e só possa ser implantado pela AWS conta que o possui. Para fazer isso, você elimina os princípios e ações da política, o que também remove as permissões de outras contas da sua AWS organização para implantar seu aplicativo.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements '[]'
```

Note

Aplicativos privados só podem ser usados na mesma AWS região em que o aplicativo foi criado.

Exemplo 4: Especificação de várias contas e permissões

Você pode conceder várias permissões e concedê-las a mais de uma AWS conta ao mesmo tempo. Para fazer isso, especifique listas como o principal e como ações, conforme exibido no exemplo a seguir.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=account-id-1,account-  
id-2,Actions=GetApplication,CreateCloudFormationChangeSet
```

Exemplo 5: compartilhar um aplicativo com todas as contas em uma AWS organização

As permissões podem ser concedidas a todos os usuários de uma AWS organização. É possível fazer isso especificando o ID da organização, como no exemplo a seguir.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=*,PrincipalOrgIDs=org-id,Actions=Deploy,UnshareApplication
```

Para obter mais informações sobre AWS organizações, consulte o [Guia AWS Organizations do usuário](#).

Note

Você só pode especificar a AWS organização da qual sua AWS conta é membro. Se você tentar especificar uma AWS organização da qual não é membro, ocorrerá um erro. Para compartilhar seu aplicativo com sua AWS organização, você deve incluir permissão para `UnshareApplication` ação, caso o compartilhamento precise ser revogado no future.

Exemplo 6: Compartilhando um aplicativo com algumas contas em uma AWS organização

As permissões podem ser concedidas a contas específicas dentro de uma AWS organização. Você faz isso especificando uma lista de AWS contas como principal e o ID da sua organização, como no exemplo a seguir.

```
aws serverlessrepo put-application-policy \  
--region region \  
--application-id application-arn \  
--statements Principals=account-id-1,account-id-2,PrincipalOrgIDs=org-  
id,Actions=Deploy,UnshareApplication
```

Note

Você só pode especificar a AWS organização da qual sua AWS conta é membro. Se você tentar especificar uma AWS organização da qual não é membro, ocorrerá um erro. Para compartilhar seu aplicativo com sua AWS organização, você deve incluir permissão para `UnshareApplication` ação, caso o compartilhamento precise ser revogado no future.

Exemplo 7: recuperar uma política de aplicativo

Para visualizar a política atual de um aplicativo, por exemplo, para ver se ele está sendo compartilhado no momento, use o comando `get-application-policy`, como no exemplo a seguir.

```
aws serverlessrepo get-application-policy \  
--region region \  

```

```
--application-id application-arn
```

Exemplo 8: permitir que o aplicativo seja aninhado por contas específicas

Aplicativos públicos podem ser aninhados por qualquer pessoa. Caso você queira permitir que seu aplicativo seja aninhado por contas específicas, você deve ajustar as permissões mínimas abaixo, conforme exibido no exemplo a seguir.

```
aws serverlessrepo put-application-policy \
--region region \
--application-id application-arn \
--statements Principals=account-id-1,account-id-2,Actions=GetApplication,CreateCloudFormationTemplate
```

AWS Serverless Application Repository Permissões da API do: Referência de ações e recursos

Ao configurar o [controle de acesso](#) e elaborar políticas de permissões que você pode associar a uma identidade do IAM (políticas baseadas em identidade), use a tabela a seguir como referência. Cada AWS Serverless Application Repository operação de API do, as ações correspondentes às quais você pode conceder permissões para executar a ação e o AWSO recurso da para você pode conceder as permissões. Você especifica as ações no campo `Action` da política e o valor do recurso no campo `Resource` da política.

Para especificar uma ação, use o prefixo `serverlessrepo:` seguido do nome da operação da API (por exemplo, `serverlessrepo:ListApplications`).

Operação	URI	Método	AWS Recursos (ARNs)
operação: ListApplications	/applications	GET	*
Permissões obrigatórias: serverlessrepo:ListApplications			
operação: CreateApplication	/applications	POST	*

Operação	URI	Método	AWSRecursos (ARNs)
Permissões obrigatórias:serverlessrepo:CreateApplication			
operação: GetApplication Permissões obrigatórias:serverlessrepo:GetApplication	/applications/ <i>id-do-aplicativo</i>	GET	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: DeleteApplication Permissões obrigatórias:serverlessrepo:DeleteApplication	/applications/ <i>id-do-aplicativo</i>	DELETE	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: UpdateApplication Permissões obrigatórias:serverlessrepo:UpdateApplication	/applications/ <i>id-do-aplicativo</i>	PATCH	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: CreateCloudFormationChangeDefinition Permissões obrigatórias:serverlessrepo:CreateCloudFormationChangeDefinition	/applications/ <i>id-do-aplicativo</i> /changesets	POST	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>

Operação	URI	Método	AWSRecursos (ARNs)
operação: GetApplicationPolítica Permissões obrigatórias: serverlessrepo:GetApplicationPolítica	/applications/ <i>id-do-aplicativo</i> /policy	GET	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: PutApplicationPolítica Permissões obrigatórias: serverlessrepo:PutApplicationPolítica	/applications/ <i>id-do-aplicativo</i> /policy	PUT	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: ListApplicationVersões do : Permissões obrigatórias: serverlessrepo:ListApplicationVersões do :	/applications/ <i>id-do-aplicativo</i> /versions	GET	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: CreateApplicationVersão Permissões obrigatórias: serverlessrepo:CreateApplicationVersão	/applications/ <i>id-do-aplicativo</i> /versions/ <i>versão-semântica</i>	PUT	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>

Operação	URI	Método	AWSRecursos (ARNs)
operação: ListApplicationDependências do Permissões obrigatórias: serverlessrepo:ListApplicationDependências do	/aplicativos/ <i>id-do-aplicativo</i> / dependências	GET	arn:aws:serverlessrepo: <i>região</i> : <i>id-da-conta</i> :applications/ <i>nome-do-aplicativo</i>
operação: SearchApplications Permissões obrigatórias: serverlessrepo:SearchApplications	n/a	n/a	*

Solução de problemas AWS Serverless Application Repository de identidade e acesso

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o AWS Serverless Application Repository e IAM

Tópicos

- [Não tenho autorização para executar uma ação no AWS Serverless Application Repository](#)
- [Não estou autorizado a realizar o meu pedido: PassRole](#)
- [Sou administrador e quero permitir que outras pessoas acessem o AWS Serverless Application Repository](#)
- [Quero permitir que pessoas fora da minha AWS conta acessem meus AWS Serverless Application Repository recursos](#)

Não tenho autorização para executar uma ação no AWS Serverless Application Repository

Se isso AWS Management Console indicar que você não está autorizado a realizar uma ação, entre em contato com o administrador para obter ajuda. O administrador é a pessoa que forneceu o seu nome de usuário e senha.

O exemplo de erro a seguir ocorre quando o mateojackson IAM usuário tenta usar o console para ver detalhes sobre um aplicativo, mas não tem `serverlessrepo:GetApplication` permissões.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
serverlessrepo:GetApplication on resource: my-example-application
```

Nesse caso, Mateo pede ao administrador para atualizar suas políticas a fim de obter acesso ao recurso `my-example-application` usando a operação `serverlessrepo:GetApplication`.

Não estou autorizado a realizar o meu pedido: PassRole

Se você receber uma mensagem de erro informando que não está autorizado a executar a ação `iam:PassRole`, as suas políticas devem ser atualizadas para permitir que você passe uma função para o AWS Serverless Application Repository.

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 AWS Serverless Application Repository. 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.

Sou administrador e quero permitir que outras pessoas acessem o AWS Serverless Application Repository

Para permitir que outras pessoas acessem AWS Serverless Application Repository, você deve conceder permissão às pessoas ou aplicativos que precisam de acesso. Se você estiver usando AWS IAM Identity Center para gerenciar pessoas e aplicativos, você atribui conjuntos de permissões a usuários ou grupos para definir seu nível de acesso. Os conjuntos de permissões criam e atribuem IAM políticas automaticamente às IAM funções associadas à pessoa ou ao aplicativo. Para obter mais informações, consulte [Conjuntos de permissões](#) no Guia AWS IAM Identity Center do usuário.

Se você não estiver usando o IAM Identity Center, deverá criar IAM entidades (usuários ou funções) para as pessoas ou aplicativos que precisam de acesso. Você deve anexar uma política à entidade que concede a eles as permissões corretas no AWS Serverless Application Repository. Depois que as permissões forem concedidas, forneça as credenciais ao usuário ou desenvolvedor do aplicativo. Eles usarão essas credenciais para acessar AWS. Para saber mais sobre a criação de IAM usuários, grupos, políticas e permissões, consulte [IAM identidades, políticas e permissões IAM no](#) Guia do IAM usuário.

Quero permitir que pessoas fora da minha AWS conta acessem meus AWS Serverless Application Repository recursos

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 é AWS Serverless Application Repository compatível com esses recursos, consulte [Como AWS Serverless Application Repository 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 IAM usuário.

- 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.

Registro em log e monitoramento no AWS Serverless Application Repository

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e performance das suas soluções da AWS. Você deve coletar dados de monitoramento de todas as partes da solução da AWS para depurar uma falha de vários pontos com mais facilidade, caso ocorra. A AWS fornece várias ferramentas para monitorar seus recursos do AWS Serverless Application Repository e responder a incidentes em potencial, tais como:

Logs do AWS CloudTrail

O AWS Serverless Application Repository é integrado ao AWS CloudTrail, um serviço que fornece um registro de ações executadas por um usuário, uma função ou um AWS serviço na AWS Serverless Application Repository. O CloudTrail captura todas as chamadas à API do AWS Serverless Application Repository como eventos.

Tópicos

- [Registro em log de chamadas à API do AWS Serverless Application Repository com o AWS CloudTrail](#)

Registro em log de chamadas à API do AWS Serverless Application Repository com o AWS CloudTrail

O AWS Serverless Application Repository é integrado ao AWS CloudTrail, que é um serviço que fornece um registro das ações realizadas por um usuário, uma função ou um AWS serviço no AWS Serverless Application Repository. O CloudTrail captura todas as chamadas à API do AWS Serverless Application Repository como eventos. As chamadas capturadas incluem as chamadas do console do AWS Serverless Application Repository e as chamadas de código para as operações da API do AWS Serverless Application Repository.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para o AWS Serverless Application Repository. Se não configurar uma

trilha, você ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history.

Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação que foi feita ao AWS Serverless Application Repository. Também é possível determinar o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, e detalhes adicionais.

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

Informações sobre o AWS Serverless Application Repository no CloudTrail

O CloudTrail está habilitado na sua conta da AWS ao criá-la. Quando ocorre uma atividade no AWS Serverless Application Repository, ela é registrada em um evento do CloudTrail com outros eventos de serviços da AWS em Event history (Histórico de eventos). Você pode visualizar, pesquisar e baixar eventos recentes em sua conta da AWS. Para obter mais informações, consulte [Visualizar eventos com o histórico de eventos do CloudTrail](#).

Para obter um registro contínuo de eventos na conta da AWS, incluindo eventos do AWS Serverless Application Repository, crie uma trilha. Uma trilha ativa CloudTrail para enviar 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 em log eventos de todas as regiões da AWS na partição da AWS e entrega os arquivos de log para o bucket do Amazon S3 especificado por você. Além disso, é possível configurar outros serviços da AWS para analisar mais profundamente e agir sobre os dados de evento coletados nos logs do CloudTrail. Para obter mais informações, consulte as informações a seguir:

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

Todas as ações do AWS Serverless Application Repository são registradas em log pelo CloudTrail e documentadas na página [Recursos do AWS Serverless Application Repository](#). Por exemplo, as chamadas para as operações `CreateApplication`, `UpdateApplications` e `ListApplications` geram entradas nos arquivos de log do CloudTrail.

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

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

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

Noções básicas das entradas dos arquivos de log do AWS Serverless Application Repository

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

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra a ação `CreateApplication`.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "Root",
    "principalId": "999999999999",
    "arn": "arn:aws:iam::999999999999:root",
    "accountId": "999999999999",
    "accessKeyId": "ASIAUVPLBDH76HEXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-07-30T16:40:42Z"
      }
    }
  },
  "invokedBy": "signin.amazonaws.com"
},
```

```

"eventTime": "2018-07-30T17:37:37Z",
"eventSource": "serverlessrepo.amazonaws.com",
"eventName": "CreateApplication",
"awsRegion": "us-east-1",
"sourceIPAddress": "72.21.217.161",
"userAgent": "signin.amazonaws.com",
"requestParameters": {
  "licenseBody": "<content of license>",
  "sourceCodeUrl": "<sample url>",
  "spdxLicenseId": "<sample license id>",
  "readmeBody": "<content of readme>",
  "author": "<author name>",
  "templateBody": "<content of SAM template>",
  "name": "<application name>",
  "semanticVersion": "<version>",
  "description": "<content of description>",
  "homePageUrl": "<sample url>",
  "labels": [
    "<label1>",
    "<label2>"
  ]
},
"responseElements": {
  "licenseUrl": "<url to access content of license>",
  "readmeUrl": "<url to access content of readme>",
  "spdxLicenseId": "<sample license id>",
  "creationTime": "2018-07-30T17:37:37.045Z",
  "author": "<author name>",
  "name": "<application name>",
  "description": "<content of description>",
  "applicationId": "arn:aws:serverlessrepo:us-east-1:999999999999:applications/<application name>",
  "homePageUrl": "<sample url>",
  "version": {
    "applicationId": "arn:aws:serverlessrepo:us-east-1:999999999999:applications/<application name>",
    "semanticVersion": "<version>",
    "sourceCodeUrl": "<sample url>",
    "templateUrl": "<url to access content of SAM template>",
    "creationTime": "2018-07-30T17:37:37.027Z",
    "parameterDefinitions": [
      {
        "name": "<parameter name>",
        "description": "<parameter description>",

```

```
        "type": "<parameter type>"
      }
    ]
  },
  "labels": [
    "<label1>",
    "<label2>"
  ]
},
"requestID": "3f50d899-941f-11e8-ab18-01063f863be5",
"eventID": "a66a6490-d388-4a4f-8c7b-9d6ec61ab262",
"readOnly": false,
"eventType": "AwsApiCall",
"recipientAccountId": "999999999999"
}
```

Validação de conformidade do AWS Serverless Application Repository

Audidores externos avaliam a segurança e a conformidade do AWS Serverless Application Repository como parte de vários programas de conformidade da AWS. Isso inclui SOC, PCI, FedRAMP e outros.

Para uma lista de AWS serviços da no escopo de programas de conformidade específicos, consulte [AWS Serviços da no escopo pelo programa de conformidade](#). Para obter informações gerais, consulte [Programas de conformidade da AWS](#).

É possível baixar os relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte [Download de relatórios no AWS Artifact](#).

Sua responsabilidade com relação à conformidade ao usar o AWS Serverless Application Repository é determinada pela confidencialidade dos dados, pelos objetivos de conformidade da empresa e pelos regulamentos e leis aplicáveis. A AWS fornece os seguintes recursos para ajudar com a conformidade:

- [Guias de início rápido de segurança e conformidade](#)— esses guias de implantação abordam as considerações de arquitetura e fornecem etapas para implantação de ambientes de linha de base focados em conformidade e segurança na AWS.
- [Recursos de compatibilidade da AWS](#) – Esta coleção de guias e pastas de trabalho pode ser aplicada ao seu setor e local.

- [AWS Config](#): esse serviço da AWS avalia até que ponto suas configurações de recursos atendem adequadamente às práticas internas e às diretrizes e regulamentações do setor.
- [AWS Security Hub](#): esse serviço da AWS fornece uma visão abrangente do estado de sua segurança na AWS que ajuda você a conferir sua conformidade com padrões e práticas recomendadas de segurança do setor.

Resiliência no AWS Serverless Application Repository

A infraestrutura global da AWS é criada com base em regiões da AWS e zonas de disponibilidade da AWS. As regiões fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, as quais são conectadas com baixa latência, altas taxas de transferência e redes altamente redundantes. Com as zonas de disponibilidade, você pode projetar e operar aplicativos e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade 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 regiões e zonas de disponibilidade da AWS, consulte [Infraestrutura global da AWS](#).

Segurança de infraestrutura no AWS Serverless Application Repository

Como serviço gerenciado, AWS Serverless Application Repository 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 AWS Serverless Application Repository 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.

Cotas do AWS Serverless Application Repository

O AWS Serverless Application Repository tem uma cota para o número de aplicativos públicos que uma conta pode ter em cada região: Essa cota é aplicada por região e pode ser aumentada. Para solicitar um aumento, use o [Console do centro de suporte](#).

Recurso	Cota padrão
Aplicativos públicos (por conta da região)	100

As cotas a seguir se aplicam ao armazenamento disponível para pacotes de códigos e políticas de aplicativos. Não é possível alterar essas cotas.

Recurso	Quota
Armazenamento gratuito do Amazon S3 para pacotes de código (por conta da região)	5 GB
Duração da política do aplicativo	6.144 caracteres

Solucionar problemas do AWS Serverless Application Repository

Ao usar o AWS Serverless Application Repository, você pode encontrar problemas ao criar, atualizar ou excluir seus aplicativos. Use esta seção para ajudar a solucionar problemas comuns que podem ocorrer. Também é possível pesquisar respostas e postar perguntas no [Fóruns do AWS Serverless Application Repository](#).

Note

Os aplicativos no AWS Serverless Application Repository são implantados usando o AWS CloudFormation. Para obter informações sobre a solução de problemas do AWS CloudFormation consulte o [AWS CloudFormation Guia de solução de problemas](#).

Tópicos

- [Não é possível tornar um aplicativo público](#)
- [Uma cota foi excedida](#)
- [Um arquivo Readme atualizado não aparece imediatamente](#)
- [Você não pode implantar um aplicativo devido a permissões do IAM insuficientes](#)
- [Não é possível implantar o mesmo aplicativo duas vezes](#)
- [Por que meu aplicativo não está publicamente disponível?](#)
- [Como entrar em contato com o Support](#)

Não é possível tornar um aplicativo público

Se não for possível tornar seu aplicativo público, pode haver um arquivo de licença ausente para o seu aplicativo aprovado pela Open Source Initiative (OSI).

Para tornar seu aplicativo público, você precisa de um arquivo de licença aprovado pela OSI e também de uma versão publicada desse aplicativo com um URL de código-fonte. Não é possível atualizar a licença de um aplicativo depois que ele é criado.

Se você não puder tornar seu aplicativo público devido à ausência de um arquivo de licença, exclua o aplicativo e crie um novo com o mesmo nome. Forneça a ele uma ou mais licenças de código aberto aprovadas pela organização Open Source Initiative (OSI).

Uma cota foi excedida

Se você receber uma mensagem de erro indicando que uma cota foi excedida, verifique se você atingiu a cota do recurso. Para as cotas do AWS Serverless Application Repository, consulte [Cotas do AWS Serverless Application Repository](#).

Um arquivo Readme atualizado não aparece imediatamente

Quando você torna seu aplicativo público, o conteúdo dele pode levar até 24 horas para ser atualizado. Se a atualização levar mais do que 24 horas, entre em contato [AWS Support](#) para ajuda. Para detalhes, veja o seguinte.

Você não pode implantar um aplicativo devido a permissões do IAM insuficientes

Para implantar um aplicativo do AWS Serverless Application Repository, você precisa de permissões para recursos do AWS Serverless Application Repository; e pilhas do AWS CloudFormation. Você também pode precisar de permissão para usar os serviços subjacentes descritos no aplicativo. Por exemplo, se estiver criando um bucket do Amazon S3 ou uma tabela do Amazon DynamoDB, você precisa de permissões para o Amazon S3 ou para o DynamoDB.

Se tiver esse tipo de problema, analise sua política do AWS Identity and Access Management (IAM) e verifique se você tem as permissões necessárias. Para obter mais informações, consulte [Como controlar o acesso com o AWS Identity and Access Management](#).

Não é possível implantar o mesmo aplicativo duas vezes

O nome de aplicativo que você fornece é usado como o nome da pilha do AWS CloudFormation. Se você tiver problemas para implantar um aplicativo, verifique há alguma pilha do AWS CloudFormation com o mesmo nome. Se houver, forneça um nome de aplicativo diferente ou exclua a pilha existente para implantar o aplicativo com o mesmo nome.

Por que meu aplicativo não está publicamente disponível?

Os aplicativos são privados por padrão. Para tornar seu aplicativo público, siga as etapas descritas [aqui](#).

Como entrar em contato com o Support

Em alguns casos, pode não ser possível encontrar soluções de problemas nesta seção ou por meio dos [fóruns do AWS Serverless Application Repository](#). Se você tiver um AWS Premium Support, você poderá criar um caso de suporte técnico em [AWS Suporte para](#).

Antes de entrar em contato AWS Support, tenha em conta o nome de recurso da Amazon (ARN) do aplicativo para o qual você precisa de ajuda. É possível encontrar o ARN do aplicativo no [console do AWS Serverless Application Repository](#).

Operações

A API REST do AWS Serverless Application Repository inclui as seguintes operações.

- [CreateApplication](#)

Cria um aplicativo, opcionalmente, incluindo umAWSArquivo SAM para criar a primeira versão do aplicativo na mesma chamada.

- [CreateApplicationVersion](#)

Cria uma versão do aplicativo.

- [CreateCloudFormationChangeSet](#)

Cria umAWS CloudFormationConjunto de alterações do aplicativo fornecido.

- [CreateCloudFormationTemplate](#)

Cria umAWS CloudFormationTemplate.

- [DeleteApplication](#)

Exclui o aplicativo especificado.

- [GetApplication](#)

Obtém o aplicativo especificado.

- [GetApplicationPolicy](#)

Recupera a política para o aplicativo.

- [GetCloudFormationTemplate](#)

Obtém o especificadoAWS CloudFormationTemplate.

- [ListApplicationDependencies](#)

Recupera a lista de aplicativos aninhados no aplicativo que os contém.

- [ListApplications](#)

Lista os aplicativos de propriedade do solicitante.

- [ListApplicationVersions](#)

Lista as versões do aplicativo especificado.

- [PutApplicationPolicy](#)

Define a política de permissão para um aplicativo. Para obter a lista de ações compatíveis com esta operação, consulte [Permissões para aplicativos](#).

- [UnshareApplication](#)

Descompartilha um aplicativo de uma AWS Organização.

Essa operação só pode ser chamada da conta de gerenciamento da organização.

- [UpdateApplication](#)

Atualiza o aplicativo especificado.

Recursos

A ferramenta AWS Serverless Application Repository RESTAPI inclui os seguintes recursos.

Tópicos

- [Applications](#)
- [Aplicações applicationId](#)
- [Conjuntos de applicationId alterações de aplicativos](#)
- [Applications applicationId Dependencies](#)
- [applicationId Política de aplicativos](#)
- [Applications applicationId Templates](#)
- [Applications applicationId Templates templateId](#)
- [Applications applicationId Unshare](#)
- [applicationId Versões de aplicativos](#)
- [applicationId Versões de aplicativos semanticVersion](#)

Applications

URI

/applications

HTTP métodos

GET

ID da operação: ListApplications

Lista os aplicativos de propriedade do solicitante.

Parâmetros de consulta

Nome	Tipo	Obrigatório	Descrição
maxItems	Cadeia de caracteres	Falso	O número total de itens a serem devolvidos.
nextToken	String	Falso	Um token para especificar onde iniciar a paginação.

Respostas

Código de status	Modelo de resposta	Descrição
200	ApplicationPage	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

POST

ID da operação: CreateApplication

Cria um aplicativo, incluindo opcionalmente um AWS SAMArquivo para criar a primeira versão do aplicativo na mesma chamada.

Respostas

Código de status	Modelo de resposta	Descrição
201	<u>Application</u>	Bem-sucedida
400	<u>BadRequestException</u>	Um dos parâmetros na solicitação é inválido.
403	<u>ForbiddenException</u>	O cliente não está autenticado.
409	<u>ConflictException</u>	O recurso já existe.
429	<u>TooManyRequestsException</u>	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	<u>InternalServerErrorException</u>	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

POST Esquema

```
{
  "name": "string",
  "description": "string",
  "author": "string",
  "spdxLicenseId": "string",
  "licenseBody": "string",
  "licenseUrl": "string",
  "readmeBody": "string",
  "readmeUrl": "string",
  "labels": [
    "string"
  ],
  "homePageUrl": "string",
  "semanticVersion": "string",
  "templateBody": "string",
  "templateUrl": "string",
  "sourceCodeUrl": "string",
  "sourceCodeArchiveUrl": "string"
}
```

Corpos da resposta

ApplicationPage Esquema

```
{
  "applications": [
    {
      "applicationId": "string",
      "name": "string",
      "description": "string",
      "author": "string",
      "spdxLicenseId": "string",
      "labels": [
        "string"
      ],
      "creationTime": "string",
      "homePageUrl": "string"
    }
  ],
  "nextToken": "string"
}
```

Application Esquema

```
{
  "applicationId": "string",
  "name": "string",
  "description": "string",
  "author": "string",
  "isVerifiedAuthor": boolean,
  "verifiedAuthorUrl": "string",
  "spdxLicenseId": "string",
  "licenseUrl": "string",
  "readmeUrl": "string",
  "labels": [
    "string"
  ],
  "creationTime": "string",
  "homePageUrl": "string",
  "version": {
    "applicationId": "string",
    "semanticVersion": "string",
    "sourceCodeUrl": "string",
    "sourceCodeArchiveUrl": "string",
    "templateUrl": "string",
    "creationTime": "string",
    "parameterDefinitions": [
      {
        "name": "string",
        "defaultValue": "string",
        "description": "string",
        "type": "string",
        "noEcho": boolean,
        "allowedPattern": "string",
        "constraintDescription": "string",
        "minValue": integer,
        "maxValue": integer,
        "minLength": integer,
        "maxLength": integer,
        "allowedValues": [
          "string"
        ]
      },
      "referencedByResources": [
        "string"
      ]
    ]
  }
}
```

```
    }
  ],
  "requiredCapabilities": [
    enum
  ],
  "resourcesSupported": boolean
}
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

NotFoundException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ConflictException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

TooManyRequestsException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

InternalServerErrorException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

Propriedades

Application

Detalhes sobre o aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

name

O nome da aplicação.

Comprimento mínimo = 1. Comprimento máximo = 140

Padrão: "[a-zA-Z0-9\-\-]+";

Tipo: string

Obrigatório: verdadeiro

description

A descrição do aplicativo.

Comprimento mínimo = 1. Comprimento máximo = 256

Tipo: string

Obrigatório: verdadeiro

author

O nome do autor que publica o aplicativo.

Comprimento mínimo = 1. Tamanho máximo = 127.

Padrão “`^[a-z0-9] (([a-z0-9] | - (?! -)) * [a-z0-9])? $`”;

Tipo: string

Obrigatório: verdadeiro

isVerifiedAuthor

Especifica se o autor desse aplicativo foi verificado. Isso significa que AWS fez uma análise de boa fé, como prestador de serviços razoável e prudente, das informações fornecidas pelo solicitante e confirmou que a identidade do solicitante é a reivindicada.

Tipo: booleano

Obrigatório: falso

verifiedAuthorUrl

O URL para o perfil público de um autor verificado. Isso URL é enviado pelo autor.

Tipo: string

Obrigatório: falso

spdxLicenseId

Um identificador válido de <https://spdx.org/licenses/>.

Tipo: string

Obrigatório: falso

licenseUrl

Um link para um arquivo de licença do aplicativo que corresponde ao valor do spdxLicense ID do seu aplicativo.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

readmeUrl

Um link para o arquivo readme na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

labels

Rótulos para melhorar a descoberta de aplicativos nos resultados de pesquisa.

Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de etiquetas: 10

Padrão: `^[a-zA-Z0-9+\\-_:\\V@]+`;

Tipo: Matriz do tipo string

Obrigatório: falso

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: falso

homePageUrl

A URL com mais informações sobre o aplicativo, por exemplo, a localização do seu GitHub repositório para o aplicativo.

Tipo: string

Obrigatório: falso

version

Informações sobre a versão do aplicativo.

Tipo: [Versão](#)

Obrigatório: falso

ApplicationPage

Uma lista dos detalhes do aplicativo.

applications

Uma série de resumos de aplicativos.

Tipo: Matriz do tipo [ApplicationSummary](#)

Obrigatório: verdadeiro

nextToken

O token para recuperação do próximo conjunto de resultados.

Tipo: string

Obrigatório: falso

ApplicationSummary

Resumo dos detalhes sobre o aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

name

O nome da aplicação.

Comprimento mínimo = 1. Comprimento máximo = 140

Padrão: "[a-zA-Z0-9\-\-]+";

Tipo: string

Obrigatório: verdadeiro

description

A descrição do aplicativo.

Comprimento mínimo = 1. Comprimento máximo = 256

Tipo: string

Obrigatório: verdadeiro

author

O nome do autor que publica o aplicativo.

Comprimento mínimo = 1. Tamanho máximo = 127.

Padrão "`^[a-z0-9]([a-z0-9] | - (?! -)) * [a-z0-9]? $`";

Tipo: string

Obrigatório: verdadeiro

spdxLicenseId

Um identificador válido de <https://spdx.org/licenses/>.

Tipo: string

Obrigatório: falso

labels

Rótulos para melhorar a descoberta de aplicativos nos resultados de pesquisa.

Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de etiquetas: 10

Padrão: `^[a-zA-Z0-9+\\-_:\\V@]+$`;

Tipo: Matriz do tipo string

Obrigatório: falso

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: falso

homePageUrl

A URL com mais informações sobre o aplicativo, por exemplo, a localização do seu GitHub repositório para o aplicativo.

Tipo: string

Obrigatório: falso

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

Capability

Valores que devem ser especificados para implantar alguns aplicativos.

CAPABILITY_IAM

CAPABILITY_NAMED_IAM

CAPABILITY_AUTO_EXPAND

CAPABILITY_RESOURCE_POLICY

ConflictException

O recurso já existe.

message

O recurso já existe.

Tipo: string

Obrigatório: falso

errorCode

409

Tipo: string

Obrigatório: falso

CreateApplicationInput

Crie uma solicitação de aplicativo.

name

O nome do aplicativo que você deseja publicar.

Comprimento mínimo = 1. Comprimento máximo = 140

Padrão: "[a-zA-Z0-9\\-]+";

Tipo: string

Obrigatório: verdadeiro

description

A descrição do aplicativo.

Comprimento mínimo = 1. Comprimento máximo = 256

Tipo: string

Obrigatório: verdadeiro

author

O nome do autor que publica o aplicativo.

Comprimento mínimo = 1. Tamanho máximo = 127.

Padrão "`^[a-z0-9]([a-z0-9]|-(?!-))*[a-z0-9]?$`";

Tipo: string

Obrigatório: verdadeiro

spdxLicenseId

Um identificador válido de <https://spdx.org/licenses/>.

Tipo: string

Obrigatório: falso

licenseBody

Um arquivo de texto local que contém a licença do aplicativo que corresponde ao valor da spdxLicense ID do seu aplicativo. O arquivo tem o formato `file://<path>/<filename>`.

Tamanho máximo 5 MB

Você pode especificar somente um dos `licenseBody` e `licenseUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

licenseUrl

Um link para o objeto do S3 que contém a licença do aplicativo que corresponde ao valor da `spdxLicense ID` do seu aplicativo.

Tamanho máximo 5 MB

Você pode especificar somente um dos `licenseBody` e `licenseUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

readmeBody

Um arquivo `readme` de texto local na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona. O arquivo tem o formato `file://<path>/<filename>`.

Tamanho máximo 5 MB

Você pode especificar somente um dos `readmeBody` e `readmeUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

readmeUrl

Um link para o objeto S3 na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona.

Tamanho máximo 5 MB

Você pode especificar somente um dos `readmeBody` e `readmeUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

labels

Rótulos para melhorar a descoberta de aplicativos nos resultados de pesquisa.

Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de etiquetas: 10

Padrão: `^[a-zA-Z0-9+\\-_:\\V@]+$`;

Tipo: Matriz do tipo string

Obrigatório: falso

homePageUrl

A URL com mais informações sobre o aplicativo, por exemplo, a localização do seu GitHub repositório para o aplicativo.

Tipo: string

Obrigatório: falso

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: falso

templateBody

O pacote bruto local AWS SAM arquivo de modelo do seu aplicativo. O arquivo tem o formato `file://<path>/<filename>`.

Você pode especificar somente um dos `templateBody` e `templateUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

templateUrl

Um link para o objeto S3 contendo o pacote AWS SAM modelo do seu aplicativo.

Você pode especificar somente um dos `templateBody` e `templateUrl`; caso contrário, ocorrerá um erro.

Tipo: string

Obrigatório: falso

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

sourceCodeArchiveUrl

Um link para o objeto S3 que contém o ZIP arquivo do código-fonte dessa versão do seu aplicativo.

Tamanho máximo 50 MB

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

ParameterDefinition

Parâmetros suportados pelo aplicativo.

name

O nome do parâmetro.

Tipo: string

Obrigatório: verdadeiro

defaultValue

Um valor do tipo apropriado para o modelo a ser usado se nenhum valor for especificado quando uma pilha é criada. Se definir restrições para o parâmetro, você deverá especificar um valor que esteja de acordo com essas restrições.

Tipo: string

Obrigatório: falso

description

Uma sequência de até 4.000 caracteres que descreve o parâmetro.

Tipo: string

Obrigatório: falso

type

O tipo do parâmetro.

Valores válidos: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

String: Uma sequência de caracteres literal.

Por exemplo, os usuários podem especificar `"MyUserName"`.

Number: Um número inteiro ou flutuante. AWS CloudFormation valida o valor do parâmetro como um número. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função `Ref` intrínseca), o valor do parâmetro se torna uma string.

Por exemplo, os usuários podem especificar `"8888"`.

List<Number>: uma matriz de números inteiros ou flutuantes separados por vírgulas. AWS CloudFormation valida o valor do parâmetro como números. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função `Ref` intrínseca), o valor do parâmetro se torna uma lista de cadeias de caracteres.

Por exemplo, os usuários podem especificar `"80,20"` e, em seguida, resultar em `Ref. ["80", "20"]`

CommaDelimitedList: uma matriz de cadeias de caracteres literais separadas por vírgulas. O total de sequências deve ser um número a mais que o número total de vírgulas. Além disso, cada sequência de membros é cortada com espaço.

Por exemplo, os usuários podem especificar `"test, dev, prod"` e, em seguida, `Ref` resultar em `["test", "dev", "prod"]`.

Tipo: string

Obrigatório: falso

noEcho

Se o valor do parâmetro deve ser mascarado sempre que alguém fizer uma chamada que descreva a pilha. Se você definir o valor como verdadeiro, o valor do parâmetro será mascarado com asteriscos (`*****`).

Tipo: booleano

Obrigatório: falso

allowedPattern

Uma expressão regular que representa os padrões a serem permitidos para tipos `String`.

Tipo: string

Obrigatório: falso

constraintDescription

Uma sequência que explica uma restrição quando a restrição é violada. Por exemplo, sem uma descrição da restrição, um parâmetro que tem um padrão permitido de `[A-Za-z0-9]+` exibe a seguinte mensagem de erro quando o usuário especifica um valor inválido:

```
Malformed input-Parameter MyParameter must match pattern [A-Za-z0-9]+
```

Ao adicionar uma descrição de restrição, como “deve conter somente letras maiúsculas e minúsculas e números”, você pode exibir a seguinte mensagem de erro personalizada:

```
Malformed input-Parameter MyParameter must contain only uppercase and lowercase letters and numbers.
```

Tipo: string

Obrigatório: falso

minValue

Um valor numérico que determina o menor valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

maxValue

Um valor numérico que determina o maior valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

minLength

Um valor inteiro que determina o menor número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

maxLength

Um valor inteiro que determina o maior número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

allowedValues

Uma matriz que contém a lista de valores permitidos para o parâmetro.

Tipo: Matriz do tipo string

Obrigatório: falso

referencedByResources

Uma lista de AWS SAM recursos que usam esse parâmetro.

Tipo: Matriz do tipo string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Version

Detalhes da versão do aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

sourceCodeArchiveUrl

Um link para o objeto S3 que contém o ZIP arquivo do código-fonte dessa versão do seu aplicativo.

Tamanho máximo 50 MB

Tipo: string

Obrigatório: falso

templateUrl

Um link para o pacote AWS SAM modelo do seu aplicativo.

Tipo: string

Obrigatório: verdadeiro

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

parameterDefinitions

Uma matriz de tipos de parâmetros suportados pelo aplicativo.

Tipo: Matriz do tipo [ParameterDefinition](#)

Obrigatório: verdadeiro

requiredCapabilities

Uma lista de valores que você deve especificar antes de poder implantar determinados aplicativos. Alguns aplicativos podem incluir recursos que podem afetar as permissões em seu AWS conta, por exemplo, criando uma nova AWS Identity and Access Management (IAM) usuários. Para esses aplicativos, você deve reconhecer explicitamente seus recursos especificando esse parâmetro.

Os únicos valores válidos são CAPABILITY_IAM

CAPABILITY_NAMED_IAMCAPABILITY_RESOURCE_POLICY,, CAPABILITY_AUTO_EXPAND e.

Os recursos a seguir exigem que você especifique CAPABILITY_IAM ou CAPABILITY_NAMED_IAM [AWS::IAM: :Group](#), [AWS::IAM:: InstanceProfile](#) :Policy e [IAM:: AWSIAM: :Role](#). AWS

Se o aplicativo contiver IAM recursos, você poderá especificar CAPABILITY_IAM ouCAPABILITY_NAMED_IAM. Se o aplicativo contiver IAM recursos com nomes personalizados, você deverá especificarCAPABILITY_NAMED_IAM.

Os recursos a seguir exigem que você especifique `CAPABILITY_RESOURCE_POLICY`:
`AWS::Lambda::Permission`, `AWS::IAM::Policy`, `AWS::ApplicationAutoScaling::ScalingPolicy`,
`AWS::S3::BucketPolicy`, `AWS::SQS::QueuePolicy` e `AWS::SNS::TopicPolicy`

Os aplicativos que contêm um ou mais aplicativos aninhados exigem que você especifique `CAPABILITY_AUTO_EXPAND`.

Se seu modelo de aplicativo contiver algum dos recursos acima, recomendamos que você revise todas as permissões associadas ao aplicativo antes da implantação. Se você não especificar esse parâmetro para um aplicativo que requer recursos, a chamada falhará.

Tipo: Matriz do tipo [Recurso](#)

Obrigatório: verdadeiro

`resourcesSupported`

Se todos os recursos AWS contidos nesse aplicativo são suportados na região em que ele está sendo recuperado.

Tipo: booleano

Obrigatório: verdadeiro

Consulte também

Para obter mais informações sobre como usar essa API em um dos idiomas específicos AWS SDKs e referências, veja o seguinte:

ListApplications

- [AWS Interface da linha de comando](#)
- [AWS SDK para .NET](#)
- [AWS SDK para C++](#)
- [AWS SDK para Go v2](#)
- [AWS SDK para Java V2](#)
- [AWS SDK para JavaScript V3](#)
- [AWS SDK para PHP V3](#)
- [AWS SDK para Python](#)

- [AWSSDKpara Ruby V3](#)

CreateApplication

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

Aplicações applicationId

URI

/applications/*applicationId*

HTTPmétodos

GET

ID da operação: GetApplication

Obtém o aplicativo especificado.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Parâmetros de consulta

Nome	Tipo	Obrigatório	Descrição
<code>semanticVersion</code>	Cadeia de caracteres	Falso	A versão semântica do aplicativo a ser obtida.

Respostas

Código de status	Modelo de resposta	Descrição
200	<u>Application</u>	Bem-sucedida
400	<u>BadRequestException</u>	Um dos parâmetros na solicitação é inválido.
403	<u>ForbiddenException</u>	O cliente não está autenticado.
404	<u>NotFoundException</u>	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	<u>TooManyRequestsException</u>	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	<u>InternalServerErrorException</u>	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

DELETE

ID da operação: `DeleteApplication`

Exclui o aplicativo especificado.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
204	Nenhum	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
409	ConflictException	O recurso já existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

PATCH

ID da operação: UpdateApplication

Atualiza o aplicativo especificado.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Application	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.

Código de status	Modelo de resposta	Descrição
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
409	ConflictException	O recurso já existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Esquemas

Corpos da solicitação

PATCH Esquema

```
{
  "description": "string",
  "author": "string",
  "readmeBody": "string",
  "readmeUrl": "string",
  "labels": [
    "string"
  ],
  "homePageUrl": "string"
}
```

Corpos da resposta

Application Esquema

```
{
  "applicationId": "string",
  "name": "string",
  "description": "string",
  "author": "string",
  "isVerifiedAuthor": boolean,
  "verifiedAuthorUrl": "string",
  "spdxLicenseId": "string",
  "licenseUrl": "string",
  "readmeUrl": "string",
  "labels": [
    "string"
  ],
  "creationTime": "string",
  "homePageUrl": "string",
  "version": {
    "applicationId": "string",
    "semanticVersion": "string",
    "sourceCodeUrl": "string",
    "sourceCodeArchiveUrl": "string",
    "templateUrl": "string",
    "creationTime": "string",
    "parameterDefinitions": [
      {
        "name": "string",
        "defaultValue": "string",
        "description": "string",
        "type": "string",
        "noEcho": boolean,
        "allowedPattern": "string",
        "constraintDescription": "string",
        "minValue": integer,
        "maxValue": integer,
        "minLength": integer,
        "maxLength": integer,
        "allowedValues": [
          "string"
        ]
      }
    ],
    "referencedByResources": [
      "string"
    ]
  }
}
```

```
    ]
  }
],
"requiredCapabilities": [
  enum
],
"resourcesSupported": boolean
}
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

NotFoundException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ConflictException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

TooManyRequestsException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

InternalServerErrorException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

Propriedades

Application

Detalhes sobre o aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

name

O nome da aplicação.

Comprimento mínimo = 1. Comprimento máximo = 140

Padrão: "[a-zA-Z0-9\-\-]+";

Tipo: string

Obrigatório: verdadeiro

description

A descrição do aplicativo.

Comprimento mínimo = 1. Comprimento máximo = 256

Tipo: string

Obrigatório: verdadeiro

author

O nome do autor que publica o aplicativo.

Comprimento mínimo = 1. Tamanho máximo = 127.

Padrão “`^[a-z0-9]([a-z0-9]|-(?!-))*[a-z0-9]?$`”;

Tipo: string

Obrigatório: verdadeiro

isVerifiedAuthor

Especifica se o autor desse aplicativo foi verificado. Isso significa que AWS fez uma análise de boa fé, como prestador de serviços razoável e prudente, das informações fornecidas pelo solicitante e confirmou que a identidade do solicitante é a reivindicada.

Tipo: booleano

Obrigatório: falso

verifiedAuthorUrl

O URL para o perfil público de um autor verificado. Isso URL é enviado pelo autor.

Tipo: string

Obrigatório: falso

spdxLicenseId

Um identificador válido de <https://spdx.org/licenses/>.

Tipo: string

Obrigatório: falso

licenseUrl

Um link para um arquivo de licença do aplicativo que corresponde ao valor do spdxLicense ID do seu aplicativo.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

readmeUrl

Um link para o arquivo readme na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

labels

Rótulos para melhorar a descoberta de aplicativos nos resultados de pesquisa.

Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de etiquetas: 10

Padrão: `^[a-zA-Z0-9+\\-_:\\V@]+$`;

Tipo: Matriz do tipo string

Obrigatório: falso

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: falso

homePageUrl

A URL com mais informações sobre o aplicativo, por exemplo, a localização do seu GitHub repositório para o aplicativo.

Tipo: string

Obrigatório: falso

version

Informações sobre a versão do aplicativo.

Tipo: [Versão](#)

Obrigatório: falso

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

Capability

Valores que devem ser especificados para implantar alguns aplicativos.

CAPABILITY_IAM

CAPABILITY_NAMED_IAM
CAPABILITY_AUTO_EXPAND
CAPABILITY_RESOURCE_POLICY

ConflictException

O recurso já existe.

message

O recurso já existe.

Tipo: string

Obrigatório: falso

errorCode

409

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

ParameterDefinition

Parâmetros suportados pelo aplicativo.

name

O nome do parâmetro.

Tipo: string

Obrigatório: verdadeiro

defaultValue

Um valor do tipo apropriado para o modelo a ser usado se nenhum valor for especificado quando uma pilha é criada. Se definir restrições para o parâmetro, você deverá especificar um valor que esteja de acordo com essas restrições.

Tipo: string

Obrigatório: falso

description

Uma sequência de até 4.000 caracteres que descreve o parâmetro.

Tipo: string

Obrigatório: falso

type

O tipo do parâmetro.

Valores válidos: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

`String`: Uma sequência de caracteres literal.

Por exemplo, os usuários podem especificar `"MyUserName"`.

`Number`: Um número inteiro ou flutuante. AWS CloudFormation valida o valor do parâmetro como um número. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função `Ref` intrínseca), o valor do parâmetro se torna uma string.

Por exemplo, os usuários podem especificar "8888".

List<Number>: uma matriz de números inteiros ou flutuantes separados por vírgulas. AWS CloudFormation valida o valor do parâmetro como números. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função Ref intrínseca), o valor do parâmetro se torna uma lista de cadeias de caracteres.

Por exemplo, os usuários podem especificar "80,20" e, em seguida, resultar em Ref. ["80", "20"]

CommaDelimitedList: uma matriz de cadeias de caracteres literais separadas por vírgulas. O total de sequências deve ser um número a mais que o número total de vírgulas. Além disso, cada sequência de membros é cortada com espaço.

Por exemplo, os usuários podem especificar "test, dev, prod" e, em seguida, Ref resultar em ["test", "dev", "prod"].

Tipo: string

Obrigatório: falso

noEcho

Se o valor do parâmetro deve ser mascarado sempre que alguém fizer uma chamada que descreva a pilha. Se você definir o valor como verdadeiro, o valor do parâmetro será mascarado com asteriscos (*****).

Tipo: booliano

Obrigatório: falso

allowedPattern

Uma expressão regular que representa os padrões a serem permitidos para tipos String.

Tipo: string

Obrigatório: falso

constraintDescription

Uma sequência que explica uma restrição quando a restrição é violada. Por exemplo, sem uma descrição da restrição, um parâmetro que tem um padrão permitido de [A-Za-z0-9]+ exibe a seguinte mensagem de erro quando o usuário especifica um valor inválido:

Malformed input-Parameter MyParameter must match pattern [A-Za-z0-9]+

Ao adicionar uma descrição de restrição, como “deve conter somente letras maiúsculas e minúsculas e números”, você pode exibir a seguinte mensagem de erro personalizada:

Malformed input-Parameter MyParameter must contain only uppercase and lowercase letters and numbers.

Tipo: string

Obrigatório: falso

minValue

Um valor numérico que determina o menor valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

maxValue

Um valor numérico que determina o maior valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

minLength

Um valor inteiro que determina o menor número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

maxLength

Um valor inteiro que determina o maior número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

allowedValues

Uma matriz que contém a lista de valores permitidos para o parâmetro.

Tipo: Matriz do tipo string

Obrigatório: falso

referencedByResources

Uma lista de AWS SAM recursos que usam esse parâmetro.

Tipo: Matriz do tipo string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

UpdateApplicationInput

Atualize a solicitação do aplicativo.

description

A descrição do aplicativo.

Comprimento mínimo = 1. Comprimento máximo = 256

Tipo: string

Obrigatório: falso

author

O nome do autor que publica o aplicativo.

Comprimento mínimo = 1. Tamanho máximo = 127.

Padrão “`^[a-z0-9] ([a-z0-9] | - (?! -)) * [a-z0-9]? $`”;

Tipo: string

Obrigatório: falso

readmeBody

Um arquivo readme de texto na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

readmeUrl

Um link para o arquivo readme na linguagem Markdown que contém uma descrição mais detalhada do aplicativo e de como ele funciona.

Tamanho máximo 5 MB

Tipo: string

Obrigatório: falso

labels

Rótulos para melhorar a descoberta de aplicativos nos resultados de pesquisa.

Comprimento mínimo = 1. Tamanho máximo = 127. Número máximo de etiquetas: 10

Padrão: `^[a-zA-Z0-9+\\-_:\\W@]+$`;

Tipo: Matriz do tipo string

Obrigatório: falso

homePageUrl

A URL com mais informações sobre o aplicativo, por exemplo, a localização do seu GitHub repositório para o aplicativo.

Tipo: string

Obrigatório: falso

Version

Detalhes da versão do aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

sourceCodeArchiveUrl

Um link para o objeto S3 que contém o ZIP arquivo do código-fonte dessa versão do seu aplicativo.

Tamanho máximo 50 MB

Tipo: string

Obrigatório: falso

templateUrl

Um link para o pacote AWS SAM modelo do seu aplicativo.

Tipo: string

Obrigatório: verdadeiro

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

parameterDefinitions

Uma matriz de tipos de parâmetros suportados pelo aplicativo.

Tipo: Matriz do tipo [ParameterDefinition](#)

Obrigatório: verdadeiro

requiredCapabilities

Uma lista de valores que você deve especificar antes de poder implantar determinados aplicativos. Alguns aplicativos podem incluir recursos que podem afetar as permissões em sua AWS conta, por exemplo, criando uma nova AWS Identity and Access Management (IAM) usuários. Para esses aplicativos, você deve reconhecer explicitamente seus recursos especificando esse parâmetro.

Os únicos valores válidos são `CAPABILITY_IAM`

`CAPABILITY_NAMED_IAM`, `CAPABILITY_RESOURCE_POLICY`, `CAPABILITY_AUTO_EXPAND` e.

Os recursos a seguir exigem que você especifique `CAPABILITY_IAM` ou `CAPABILITY_NAMED_IAM`

[AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), [AWS::IAM::Role](#).

Se o aplicativo contiver IAM recursos, você poderá especificar `CAPABILITY_IAM`

ou `CAPABILITY_NAMED_IAM`. Se o aplicativo contiver IAM recursos com nomes personalizados, você deverá especificar `CAPABILITY_NAMED_IAM`.

Os recursos a seguir exigem que você especifique `CAPABILITY_RESOURCE_POLICY`:

[AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#):

[AWS::S3::BucketPolicy](#), [AWS::SQS::QueuePolicy](#), [AWS::SNS::TopicPolicy](#)

Os aplicativos que contêm um ou mais aplicativos aninhados exigem que você especifique `CAPABILITY_AUTO_EXPAND`.

Se seu modelo de aplicativo contiver algum dos recursos acima, recomendamos que você revise todas as permissões associadas ao aplicativo antes da implantação. Se você não especificar esse parâmetro para um aplicativo que requer recursos, a chamada falhará.

Tipo: Matriz do tipo [Recurso](#)

Obrigatório: verdadeiro

resourcesSupported

Se todos os recursos contidos nesse aplicativo são suportados na região em que ele está sendo recuperado.

Tipo: booleano

Obrigatório: verdadeiro

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

GetApplication

- [AWS Interface da linha de comando](#)
- [AWSSDKpara. NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

DeleteApplication

- [AWS Interface da linha de comando](#)
- [AWSSDKpara. NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

UpdateApplication

- [AWS Interface da linha de comando](#)
- [AWSSDKpara. NET](#)

- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

Conjuntos de applicationId alterações de aplicativos

URI

/applications/*applicationId*/changesets

HTTPmétodos

POST

ID da operação: CreateCloudFormationChangeSet

Cria um AWS CloudFormation conjunto de alterações para o aplicativo em questão.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
201	ChangeSetDetails	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.

Código de status	Modelo de resposta	Descrição
403	ForbiddenException	O cliente não está autenticado.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

POST Esquema

```
{
```

```
"stackName": "string",
"semanticVersion": "string",
"templateId": "string",
"parameterOverrides": [
  {
    "name": "string",
    "value": "string"
  }
],
"capabilities": [
  "string"
],
"changeSetName": "string",
"clientToken": "string",
"description": "string",
"notificationArns": [
  "string"
],
"resourceTypes": [
  "string"
],
"rollbackConfiguration": {
  "rollbackTriggers": [
    {
      "arn": "string",
      "type": "string"
    }
  ],
  "monitoringTimeInMinutes": integer
},
"tags": [
  {
    "key": "string",
    "value": "string"
  }
]
}
```

Corpos da resposta

ChangeSetDetails Esquema

```
{
```

```
"applicationId": "string",
"semanticVersion": "string",
"changeSetId": "string",
"stackId": "string"
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

TooManyRequestsException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

InternalServerErrorException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

Propriedades

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ChangeSetDetails

Detalhes do conjunto de alterações.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

changeSetId

O Amazon Resource Name (ARN) do conjunto de alterações.

Restrições de tamanho: tamanho mínimo 1.

PadrãoARN: [-a-zA-z0-9:/] *

Tipo: string

Obrigatório: verdadeiro

stackId

O ID exclusivo da pilha.

Tipo: string

Obrigatório: verdadeiro

CreateCloudFormationChangeSetInput

Crie uma solicitação de conjunto de alterações do aplicativo.

stackName

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: falso

templateId

O UUID retornado por CreateCloudFormationTemplate.

Padrão: [0-9a-fa-f] {8}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {12}

Tipo: string

Obrigatório: falso

parameterOverrides

Uma lista de valores de parâmetros para os parâmetros do aplicativo.

Tipo: Matriz do tipo [ParameterValue](#)

Obrigatório: falso

capabilities

Uma lista de valores que você deve especificar antes de poder implantar determinados aplicativos. Alguns aplicativos podem incluir recursos que podem afetar as permissões em seu AWS conta, por exemplo, criando uma nova AWS Identity and Access Management (IAM) usuários. Para esses aplicativos, você deve reconhecer explicitamente seus recursos especificando esse parâmetro.

Os únicos valores válidos são CAPABILITY_IAM

CAPABILITY_NAMED_IAMCAPABILITY_RESOURCE_POLICY,, CAPABILITY_AUTO_EXPAND e.

Os recursos a seguir exigem que você especifique CAPABILITY_IAM ou CAPABILITY_NAMED_IAM [AWS::IAM: :Group](#), [AWS::IAM:: InstanceProfile](#) :Policy eIAM:: AWSIAM: :Role. AWS

Se o aplicativo contiver IAM recursos, você poderá especificar CAPABILITY_IAM ouCAPABILITY_NAMED_IAM. Se o aplicativo contiver IAM recursos com nomes personalizados, você deverá especificarCAPABILITY_NAMED_IAM.

Os recursos a seguir exigem que você especifiqueCAPABILITY_RESOURCE_POLICY:

[AWS: :Lambda: :Permission](#), [AWS: ::PolicyIAM,:: AWS::ApplicationAutoScaling:,ScalingPolicy:](#)

[AWSS3:,::: eBucketPolicy: AWS: SQS: QueuePolicy AWS SNS TopicPolicy](#)

Os aplicativos que contêm um ou mais aplicativos aninhados exigem que você especifique CAPABILITY_AUTO_EXPAND.

Se seu modelo de aplicativo contiver algum dos recursos acima, recomendamos que você revise todas as permissões associadas ao aplicativo antes da implantação. Se você não especificar esse parâmetro para um aplicativo que requer recursos, a chamada falhará.

Tipo: Matriz do tipo string

Obrigatório: falso

changeSetName

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: string

Obrigatório: falso

clientToken

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: string

Obrigatório: falso

description

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: string

Obrigatório: falso

notificationArns

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: Matriz do tipo string

Obrigatório: falso

resourceTypes

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: Matriz do tipo string

Obrigatório: falso

rollbackConfiguration

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: [RollbackConfiguration](#)

Obrigatório: falso

tags

Essa propriedade corresponde ao parâmetro de mesmo nome para o AWS CloudFormation [CreateChangeSet](#) API.

Tipo: Matriz do tipo [Tag](#)

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

ParameterValue

Valor do parâmetro do aplicativo.

name

A chave associada ao parâmetro. Se você não especificar uma chave e um valor para um parâmetro específico, AWS CloudFormation usa o valor padrão especificado no seu modelo.

Tipo: string

Obrigatório: verdadeiro

value

O valor de entrada associado ao parâmetro.

Tipo: string

Obrigatório: verdadeiro

RollbackConfiguration

Esta propriedade corresponde ao AWS CloudFormation [RollbackConfiguration](#) Tipo de dados.

rollbackTriggers

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation [RollbackConfiguration](#) Tipo de dados.

Tipo: Matriz do tipo [RollbackTrigger](#)

Obrigatório: falso

monitoringTimeInMinutes

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation [RollbackConfiguration](#) Tipo de dados.

Tipo: inteiro

Obrigatório: falso

RollbackTrigger

Esta propriedade corresponde ao AWS CloudFormation [RollbackTrigger](#) Tipo de dados.

arn

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation [RollbackTrigger](#) Tipo de dados.

Tipo: string

Obrigatório: verdadeiro

type

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation [RollbackTrigger](#) Tipo de dados.

Tipo: string

Obrigatório: verdadeiro

Tag

Esta propriedade corresponde ao AWS CloudFormation Tipo de dados da [tag](#).

key

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation Tipo de dados da [tag](#).

Tipo: string

Obrigatório: verdadeiro

value

Essa propriedade corresponde ao conteúdo do mesmo nome para o AWS CloudFormation Tipo de dados da [tag](#).

Tipo: string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

CreateCloudFormationChangeSet

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

Applications applicationId Dependencies

URI

/applications/*applicationId*/dependencies

HTTPmétodos

GET

ID da operação: ListApplicationDependencies

Recupera a lista de aplicativos aninhados no aplicativo que o contém.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Parâmetros de consulta

Nome	Tipo	Obrigatório	Descrição
nextToken	Cadeia de caracteres	Falso	Um token para especificar onde iniciar a paginação.
maxItems	String	Falso	O número total de itens a serem devolvidos.
semanticVersion	String	Falso	A versão semântica do aplicativo a ser obtida.

Respostas

Código de status	Modelo de resposta	Descrição
200	ApplicationDependencyPage	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Código de status	Modelo de resposta	Descrição
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da resposta

ApplicationDependencyPage Esquema

```
{
  "dependencies": [
    {
      "applicationId": "string",
      "semanticVersion": "string"
    }
  ],
  "nextToken": "string"
}
```

BadRequestException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

ForbiddenException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

NotFoundException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

Propriedades

ApplicationDependencyPage

Uma lista de resumos de aplicativos aninhados no aplicativo.

dependencies

Uma matriz de resumos de aplicativos aninhados no aplicativo.

Tipo: Matriz do tipo [ApplicationDependencySummary](#)

Obrigatório: verdadeiro

nextToken

O token para recuperação do próximo conjunto de resultados.

Tipo: string

Obrigatório: falso

ApplicationDependencySummary

Um resumo aninhado do aplicativo.

applicationId

O nome do recurso Amazon (ARN) do aplicativo aninhado.

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo aninhado.

Tipo: string

Obrigatório: verdadeiro

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

ListApplicationDependencies

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

applicationId Política de aplicativos

URI

/applications/*applicationId*/policy

HTTPmétodos

GET

ID da operação: `GetApplicationPolicy`

Recupera a política do aplicativo.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i><code>applicationId</code></i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	ApplicationPolicy	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o

Código de status	Modelo de resposta	Descrição
		serviço encontrou um erro interno.

PUT

ID da operação: PutApplicationPolicy

Define a política de permissão para um aplicativo. Para ver a lista de ações suportadas por essa operação, consulte [Permissões do aplicativo](#).

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	ApplicationPolicy	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de

Código de status	Modelo de resposta	Descrição
500	InternalServerErrorException	solicições por unidade de tempo. A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

PUT Esquema

```
{
  "statements": [
    {
      "statementId": "string",
      "principals": [
        "string"
      ],
    },
  ],
}
```

```
    "actions": [
      "string"
    ],
    "principalOrgIDs": [
      "string"
    ]
  }
]
```

Corpos da resposta

ApplicationPolicy Esquema

```
{
  "statements": [
    {
      "statementId": "string",
      "principals": [
        "string"
      ],
      "actions": [
        "string"
      ],
      "principalOrgIDs": [
        "string"
      ]
    }
  ]
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
```

```
"message": "string",  
"errorCode": "string"  
}
```

NotFoundException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

Propriedades

ApplicationPolicy

Declarações de política aplicadas ao aplicativo.

statements

Uma série de declarações de política aplicadas ao aplicativo.

Tipo: Matriz do tipo [ApplicationPolicyStatement](#)

Obrigatório: verdadeiro

ApplicationPolicyStatement

Declaração de política aplicada ao aplicativo.

statementId

Um ID exclusivo para a declaração.

Tipo: string

Obrigatório: falso

principals

Uma variedade de AWS conta com IDs a qual compartilhar o aplicativo ou * para tornar o aplicativo público.

Tipo: Matriz do tipo string

Obrigatório: verdadeiro

actions

Para ver a lista de ações suportadas por essa operação, consulte [Permissões do aplicativo](#).

Tipo: Matriz do tipo string

Obrigatório: verdadeiro

principalOrgIDs

A ferramenta AWS Organizations ID com a qual compartilhar o aplicativo.

Tipo: Matriz do tipo string

Obrigatório: falso

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

GetApplicationPolicy

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

PutApplicationPolicy

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)

- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

Applications applicationId Templates

URI

/applications/*applicationId*/templates

HTTPmétodos

POST

ID da operação: CreateCloudFormationTemplate

Cria um AWS CloudFormation modelo.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
201	TemplateDetails	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política

Código de status	Modelo de resposta	Descrição
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

POST Esquema

```
{
```

```
"semanticVersion": "string"
}
```

Corpos da resposta

TemplateDetails Esquema

```
{
  "templateId": "string",
  "templateUrl": "string",
  "applicationId": "string",
  "semanticVersion": "string",
  "status": enum,
  "creationTime": "string",
  "expirationTime": "string"
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

NotFoundException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

TooManyRequestsException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

InternalServerErrorException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

Propriedades

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

CreateCloudFormationTemplateInput

Crie uma solicitação de modelo.

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TemplateDetails

Detalhes do modelo.

templateId

O UUID retornado por `CreateCloudFormationTemplate`.

Padrão: `[0-9a-fa-f] {8}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {12}`

Tipo: string

Obrigatório: verdadeiro

templateUrl

Um link para o modelo que pode ser usado para implantar o aplicativo usando AWS CloudFormation.

Tipo: string

Obrigatório: verdadeiro

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

status

Status do fluxo de trabalho de criação do modelo.

Valores possíveis: PREPARING | ACTIVE | EXPIRED

Tipo: string

Obrigatório: verdadeiro

Valores: PREPARING | ACTIVE | EXPIRED

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

expirationTime

A data e a hora em que esse modelo expira. Os modelos expiram 1 hora após a criação.

Tipo: string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

CreateCloudFormationTemplate

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)

- [AWSSDKpara Ruby V3](#)

Applications applicationId Templates templateId

URI

/applications/*applicationId*/templates/*templateId*

HTTPmétodos

GET

ID da operação: GetCloudFormationTemplate

Obtém o especificado AWS CloudFormation modelo.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.
<i>templateId</i>	String	Verdadeiro	O UUID retornado por CreateCloudFormationTemplate. Padrão: [0-9a-fa-f]{8}\ - [0-9a-fa-f]{4}\ - [0-9a-fa-f]{4}\ - [0-9a-fa-f]{4}\ - [0-9a-fa-f]{12}

Respostas

Código de status	Modelo de resposta	Descrição
200	TemplateDetails	Bem-sucedida

Código de status	Modelo de resposta	Descrição
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.
<i>templateId</i>	String	Verdadeiro	O UUID retornado por <code>CreateCloudFormationTemplate</code> . Padrão: [0-9a-fa-f]{8} - [0-9a-fa-f]{4} - [0-9a-fa-f]{4} - [0-9a-fa-f]{8}

Nome	Tipo	Obrigatório	Descrição
			[0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {12}
Respostas			
Código de status		Modelo de resposta	Descrição
200		Nenhum	200 respostas

Esquemas

Corpos da resposta

TemplateDetails Esquema

```
{
  "templateId": "string",
  "templateUrl": "string",
  "applicationId": "string",
  "semanticVersion": "string",
  "status": enum,
  "creationTime": "string",
  "expirationTime": "string"
}
```

BadRequestException Esquema

```
{
  "message": "string",
  "errorCode": "string"
}
```

ForbiddenException Esquema

```
{
```

```
"message": "string",  
"errorCode": "string"  
}
```

NotFoundException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

Propriedades

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TemplateDetails

Detalhes do modelo.

templateId

O UUID retornado por `CreateCloudFormationTemplate`.

Padrão: `[0-9a-fa-f] {8}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {4}\ - [0-9a-fa-f] {12}`

Tipo: string

Obrigatório: verdadeiro

templateUrl

Um link para o modelo que pode ser usado para implantar o aplicativo usando AWS CloudFormation.

Tipo: string

Obrigatório: verdadeiro

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

status

Status do fluxo de trabalho de criação do modelo.

Valores possíveis: PREPARING | ACTIVE | EXPIRED

Tipo: string

Obrigatório: verdadeiro

Valores: PREPARING | ACTIVE | EXPIRED

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

expirationTime

A data e a hora em que esse modelo expira. Os modelos expiram 1 hora após a criação.

Tipo: string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDK e referências, veja o seguinte:

GetCloudFormationTemplate

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)

- [AWSSDKpara Ruby V3](#)

Applications applicationId Unshare

URI

/applications/*applicationId*/unshare

HTTPmétodos

POST

ID da operação: UnshareApplication

Cancela o compartilhamento de um aplicativo de um AWS Organização.

Essa operação só pode ser chamada da conta de gerenciamento da organização.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
204	Nenhum	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política

Código de status	Modelo de resposta	Descrição
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

POST Esquema

```
{
```

```
"organizationId": "string"  
}
```

Corpos da resposta

BadRequestException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

ForbiddenException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

NotFoundException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",
```

```
"errorCode": "string"  
}
```

Propriedades

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

UnshareApplicationInput

Cancelar o compartilhamento da solicitação do aplicativo.

organizationId

A ferramenta AWS Organizations ID da qual cancelar o compartilhamento do aplicativo.

Tipo: string

Obrigatório: verdadeiro

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

UnshareApplication

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)

- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

applicationId Versões de aplicativos

URI

/applications/*applicationId*/versions

HTTPmétodos

GET

ID da operação: ListApplicationVersions

Lista as versões do aplicativo especificado.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Parâmetros de consulta

Nome	Tipo	Obrigatório	Descrição
maxItems	Cadeia de caracteres	Falso	O número total de itens a serem devolvidos.

Nome	Tipo	Obrigatório	Descrição
nextToken	String	Falso	Um token para especificar onde iniciar a paginação.

Respostas

Código de status	Modelo de resposta	Descrição
200	ApplicationVersionPage	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
404	NotFoundException	O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da resposta

ApplicationVersionPage Esquema

```
{
  "versions": [
    {
      "applicationId": "string",
      "semanticVersion": "string",
      "sourceCodeUrl": "string",
      "creationTime": "string"
    }
  ],
  "nextToken": "string"
}
```

BadRequestException Esquema

```
{
  "message": "string",
}
```

```
"errorCode": "string"  
}
```

ForbiddenException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

NotFoundException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

Propriedades

ApplicationVersionPage

Uma lista de resumos de versões do aplicativo.

versions

Uma série de resumos de versões do aplicativo.

Tipo: Matriz do tipo [VersionSummary](#)

Obrigatório: verdadeiro

nextToken

O token para recuperação do próximo conjunto de resultados.

Tipo: string

Obrigatório: falso

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

NotFoundException

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

message

O recurso (por exemplo, uma declaração de política de acesso) especificado na solicitação não existe.

Tipo: string

Obrigatório: falso

errorCode

404

Tipo: string

Obrigatório: falso

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

VersionSummary

Um resumo da versão do aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

ListApplicationVersions

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)

- [AWSSDKpara Ruby V3](#)

applicationId Versões de aplicativos semanticVersion

URI

/applications/*applicationId*/versions/*semanticVersion*

HTTPmétodos

PUT

ID da operação: CreateApplicationVersion

Cria uma versão do aplicativo.

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.
<i>semanticVersion</i>	String	Verdadeiro	A versão semântica da nova versão.

Respostas

Código de status	Modelo de resposta	Descrição
201	Version	Bem-sucedida
400	BadRequestException	Um dos parâmetros na solicitação é inválido.
403	ForbiddenException	O cliente não está autenticado.
409	ConflictException	O recurso já existe.

Código de status	Modelo de resposta	Descrição
429	TooManyRequestsException	O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.
500	InternalServerErrorException	A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

OPTIONS

Parâmetros de caminho

Nome	Tipo	Obrigatório	Descrição
<i>applicationId</i>	Cadeia de caracteres	Verdadeiro	O nome do recurso Amazon (ARN) do aplicativo.
<i>semanticVersion</i>	String	Verdadeiro	A versão semântica da nova versão.

Respostas

Código de status	Modelo de resposta	Descrição
200	Nenhum	200 respostas

Esquemas

Corpos da solicitação

PUT Esquema

```
{
```

```
"templateBody": "string",
"templateUrl": "string",
"sourceCodeUrl": "string",
"sourceCodeArchiveUrl": "string"
}
```

Corpos da resposta

Version Esquema

```
{
  "applicationId": "string",
  "semanticVersion": "string",
  "sourceCodeUrl": "string",
  "sourceCodeArchiveUrl": "string",
  "templateUrl": "string",
  "creationTime": "string",
  "parameterDefinitions": [
    {
      "name": "string",
      "defaultValue": "string",
      "description": "string",
      "type": "string",
      "noEcho": boolean,
      "allowedPattern": "string",
      "constraintDescription": "string",
      "minValue": integer,
      "maxValue": integer,
      "minLength": integer,
      "maxLength": integer,
      "allowedValues": [
        "string"
      ],
      "referencedByResources": [
        "string"
      ]
    }
  ],
  "requiredCapabilities": [
    enum
  ],
  "resourcesSupported": boolean
}
```

BadRequestException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

ForbiddenException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

ConflictException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

TooManyRequestsException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

InternalServerErrorException Esquema

```
{  
  "message": "string",  
  "errorCode": "string"  
}
```

Propriedades

BadRequestException

Um dos parâmetros na solicitação é inválido.

message

Um dos parâmetros na solicitação é inválido.

Tipo: string

Obrigatório: falso

errorCode

400

Tipo: string

Obrigatório: falso

Capability

Valores que devem ser especificados para implantar alguns aplicativos.

CAPABILITY_IAM

CAPABILITY_NAMED_IAM

CAPABILITY_AUTO_EXPAND

CAPABILITY_RESOURCE_POLICY

ConflictException

O recurso já existe.

message

O recurso já existe.

Tipo: string

Obrigatório: falso

errorCode

409

Tipo: string

Obrigatório: falso

CreateApplicationVersionInput

Crie uma solicitação de versão.

templateBody

O cru embalado AWS SAM modelo do seu aplicativo.

Tipo: string

Obrigatório: falso

templateUrl

Um link para o pacote AWS SAM modelo do seu aplicativo.

Tipo: string

Obrigatório: falso

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

sourceCodeArchiveUrl

Um link para o objeto S3 que contém o ZIP arquivo do código-fonte dessa versão do seu aplicativo.

Tamanho máximo 50 MB

Tipo: string

Obrigatório: falso

ForbiddenException

O cliente não está autenticado.

message

O cliente não está autenticado.

Tipo: string

Obrigatório: falso

errorCode

403

Tipo: string

Obrigatório: falso

InternalServerErrorException

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

message

A ferramenta AWS Serverless Application Repository o serviço encontrou um erro interno.

Tipo: string

Obrigatório: falso

errorCode

500

Tipo: string

Obrigatório: falso

ParameterDefinition

Parâmetros suportados pelo aplicativo.

name

O nome do parâmetro.

Tipo: string

Obrigatório: verdadeiro

defaultValue

Um valor do tipo apropriado para o modelo a ser usado se nenhum valor for especificado quando uma pilha é criada. Se definir restrições para o parâmetro, você deverá especificar um valor que esteja de acordo com essas restrições.

Tipo: string

Obrigatório: falso

description

Uma sequência de até 4.000 caracteres que descreve o parâmetro.

Tipo: string

Obrigatório: falso

type

O tipo do parâmetro.

Valores válidos: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

`String`: Uma sequência de caracteres literal.

Por exemplo, os usuários podem especificar `"MyUserName"`.

`Number`: Um número inteiro ou flutuante. AWS CloudFormation valida o valor do parâmetro como um número. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função `Ref` intrínseca), o valor do parâmetro se torna uma string.

Por exemplo, os usuários podem especificar "8888".

List<Number>: uma matriz de números inteiros ou flutuantes separados por vírgulas. AWS CloudFormation valida o valor do parâmetro como números. No entanto, quando você usa o parâmetro em outro lugar do seu modelo (por exemplo, usando a função Ref intrínseca), o valor do parâmetro se torna uma lista de cadeias de caracteres.

Por exemplo, os usuários podem especificar "80,20" e, em seguida, resultar em Ref. ["80", "20"]

CommaDelimitedList: uma matriz de cadeias de caracteres literais separadas por vírgulas. O total de sequências deve ser um número a mais que o número total de vírgulas. Além disso, cada sequência de membros é cortada com espaço.

Por exemplo, os usuários podem especificar "test, dev, prod" e, em seguida, Ref resultar em ["test", "dev", "prod"].

Tipo: string

Obrigatório: falso

noEcho

Se o valor do parâmetro deve ser mascarado sempre que alguém fizer uma chamada que descreva a pilha. Se você definir o valor como verdadeiro, o valor do parâmetro será mascarado com asteriscos (*****).

Tipo: booliano

Obrigatório: falso

allowedPattern

Uma expressão regular que representa os padrões a serem permitidos para tipos String.

Tipo: string

Obrigatório: falso

constraintDescription

Uma sequência que explica uma restrição quando a restrição é violada. Por exemplo, sem uma descrição da restrição, um parâmetro que tem um padrão permitido de [A-Za-z0-9]+ exibe a seguinte mensagem de erro quando o usuário especifica um valor inválido:

Malformed input-Parameter MyParameter must match pattern [A-Za-z0-9]+

Ao adicionar uma descrição de restrição, como “deve conter somente letras maiúsculas e minúsculas e números”, você pode exibir a seguinte mensagem de erro personalizada:

Malformed input-Parameter MyParameter must contain only uppercase and lowercase letters and numbers.

Tipo: string

Obrigatório: falso

minValue

Um valor numérico que determina o menor valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

maxValue

Um valor numérico que determina o maior valor numérico que você deseja permitir para `Number` tipos.

Tipo: inteiro

Obrigatório: falso

minLength

Um valor inteiro que determina o menor número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

maxLength

Um valor inteiro que determina o maior número de caracteres que você deseja permitir para `String` tipos.

Tipo: inteiro

Obrigatório: falso

allowedValues

Uma matriz que contém a lista de valores permitidos para o parâmetro.

Tipo: Matriz do tipo string

Obrigatório: falso

referencedByResources

Uma lista de AWS SAM recursos que usam esse parâmetro.

Tipo: Matriz do tipo string

Obrigatório: verdadeiro

TooManyRequestsException

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

message

O cliente está enviando mais do que o número permitido de solicitações por unidade de tempo.

Tipo: string

Obrigatório: falso

errorCode

429

Tipo: string

Obrigatório: falso

Version

Detalhes da versão do aplicativo.

applicationId

O nome de recurso da Amazon do aplicativo (ARN).

Tipo: string

Obrigatório: verdadeiro

semanticVersion

A versão semântica do aplicativo:

<https://semver.org/>

Tipo: string

Obrigatório: verdadeiro

sourceCodeUrl

Um link para um repositório público para o código-fonte do seu aplicativo, por exemplo, o URL de um GitHub commit específico.

Tipo: string

Obrigatório: falso

sourceCodeArchiveUrl

Um link para o objeto S3 que contém o ZIP arquivo do código-fonte dessa versão do seu aplicativo.

Tamanho máximo 50 MB

Tipo: string

Obrigatório: falso

templateUrl

Um link para o pacote AWS SAM modelo do seu aplicativo.

Tipo: string

Obrigatório: verdadeiro

creationTime

A data e a hora em que esse recurso foi criado.

Tipo: string

Obrigatório: verdadeiro

parameterDefinitions

Uma matriz de tipos de parâmetros suportados pelo aplicativo.

Tipo: Matriz do tipo [ParameterDefinition](#)

Obrigatório: verdadeiro

requiredCapabilities

Uma lista de valores que você deve especificar antes de poder implantar determinados aplicativos. Alguns aplicativos podem incluir recursos que podem afetar as permissões em seu AWS conta, por exemplo, criando uma nova AWS Identity and Access Management (IAM) usuários. Para esses aplicativos, você deve reconhecer explicitamente seus recursos especificando esse parâmetro.

Os únicos valores válidos são CAPABILITY_IAM

CAPABILITY_NAMED_IAMCAPABILITY_RESOURCE_POLICY,, CAPABILITY_AUTO_EXPAND e.

Os recursos a seguir exigem que você especifique CAPABILITY_IAM ou CAPABILITY_NAMED_IAM

[AWS::IAM: :Group](#), [AWS::IAM:: InstanceProfile :Policy](#) e [IAM:: AWSIAM: :Role](#). AWS

Se o aplicativo contiver IAM recursos, você poderá especificar CAPABILITY_IAM

ouCAPABILITY_NAMED_IAM. Se o aplicativo contiver IAM recursos com nomes personalizados, você deverá especificarCAPABILITY_NAMED_IAM.

[Os recursos a seguir exigem que você especifiqueCAPABILITY_RESOURCE_POLICY:](#)

[AWS: :Lambda: :Permission](#), [AWS: :PolicyIAM,:: AWS::ApplicationAutoScaling:,ScalingPolicy:](#)

[AWSS3::,,: e BucketPolicy AWS::SQS:. QueuePolicy AWS SNS TopicPolicy](#)

Os aplicativos que contêm um ou mais aplicativos aninhados exigem que você especifique CAPABILITY_AUTO_EXPAND.

Se seu modelo de aplicativo contiver algum dos recursos acima, recomendamos que você revise todas as permissões associadas ao aplicativo antes da implantação. Se você não especificar esse parâmetro para um aplicativo que requer recursos, a chamada falhará.

Tipo: Matriz do tipo [Recurso](#)

Obrigatório: verdadeiro

resourcesSupported

Se todos os AWS os recursos contidos nesse aplicativo são suportados na região em que ele está sendo recuperado.

Tipo: booliano

Obrigatório: verdadeiro

Consulte também

Para obter mais informações sobre como usar isso API em um dos idiomas específicos AWS SDKse referências, veja o seguinte:

CreateApplicationVersion

- [AWS Interface da linha de comando](#)
- [AWSSDKpara .NET](#)
- [AWSSDKpara C++](#)
- [AWSSDKpara Go v2](#)
- [AWSSDKpara Java V2](#)
- [AWSSDKpara JavaScript V3](#)
- [AWSSDKpara PHP V3](#)
- [AWSSDKpara Python](#)
- [AWSSDKpara Ruby V3](#)

Histórico do documento

- Versão da API: mais recente
- Última atualização da documentação: 10 de março de 2020

A tabela a seguir descreve as alterações importantes em cada versão do Guia do AWS Serverless Application Repository desenvolvedor do. Para receber notificações sobre atualizações dessa documentação, você pode se inscrever em um feed RSS.

Alteração	Descrição	Data
Atualizações para compartilhar e restringir o acesso a aplicativos	Foi adicionado suporte para compartilhar aplicativos em contas em uma AWS organização e restringir o acesso a aplicativos públicos para AWS contas e AWS Organizations. Para obter mais exemplos de compartilhamento de aplicativos com usuários em uma organização, consulte Exemplos de políticas de AWS Serverless Application Repository aplicativos . Para obter exemplos de restrição de acesso a aplicativos públicos, consulte Exemplos de políticas baseadas em identidade do AWS Serverless Application Repository .	10 de março de 2020
Novos recursos suportados	Suporte adicionado para diversos recursos agregados. Para obter uma lista completa de recursos compatíveis	17 de janeiro de 2020

	is, consulte Lista deAWS recursos compatíveis do .	
regiões da China	O agoraAWS Serverless Application Repository está disponível nas regiões da China, Pequim e do. Para obter mais informações sobre regiões e endpoints do AWS Serverless Application Repository, consulte Regiões e endpoints no Referência geral da AWS.	15 de janeiro de 2020
Seção de segurança atualizada para consistência com outrosAWS serviços.	Para obter mais informações, consulte Segurança .	2 de janeiro de 2020
Processo simplificado para publicação de aplicativos	O novo comando <code>aws sam publish</code> na CLI do AWS SAM simplifica o processo para publicar aplicativos sem servidor no AWS Serverless Application Repository. Para ver um end-to-end tutorial sobre como baixar e publicar um aplicativo de amostra, consulte Início rápido: publicação de aplicativos . Para obter instruções sobre como publicar um aplicativo que você já desenvolveu e testou naAWS nuvem, consulte Publicando um aplicativo por meio daAWS SAM CLI .	21 de dezembro de 2018

[Suporte para aplicativos aninhados e camadas](#)

Suporte adicional a aplicativos aninhados e layers Isso inclui atualizações nos [AWSrecursos suportados](#) e o [reconhecimento dos recursos do aplicativo](#).

29 de novembro de 2018

[Publicando aplicativos com funções e políticas de recursos personalizadas do IAM](#)

Suporte adicional para publicação de aplicativos com funções do IAM e política de recursos personalizadas. Isso inclui atualizações nos fluxos de trabalho de [consumo de aplicativos e publicação de aplicativos](#) e atualizações [AWSdos recursos suportados e da referência de API](#) no Guia doAWS Serverless Application Repository desenvolvedor.

16 de novembro de 2018

[Atualizações do modelo de política](#)

Atualizações dos [modelos de políticas](#) compatíveis no Guia doAWS Serverless Application Repository desenvolvedor.

26 de setembro de 2018

[Atualizações da documentação](#)

Foi adicionado o tópico Autenticação e Controle de Acesso ao Guia doAWS Serverless Application Repository Desenvolvedor.

2 de julho de 2018

Lançamento público

Lançamento público do AWS Serverless Application Repository, que agora está disponível em 14 AWS regiões. Para obter mais informações sobre as AWS regiões em que o AWS Serverless Application Repository está disponível e AWS Serverless Application Repository endpoints, consulte [Regiões e endpoints](#) no Referência geral da AWS.

20 de fevereiro de 2018

Novo guia

Esta é a primeira versão prévia do Guia do AWS Serverless Application Repository Desenvolvedor.

30 de novembro de 2017

Glossário da AWS

Para obter a terminologia mais recente da AWS, consulte o [glossário da AWS](#) na Referência do Glossário da AWS.

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.