



Entwicklerhandbuch

# AWS Serverless Application Repository



# AWS Serverless Application Repository: Entwicklerhandbuch

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

---

# Table of Contents

Was ist der AWS Serverless Application Repository? .....	1
Nächste Schritte .....	1
Quick Start: Veröffentlichen von Anwendungen .....	3
Übersicht .....	3
Hello World-Anwendung .....	3
Bevor Sie beginnen .....	4
Schritt 1: Initialisieren der Anwendung .....	4
Schritt 2: Lokales Testen der Anwendung .....	5
Schritt 3: Verpacken der Anwendung .....	6
Schritt 4: Veröffentlichen der Anwendung .....	8
Nächste Schritte .....	8
Weitere Informationen .....	9
Veröffentlichen von Anwendungen .....	10
Verwenden AWS SAM mit dem AWS Serverless Application Repository .....	11
Unterstützte AWS Ressourcen in der AWS Serverless Application Repository .....	11
Richtlinienvorlagen .....	12
Liste der unterstützten AWS Ressourcen .....	12
So veröffentlichen Sie Anwendungen .....	19
Veröffentlichen einer Anwendung (AWS CLI) .....	20
Veröffentlichen einer neuen Anwendung (Konsole) .....	20
Freigeben einer Anwendung .....	27
Aufheben der Freigabe einer Anwendung .....	29
Löschen einer Anwendung .....	31
Veröffentlichen von neuen Anwendungsversionen .....	32
Verifiziertes Autoren-Badge .....	33
Anfordern eines verifizierten Autoren-Badge .....	33
Lambda-Layer teilen .....	34
So funktioniert's .....	34
Beispiel .....	35
Bereitstellen von Anwendungen .....	36
Anwendungsbereitstellungsberechtigungen .....	36
Anwendungsfunktionen .....	37
Suchen und Bestätigen von Anwendungsfunktionen (Konsole) .....	38
Anzeigen von Anwendungsfunktionen (AWS CLI) .....	38

So stellen Sie Anwendungen bereit .....	39
Bereitstellen einer neuen Anwendungsversion (Konsole) .....	39
Bereitstellen einer neuen Anwendungsversion (AWS CLI) .....	41
Löschen von Anwendungs-Stacks .....	42
Aktualisieren von Anwendungen .....	42
Sicherheit .....	44
Datenschutz .....	45
Verschlüsselung während der Übertragung .....	46
Verschlüsselung im Ruhezustand .....	46
Identitäts- und Zugriffsverwaltung .....	46
Zielgruppe .....	47
Authentifizieren mit Identitäten .....	48
Verwalten des Zugriffs mit Richtlinien .....	51
So AWS Serverless Application Repository funktioniert das mit IAM .....	54
Beispiele für identitätsbasierte Richtlinien .....	60
Beispiele für Anwendungsrichtlinien .....	70
AWS Serverless Application Repository Referenz zu API-Berechtigungen .....	76
Fehlerbehebung .....	79
Protokollieren und Überwachen .....	82
AWS Serverless Application Repository API-Aufrufe protokollieren mit AWS CloudTrail .....	82
Compliance-Validierung .....	86
Ausfallsicherheit .....	87
Sicherheit der Infrastruktur .....	87
AWS PrivateLink .....	88
Überlegungen .....	88
Erstellen eines Schnittstellenendpunkts .....	88
Erstellen einer Endpunktrichtlinie .....	89
Kontingente .....	91
Fehlerbehebung .....	92
Sie können eine Anwendung nicht veröffentlichen .....	92
Ein Kontingent wurde überschritten .....	93
Eine aktualisierte Readme-Datei erscheint nicht sofort .....	93
Sie können eine Anwendung infolge von unzureichenden IAM-Berechtigungen nicht bereitstellen .....	93
Sie können eine Anwendung nicht zweimal bereitstellen .....	93
Warum ist meine Anwendung nicht öffentlich verfügbar .....	94

Den -Support kontaktieren .....	94
Operationen .....	95
Ressourcen .....	97
Applications .....	97
URI .....	97
HTTP-Methoden .....	97
Schemata .....	99
Eigenschaften .....	103
Weitere Informationen finden Sie auch unter .....	121
applicationId .....	122
URI .....	122
HTTP-Methoden .....	122
Schemata .....	126
Eigenschaften .....	129
Weitere Informationen finden Sie auch unter .....	143
Anwendungs-ID-Änderungssätze .....	144
URI .....	144
HTTP-Methoden .....	144
Schemata .....	146
Eigenschaften .....	148
Weitere Informationen finden Sie auch unter .....	156
Applications applicationId Dependencies .....	157
URI .....	157
HTTP-Methoden .....	157
Schemata .....	159
Eigenschaften .....	160
Weitere Informationen finden Sie auch unter .....	163
ApplicationID-Richtlinie für Anwendungen .....	164
URI .....	164
HTTP-Methoden .....	164
Schemata .....	167
Eigenschaften .....	169
Weitere Informationen finden Sie auch unter .....	172
Applications applicationId Templates .....	173
URI .....	173
HTTP-Methoden .....	173

Schemata .....	175
Eigenschaften .....	177
Weitere Informationen finden Sie auch unter .....	181
Applications applicationId Templates templateId .....	181
URI .....	181
HTTP-Methoden .....	181
Schemata .....	184
Eigenschaften .....	185
Weitere Informationen finden Sie auch unter .....	189
Applications applicationId Unshare .....	189
URI .....	189
HTTP-Methoden .....	189
Schemata .....	191
Eigenschaften .....	192
Weitere Informationen finden Sie auch unter .....	195
Anwendungs-ID-Versionen .....	195
URI .....	195
HTTP-Methoden .....	196
Schemata .....	197
Eigenschaften .....	199
Weitere Informationen finden Sie auch unter .....	202
Anwendungen ApplicationID-Versionen SemanticVersion .....	203
URI .....	203
HTTP-Methoden .....	203
Schemata .....	205
Eigenschaften .....	207
Weitere Informationen finden Sie auch unter .....	216
Dokumentverlauf .....	218
AWS Glossar .....	223
.....	ccxxiv

# Was ist der AWS Serverless Application Repository?

Das AWS Serverless Application Repository macht es Entwicklern und Unternehmen leicht, serverlose Anwendungen schnell in der Cloud zu finden, bereitzustellen und zu veröffentlichen. AWS Weitere Informationen zu serverlosen Anwendungen finden Sie auf der Website unter [Serverless Computing and Applications](#). AWS

Sie können Anwendungen einfach veröffentlichen, indem Sie sie für die ganze Community oder privat für Ihr Team oder in Ihrer Organisation freigeben. Um eine serverlose Anwendung (oder App) zu veröffentlichen, können Sie die AWS Management Console AWS SAM Befehlszeilenschnittstelle (AWS SAM CLI) verwenden oder AWS SDKs Ihren Code hochladen. Zusammen mit Ihrem Code laden Sie eine einfache Manifestdatei hoch, die auch als Vorlage AWS Serverless Application Model (AWS SAM) bezeichnet wird. Weitere Informationen AWS SAM dazu finden Sie im [AWS Serverless Application Model Entwicklerhandbuch](#).

Der AWS Serverless Application Repository ist tief in die AWS Lambda Konsole integriert. Diese Integration bedeutet, dass Entwickler aller Stufen mit serverlosem Computing beginnen können, ohne etwas Neues erlernen zu müssen. Sie können mithilfe von Kategorie-Schlüsselwörtern nach Anwendungen wie mobilen und Web-Backends, Anwendungen zur Datenverarbeitung oder Chatbots suchen. Außerdem können Sie über Name, Herausgeber oder Ereignisquelle nach Anwendungen suchen. Wenn Sie eine Anwendung verwenden möchten, wählen Sie sie einfach aus, konfigurieren Sie die erforderlichen Felder und stellen Sie sie mit ein paar Klicks bereit.

In diesem Leitfaden erfahren Sie mehr zu den beiden Möglichkeiten zur Arbeit mit dem AWS Serverless Application Repository:

- [Veröffentlichen von Anwendungen](#)— Konfigurieren und laden Sie Anwendungen hoch, um sie anderen Entwicklern zur Verfügung zu stellen, und veröffentlichen Sie neue Versionen von Anwendungen.
- [Bereitstellen von Anwendungen](#)— Suchen Sie nach Anwendungen und sehen Sie sich Informationen zu ihnen an, einschließlich Quellcode und Readme-Dateien. Außerdem können Sie Anwendungen Ihrer Wahl installieren, konfigurieren und bereitstellen.

## Nächste Schritte

- Ein Tutorial zum Veröffentlichen einer Beispielanwendung in der finden Sie AWS Serverless Application Repository unter [Quick Start: Veröffentlichen von Anwendungen](#).

- Anweisungen zum Bereitstellen von Anwendungen aus dem AWS Serverless Application Repository finden Sie unter [So stellen Sie Anwendungen bereit](#).

# Quick Start: Veröffentlichen von Anwendungen

Dieses Handbuch führt Sie durch die Schritte zum Herunterladen, Erstellen, Testen und Veröffentlichen einer serverlosen Beispielanwendung auf der AWS Serverless Application Repository AWS SAM CLI. Sie können diese Beispielanwendung als Ausgangspunkt für die Entwicklung und Veröffentlichung Ihrer eigenen serverlosen Anwendung verwenden.

## Übersicht

In den folgenden Schritten wird beschrieben, wie Sie eine serverlose Beispielanwendung herunterladen, erstellen und veröffentlichen:

1. Initialisieren. Laden Sie eine Beispielanwendung aus der Vorlage mithilfe von `sam init` herunter.
2. Ausführen lokaler Tests. Testen Sie die Anwendung lokal mit `sam local invoke` und/oder `sam local start-api`. Beachten Sie, dass Ihre Lambda-Funktion mit diesen Befehlen zwar lokal aufgerufen wird, aber von AWS Ressourcen in der AWS Cloud liest und in sie schreibt.
3. Paket. Wenn Sie mit Ihrer Lambda-Funktion zufrieden sind, bündeln Sie die Lambda-Funktion, die AWS SAM Vorlage und alle Abhängigkeiten in einem AWS CloudFormation Bereitstellungspaket mit `sam package`. In diesem Schritt fügen Sie auch Informationen über die Anwendung hinzu, die in das AWS Serverless Application Repository hochgeladen wird.
4. Veröffentlichen. Veröffentlichen Sie die Anwendung im AWS Serverless Application Repository mithilfe von `sam publish`. Am Ende dieses Schritts können Sie Ihre Anwendung in der AWS Cloud anzeigen AWS Serverless Application Repository und mithilfe von `AWS Serverless Application Repository`

Das Beispiel [Hello World-Anwendung](#) im nächsten Abschnitt führt Sie durch diese Schritte beim Erstellen und Veröffentlichen einer serverlosen Anwendung.

## Hello World-Anwendung

In dieser Übung laden Sie eine serverlose Hello World-Anwendung, die ein einfaches API-Backend darstellt, herunter und testen Sie diese. Es hat einen Amazon API Gateway Gateway-Endpunkt, der eine GET-Operation und eine Lambda-Funktion unterstützt. Wenn eine GET-Anfrage an den Endpunkt gesendet wird, ruft API Gateway die Lambda-Funktion auf. AWS Lambda führt dann die Funktion aus, die einfach eine Nachricht zurückgibt. `hello world`

Die Anwendung besteht aus folgenden Komponenten:

- Eine AWS SAM Vorlage, die zwei AWS Ressourcen für die Hello World-Anwendung definiert: einen API-Gateway-Dienst mit einer GET-Operation und eine Lambda-Funktion. Die Vorlage definiert auch die Zuordnung zwischen dem API-Gateway-GET-Vorgang und der Lambda-Funktion.
- Anwendungscode, der in Python geschrieben ist.

## Bevor Sie beginnen

Stellen Sie sicher, dass Sie über die erforderliche Einrichtung für diese Übung verfügen:

- Sie müssen über ein AWS Konto bei einem IAM-Benutzer mit Administratorrechten verfügen. Weitere Informationen finden [Sie unter AWS Konto einrichten](#).
- Sie müssen die AWS SAM CLI (Befehlszeilenschnittstelle) installiert haben. Siehe [AWS SAM CLI installieren](#).
- Sie müssen Version 1.16.77 oder höher von installiert haben. AWS CLI Siehe [Installieren der AWS Command Line Interface](#).

## Schritt 1: Initialisieren der Anwendung

In diesem Abschnitt laden Sie die Beispielanwendung herunter, die aus einer AWS SAM -Vorlage und einem Anwendungscode besteht.

So initialisieren Sie die Anwendung

1. Führen Sie den folgenden Befehl an einer AWS SAM CLI-Eingabeaufforderung aus.

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

2. Überprüfen Sie den Inhalt des Verzeichnisses, das der Befehl erstellt hat (sam-app/):
  - `template.yaml`— Definiert zwei AWS Ressourcen, die die Hello World-Anwendung benötigt: eine Lambda-Funktion und einen API-Gateway-Endpunkt, der eine GET-Operation unterstützt. Die Vorlage definiert auch die Zuweisung zwischen den beiden Ressourcen.
  - Inhalt im Zusammenhang mit dem Hello World-Anwendungscode:
    - `hello_world/Verzeichnis` — Enthält den Anwendungscode, der zurückgegeben wird, `hello world` wenn Sie ihn ausführen.

**Note**

Für diese Übung ist der Anwendungscode in Python geschrieben, und Sie geben die Laufzeit im `init` Befehl an. AWS Lambda unterstützt zusätzliche Sprachen für die Erstellung von Anwendungscode. Wenn Sie eine andere unterstützte Laufzeit angeben, enthält der `init`-Befehl den Hello World-Code in der angegebenen Sprache und eine `README.md`-Datei, die Sie für diese Sprache befolgen können. Informationen zu unterstützten Laufzeiten finden Sie unter [Lambda-Ausführungsumgebung und verfügbare Bibliotheken](#).

## Schritt 2: Lokales Testen der Anwendung

Nachdem Sie die AWS SAM Anwendung nun auf Ihrem lokalen Computer installiert haben, gehen Sie wie folgt vor, um sie lokal zu testen.

So testen Sie die Anwendung lokal

1. Starten Sie den API-Gateway-Endpunkt lokal. Sie müssen den folgenden Befehl aus dem Verzeichnis ausführen, das die `template.yaml`-Datei enthält.

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

Der Befehl gibt einen API-Gateway-Endpunkt zurück, an den Sie Anfragen für lokale Tests senden können.

2. Testen Sie die Anwendung. Kopieren Sie die API Gateway Gateway-Endpunkt-URL, fügen Sie sie in den Browser ein und wählen Sie Enter. Ein Beispiel für eine API Gateway Gateway-Endpunkt-URL ist `http://127.0.0.1:3000/hello`.

API Gateway ruft lokal die Lambda-Funktion auf, der der Endpunkt zugeordnet ist. Die Lambda-Funktion wird im lokalen Docker-Container ausgeführt und kehrt zurück. `hello world` API Gateway gibt eine Antwort an den Browser zurück, der den Text enthält.

Übung: Ändern der Nachrichtenzeichenfolge

Nachdem Sie die Beispielanwendung erfolgreich getestet haben, können Sie mit einer einfachen Änderung experimentieren: Ändern Sie die zurückgegebene Nachrichtenzeichenfolge.

1. Bearbeiten Sie die `/hello_world/app.py`-Datei, um die Nachrichtenzeichenfolge von `'hello world'` in `'Hello World!'` zu ändern.
2. Laden Sie die Test-URL in Ihrem Browser neu und beachten Sie die neue Zeichenfolge.

Sie werden feststellen, dass Ihr neuer Code dynamisch geladen wird, ohne dass Sie den Prozess `local` neu starten müssen.

## Schritt 3: Verpacken der Anwendung

Nachdem Sie Ihre Anwendung lokal getestet haben, verwenden Sie die AWS SAM CLI, um ein Bereitstellungspaket und eine verpackte AWS SAM Vorlage zu erstellen.

### Note

In den folgenden Schritten erstellen Sie eine ZIP-Datei für den Inhalt des `hello_world/`-Verzeichnisses, das den Anwendungscode enthält. Diese ZIP-Datei ist das Bereitstellungspaket für Ihre serverlose Anwendung. Weitere Informationen finden Sie unter [Creating a Deployment Package \(Python\)](#) im AWS Lambda Developer Guide.

So erstellen Sie ein Lambda-Bereitstellungspaket

1. Fügen Sie Ihrer AWS SAM Vorlagendatei einen Metadata Abschnitt mit den erforderlichen Anwendungsinformationen hinzu. Weitere Informationen zum Metadata Abschnitt mit AWS SAM Vorlagen finden Sie unter [Eigenschaften des Abschnitts „AWS SAM Vorlagen-Metadaten“](#) im AWS Serverless Application Model Entwicklerhandbuch.

Hier ist ein Metadata-Beispielabschnitt:

```
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
```

Die `ReadmeUrl` Eigenschaften `LicenseUrl` und können entweder Verweise auf lokale Dateien sein (wie im obigen Beispiel), oder sie können Links zu Amazon S3 S3-Buckets sein, die diese Artefakte bereits hosten.

- Erstellen Sie einen S3-Bucket an dem Speicherort, an dem Sie den verpackten Code speichern möchten. Wenn Sie einen vorhandenen S3-Bucket verwenden möchten, überspringen Sie diesen Schritt.

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

- Erstellen Sie das Lambda-Funktionsbereitstellungspaket, indem Sie den folgenden package AWS SAM CLI-Befehl ausführen.

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

Der Befehl hat folgende Auswirkungen:

- Komprimiert den Inhalt des `aws-sam/hello_world/` Verzeichnisses und lädt ihn auf Amazon S3 hoch.
- Lädt das Bereitstellungspaket, die README-Datei und die LICENSE-Datei in den durch die Option angegebenen Amazon S3 S3-Bucket hoch. `--s3-bucket`
- Gibt eine neue Vorlagendatei mit dem Namen `packaged.yaml` aus, die Sie im nächsten Schritt zum Veröffentlichen der Anwendung im AWS Serverless Application Repository verwenden. Die `packaged.yaml` Vorlagendatei ähnelt der ursprünglichen Vorlagendatei (`template.yaml`), weist jedoch einen wesentlichen Unterschied auf: Die `ReadmeUrl` Eigenschaften `CodeUri``LicenseUrl`, und verweisen auf den Amazon S3 S3-Bucket und die Objekte, die die jeweiligen Artefakte enthalten. Der folgende Ausschnitt aus einer `packaged.yaml`-Beispielvorlagendatei zeigt die `CodeUri`-Eigenschaft:

```
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/fbd77a3647a4f47a352fc0bjectGUID
```

```
...
```

## Schritt 4: Veröffentlichen der Anwendung

Nachdem Sie das Bereitstellungspaket erstellt haben, verwenden Sie es, um die Anwendung im AWS Serverless Application Repository zu veröffentlichen.

Um die serverlose Anwendung auf dem zu veröffentlichen AWS Serverless Application Repository

- Führen Sie den folgenden Befehl aus, um die neue Anwendung mit der ersten Version in AWS Serverless Application Repository zu veröffentlichen, die als 0.0.1 erstellt wurde.

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

### Note

Die Anwendung wird standardmäßig als privat erstellt. Sie müssen die Anwendung gemeinsam nutzen, bevor andere AWS Konten Ihre Anwendung anzeigen und bereitstellen können. Weitere Informationen zum Freigeben Ihrer Anwendung finden Sie nachstehend in den nächsten Schritten.

## Nächste Schritte

Nachdem Sie Ihre Beispielanwendung veröffentlicht haben, folgen einige Dinge, die Sie möglicherweise damit tun möchten.

- Ihre Anwendung anzeigen in AWS Serverless Application Repository — Die Ausgabe des `sam publish` Befehls enthält einen Link, der AWS Serverless Application Repository direkt zur

Detailseite Ihrer Anwendung führt. Sie können auch auf der AWS Serverless Application Repository Landingpage nach Ihrer Anwendung suchen.

- Teilen Sie Ihre Bewerbung — Da Ihre Bewerbung standardmäßig auf privat gesetzt ist, ist sie für andere AWS Konten nicht sichtbar. Um Ihre Anwendung mit anderen zu teilen, müssen Sie sie entweder öffentlich machen oder einer bestimmten Liste von AWS Konten die Erlaubnis erteilen. Informationen zur gemeinsamen Nutzung Ihrer Anwendung AWS CLI finden Sie unter [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#). Informationen zum Freigeben Ihrer Anwendung über die Konsole finden Sie unter [Freigeben einer Anwendung](#).

## Weitere Informationen

Weitere Informationen zum Metadata Abschnitt mit AWS SAM Vorlagen `sam package` und `sam publish` Befehlen von AWS SAM CLI finden Sie unter [Publishing Applications Using AWS SAM CLI](#) im AWS Serverless Application Model Developer Guide.

# Veröffentlichen von Anwendungen

Wenn Sie eine serverlose Anwendung auf dem veröffentlichen AWS Serverless Application Repository, stellen Sie sie anderen zur Verfügung, damit sie gefunden und bereitgestellt werden können.

Zuerst definieren Sie Ihre Anwendung mit einer AWS Serverless Application Model (AWS SAM)-Vorlage. Wenn Sie Ihre Anwendung definieren, müssen Sie überlegen, ob Benutzer Ihrer Anwendung die Funktionen der Anwendung bestätigen müssen. Weitere Informationen zur Verwendung AWS SAM und Bestätigung von Funktionen finden Sie unter [Verwenden AWS SAM mit dem AWS Serverless Application Repository](#)

Sie können serverlose Anwendungen mithilfe der AWS Management Console AWS SAM Befehlszeilenschnittstelle (AWS SAM CLI) oder eines AWS SDK veröffentlichen. Weitere Informationen zu den Verfahren zum Veröffentlichen von Anwendungen auf dem finden Sie AWS Serverless Application Repository unter [So veröffentlichen Sie Anwendungen](#).

Wenn Sie Ihre Anwendung veröffentlichen, ist sie zunächst auf privat gesetzt, was bedeutet, dass sie nur für das AWS Konto verfügbar ist, mit dem sie erstellt wurde. Um Ihre Anwendung mit anderen zu teilen, müssen Sie sie entweder auf privat geteilt (nur mit einer bestimmten Gruppe von AWS Konten geteilt) oder öffentlich geteilt (für alle geteilt) einstellen.

Wenn Sie eine Anwendung auf der veröffentlichen AWS Serverless Application Repository und sie auf öffentlich setzen, stellt der Dienst die Anwendung Verbrauchern in allen Regionen zur Verfügung. Wenn ein Verbraucher eine öffentliche Anwendung in einer anderen Region als der Region bereitstellt, in der die Anwendung zuerst veröffentlicht wurde, AWS Serverless Application Repository kopiert er die Bereitstellungsartefakte der Anwendung in einen Amazon S3 S3-Bucket in der Zielregion. Es aktualisiert alle Ressourcen in der AWS SAM Vorlage, die diese Artefakte verwenden, um stattdessen auf die Dateien im Amazon S3 S3-Bucket für die Zielregion zu verweisen. Bereitstellungsartefakte können Lambda-Funktionscode, API-Definitionsdateien usw. enthalten.

## Note

Private und privat gemeinsam genutzte Anwendungen sind nur in der AWS Region verfügbar, in der sie erstellt wurden. Öffentlich geteilte Anwendungen sind in allen AWS Regionen verfügbar. Weitere Informationen zum Freigeben von Anwendungen finden Sie unter [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#).

## Themen

- [Verwenden AWS SAM mit dem AWS Serverless Application Repository](#)
- [So veröffentlichen Sie Anwendungen](#)
- [Verifiziertes Autoren-Badge](#)
- [Lambda-Layer teilen](#)

# Verwenden AWS SAM mit dem AWS Serverless Application Repository

The AWS Serverless Application Model (AWS SAM) ist ein Open-Source-Framework, auf dem Sie [serverlose Anwendungen](#) erstellen können. [AWS Weitere Informationen AWS SAM zur Erstellung Ihrer serverlosen Anwendung finden Sie im Entwicklerhandbuch.AWS Serverless Application Model](#)

Bei der Entwicklung von Anwendungen AWS Serverless Application Repository, die auf der veröffentlicht werden, müssen Sie die verfügbaren verfügbaren AWS Ressourcen und Richtlinienvorlagen berücksichtigen. In den folgenden Abschnitten werden diese Themen ausführlicher beschrieben.

## Unterstützte AWS Ressourcen in der AWS Serverless Application Repository

Das AWS Serverless Application Repository unterstützt serverlose Anwendungen, die aus vielen AWS SAM AWS CloudFormation Ressourcen bestehen. Eine vollständige Liste der AWS Ressourcen, die von unterstützt werden AWS Serverless Application Repository, finden Sie unter [Liste der unterstützten AWS Ressourcen](#).

Wenn Sie Support für eine zusätzliche AWS Ressource anfordern möchten, wenden Sie sich an den [AWS Support](#).

### Important

Wenn Ihre Anwendungsvorlage eine der folgenden benutzerdefinierten IAM-Rollen oder Ressourcenrichtlinien enthält, wird die Anwendung standardmäßig nicht in den Suchergebnissen aufgeführt. Darüber hinaus müssen Kunden die benutzerdefinierten IAM-Rollen und Ressourcenrichtlinien der Anwendung bestätigen, ehe sie die

Anwendung bereitstellen können. Weitere Informationen finden Sie unter [Bestätigen von Anwendungsfunktionen](#).

Die Liste der Ressourcen, für die dieses gilt, sieht wie folgt aus:

- IAM-Rollen: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), und [AWS::IAM::Role](#).
- Ressourcenrichtlinien: [AWS::Lambda::LayerVersion](#) [Permission](#), [AWS::Events::EventBusPolicy](#), [AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#), [AWS::S3::BucketPolicy](#), [AWS::SQS::QueuePolicy](#), und [AWS::SNS::TopicPolicy](#)

Wenn Ihre Anwendung die [AWS::Serverless::Application](#) Ressource enthält, müssen Kunden bestätigen, dass die Anwendung eine verschachtelte Anwendung enthält, bevor sie die Anwendung bereitstellen können. Weitere Informationen über verschachtelte Anwendungen finden Sie im Thema [Verschachtelte Anwendungen](#) im AWS Serverless Application Model -Entwicklerhandbuch. Weitere Informationen zur Funktionsbestätigung finden Sie unter [Bestätigen von Anwendungsfunktionen](#).

## Richtlinienvorlagen

AWS SAM stellt Ihnen eine Liste von Richtlinienvorlagen zur Verfügung, mit denen Sie die Berechtigungen Ihrer Lambda-Funktionen auf die Ressourcen beschränken können, die von Ihrer Anwendung verwendet werden. Für die Verwendung von Richtlinienvorlagen sind keine zusätzlichen Kundenbestätigungen erforderlich, um die Anwendung zu durchsuchen oder bereitzustellen.

Eine Liste der AWS SAM Standardrichtlinienvorlagen finden Sie unter [AWS SAM Richtlinienvorlagen](#) im [AWS Serverless Application Model Entwicklerhandbuch](#).

## Liste der unterstützten AWS Ressourcen

Dies ist die vollständige Liste der AWS Ressourcen, die von der unterstützt werden 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::VPCElasticGatewayAttachment`
- `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`

## So veröffentlichen Sie Anwendungen

In diesem Abschnitt finden Sie Verfahren zum Veröffentlichen Ihrer serverlosen Anwendung auf der AWS Serverless Application Repository mithilfe der AWS SAM CLI oder der AWS Management

Console. Außerdem wird gezeigt, wie Sie Ihre Anwendung freigeben, damit andere Benutzer sie bereitstellen können, und aus dem AWS Serverless Application Repository löschen.

### Important

Die Informationen, die Sie beim Veröffentlichen einer Anwendung eingeben, sind nicht verschlüsselt. Diese Informationen umfassen Daten wie den Namen des Autors. Wenn Sie nicht möchten, dass personenbezogene Daten gespeichert oder veröffentlicht werden, empfehlen wir, solche Informationen bei der Veröffentlichung Ihrer Anwendung nicht einzugeben.

## Veröffentlichen einer Anwendung (AWS CLI)

Der einfachste Weg, eine Anwendung in der zu veröffentlichen, AWS Serverless Application Repository besteht darin, eine Reihe von AWS SAM CLI-Befehlen zu verwenden. Weitere Informationen finden Sie unter [Veröffentlichen einer Anwendung mit der AWS SAM CLI](#) im AWS Serverless Application Model (AWS SAM) Developer Guide.

## Veröffentlichen einer neuen Anwendung (Konsole)

In diesem Abschnitt erfahren Sie, wie Sie AWS Management Console mit dem eine neue Anwendung auf dem veröffentlichen AWS Serverless Application Repository. Anweisungen zum Veröffentlichen einer neuen Version einer vorhandenen Anwendung finden Sie unter [Veröffentlichen der neuen Version einer vorhandenen Anwendung](#).

## Voraussetzungen

Bevor Sie eine Anwendung auf dem veröffentlichen AWS Serverless Application Repository, benötigen Sie Folgendes:

- Ein gültiges AWS Konto.
- Eine gültige Vorlage AWS Serverless Application Model (AWS SAM), die die verwendeten AWS Ressourcen definiert. Weitere Informationen zu AWS SAM Vorlagen finden Sie unter [Grundlagen von AWS SAM Vorlagen](#).
- Ein Paket für Ihre Anwendung, das Sie mit dem AWS CloudFormation package Befehl für erstellt haben AWS CLI. Dieser Befehl packt die lokalen Artefakte (lokale Pfade), auf die Ihre AWS SAM

Vorlage verweist. Weitere Informationen finden Sie in der AWS CloudFormation Dokumentation unter [Paket](#).

- Eine URL, die auf den Quellcode Ihrer Anwendung verweist, falls Sie Ihre Anwendung öffentlich veröffentlichen möchten.
- Eine readme.txt-Datei. In dieser Datei sollte beschrieben werden, wie Kunden Ihre Anwendung verwenden können und wie sie konfiguriert wird, bevor sie sie in ihren eigenen AWS Konten bereitstellen.
- Eine license.txt-Datei oder eine gültige Lizenzkennung von der [SPDX-Website](#). Beachten Sie, dass eine Lizenz nur erforderlich ist, wenn Sie Ihre Anwendung öffentlich freigeben möchten. Wenn Sie Ihre Anwendung privat halten oder nur privat freigeben möchten, müssen Sie keine Lizenz angeben.
- Eine gültige Amazon S3-Bucket-Richtlinie, die dem Service Leseberechtigungen für Artefakte gewährt, die beim Paketieren Ihrer Anwendung auf Amazon S3 hochgeladen wurden. Gehen Sie folgendermaßen vor, um diese Richtlinie festzulegen:
  1. Öffnen Sie die Amazon S3 S3-Konsole unter <https://console.aws.amazon.com/s3/>.
  2. Wählen Sie den Amazon S3 S3-Bucket aus, den Sie zum Verpacken Ihrer Anwendung verwendet haben.
  3. Wählen Sie die Registerkarte Berechtigungen.
  4. Wählen Sie Bucket Policy (Bucket-Richtlinie) aus.
  5. Fügen Sie die folgende Richtlinienanweisung in den Bucket-Richtlinien-Editor ein. Achten Sie darauf, Ihren Bucket-Namen im Resource Element und Ihre AWS Konto-ID im Condition Element zu ersetzen. Der Ausdruck im Condition Element stellt sicher, dass Sie AWS Serverless Application Repository nur über das angegebene AWS Konto Zugriff auf Anwendungen haben. Weitere Informationen zu Richtlinienenerklärungen finden Sie in der [Referenz zu den IAM-JSON-Richtlinienelementen](#) im IAM-Benutzerhandbuch.

```
{
  "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"
            }
        }
    ]
}

```

- Wählen Sie die Schaltfläche Save (Speichern) aus.

## Verfahren

Erstellen Sie mithilfe des AWS Serverless Application Repository folgenden Verfahrens eine neue Anwendung in der.

Um eine neue Anwendung in der AWS Serverless Application Repository

- Öffnen Sie die [AWS Serverless Application Repository -Konsole](#) und wählen Sie Publish applications (Anwendungen veröffentlichen) aus.
- Geben Sie auf der Seite Publish an application (Anwendung veröffentlichen) die folgenden Anwendungsinformationen ein und wählen Sie dann Publish application (Anwendung veröffentlichen) aus:

Eigenschaft	Erforderlich	Beschreibung
Anwendungsname	TRUE	Der Name der Anwendung.  Minimale Länge = 1. Maximale Länge = 140.  Pattern: "[a-zA-Z0-9\-\-]+";
Autor	TRUE	Der Name des Autors, der die Anwendung veröffentlicht.  Minimale Länge = 1. Maximale Länge = 127.

Eigenschaft	Erforderlich	Beschreibung
		Pattern: <code>"^[a-z0-9]([a-z0-9](-?!-))*[a-z0-9]?\$"</code> ;
Homepage	FALSE	Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.
Beschreibung	TRUE	Die Beschreibung der Anwendung.  Minimale Länge = 1. Maximale Länge = 256.
Labels	FALSE	Die Beschriftungen, die die Erkennung von Anwendungen in Suchergebnissen verbessern.  Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl der Etiketten: 10.  Pattern: <code>"^[a-zA-Z0-9+\\-._:\\/ @]+\$"</code> ;

Eigenschaft	Erforderlich	Beschreibung
Spdx-Lizenz (Dropdown-Liste)	FALSE	<p>Wählen Sie eine gültige Lizenzkennung aus der Dropdown-Liste aus, die Lizenzen enthält, die auf der <a href="#">SPDX-Website</a> verfügbar sind. Wenn Sie ein Element in der Dropdown-Liste auswählen, wird das Textfeld Lizenz darunter aufgefüllt. Hinweis: Wenn Sie eine Lizenz in der Dropdown-Liste auswählen, wird der Inhalt des Textfelds Lizenz ersetzt und alle manuellen Änderungen verworfen, die Sie vorgenommen haben.</p>

Eigenschaft	Erforderlich	Beschreibung
License	FALSE	<p>Laden Sie eine TXT-Lizenzdatei hoch oder wählen Sie eine Lizenz aus dem Dropdown-Menü SPDX license (SPDX-Lizenz), das in der vorherigen Zeile beschrieben ist. Wenn Sie eine Lizenz aus dem Dropdown-Menü SPDX license (SPDX-Lizenz) auswählen, wird automatisch das Textfeld Lizenz ausgefüllt. Sie können den Inhalt dieses Textfelds manuell bearbeiten, nachdem Sie eine Lizenzdatei hochgeladen haben oder eine aus dem Dropdown-Menü Spdx-Lizenz auswählen. Wenn jedoch eine andere Spdx-Lizenz aus dem Dropdown-Menü ausgewählt wird, werden alle manuellen Änderungen, die Sie vorgenommen haben, verworfen.</p> <p>Dies ist ein optionales Feld, aber Sie müssen eine Lizenz bereitstellen, um die Anwendung öffentlich freigeben zu können.</p>

Eigenschaft	Erforderlich	Beschreibung
Readme	FALSE	Laden Sie den Inhalt der Readme-Datei hoch, die im Text- oder Markdown-Format vorliegen kann. Diese Inhalte werden auf der Detailseite der Anwendung im AWS Serverless Application Repository angezeigt. Sie können den Inhalt dieses Textfelds nach dem Hochladen einer Datei manuell bearbeiten.
Semantic version (Semantische Version)	FALSE	Die semantische Version der Anwendung. Weitere Informationen finden Sie auf der <a href="#">Semantic Versioning-Website</a> .  Sie müssen einen Wert für diese Eigenschaft angeben, um Ihre Anwendung öffentlich zu machen.
Quellcode-URL	FALSE	Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung.
SAM-Vorlage	TRUE	Eine gültige Vorlage AWS Serverless Application Model (AWS SAM), die die verwendeten AWS Ressourcen definiert.

## Freigeben einer Anwendung

Für veröffentlichte Anwendungen können Berechtigungen in einer der drei folgenden Kategorien festgelegt werden:

- **Privat (Standard)** — Anwendungen, die mit demselben Konto erstellt und mit keinem anderen AWS Konto geteilt wurden. Nur Verbraucher, die Ihr AWS Konto teilen, sind berechtigt, private Anwendungen bereitzustellen.
- **Privat geteilt** — Anwendungen, die der Herausgeber ausdrücklich mit einer bestimmten Gruppe von AWS Konten oder mit AWS Konten in einer AWS Organisation geteilt hat. Verbraucher sind berechtigt, Anwendungen bereitzustellen, die mit ihrem AWS Konto oder ihrer AWS Organisation geteilt wurden. Weitere Informationen zu AWS Organizations finden Sie im [AWS Organizations Benutzerhandbuch](#).
- **Öffentlich geteilt** — Anwendungen, die der Herausgeber für alle freigegeben hat. Alle Verbraucher haben die Berechtigung, alle öffentlich freigegebenen Anwendungen bereitzustellen.

Nachdem Sie eine Anwendung auf dem veröffentlicht haben AWS Serverless Application Repository, ist sie standardmäßig auf privat gesetzt. In diesem Abschnitt erfahren Sie, wie Sie eine Anwendung privat mit bestimmten AWS Konten oder einer AWS Organisation teilen oder sie öffentlich für alle freigeben können.

### Freigeben einer Anwendung über die Konsole

Sie haben zwei Möglichkeiten, Ihre Anwendung mit anderen zu teilen: 1) Teilen Sie sie mit bestimmten AWS Konten oder AWS Konten innerhalb Ihrer AWS Organisation oder 2) Teilen Sie sie öffentlich für alle. Weitere Informationen dazu AWS Organizations finden Sie im [AWS Organizations Benutzerhandbuch](#).

Option 1: Um Ihre Anwendung mit bestimmten AWS Konten oder Konten innerhalb Ihrer AWS Organisation zu teilen

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).
2. Wählen Sie im Navigationsbereich Published Applications (Veröffentlichte Anwendungen) aus, um eine Liste der von Ihnen erstellten Anwendungen anzuzeigen.
3. Wählen Sie die Anwendung aus, die Sie freigeben möchten.
4. Wählen Sie die Registerkarte Sharing (Freigabe).

5. Wählen Sie im Abschnitt **Application policy statements** (Anwendungsrichtlinienanweisungen) die Schaltfläche **Create statement** (Anweisung erstellen).
6. Füllen Sie im Fenster **Statement Configuration** (Anweisungskonfiguration) die Felder aus, die darauf basieren, wie Sie Ihre Anwendung freigeben möchten.

 Note

Wenn Sie Inhalte mit einer Organisation teilen, können Sie nur die Organisation angeben, der Ihr AWS Konto angehört. Wenn Sie versuchen, eine AWS Organisation anzugeben, der Sie nicht angehören, wird ein Fehler angezeigt.

Um Ihre Bewerbung mit Ihrer AWS Organisation zu teilen, müssen Sie bestätigen, dass die `UnshareApplication` Maßnahme zu Ihrer Grundsatzerklärung hinzugefügt wird, falls die Weitergabe in future widerrufen werden muss.

7. Wählen Sie die Schaltfläche **Save** (Speichern) aus.

Option 2: So geben Sie Ihre Anwendung öffentlich für alle frei

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).
2. Wählen Sie im Navigationsbereich **Published Applications** (Veröffentlichte Anwendungen) aus, um eine Liste der von Ihnen erstellten Anwendungen anzuzeigen.
3. Wählen Sie die Anwendung aus, die Sie freigeben möchten.
4. Wählen Sie die Registerkarte **Sharing** (Freigabe).
5. Wählen Sie im Abschnitt **Public Sharing** (Öffentliche Freigabe) die Schaltfläche **Edit** (Bearbeiten).
6. Wählen Sie unter **Public sharing** (Öffentliche Freigabe) das Optionsfeld **Enabled** (Aktiviert) aus.
7. Geben Sie im Textfeld den Namen Ihrer Anwendung ein und wählen Sie dann die Schaltfläche **Save** (Speichern) aus.

 Note

Um eine Anwendung öffentlich freigeben zu können, müssen die Eigenschaften `SemanticVersion` und `LicenseUrl` festgelegt sein.

## Freigeben einer Anwendung über die AWS CLI

Um eine Anwendung mithilfe von zu teilen, gewähren AWS CLI Sie Berechtigungen, indem Sie den [put-application-policy](#) Befehl verwenden, um die AWS Konten anzugeben, für die Sie die gemeinsame Nutzung als Hauptbenutzer verwenden möchten.

Weitere Informationen zum Teilen Ihrer Anwendung mithilfe der AWS CLI finden Sie unter [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#).

## Aufheben der Freigabe einer Anwendung

Es gibt zwei Möglichkeiten, die gemeinsame Nutzung einer Anwendung für eine AWS Organisation aufzuheben:

1. Der Herausgeber der Anwendung kann Berechtigungen mit dem [put-application-policy](#)-Befehl entfernen.
2. Ein Benutzer aus dem Verwaltungskonto einer AWS Organisation kann für jede [Anwendung, die für die Organisation freigegeben ist, den Vorgang rückgängig](#) machen, selbst wenn die Anwendung von einem Benutzer mit einem anderen Konto veröffentlicht wurde.

### Note

Wenn die gemeinsame Nutzung einer Anwendung mit dem Vorgang „Anwendung rückgängig machen“ für eine AWS Organisation aufgehoben wird, kann sie nicht erneut für die Organisation freigegeben AWS werden.

Weitere Informationen zu AWS Organizations finden Sie im [AWS Organizations Benutzerhandbuch](#).

## Entfernen von Berechtigungen durch den Herausgeber

Entfernen von Berechtigungen durch den Herausgeber mithilfe der Konsole

Um die gemeinsame Nutzung einer Anwendung über den rückgängig zu machen AWS Management Console, entfernen Sie die Richtlinienerklärung, in der die Anwendung mit anderen AWS Konten geteilt wird. Führen Sie dazu die folgenden Schritte aus:

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).

2. Wählen Sie im linken Navigationsbereich die Option Available Applications (Verfügbare Anwendungen).
3. Wählen Sie die Anwendung aus, deren Freigabe Sie aufheben möchten.
4. Wählen Sie die Registerkarte Sharing (Freigabe).
5. Wählen Sie im Abschnitt Application policy statements (Anwendungsrichtlinienanweisungen) die Richtlinienanweisung aus, die die Anwendung für die Konten freigibt, für die Sie die Freigabe aufheben möchten.
6. Wählen Sie Löschen.
7. Es wird eine Bestätigungsmeldung angezeigt. Klicken Sie erneut auf Delete (Löschen).

Der Herausgeber entfernt Berechtigungen über den AWS CLI

Um die Freigabe einer Anwendung über den rückgängig zu machen AWS CLI, kann der Herausgeber die Berechtigungen entfernen oder anderweitig ändern, indem er den [put-application-policy](#) Befehl verwendet, um die Anwendung privat zu machen oder sie für eine andere Gruppe von AWS Konten freizugeben.

Weitere Informationen zum Ändern von Berechtigungen mithilfe der AWS CLI finden Sie unter [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#).

## Verwaltungskonto: Aufheben der gemeinsamen Nutzung einer Anwendung

Verwaltungskonto Aufheben der gemeinsamen Nutzung einer Anwendung für eine AWS Organisation über die Konsole

Um die Freigabe einer Anwendung für eine AWS Organisation über die aufzuheben AWS Management Console, kann ein Benutzer des Verwaltungskontos wie folgt vorgehen:

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).
2. Wählen Sie im linken Navigationsbereich die Option Available Applications (Verfügbare Anwendungen).
3. Wählen Sie in der Kachel der Anwendung die Option Unshare (Freigabe aufheben).
4. Bestätigen Sie im Meldungsfeld zum Aufheben der Freigabe, dass Sie die Freigabe der Anwendung aufheben möchten, indem Sie die Organisations-ID und den Anwendungsnamen eingeben und dann Save (Speichern) auswählen.

## Verwaltungskonto: Aufheben der Freigabe einer Anwendung für eine AWS Organisation über AWS CLI

Um die Freigabe einer Anwendung für eine AWS Organisation aufzuheben, kann ein Benutzer mit dem Verwaltungskonto den `aws serverlessrepo unshare-application` Befehl ausführen.

Mit dem folgenden Befehl *application-id* wird die Freigabe einer Anwendung von einer AWS Organisation aufgehoben, wobei der Amazon-Ressourcenname (ARN) der Anwendung und *organization-id* die AWS Organisations-ID angegeben sind:

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

## Löschen einer Anwendung

Sie können Anwendungen aus dem löschen, AWS Serverless Application Repository indem Sie entweder die AWS Management Console oder die AWS SAM CLI verwenden.

### Löschen einer Anwendung (Konsole)

Gehen Sie wie folgt vor AWS Management Console, um eine veröffentlichte Anwendung über die zu löschen.

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).
2. Wählen Sie für My Applications (Eigene Anwendungen) die Anwendung aus, die Sie löschen möchten.
3. Wählen Sie auf der Detailseite der Anwendung Delete application (Anwendung löschen) aus.
4. Wählen Sie Delete application (Anwendung löschen) aus, um den Löschvorgang abzuschließen.

### Löschen einer Anwendung (AWS CLI)

Um eine veröffentlichte Anwendung mit dem zu löschen AWS CLI, führen Sie den [aws serverlessrepo delete-application](#) Befehl aus.

Der folgende Befehl löscht eine Anwendung, wobei *application-id* der Amazon-Ressourcenname (ARN) der Anwendung ist:

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

## Veröffentlichen der neuen Version einer vorhandenen Anwendung

In diesem Abschnitt erfahren Sie, wie Sie mithilfe der AWS SAM CLI oder der eine neue Version einer vorhandenen Anwendung auf der veröffentlichen AWS Management Console. AWS Serverless Application Repository Anweisungen zum Veröffentlichen einer neuen Anwendung finden Sie unter [So veröffentlichen Sie Anwendungen](#).

### Veröffentlichen der neuen Version einer vorhandenen Anwendung (AWS CLI)

Der einfachste Weg, eine neue Version einer vorhandenen Anwendung zu veröffentlichen, besteht darin, eine Reihe von AWS SAM CLI-Befehlen zu verwenden. Weitere Informationen finden Sie unter [Veröffentlichen einer Anwendung mit der AWS SAM CLI](#) im AWS Serverless Application Model (AWS SAM) Developer Guide.

### Veröffentlichen der neuen Version einer vorhandenen Anwendung (Konsole)

Gehen Sie folgendermaßen vor, um eine neue Version einer Anwendung zu veröffentlichen, die Sie zuvor veröffentlicht haben:

1. Öffnen Sie die [AWS Serverless Application Repository -Konsole](#).
2. Wählen Sie im Navigationsbereich My Applications (Meine Anwendungen) aus, um eine Liste der von Ihnen erstellten Anwendungen anzuzeigen.
3. Wählen Sie die Anwendung aus, für die Sie eine neue Version veröffentlichen möchten.
4. Wählen Sie Publish new version (Neue Version veröffentlichen) aus.
5. Geben Sie unter Versions (Versionen) die folgenden Anwendungsinformationen ein:

Eigenschaft	Erforderlich	Beschreibung
Semantic version (Semantische Version)	TRUE	Die semantische Version der Anwendung. Weitere Informationen finden Sie auf der <a href="#">Semantic Versioning-Website</a> .  Sie müssen einen Wert für diese Eigenschaft angeben, um Ihre Anwendung öffentlich zu machen.

Eigenschaft	Erforderlich	Beschreibung
Quellcode-URL	FALSE	Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung.
SAM-Vorlage	TRUE	Eine gültige Vorlage AWS Serverless Application Model (AWS SAM), die die verwendeten AWS Ressourcen definiert.

6. Wählen Sie Publish version (Version veröffentlichen) aus.

## Verifiziertes Autoren-Badge

Bei den verifizierten Autoren AWS Serverless Application Repository handelt es sich um solche, die AWS als vernünftiger und umsichtiger Dienstleister die vom Antragsteller bereitgestellten Informationen nach bestem Wissen und Gewissen geprüft und bestätigt haben, dass die Identität des Anfragenden den Angaben entspricht.

Anwendungen verifizierter Autoren zeigen ein verifiziertes Autoren-Badge sowie einen Link zum öffentlichen Profil des Autors an. Das verifizierte Autoren-Badge wird sowohl in den Suchergebnissen als auch auf der Anwendungsdetailseite angezeigt.

## Anfordern eines verifizierten Autoren-Badge

[Sie können in der beantragen, als verifizierter Autor zugelassen zu werden, AWS Serverless Application Repository indem Sie eine E-Mail an `serverlessrepo-verified-author@amazon.com` senden.](#) Sie müssen die folgenden Informationen angeben:

- Name des Autors
- AWS Konto-ID
- Öffentlich zugänglicher Profillink, z. B. Ihr LinkedIn Profil GitHub oder Ihr Profil

Nachdem Sie eine Anfrage für einen verifizierten Autorausweis eingereicht haben, können Sie AWS innerhalb weniger Tage mit einer Antwort rechnen. Möglicherweise werden Sie nach zusätzlichen Informationen gefragt, bevor Ihre Anfrage genehmigt wird.

Nachdem Ihre Anfrage genehmigt wurde, können Sie erwarten, dass das verifizierte Autoren-Badge für Ihre Anwendungen innerhalb eines Tages angezeigt wird.

#### Note

Das Badge für verifizierte Autoren wird für alle Anwendungen angezeigt, die sowohl dem AWS Konto als auch dem Autorennamen entsprechen. Da AWS Konten mehrere Autoren haben können, werden Badges in Anwendungen, die einen anderen Autorennamen haben, nicht angezeigt. Damit Autoren-Badges in Anwendungen mit unterschiedlichen Autorennamen angezeigt werden, müssen Sie eine andere Anfrage für diesen Autor senden.

## Lambda-Layer teilen

Wenn Sie Funktionen in einer Lambda-Schicht implementiert haben, möchten Sie Ihre Ebene möglicherweise gemeinsam nutzen, ohne eine globale Instanz davon zu hosten. Wenn Sie Ebenen auf diese Weise freigeben, können andere Benutzer eine Instance Ihrer Ebene in ihrem eigenen Konto bereitstellen. Dadurch wird verhindert, dass Clientanwendungen von einer globalen Instance Ihrer Ebene abhängig sind. Das AWS Serverless Application Repository ermöglicht es Ihnen, Lambda-Schichten auf diese Weise einfach gemeinsam zu nutzen.

Weitere Informationen zu Lambda-Layern finden Sie unter [AWS Lambda Layers](#) im Developer Guide.

## So funktioniert's

Im Folgenden finden Sie die Schritte zum Freigeben der Ebene mit dem AWS Serverless Application Repository. Auf diese Weise kann eine Kopie Ihres Layers im AWS Konto des Benutzers erstellt werden.

1. Definieren Sie eine serverlose Anwendung mit einer AWS SAM Vorlage, die Ihren Layer als Ressource enthält, also entweder als Ressource [AWS::Serverless::LayerVersion](#) oder als [AWS::Lambda::LayerVersion](#) Ressource.
2. Veröffentlichen Sie Ihre Anwendung auf der AWS Serverless Application Repository und teilen Sie sie (entweder öffentlich oder privat).

3. Ein Kunde stellt Ihre Anwendung bereit, wodurch eine Kopie Ihres Layers in seinem eigenen AWS Konto erstellt wird. Der Kunde kann jetzt in seiner Client-Anwendung auf den Amazon-Ressourcennamen (ARN) des Layers in seinem AWS Konto verweisen.

## Beispiel

Im Folgenden finden Sie eine AWS SAM Beispielvorlage für eine Anwendung, die die Lambda-Schicht enthält, die Sie teilen möchten:

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

Wenn ein Kunde Ihre Anwendung von aus bereitstellt AWS Serverless Application Repository, wird in seinem AWS Konto eine Ebene erstellt. Der ARN der Ebene sieht ungefähr wie folgt aus:

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

Der Kunde kann diesen ARN nun in seiner eigenen Client-Anwendung referenzieren, wie in diesem Beispiel:

```
Resources:
  MyFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.7
      CodeUri: source/app-code/
      Layers:
        - arn:aws:lambda:us-east-1:012345678901:layer:shared-layer:1
```

# Bereitstellen von Anwendungen

Der folgende Abschnitt hilft Ihnen dabei, mehr über die Suche und Bereitstellung von serverlosen Anwendungen zu erfahren, die im AWS Serverless Application Repository veröffentlicht wurden. Sie können auf der öffentlichen [Website](#) nach Anwendungen suchen, die öffentlich verfügbar sind, ohne über ein AWS Konto zu verfügen. Alternativ können Sie von der AWS Lambda Konsole aus nach Anwendungen suchen.

Einige Anwendungen verfügen über ein verifiziertes Autoren-Badge mit einem Link zum Profil des Autors. Ein Autor gilt als verifizierter Autor, wenn AWS er als vernünftiger und umsichtiger Dienstleister die vom Antragsteller bereitgestellten Informationen nach bestem Wissen und Gewissen überprüft und bestätigt hat, dass die Identität des Anfragenden den Angaben entspricht.

Bevor Sie Anwendungen vom bereitstellen, lesen Sie sich die folgenden Themen durch AWS Serverless Application Repository, um mehr über Berechtigungen und Anwendungsfunktionen für die Anwendungsbereitstellung zu erfahren.

Themen

- [Anwendungsbereitstellungsberechtigungen](#)
- [Anwendungsfunktionen: IAM-Rollen, Ressourcenrichtlinien und verschachtelte Anwendungen](#)
- [So stellen Sie Anwendungen bereit](#)

## Anwendungsbereitstellungsberechtigungen

Um eine Anwendung in der bereitstellen zu können AWS Serverless Application Repository, benötigen Sie die entsprechenden Berechtigungen. Es gibt drei Kategorien von Anwendungen, für die Sie über Berechtigungen zum Bereitstellen verfügen:

- Privat — Anwendungen, die mit demselben Konto erstellt und mit keinem anderen Konto geteilt wurden. Sie sind berechtigt, Anwendungen bereitzustellen, die mit Ihrem AWS Konto erstellt wurden.
- Privat geteilt — Anwendungen, die der Herausgeber ausdrücklich für eine bestimmte Gruppe von AWS Konten freigegeben hat. Sie sind berechtigt, Anwendungen bereitzustellen, die mit Ihrem AWS Konto geteilt wurden.
- Öffentlich geteilt — Anwendungen, die der Herausgeber mit allen geteilt hat. Sie haben die Berechtigung, alle öffentlich freigegebenen Anwendungen bereitzustellen.

Sie können nur nach Anwendungen suchen, für die Sie über Berechtigungen verfügen. Dazu gehören Anwendungen, die mit Ihrem AWS Konto erstellt, privat mit Ihrem AWS Konto geteilt und öffentlich geteilt wurden. Alle anderen Anwendungen werden nicht für Sie angezeigt.

### Important

Anwendungen, die verschachtelte Anwendungen enthalten, erben die Freigabeeinschränkungen der verschachtelten Anwendungen. Nehmen wir zum Beispiel an, dass eine Anwendung öffentlich geteilt wird, aber sie enthält eine verschachtelte Anwendung, die nur privat mit dem AWS Konto geteilt wird, mit dem die übergeordnete Anwendung erstellt wurde. In diesem Fall können Sie die übergeordnete Anwendung nicht bereitstellen, wenn Ihr AWS Konto nicht berechtigt ist, die verschachtelte Anwendung bereitzustellen. Weitere Informationen über verschachtelte Anwendungen finden Sie im Thema [Verschachtelte Anwendungen](#) im AWS Serverless Application Model -Entwicklerhandbuch.

## Anwendungsfunktionen: IAM-Rollen, Ressourcenrichtlinien und verschachtelte Anwendungen

Bevor Sie eine Anwendung bereitstellen können, AWS Serverless Application Repository überprüft die Vorlage der Anwendung auf IAM-Rollen, AWS Ressourcenrichtlinien und verschachtelte Anwendungen, die laut Vorlage erstellt werden sollen. IAM-Ressourcen, z. B. eine IAM-Rolle mit Vollzugriff, können jede Ressource in Ihrem Konto ändern. AWS Daher wird empfohlen, die mit der Anwendung verknüpften Berechtigungen zu prüfen, bevor Sie fortfahren, damit Sie nicht versehentlich Ressourcen mit eskalierten Berechtigungen erstellen. Um sicherzustellen, dass Sie dies getan haben, müssen Sie bestätigen, dass die Anwendung Funktionen enthält, bevor wir die Anwendung in Ihrem Namen bereitstellen AWS Serverless Application Repository können.

Anwendungen können folgende vier Funktionen enthalten: `CAPABILITY_IAM`, `CAPABILITY_NAMED_IAM`, `CAPABILITY_RESOURCE_POLICY` und `CAPABILITY_AUTO_EXPAND`.

Für die folgenden Ressourcen müssen Sie `CAPABILITY_IAM` oder `CAPABILITY_NAMED_IAM` angeben: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), und [AWS::IAM::Role](#). Beinhaltet die Anwendung IAM-Ressourcen mit benutzerdefinierten Namen, müssen Sie `CAPABILITY_NAMED_IAM` angeben. Ein Beispiel dazu, wie Sie bestimmte Funktionen angeben, finden Sie unter [Suchen und Bestätigen von Anwendungsfunktionen \(AWS CLI\)](#).

Für die folgenden Ressourcen müssen Sie Folgendes angeben: `CAPABILITY_RESOURCE_POLICY`: [AWS::Lambda::LayerVersionPermission](#), [AWS::Events::EventBusPolicy](#), [AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#), [AWS::S3::BucketPolicy](#) und [AWS::SQS::QueuePolicy](#), [AWS::SNS::TopicPolicy](#).

Bei Anwendungen, die eine oder mehrere verschachtelte Anwendungen enthalten, müssen Sie `CAPABILITY_AUTO_EXPAND` angeben. Weitere Informationen über verschachtelte Anwendungen finden Sie im Thema [Verschachtelte Anwendungen](#) im AWS Serverless Application Model - Entwicklerhandbuch.

## Suchen und Bestätigen von Anwendungsfunktionen (Konsole)

Sie finden Anwendungen, die AWS Serverless Application Repository auf der [AWS Serverless Application Repository Website](#) oder über die [Lambda-Konsole \(auf der Seite Funktion erstellen unter dem AWS Serverless Application Repository Tab\)](#) verfügbar sind.

Anwendungen, die eine Bestätigung von Funktionen für das Erstellen von benutzerdefinierten IAM-Rollen oder Ressourcenrichtlinien benötigen, werden standardmäßig nicht in den Suchergebnissen angezeigt. Zum Suchen nach Anwendungen, die diese Funktionen enthalten, müssen Sie das Kontrollkästchen `Show apps that create custom IAM roles or resource policies` (Apps zeigen, die benutzerdefinierte IAM-Rollen oder Ressourcenrichtlinien erstellen) aktivieren.

Sie können den Funktionsumfang einer Anwendung auf der Registerkarte `Permissions` (Berechtigungen) überprüfen, wenn Sie eine Anwendung auswählen. Zum Bereitstellen der Anwendung müssen Sie das Kontrollkästchen `I acknowledge this application creates custom IAM roles or resource policies` (Ich bestätige, dass diese Anwendung benutzerdefinierte IAM-Rollen und Ressourcenrichtlinien erstellt) aktivieren. Wenn Sie diese Funktionen nicht bestätigen, wird die folgende Fehlermeldung angezeigt: `Bestätigung erforderlich`. Aktivieren Sie zur Bereitstellung das Kontrollkästchen im Abschnitt `Anwendungsparameter konfigurieren`.

## Anzeigen von Anwendungsfunktionen (AWS CLI)

Um die Funktionen einer Anwendung mithilfe von anzuzeigen AWS CLI, benötigen Sie zunächst den Amazon-Ressourcennamen (ARN) der Anwendung. Sie können dann den folgenden Befehl ausführen:

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

Die Eigenschaft [requiredCapabilities](#) enthält die Liste der Anwendungsfunktionen, die Sie vor der Bereitstellung der Anwendung bestätigen müssen. Beachten Sie, dass die Anwendung keine erforderlichen Funktionen besitzt, wenn die Eigenschaft [requiredCapabilities](#) leer ist.

## So stellen Sie Anwendungen bereit

In diesem Abschnitt finden Sie Verfahren zum Bereitstellen serverloser Anwendungen vom mithilfe AWS Serverless Application Repository von AWS Management Console oder AWS CLI

### Bereitstellen einer neuen Anwendungsversion (Konsole)

In diesem Abschnitt erfahren Sie, wie Sie eine neue Anwendung AWS Serverless Application Repository mithilfe von bereitstellen. AWS Management Console Anweisungen zum Bereitstellen einer neuen Version einer vorhandenen Anwendung finden Sie unter [Aktualisieren von Anwendungen](#).

### Suchen und Entwickeln von Anwendungen

Verwenden Sie das folgende Verfahren, um eine Anwendung in der AWS Serverless Application Repository zu suchen, zu konfigurieren und bereitzustellen.

So suchen und konfigurieren Sie eine Anwendung in der AWS Serverless Application Repository

1. Öffnen Sie die [öffentliche AWS Serverless Application Repository -Startseite](#) oder öffnen Sie die [AWS Lambda -Konsole](#). Wählen Sie Create function (Funktion erstellen) aus und dann Browse serverless app repository (Serverloses App-Repository durchsuchen).
2. Suchen Sie nach einer Anwendung.

#### Note

Zum Anzeigen von Anwendungen, die benutzerdefinierte IAM-Rollen oder Ressourcenrichtlinien enthalten, aktivieren Sie das Kontrollkästchen Show apps that create custom IAM roles or resource policies (Apps zeigen, die benutzerdefinierte IAM-Rollen oder Ressourcenrichtlinien erstellen). Weitere Informationen zu benutzerdefinierten IAM-Rollen und Ressourcenrichtlinien finden Sie unter [Bestätigen von Anwendungsfunktionen](#).

3. Wählen Sie eine Anwendung aus, um Details wie ihre Berechtigungen, Funktionen und die Häufigkeit, mit der sie von AWS Kunden bereitgestellt wurde, einzusehen.

Die Anzahl der Bereitstellungen wird für die AWS Region angezeigt, in der Sie die Anwendung bereitstellen möchten.

4. Sehen Sie sich auf der Seite mit den Anwendungsdetails die Berechtigungen und Anwendungsressourcen der Anwendung an, indem Sie sich die AWS SAM Vorlagen-, Lizenz- und Readme-Datei ansehen. Auf dieser Seite können Sie bei öffentlich freigegebenen Anwendungen auch nach dem Link der Source code URL (Quellcode-URL) suchen. Wenn die Anwendung verschachtelte Anwendungen enthält, können Sie auch die Details der verschachtelten Anwendungen auf dieser Seite sehen.
5. Konfigurieren Sie die Anwendung im Abschnitt Application Settings (Anwendungseinstellungen). Eine Anleitung zur Konfiguration einer bestimmten Anwendung finden Sie in der Readme-Datei der Anwendung.

Konfigurationsvoraussetzungen können unter anderem die Angabe des Namens einer Ressource sein, wenn die Anwendung Zugriff darauf erhalten soll. Eine solche Ressource kann eine Amazon DynamoDB-Tabelle, ein Amazon S3 S3-Bucket oder eine Amazon API Gateway Gateway-API sein.

6. Wählen Sie Bereitstellen. Damit gelangen Sie auf die Seite Deployment status (Bereitstellungsstatus).

#### Note

Wenn die Anwendung über Funktionen verfügt, die eine Bestätigung erfordern, müssen Sie das Kontrollkästchen `I acknowledge this application creates custom IAM roles or resource policies` (Ich bestätige, dass diese Anwendung benutzerdefinierte IAM-Rollen oder Ressourcenrichtlinien erstellt) aktivieren, bevor Sie die Anwendung bereitstellen. Andernfalls wird ein Fehler ausgegeben. Weitere Informationen zu benutzerdefinierten IAM-Rollen und Ressourcenrichtlinien finden Sie unter [Bestätigen von Anwendungsfunktionen](#).

7. Auf der Seite Deployment status (Bereitstellungsstatus) können Sie den Fortschritt Ihrer Bereitstellung verfolgen. Während Sie auf den Abschluss Ihrer Bereitstellung warten, können Sie nach anderen Anwendungen suchen und suchen und über die Lambda-Konsole zu dieser Seite zurückkehren.

Nachdem Ihre Anwendung erfolgreich bereitgestellt wurde, können Sie die Ressourcen überprüfen und verwalten, die mithilfe vorhandener AWS Tools erstellt wurden.

## Bereitstellen einer neuen Anwendungsversion (AWS CLI)

In diesem Abschnitt erfahren Sie, wie Sie mithilfe von eine neue Anwendung AWS Serverless Application Repository von bereitstellen AWS CLI. Anweisungen zum Bereitstellen einer neuen Version einer vorhandenen Anwendung finden Sie unter [Aktualisieren von Anwendungen](#).

### Suchen und Bestätigen von Anwendungsfunktionen (AWS CLI)

Gehen Sie wie folgt vor AWS CLI, um die Funktionen einer Anwendung mithilfe von zu bestätigen:

1. Überprüfen Sie die Funktionen der Anwendung. Verwenden Sie den folgenden AWS CLI Befehl, um die Funktionen einer Anwendung zu überprüfen:

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

Die Antworteigenschaft [requiredCapabilities](#) enthält die Liste der Anwendungsfunktionen, die Sie vor der Bereitstellung der Anwendung bestätigen müssen. Sie können auch die [GetApplication API](#) in verwenden AWS SDKs , um diese Daten abzurufen.

2. Erstellen Sie das Changeset. Sie müssen die erforderlichen [Funktionen](#) bereitstellen, wenn Sie das Changeset erstellen. AWS CloudFormation Verwenden Sie beispielsweise den folgenden AWS CLI Befehl, um eine Anwendung bereitzustellen, indem Sie deren Funktionen bestätigen:

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

Die Änderungsatz-ID wird zurückgegeben, wenn dieser Befehl erfolgreich ausgeführt wird. Sie benötigen die Änderungsatz-ID für den nächsten Schritt. Sie können auch die [CreateCloudFormationChangeSet API](#) in verwenden, AWS SDKs um den Changeset zu erstellen.

Mit dem folgenden AWS CLI Befehl wird beispielsweise eine Anwendung bestätigt, die eine [AWS::IAM::Role](#) Ressource mit einem benutzerdefinierten Namen und eine oder mehrere verschachtelte Anwendungen enthält:

```
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. Führen Sie das Changeset aus. Durch das Ausführen des Änderungssatzes wird die Bereitstellung tatsächlich ausgeführt. Geben Sie die Änderungssatz-ID an, die beim Erstellen des Änderungssatzes im vorherigen Schritt zurückgegeben wurde.

Der folgende AWS CLI Beispielbefehl führt das Changeset der Anwendung aus, um die Anwendung bereitzustellen:

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

Sie können auch die [ExecuteChangeSet API](#) verwenden, AWS SDKs um das Changeset auszuführen.

## Löschen von Anwendungs-Stacks

Um eine Anwendung zu löschen, die Sie zuvor mit dem bereitgestellt haben AWS Serverless Application Repository, gehen Sie genauso vor wie beim Löschen eines AWS CloudFormation Stacks:

- AWS Management Console: Informationen zum Löschen einer Anwendung mit dem AWS Management Console finden Sie unter [Löschen eines Stacks auf der AWS CloudFormation Konsole](#) im AWS CloudFormation Benutzerhandbuch.
- AWS CLI: Informationen zum Löschen einer Anwendung mithilfe [von finden Sie unter Löschen eines Stacks](#) im AWS CloudFormation Benutzerhandbuch. AWS CLI

## Aktualisieren von Anwendungen

Nachdem Sie eine Anwendung aus dem bereitgestellt haben AWS Serverless Application Repository, möchten Sie sie möglicherweise aktualisieren. Sie möchten beispielsweise eine Anwendungseinstellung ändern oder die Anwendung auf die neueste Version aktualisieren, die veröffentlicht wurde.

In den folgenden Abschnitten wird beschrieben, wie Sie eine neue Version einer Anwendung bereitstellen, indem Sie entweder den AWS Management Console oder den verwenden AWS CLI.

## Aktualisieren von Anwendungen (Konsole)

Verwenden Sie zum Aktualisieren einer zuvor bereitgestellten Anwendung dasselbe Verfahren wie beim Bereitstellen einer neuen Anwendung und geben Sie denselben Anwendungsnamen an, mit dem Sie sie ursprünglich bereitgestellt haben. Insbesondere wird der Ihrem AWS Serverless Application Repository Anwendungsnamen `serverlessrepo-` vorangestellt. Um jedoch eine neue Version Ihrer Anwendung bereitzustellen, geben Sie den ursprünglichen Anwendungsnamen an, ohne `serverlessrepo-` voranzustellen.

Wenn Sie beispielsweise eine Anwendung mit dem Namen `MyApplication` bereitgestellt haben, wäre der Stack-Name `serverlessrepo-MyApplication`. Um diese Anwendung zu aktualisieren, würden Sie den Namen `MyApplication` erneut angeben. Geben Sie nicht den vollständigen Stacknamen von an. `serverlessrepo-MyApplication`

Für alle anderen Anwendungseinstellungen können Sie die Werte der vorherigen Bereitstellung übernehmen oder neue Werte bereitstellen.

## Aktualisieren von Anwendungen (AWS CLI)

Verwenden Sie zum Aktualisieren einer zuvor bereitgestellten Anwendung dasselbe Verfahren wie beim Bereitstellen einer neuen Anwendung und geben Sie denselben `--stack-name` an, mit dem Sie sie ursprünglich bereitgestellt haben. Insbesondere steht er vor AWS Serverless Application Repository Ihrem Stack-Namens `serverlessrepo-`. Um jedoch eine neue Version Ihrer Anwendung bereitzustellen, geben Sie den ursprünglichen Stack-Namen an, ohne `serverlessrepo-` voranzustellen.

Wenn Sie beispielsweise eine Anwendung mit dem Stack-Namen `MyApplication` bereitgestellt haben, lautet der erstellte Stack-Name `serverlessrepo-MyApplication`. Um diese Anwendung zu aktualisieren, würden Sie den Namen `MyApplication` erneut angeben — geben Sie nicht den vollständigen Stacknamen von an. `serverlessrepo-MyApplication`

# Sicherheit in der AWS Serverless Application Repository

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS Dienste in der AWS Cloud ausführt. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der [AWS -Compliance-Programme](#) regelmäßig. Informationen zu den Compliance-Programmen, die für das AWS Serverless Application Repository gelten, finden Sie unter [Vom Compliance-Programm abgedeckte AWS -Services](#).
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Dienst, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

In dieser Dokumentation wird erläutert, wie das Modell der übergreifenden Verantwortlichkeit bei der Verwendung von AWS Serverless Application Repository zum Tragen kommt. In den folgenden Themen erfahren Sie, wie Sie den konfigurieren AWS Serverless Application Repository , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AWS Serverless Application Repository Ressourcen unterstützen.

## Themen

- [Datenschutz in der AWS Serverless Application Repository](#)
- [Identity and Access Management für das AWS Serverless Application Repository](#)
- [Protokollierung und Überwachung im AWS Serverless Application Repository](#)
- [Konformitätsvalidierung für die AWS Serverless Application Repository](#)
- [Resilienz in der AWS Serverless Application Repository](#)
- [Infrastruktursicherheit in der AWS Serverless Application Repository](#)
- [Zugriff AWS Serverless Application Repository über einen Schnittstellenendpunkt \(AWS PrivateLink\)](#)

# Datenschutz in der AWS Serverless Application Repository

Das AWS [Modell](#) der gilt für den Datenschutz in AWS Serverless Application Repository. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter [Häufig gestellte Fragen zum Datenschutz](#). Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag [AWS -Modell der geteilten Verantwortung und in der DSGVO](#) im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein. AWS CloudTrail Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter [Arbeiten mit CloudTrail Pfaden](#) im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-3](#).

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit der Konsole, der AWS Serverless Application Repository API oder auf andere AWS-Services Weise arbeiten oder diese verwenden. AWS CLI AWS SDKs Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle

verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

## Verschlüsselung während der Übertragung

AWS Serverless Application Repository API-Endpunkte unterstützen nur sichere Verbindungen über HTTPS. Wenn Sie AWS Serverless Application Repository Ressourcen mit dem AWS Management Console AWS SDK oder der AWS Serverless Application Repository API verwalten, wird die gesamte Kommunikation mit Transport Layer Security (TLS) verschlüsselt.

Eine vollständige Liste der API-Endpunkte finden Sie unter [AWS Regionen und Endpunkte](#) in Allgemeine AWS-Referenz.

## Verschlüsselung im Ruhezustand

Das AWS Serverless Application Repository verschlüsselt Dateien, die Sie hochladen AWS Serverless Application Repository, einschließlich Bereitstellungspaketen und Layer-Archiven.

## Identity and Access Management für das AWS Serverless Application Repository

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. AWS Serverless Application Repository IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Einen Überblick über die Funktionsweise von IAM finden Sie unter [Grundlegendes zur Funktionsweise von IAM](#) im IAM-Benutzerhandbuch.

### Themen

- [Zielgruppe](#)
- [Authentifizieren mit Identitäten](#)
- [Verwalten des Zugriffs mit Richtlinien](#)
- [So AWS Serverless Application Repository funktioniert das mit IAM](#)

- [AWS Serverless Application Repository Beispiele für identitätsbasierte Richtlinien](#)
- [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#)
- [AWS Serverless Application Repository API-Berechtigungen: Referenz zu Aktionen und Ressourcen](#)
- [Problembehandlung bei AWS Serverless Application Repository Identität und Zugriff](#)

## Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Art der Arbeit ab, in der Sie tätig sind. AWS Serverless Application Repository

**Dienstbenutzer** — Wenn Sie den AWS Serverless Application Repository Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr AWS Serverless Application Repository Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Unter [Problembehandlung bei AWS Serverless Application Repository Identität und Zugriff](#) finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in AWS Serverless Application Repository haben.

**Serviceadministrator** — Wenn Sie in Ihrem Unternehmen für die AWS Serverless Application Repository Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AWS Serverless Application Repository. Es ist Ihre Aufgabe, zu bestimmen, auf welche AWS Serverless Application Repository Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM nutzen kann AWS Serverless Application Repository, finden Sie unter [So AWS Serverless Application Repository funktioniert das mit IAM](#).

**IAM-Administrator:** Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf AWS Serverless Application Repository verfassen können. Beispiele für AWS Serverless Application Repository identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter [AWS Serverless Application Repository Beispiele für identitätsbasierte Richtlinien](#)

## Authentifizieren mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS, übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter [So melden Sie sich bei Ihrem an AWS-Konto](#) im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, mit denen Sie Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter [AWS Signature Version 4 für API-Anforderungen](#) im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter [Multi-Faktor-Authentifizierung](#) im AWS IAM Identity Center - Benutzerhandbuch und [AWS Multi-Faktor-Authentifizierung \(MFA\) in IAM](#) im IAM-Benutzerhandbuch.

### AWS-Konto Root-Benutzer

Wenn Sie ein AWS-Konto erstellen, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-

Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter [Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

## IAM-Benutzer und -Gruppen

Ein [IAM-Benutzer](#) ist eine Identität innerhalb von Ihrem AWS-Konto, die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter [Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

Eine [IAM-Gruppe](#) ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdmins und dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter [Anwendungsfälle für IAM-Benutzer](#) im IAM-Benutzerhandbuch.

## IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität innerhalb von Ihrem AWS-Konto, die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie [von einer Benutzer- zu einer IAM-Rolle \(Konsole\) wechseln](#). Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter [Methoden für die Übernahme einer Rolle](#) im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- **Verbundbenutzerzugriff** – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter [Erstellen von Rollen für externe Identitätsanbieter \(Verbund\)](#) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter [Berechtigungssätze](#) im AWS IAM Identity Center -Benutzerhandbuch.
- **Temporäre IAM-Benutzerberechtigungen** – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- **Kontoübergreifender Zugriff** – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.
- **Serviceübergreifender Zugriff** — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - **Forward Access Sessions (FAS)** — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter [Zugriffssitzungen weiterleiten](#).
- **Servicerolle** – Eine Servicerolle ist eine [IAM-Rolle](#), die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM

erstellen, ändern und löschen. Weitere Informationen finden Sie unter [Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service](#) im IAM-Benutzerhandbuch.

- **Dienstbezogene Rolle** — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer Service-Verknüpfung verbunden ist. Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-Verknüpfte Rollen anzeigen, aber nicht bearbeiten.
- **Auf Amazon ausgeführte Anwendungen EC2** — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter [Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden](#).

## Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter [Übersicht über JSON-Richtlinien](#) im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console, der AWS CLI, oder der AWS API abrufen.

## Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter [Auswählen zwischen verwalteten und eingebundenen Richtlinien](#) im IAM-Benutzerhandbuch.

## Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#). Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

## Zugriffskontrolllisten (ACLs)

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF  
Weitere Informationen finden Sie unter [Übersicht über ACLs die Zugriffskontrollliste \(ACL\)](#) im Amazon Simple Storage Service Developer Guide.

## Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- **Berechtigungsgrenzen** – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter [Berechtigungsgrenzen für IAM-Entitäten](#) im IAM-Benutzerhandbuch.
- **Dienststeuerungsrichtlinien (SCPs)** — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter [Richtlinien zur Servicesteuerung](#) im AWS Organizations Benutzerhandbuch.
- **Ressourcenkontrollrichtlinien (RCPs)** — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich

auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter [Resource Control Policies \(RCPs\)](#) im AWS Organizations Benutzerhandbuch.

- **Sitzungsrichtlinien** – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter [Sitzungsrichtlinien](#) im IAM-Benutzerhandbuch.

## Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter [Bewertungslogik für Richtlinien](#).

## So AWS Serverless Application Repository funktioniert das mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf die zu verwalten AWS Serverless Application Repository, sollten Sie wissen, welche IAM-Funktionen für die Verwendung mit dem verfügbar sind. AWS Serverless Application Repository

Einen Überblick über die Funktionsweise von IAM finden Sie unter [Grundlegendes zur Funktionsweise von IAM im IAM-Benutzerhandbuch](#). Einen allgemeinen Überblick darüber, wie diese AWS Serverless Application Repository und andere AWS Dienste mit IAM funktionieren, finden Sie im IAM-Benutzerhandbuch unter [AWS Services That Work with IAM](#).

### Themen

- [Identitätsbasierte AWS Serverless Application Repository -Richtlinien](#)
- [AWS Serverless Application Repository Anwendungsrichtlinien](#)
- [Autorisierung auf der Basis von AWS Serverless Application Repository -Tags](#)
- [AWS Serverless Application Repository IAM-Rollen](#)

## Identitätsbasierte AWS Serverless Application Repository -Richtlinien

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. AWS Serverless Application Repository unterstützt bestimmte Aktionen, Ressourcen und Bedingungsschlüssel. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der [IAM-Referenz für JSON-Richtlinienelemente](#) im IAM-Benutzerhandbuch.

Dies ist ein Beispiel für eine Berechtigungsrichtlinie.

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

Die Richtlinie enthält zwei Anweisungen:

- Die erste Anweisung erteilt Berechtigungen für die AWS Serverless Application Repository Aktion `serverlessrepo:CreateApplication` für alle AWS Serverless Application Repository Ressourcen, wie durch das Platzhalterzeichen (\*) als Wert angegeben. Resource

- Die zweite Anweisung erteilt die Genehmigung für die AWS Serverless Application Repository Aktion `serverlessrepo:CreateApplicationVersion` an einer AWS Ressource, indem der Amazon-Ressourcenname (ARN) für eine AWS Serverless Application Repository Anwendung verwendet wird. Die Anwendung wird durch den `Resource`-Wert angegeben.

Das Element `Principal` ist in der Richtlinie nicht angegeben, da in identitätsbasierten Richtlinien die Angabe des Prinzipals als Empfänger der Berechtigung nicht erforderlich ist. Wenn Sie einem Benutzer eine Richtlinie zuweisen, ist der Benutzer automatisch der Prinzipal. Wird die Berechtigungsrichtlinie einer IAM-Rolle zugewiesen, erhält der in der Vertrauensrichtlinie der Rolle angegebene Prinzipal die Berechtigungen.

Eine Tabelle mit allen AWS Serverless Application Repository API-Vorgängen und den AWS Ressourcen, für die sie gelten, finden Sie unter [AWS Serverless Application Repository API-Berechtigungen: Referenz zu Aktionen und Ressourcen](#).

## Aktionen

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix AWS Serverless Application Repository verwendet: `serverlessrepo:`. Um beispielsweise jemandem die Erlaubnis zu erteilen, eine AWS Serverless Application Repository Instanz mit dem AWS Serverless Application Repository `SearchApplications` API-Vorgang auszuführen, nehmen Sie die `serverlessrepo:SearchApplications` Aktion in seine Richtlinie auf. Richtlinienanweisungen müssen entweder ein `Action` oder ein `NotAction`-Element enthalten. Die AWS Serverless Application Repository definiert ihre eigenen Aktionen, die Aufgaben beschreiben, die Sie mit diesem Dienst ausführen können.

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie wie folgt durch Kommata:

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

Sie können auch Platzhalter verwenden, um mehrere Aktionen anzugeben. Beispielsweise können Sie alle Aktionen festlegen, die mit dem Wort `List` beginnen, einschließlich der folgenden Aktion:

```
"Action": "serverlessrepo:List*"
```

Eine Liste der AWS Serverless Application Repository [Aktionen finden Sie AWS Serverless Application Repository im IAM-Benutzerhandbuch unter Definierte Aktionen von](#).

## Ressourcen

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein `Resource` oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen [Amazon-Ressourcennamen \(ARN\)](#) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (\*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

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

In der AWS Serverless Application Repository ist die primäre AWS Ressource eine AWS Serverless Application Repository Anwendung. AWS Serverless Application Repository Anwendungen sind mit eindeutigen Amazon-Ressourcennamen (ARNs) verknüpft, wie in der folgenden Tabelle dargestellt.

AWS Art der Ressource	Format des Amazon-Ressourcennamens (ARN)
Anwendung	<code>arn ::serverlessrepo: ::applications/ <i>partition</i> <i>region</i> <i>account-id</i> <i>application-name</i></code>

Weitere Informationen zum Format von ARNs finden Sie unter [Amazon Resource Names \(ARNs\) und AWS Service Namespaces](#).

Im Folgenden finden Sie ein Beispiel für eine Richtlinie, die Berechtigungen für die `serverlessrepo:ListApplications` Aktion für alle Ressourcen gewährt. AWS In der aktuellen Implementierung unterstützt die AWS Serverless Application Repository nicht die Identifizierung bestimmter AWS Ressourcen mithilfe der AWS Ressource ARNs (auch als Berechtigungen auf Ressourcenebene bezeichnet) für einige API-Aktionen. In diesen Fällen müssen Sie ein Platzhalterzeichen (\*) angeben.

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

Eine Tabelle mit allen AWS Serverless Application Repository API-Aktionen und den AWS Ressourcen, für die sie gelten, finden Sie unter [AWS Serverless Application Repository API-Berechtigungen: Referenz zu Aktionen und Ressourcen](#)

### Bedingungsschlüssel

Der stellt AWS Serverless Application Repository keine dienstspezifischen Bedingungsschlüssel bereit, unterstützt aber die Verwendung einiger globaler Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [AWS Globale Bedingungskontextschlüssel](#) im IAM-Benutzerhandbuch.

## Beispiele

Beispiele für AWS Serverless Application Repository identitätsbasierte Richtlinien finden Sie unter [AWS Serverless Application Repository Beispiele für identitätsbasierte Richtlinien](#)

## AWS Serverless Application Repository Anwendungsrichtlinien

Anwendungsrichtlinien bestimmen die Aktionen, die ein bestimmter Principal oder eine PrincipalOrg für eine AWS Serverless Application Repository Anwendung ausführen kann.

Sie können der mit einer AWS Serverless Application Repository Anwendung verknüpften Richtlinie Berechtigungen hinzufügen. Mit AWS Serverless Application Repository Anwendungen verknüpfte Berechtigungsrichtlinien werden als Anwendungsrichtlinien bezeichnet. [Anwendungsrichtlinien](#) sind Erweiterungen von [ressourcenbasierten IAM-Richtlinien](#). Die primäre Ressource ist die Anwendung. AWS Serverless Application Repository Sie können AWS Serverless Application Repository Anwendungsrichtlinien verwenden, um die Berechtigungen für die Anwendungsbereitstellung zu verwalten.

AWS Serverless Application Repository Anwendungsrichtlinien werden hauptsächlich von Herausgebern verwendet, um Verbrauchern die Erlaubnis zu erteilen, ihre Anwendungen bereitzustellen und damit verbundene Vorgänge durchzuführen, z. B. um nach diesen Anwendungen zu suchen und Details zu diesen Anwendungen anzuzeigen. Herausgeber können Anwendungsberechtigungen mit den folgenden drei Kategorien festlegen:

- Privat — Anwendungen, die mit demselben Konto erstellt und mit keinem anderen Konto geteilt wurden. Sie sind berechtigt, Anwendungen bereitzustellen, die mit Ihrem AWS Konto erstellt wurden.
- Privat geteilt — Anwendungen, die der Herausgeber ausdrücklich mit einer bestimmten Gruppe von AWS Konten oder AWS Organizations geteilt hat. Sie sind berechtigt, Anwendungen bereitzustellen, die mit Ihrem AWS Konto oder Ihrer AWS Organisation geteilt wurden.
- Öffentlich geteilt — Anwendungen, die der Herausgeber mit allen geteilt hat. Sie haben die Berechtigung, alle öffentlich freigegebenen Anwendungen bereitzustellen.

Sie können Berechtigungen erteilen, indem Sie den AWS CLI AWS SDKs, oder den verwenden AWS Management Console.

## Beispiele

Beispiele für die Verwaltung von AWS Serverless Application Repository Anwendungsrichtlinien finden Sie unter [AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien](#).

## Autorisierung auf der Basis von AWS Serverless Application Repository -Tags

Das unterstützt AWS Serverless Application Repository nicht die Steuerung des Zugriffs auf Ressourcen oder Aktionen auf der Grundlage von Tags.

## AWS Serverless Application Repository IAM-Rollen

Eine [IAM-Rolle](#) ist eine Entität in Ihrem AWS Konto, die über bestimmte Berechtigungen verfügt.

Verwenden temporärer Anmeldeinformationen mit dem AWS Serverless Application Repository

Sie können temporäre Anmeldeinformationen verwenden, um sich über einen Verbund anzumelden, eine IAM-Rolle anzunehmen oder eine kontenübergreifende Rolle anzunehmen. Sie erhalten temporäre Sicherheitsanmeldedaten, indem Sie AWS STS API-Operationen wie [AssumeRole](#) oder aufrufen [GetFederationToken](#).

Das AWS Serverless Application Repository unterstützt die Verwendung temporärer Anmeldeinformationen.

## Serviceverknüpfte Rollen

Das unterstützt AWS Serverless Application Repository keine dienstbezogenen Rollen.

## Servicerollen

Das unterstützt AWS Serverless Application Repository keine Servicerollen.

## AWS Serverless Application Repository Beispiele für identitätsbasierte Richtlinien

IAM-Benutzer besitzen keine Berechtigungen zum Erstellen oder Ändern von AWS Serverless Application Repository -Ressourcen. Sie können auch keine Aufgaben mit der AWS Management Console AWS CLI, oder AWS API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern und Rollen die Berechtigung zum Ausführen bestimmter API-Operationen für die

angegebenen Ressourcen gewähren, die diese benötigen. Der Administrator muss diese Richtlinien anschließend den IAM-Benutzern oder -Gruppen anfügen, die diese Berechtigungen benötigen.

Informationen zum Erstellen einer identitätsbasierten IAM-Richtlinie mithilfe dieser Beispieldokumente zu JSON-Richtlinien finden Sie unter [Creating Policies on the JSON Tab](#) im IAM-Benutzerhandbuch.

## Themen

- [Bewährte Methoden für Richtlinien](#)
- [Verwenden der AWS Serverless Application Repository -Konsole](#)
- [Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer](#)
- [Beispiele für vom Kunden verwaltete Richtlinien](#)

## Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien sind sehr leistungsfähig. Sie bestimmen, ob jemand AWS Serverless Application Repository Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Diese Aktionen können Kosten für Ihr AWS Konto verursachen. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Gewähren von geringsten Rechten – Gewähren Sie beim Erstellen benutzerdefinierter Richtlinien nur die Berechtigungen, die zum Ausführen einer Aufgabe erforderlich sind. Beginnen Sie mit einem Mindestsatz von Berechtigungen und gewähren Sie zusätzliche Berechtigungen wie erforderlich. Dies ist sicherer, als mit Berechtigungen zu beginnen, die zu weit gefasst sind, und dann später zu versuchen, sie zu begrenzen. Weitere Informationen finden Sie unter [Gewähren von geringsten Rechten](#) im IAM-Benutzerhandbuch.
- Aktivieren von MFA für sensible Vorgänge – Sie sollten die Verwendung der Multi-Faktor-Authentifizierung (MFA) von IAM-Benutzern fordern, um beim Zugriff auf sensible Ressourcen oder API-Operationen zusätzliche Sicherheit zu erhalten. Weitere Informationen finden Sie unter [Verwenden der Multi-Faktor-Authentifizierung \(MFA\) in AWS](#) im IAM-Benutzerhandbuch.
- Verwenden von Richtlinienbedingungen für zusätzliche Sicherheit – Definieren Sie die Bedingungen, unter denen Ihre identitätsbasierten Richtlinien den Zugriff auf eine Ressource zulassen, soweit praktikabel. Beispielsweise können Sie Bedingungen schreiben, die eine Reihe von zulässigen IP-Adressen festlegen, von denen eine Anforderung stammen muss. Sie können auch Bedingungen schreiben, die Anforderungen nur innerhalb eines bestimmten Datums- oder Zeitbereichs zulassen oder die Verwendung von SSL oder MFA fordern. Weitere Informationen finden Sie unter [IAM-JSON-Richtlinienelemente: Bedingung](#) im IAM-Benutzerhandbuch.

## Verwenden der AWS Serverless Application Repository -Konsole

Die AWS Serverless Application Repository Konsole bietet eine integrierte Umgebung, in der Sie AWS Serverless Application Repository Anwendungen entdecken und verwalten können. Die Konsole bietet Funktionen und Workflows, für die häufig zusätzlich zu den API-spezifischen Berechtigungen, die in der dokumentiert sind, Berechtigungen für die Verwaltung einer AWS Serverless Application Repository Anwendung erforderlich sind. [AWS Serverless Application Repository API-Berechtigungen: Referenz zu Aktionen und Ressourcen](#)

Weitere Informationen zu den für die Verwendung der AWS Serverless Application Repository Konsole erforderlichen Berechtigungen finden Sie unter. [Beispiele für vom Kunden verwaltete Richtlinien](#)

### Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder AWS .

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "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": "*"
}
]
```

## Beispiele für vom Kunden verwaltete Richtlinien

In den Beispielen in diesem Abschnitt finden Sie eine Gruppe von Beispielrichtlinien, die Sie Benutzern zuweisen können. Wenn Sie mit dem Erstellen von Richtlinien noch nicht vertraut sind, sollten Sie zunächst einen IAM-Benutzer in Ihrem Konto erstellen und ihm dann die Richtlinien der Reihe nach zuweisen. Sie können diese Beispiele auch verwenden, um eine einzelne benutzerdefinierte Richtlinie zu erstellen, die Berechtigungen zum Ausführen mehrerer Aktionen enthält, und sie dann an den Benutzer anfügen.

Weitere Informationen zum Anhängen von Richtlinien an Benutzer finden Sie unter [Hinzufügen von Berechtigungen für einen Benutzer](#) im IAM-Benutzerhandbuch.

### Beispiele

- [Publisher-Beispiel 1: Berechtigen eines Publishers zum Auflisten von Anwendungen](#)
- [Publisher-Beispiel 2: Berechtigen eines Publishers zum Anzeigen von Details zu einer Anwendung oder der Anwendungsversion](#)
- [Publisher-Beispiel 3: Berechtigen eines Publishers zum Erstellen einer Anwendung oder Anwendungsversion](#)
- [Publisher-Beispiel 4: Berechtigen eines Publishers zum Erstellen einer Anwendungsrichtlinie zur gemeinsamen Nutzung von Anwendungen mit anderen](#)
- [Konsumenten-Beispiel 1: Berechtigen eines Konsumenten zum Suchen nach Anwendungen](#)
- [Konsumenten-Beispiel 2: Berechtigen eines Konsumenten zum Anzeigen von Details zu einer Anwendung](#)
- [Konsumenten-Beispiel 3: Berechtigen eines Konsumenten zum Bereitstellen einer Anwendung](#)
- [Beispiel 4: Verweigern von Zugriff auf Bereitstellungsressourcen](#)
- [Beispiel 5: Verhindern, dass Konsumenten öffentliche Anwendungen suchen und bereitstellen](#)

## Publisher-Beispiel 1: Berechtigen eines Publishers zum Auflisten von Anwendungen

Ein IAM-Benutzer in Ihrem Konto muss über Berechtigungen für die `serverlessrepo:ListApplications`-Operation verfügen, bevor der Benutzer etwas in der Konsole sehen kann. Wenn Sie diese Berechtigungen gewähren, kann die Konsole die Liste der AWS Serverless Application Repository Anwendungen in dem AWS Konto anzeigen, das in der jeweiligen AWS Region erstellt wurde, zu der der Benutzer gehört.

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

## Publisher-Beispiel 2: Berechtigen eines Publishers zum Anzeigen von Details zu einer Anwendung oder der Anwendungsversion

Ein Benutzer kann eine AWS Serverless Application Repository Anwendung auswählen und Details der Anwendung anzeigen. Diese Details umfassen Autor, Beschreibung, Versionen und andere Konfigurationsinformationen. Zu diesem Zweck benötigt der Benutzer Berechtigungen für die API-Operationen `serverlessrepo:GetApplication` und `serverlessrepo:ListApplicationVersions` für das AWS Serverless Application Repository.

Im folgenden Beispiel werden diese Berechtigungen für die betreffende Anwendung gewährt, deren Amazon-Ressourcenname (ARN) als `Resource`-Wert angegeben wird.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewApplication",
      "Effect": "Allow",

```

```

    "Action": [
      "serverlessrepo:GetApplication",
      "serverlessrepo:ListApplicationVersions"
    ],
    "Resource": "arn:aws:serverlessrepo:region:account-id:applications/application-name"
  }
]
}

```

### Publisher-Beispiel 3: Berechtigen eines Publishers zum Erstellen einer Anwendung oder Anwendungsversion

Wenn Sie einem Benutzer Berechtigungen zum Erstellen von AWS Serverless Application Repository Anwendungen gewähren möchten, müssen Sie Berechtigungen für die `serverlessrepo:CreateApplicationVersions` Operationen `serverlessrepo:CreateApplication` und erteilen, wie in der folgenden Richtlinie beschrieben.

```

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

```

### Publisher-Beispiel 4: Berechtigen eines Publishers zum Erstellen einer Anwendungsrichtlinie zur gemeinsamen Nutzung von Anwendungen mit anderen

Damit Benutzer Anwendungen gemeinsam mit anderen nutzen können, müssen Sie ihnen, wie in der folgenden Richtlinie gezeigt, Berechtigungen zum Erstellen von Anwendungsrichtlinien gewähren.

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

### Konsumenten-Beispiel 1: Berechtigen eines Konsumenten zum Suchen nach Anwendungen

Damit Konsumenten nach Anwendungen suchen können, müssen Sie ihnen die folgenden Berechtigungen gewähren.

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

## Konsumenten-Beispiel 2: Berechtigen eines Konsumenten zum Anzeigen von Details zu einer Anwendung

Ein Benutzer kann eine AWS Serverless Application Repository Anwendung auswählen und Details der Anwendung wie Autor, Beschreibung, Versionen und andere Konfigurationsinformationen einsehen. Dazu muss der Benutzer über Berechtigungen für die folgenden AWS Serverless Application Repository Vorgänge verfügen.

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

## Konsumenten-Beispiel 3: Berechtigen eines Konsumenten zum Bereitstellen einer Anwendung

Damit Kunden Anwendungen bereitstellen können, müssen Sie ihnen Berechtigungen für das Durchführen einer Reihe von Operationen gewähren. Die folgende Richtlinie erteilt den Kunden die erforderlichen Berechtigungen.

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

```

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

```

### Note

Für die Bereitstellung einer Anwendung sind möglicherweise Berechtigungen zur Nutzung zusätzlicher AWS Ressourcen erforderlich. Da der denselben zugrunde liegenden Bereitstellungsmechanismus AWS Serverless Application Repository verwendet wie AWS CloudFormation, finden Sie weitere Informationen unter [Zugriffskontrolle mit AWS Identity and Access Management](#). Hilfe zu Bereitstellungsproblemen im Zusammenhang mit Berechtigungen finden Sie unter [Fehlerbehebung: Unzureichende IAM-Berechtigungen](#).

### Beispiel 4: Verweigern von Zugriff auf Bereitstellungsressourcen

Wenn eine Anwendung privat mit einem AWS Konto geteilt wird, können standardmäßig alle Benutzer in diesem Konto auf die Bereitstellungsressourcen aller anderen Benutzer desselben Kontos zugreifen. Die folgende Richtlinie verhindert, dass Benutzer in einem Konto auf Bereitstellungsressourcen zugreifen, die im Amazon S3 S3-Bucket für die gespeichert sind AWS Serverless Application Repository.

```

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

```

## Beispiel 5: Verhindern, dass Konsumenten öffentliche Anwendungen suchen und bereitstellen

Sie können verhindern, dass Benutzer bestimmte Aktionen für Anwendungen ausführen.

Die folgende Richtlinie gilt für öffentliche Anwendungen, indem für `serverlessrepo:applicationType public` angegeben wird. Sie verhindert, dass Benutzer eine Reihe von Aktionen ausführen, indem für `Effect Deny` angegeben wird. Weitere Informationen zu den für AWS Serverless Application Repository verfügbaren Bedingungsschlüsseln finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für 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

Diese Richtlinienerklärung kann auch als Servicesteuerungsrichtlinie verwendet und auf eine AWS Organisation angewendet werden. Weitere Informationen zu Service Control-Richtlinien finden Sie unter [Service Control-Richtlinien](#) im AWS Organizations Benutzerhandbuch.

# AWS Serverless Application Repository Beispiele für Anwendungsrichtlinien

Mit AWS Serverless Application Repository Anwendungen verknüpfte Berechtigungsrichtlinien werden als Anwendungsrichtlinien bezeichnet. Anwendungsrichtlinien bestimmen die Aktionen, die ein bestimmter Principal oder eine PrincipalOrg für eine AWS Serverless Application Repository Anwendung ausführen kann.

Eine AWS Serverless Application Repository Anwendung ist die primäre AWS Ressource in der. AWS Serverless Application Repository Anwendungsrichtlinien werden hauptsächlich von Herausgebern verwendet, um Verbrauchern die Erlaubnis zu erteilen, ihre Anwendungen bereitzustellen und damit verbundene Vorgänge durchzuführen, z. B. um nach diesen Anwendungen zu suchen und Details zu diesen Anwendungen anzuzeigen.

Herausgeber können Anwendungsberechtigungen mit den folgenden drei Kategorien festlegen:

- **Privat** — Anwendungen, die mit demselben Konto erstellt und mit keinem anderen Konto geteilt wurden. Nur Verbraucher, die Ihr AWS Konto gemeinsam nutzen, sind berechtigt, private Anwendungen bereitzustellen.
- **Privat geteilt** — Anwendungen, die der Herausgeber ausdrücklich mit einer bestimmten Gruppe von AWS Konten oder mit AWS Konten in einer AWS Organisation geteilt hat. Verbraucher sind berechtigt, Anwendungen bereitzustellen, die mit ihrem AWS Konto oder ihrer AWS Organisation geteilt wurden. Weitere Informationen zu AWS Organisationen finden Sie im [AWS Organizations Benutzerhandbuch](#).
- **Öffentlich geteilt** — Anwendungen, die der Herausgeber für alle freigegeben hat. Alle Verbraucher haben die Berechtigung, alle öffentlich freigegebenen Anwendungen bereitzustellen.

## Note

Für privat gemeinsam genutzte Anwendungen werden AWS Serverless Application Repository nur AWS Konten als Prinzipale unterstützt. Herausgeber können allen Benutzern innerhalb eines AWS Kontos eine einzige Gruppe für eine AWS Serverless Application Repository Anwendung zuweisen oder sie verweigern. Herausgeber können einer AWS Serverless Application Repository Anwendung nicht einzelne Benutzer innerhalb eines AWS Kontos gewähren oder verweigern.

Anweisungen zum Einstellen von Anwendungsberechtigungen mithilfe von finden Sie unter [Freigeben einer Anwendung](#). AWS Management Console

Anweisungen zum Einstellen von Anwendungsberechtigungen anhand der Beispiele AWS CLI und finden Sie in den folgenden Abschnitten.

## Anwendungsberechtigungen (AWS CLI und AWS SDKs)

Wenn Sie das AWS CLI oder das verwenden AWS SDKs , um Berechtigungen für eine AWS Serverless Application Repository Anwendung festzulegen, können Sie die folgenden Aktionen angeben:

Aktion	Beschreibung
GetApplication	Erteilt die Berechtigung zum Anzeigen von Informationen zur Anwendung.
CreateCloudFormationChangeSet	Erteilt die Berechtigung zum Bereitstellen der Anwendung. Hinweis: Diese Aktion gewährt keine anderen Berechtigungen außer der Bereitstellung.
CreateCloudFormationTemplate	Erteilt die Berechtigung, eine AWS CloudFormation Vorlage für die Anwendung zu erstellen.
ListApplicationVersions	Erteilt die Berechtigung zum Auflisten der Versionen der Anwendung.
ListApplicationDependencies	Erteilt die Berechtigung zum Auflisten der Listenanwendungen, die in der enthaltenen Anwendung verschachtelt sind.
SearchApplications	Erteilt die Berechtigung zum Suchen nach der Anwendung.
Bereitstellen	Mit dieser Aktion werden alle zuvor in der Tabelle aufgeführten Aktionen aktiviert. Das heißt, sie erteilt die Berechtigung für die Anzeige der Anwendung, für die Bereitstellung, für die Liste der Versionen und für die Suche.

## Beispiele für Anwendungsrichtlinien

In den folgenden Beispielen wird gezeigt, wie Berechtigungen mithilfe der AWS CLI erteilt werden. Informationen zum Erteilen von Berechtigungen mithilfe von finden Sie unter [Freigeben einer Anwendung](#). AWS Management Console

In allen Beispielen in diesem Abschnitt werden die folgenden AWS CLI Befehle verwendet, um die mit AWS Serverless Application Repository Anwendungen verknüpften Berechtigungsrichtlinien zu verwalten:

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

### Themen

- [Beispiel 1: Freigeben einer Anwendung für ein anderes Konto](#)
- [Beispiel 2: Freigeben einer Anwendung zur öffentlichen gemeinsamen Nutzung](#)
- [Beispiel 3: Einschränken einer Anwendung auf private Nutzung](#)
- [Beispiel 4: Angeben mehrerer Konten und Berechtigungen](#)
- [Beispiel 5: Teilen Sie eine Anwendung mit allen Konten in einer AWS Organisation](#)
- [Beispiel 6: Teilen einer Anwendung mit einigen Konten in einer Organisation AWS](#)
- [Beispiel 7: Abrufen einer Anwendungsrichtlinie](#)
- [Beispiel 8: Zulassen, dass die Anwendung nach bestimmten Konten verschachtelt wird](#)

### Beispiel 1: Freigeben einer Anwendung für ein anderes Konto

Um eine Anwendung für ein anderes bestimmtes Konto gemeinsam zu nutzen, sie aber nicht mit anderen zu teilen, geben Sie die AWS Konto-ID, mit der Sie sie teilen möchten, als Hauptbenutzer an. Dies wird auch als Festlegen der Anwendung auf privat freigegeben bezeichnet. Verwenden Sie dazu den folgenden AWS CLI Befehl.

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

**Note**

Privat gemeinsam genutzte Anwendungen können nur in derselben AWS Region verwendet werden, in der die Anwendung erstellt wurde.

**Beispiel 2: Freigeben einer Anwendung zur öffentlichen gemeinsamen Nutzung**

Um eine Anwendung öffentlich gemeinsam nutzbar zu machen, müssen Sie sie wie im folgenden Beispiel durch Angabe von "\*" als Prinzipal für jeden freigeben. Anwendungen, die öffentlich freigegeben werden, sind in allen Regionen verfügbar.

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

**Note**

Um eine Anwendung öffentlich freigeben zu können, müssen die Eigenschaften `SemanticVersion` und `LicenseUrl` festgelegt sein.

**Beispiel 3: Einschränken einer Anwendung auf private Nutzung**

Sie können eine Anwendung als privat kennzeichnen, sodass sie mit niemandem geteilt wird und nur von dem AWS Konto bereitgestellt werden kann, dem sie gehört. Dazu löschen Sie die Prinzipale und Aktionen aus der Richtlinie, wodurch auch Berechtigungen von anderen Konten innerhalb Ihrer AWS Organisation von der Bereitstellung Ihrer Anwendung ausgeschlossen werden.

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

**Note**

Private Anwendungen können nur in derselben AWS Region verwendet werden, in der die Anwendung erstellt wurde.

## Beispiel 4: Angeben mehrerer Konten und Berechtigungen

Sie können mehrere Berechtigungen gewähren, und Sie können sie mehreren AWS Konten gleichzeitig gewähren. Dazu geben Sie Listen als Prinzipal und Aktionen an, wie im folgenden Beispiel gezeigt.

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

## Beispiel 5: Teilen Sie eine Anwendung mit allen Konten in einer AWS Organisation

Berechtigungen können allen Benutzern innerhalb einer AWS Organisation erteilt werden. Dazu geben Sie Ihre Organisations-ID an, wie im folgenden Beispiel.

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

Weitere Informationen zu AWS Organisationen finden Sie im [AWS Organizations Benutzerhandbuch](#).

### Note

Sie können nur die AWS Organisation angeben, der Ihr AWS Konto angehört. Wenn Sie versuchen, eine AWS Organisation anzugeben, der Sie nicht angehören, wird ein Fehler angezeigt.

Um Ihre Anwendung mit Ihrer AWS Organisation zu teilen, müssen Sie die Genehmigung für die `UnshareApplication` Aktion angeben, falls die gemeinsame Nutzung in future aufgehoben werden muss.

## Beispiel 6: Teilen einer Anwendung mit einigen Konten in einer Organisation AWS

Berechtigungen können bestimmten Konten innerhalb einer AWS Organisation erteilt werden. Sie tun dies, indem Sie eine Liste von AWS Konten als Hauptbenutzer und Ihre Organisations-ID angeben, wie im folgenden Beispiel.

```
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

Sie können nur die AWS Organisation angeben, der Ihr AWS Konto angehört. Wenn Sie versuchen, eine AWS Organisation anzugeben, der Sie nicht angehören, wird ein Fehler angezeigt.

Um Ihre Anwendung mit Ihrer AWS Organisation zu teilen, müssen Sie die Genehmigung für die UnshareApplication Aktion angeben, falls die gemeinsame Nutzung in future aufgehoben werden muss.

## Beispiel 7: Abrufen einer Anwendungsrichtlinie

Um die aktuelle Richtlinie einer Anwendung anzuzeigen, etwa um nachzusehen, ob sie derzeit freigegeben ist, verwenden Sie den Befehl `get-application-policy`, wie im folgenden Beispiel gezeigt.

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

## Beispiel 8: Zulassen, dass die Anwendung nach bestimmten Konten verschachtelt wird

Öffentliche Anwendungen dürfen von jedem Benutzer verschachtelt werden. Wenn Sie möchten, dass Ihre Anwendung nur von bestimmten Konten verschachtelt werden darf, müssen Sie die folgenden minimalen Berechtigungen festlegen, wie im folgenden Beispiel gezeigt.

```
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 API-Berechtigungen: Referenz zu Aktionen und Ressourcen

Wenn Sie die [Zugriffskontrolle](#) einrichten und Berechtigungsrichtlinien für eine IAM-Identität (identitätsbasierte Richtlinie) verfassen, können Sie die folgende Tabelle als Referenz verwenden. Die jeden AWS Serverless Application Repository API-Vorgang, die entsprechenden Aktionen, denen Sie Berechtigungen zur Ausführung der Aktion erteilen können, und die AWS Ressource, der Sie die Berechtigungen erteilen können. Die Aktionen geben Sie im Feld `Action` und den Wert für die Ressource im Feld `Resource` der Richtlinie an.

Um eine Aktion anzugeben, verwenden Sie das Präfix `serverlessrepo:` gefolgt vom Namen der API-Operation (z. B. `serverlessrepo:ListApplications`).

Operation	URI	Methode	AWS Ressourcen (ARNs)
Betrieb: ListApplications  Erforderliche Berechtigungen: <code>serverlessrepo:ListApplications</code>	<code>/applications</code>	GET	*
Betrieb: CreateApplication  Erforderliche Berechtigungen: <code>serverlessrepo:CreateApplication</code>	<code>/applications</code>	POST	*
Betrieb: GetApplication  Erforderliche Berechtigungen:	<code>/Anwendungen/ <i>application-id</i></code>	GET	<code>arn:aws:serverlessrepo::applications/ <i>region account-id application-name</i></code>

Operation	URI	Methode	AWS Ressourcen (ARNs)
serverlessrepo: GetApplication			
Bedienung: DeleteApplication  Erforderliche Berechtigungen: serverlessrepo: DeleteApplication	/Anwendungen/ <i>application-id</i>	DELETE	arn:aws:serverless repo: ::applications/ <i>region account-id application-name</i>
Bedienung: UpdateApplication  Erforderliche Berechtigungen: serverlessrepo: UpdateApplication	/Anwendungen/ <i>application-id</i>	PATCH	arn:aws:serverless repo: ::applications/ <i>region account-id application-name</i>
Bedienung: CreateCloudFormationChangeSet	<i>application-id</i> /Anwendungen/ changesets	POST	arn:aws:serverless repo: :applikations/ <i>region account-id application-name</i>
Bedienung: GetApplicationPolicy  Erforderliche Berechtigungen: serverlessrepo: GetApplicationPolicy	<i>application-id</i> /Anwendungen/ Richtlinie	GET	arn:aws:serverless repo: ::applications/ <i>region account-id application-name</i>

Operation	URI	Methode	AWS Ressourcen (ARNs)
Bedienung: PutApplicationPolicy  Erforderliche Berechtigungen: serverlessrepo: PutApplicationPolicy	<i>application-id</i> /Anwendungen//Richtlinie	PUT	arn:aws:serverlessrepo::applications/ <i>region</i> <i>account-id</i> <i>application-name</i>
Bedienung: ListApplicationVersions  Erforderliche Berechtigungen: serverlessrepo: ListApplicationVersions	<i>application-id</i> /Anwendungen//Versionen	GET	arn:aws:serverlessrepo::applications/ <i>region</i> <i>account-id</i> <i>application-name</i>
Bedienung: CreateApplicationVersion  Erforderliche Berechtigungen: serverlessrepo: CreateApplicationVersion	<i>application-id</i> /Anwendungen//Versionen/ <i>semantic-version</i>	PUT	arn:aws:serverlessrepo::applikations/ <i>region</i> <i>account-id</i> <i>application-name</i>
Bedienung: ListApplicationDependencies  Erforderliche Berechtigungen: serverlessrepo: ListApplicationDependencies	<i>application-id</i> /Anwendungen//Abhängigkeiten	GET	arn:aws:serverlessrepo::applications/ <i>region</i> <i>account-id</i> <i>application-name</i>

Operation	URI	Methode	AWS Ressourcen (ARNs)
Bedienung: SearchApplications	–	–	*
Erforderliche Berechtigungen: serverlessrepo: SearchApplications			

## Problembehandlung bei AWS Serverless Application Repository Identität und Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit dem AWS Serverless Application Repository und IAM auftreten können.

### Themen

- [Ich bin nicht autorisiert, eine Aktion in AWS Serverless Application Repository auszuführen](#)
- [Ich bin nicht berechtigt, IAM auszuführen: PassRole](#)
- [Ich bin Administrator und möchte anderen den Zugriff auf das ermöglichen AWS Serverless Application Repository](#)
- [Ich möchte Personen außerhalb meines Kontos den Zugriff auf meine AWS Ressourcen ermöglichen AWS Serverless Application Repository](#)

### Ich bin nicht autorisiert, eine Aktion in AWS Serverless Application Repository auszuführen

Wenn Ihnen AWS Management Console mitgeteilt wird, dass Sie nicht berechtigt sind, eine Aktion auszuführen, müssen Sie sich an Ihren Administrator wenden, um Unterstützung zu erhalten. Ihr Administrator ist die Person, die Ihnen Ihren Benutzernamen und Ihr Passwort bereitgestellt hat.

Der folgende Beispielfehler tritt auf, wenn der `mateojackson` IAM-Benutzer versucht, die Konsole zu verwenden, um Details zu einer Anwendung anzuzeigen, aber nicht über die `serverlessrepo:GetApplication` entsprechenden Berechtigungen verfügt.

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

In diesem Fall bittet Mateo seinen Administrator, seine Richtlinien zu aktualisieren, um ihm den Zugriff auf die *my-example-application*-Ressource mit der *serverlessrepo:GetApplication*-Operation zu ermöglichen.

## Ich bin nicht berechtigt, IAM auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an AWS Serverless Application Repository übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in AWS Serverless Application Repository auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

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

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

## Ich bin Administrator und möchte anderen den Zugriff auf das ermöglichen AWS Serverless Application Repository

Um anderen den Zugriff zu ermöglichen AWS Serverless Application Repository, müssen Sie den Personen oder Anwendungen, die Zugriff benötigen, die entsprechenden Berechtigungen erteilen. Wenn Sie Personen und Anwendungen verwalten, weisen Sie Benutzern oder Gruppen Berechtigungssätze zu, um deren Zugriffsebene zu definieren. AWS IAM Identity Center Mit

Berechtigungssätzen werden automatisch IAM-Richtlinien erstellt und den IAM-Rollen zugewiesen, die der Person oder Anwendung zugeordnet sind. Weitere Informationen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter [Berechtigungssätze](#).

Wenn Sie IAM Identity Center nicht verwenden, müssen Sie IAM-Entitäten (Benutzer oder Rollen) für die Personen oder Anwendungen erstellen, die Zugriff benötigen. Anschließend müssen Sie der Entität eine Richtlinie anfügen, die dieser die korrekten Berechtigungen in AWS Serverless Application Repository gewährt. Nachdem die Berechtigungen erteilt wurden, stellen Sie dem Benutzer oder Anwendungsentwickler die Anmeldeinformationen zur Verfügung. Sie werden diese Anmeldeinformationen für den Zugriff verwenden AWS. Weitere Informationen zum Erstellen von IAM-Benutzern, -Gruppen, -Richtlinien und -Berechtigungen finden Sie im [IAM-Benutzerhandbuch unter IAM-Identitäten sowie Richtlinien und Berechtigungen in IAM](#).

## Ich möchte Personen außerhalb meines Kontos den Zugriff auf meine AWS Ressourcen ermöglichen AWS Serverless Application Repository

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen AWS Serverless Application Repository unterstützt werden, finden Sie unter [So AWS Serverless Application Repository funktioniert das mit IAM](#)
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter [Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen](#).
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie [AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte](#).
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter [Gewähren von Zugriff für extern authentifizierte Benutzer \(Identitätsverbund\)](#) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

# Protokollierung und Überwachung im AWS Serverless Application Repository

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung Ihrer AWS Lösungen. Sie sollten Überwachungsdaten aus allen Teilen Ihrer AWS Lösung sammeln, damit Sie einen etwaigen Ausfall an mehreren Punkten leichter debuggen können. AWS bietet verschiedene Tools zur Überwachung Ihrer AWS Serverless Application Repository Ressourcen und zur Reaktion auf potenzielle Vorfälle, z. B. die folgenden:

## AWS CloudTrail Logs

Der AWS Serverless Application Repository ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst in der ausgeführt wurden AWS Serverless Application Repository. CloudTrail erfasst alle API-Aufrufe für die AWS Serverless Application Repository AS-Ereignisse.

## Themen

- [AWS Serverless Application Repository API-Aufrufe protokollieren mit AWS CloudTrail](#)

## AWS Serverless Application Repository API-Aufrufe protokollieren mit AWS CloudTrail

AWS Serverless Application Repository ist in integriert. AWS CloudTrail Dabei handelt es sich um einen Dienst, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst in der ausgeführt wurden AWS Serverless Application Repository. CloudTrail erfasst alle API-Aufrufe für die AWS Serverless Application Repository AS-Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der AWS Serverless Application Repository Konsole und Codeaufrufen für die AWS Serverless Application Repository API-Operationen.

Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für den AWS Serverless Application Repository. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen.

Anhand der von gesammelten Informationen können Sie feststellen CloudTrail, welche Anfrage an den gestellt wurde AWS Serverless Application Repository. Sie können auch die IP-Adresse,

von der die Anforderung ausging, den Ersteller und den Erstellungszeitpunkt sowie weitere Details bestimmen.

Weitere Informationen CloudTrail dazu finden Sie im [AWS CloudTrail Benutzerhandbuch](#).

## AWS Serverless Application Repository Informationen in CloudTrail

CloudTrail ist in Ihrem AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn eine Aktivität in der stattfindet AWS Serverless Application Repository, wird diese Aktivität zusammen mit anderen CloudTrail AWS Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto ansehen, suchen und herunterladen. Weitere Informationen finden Sie unter [Ereignisse mit CloudTrail Ereignisverlauf anzeigen](#).

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich der Ereignisse für den AWS Serverless Application Repository, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole erstellen, gilt der Trail standardmäßig für alle AWS Regionen. Der Trail protokolliert Ereignisse aus allen AWS Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie hier:

- [Übersicht zum Erstellen eines Trails](#)
- [CloudTrail Unterstützte Dienste und Integrationen](#)
- [Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail](#)
- [Empfangen von CloudTrail Protokolldateien aus mehreren Regionen](#) und [Empfangen von CloudTrail Protokolldateien von mehreren Konten](#)

Alle AWS Serverless Application Repository Aktionen werden von der Seite [AWS Serverless Application Repository Ressourcen](#) protokolliert CloudTrail und dort dokumentiert. Beispielsweise generieren Aufrufe der ListApplications Operationen CreateApplicationUpdateApplications, und Einträge in den CloudTrail Protokolldateien.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Root- oder AWS Identity and Access Management (IAM-) Benutzeranmeldedaten gestellt wurde.

- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Weitere Informationen finden Sie unter [CloudTrail userIdentity-Element](#).

## Grundlegendes zu AWS Serverless Application Repository Protokolldateieinträgen

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `CreateApplication` Aktion demonstriert.

```
{
  "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"
}
```

## Konformitätsvalidierung für die AWS Serverless Application Repository

Externe Prüfer bewerten die Sicherheit und Konformität von im AWS Serverless Application Repository Rahmen mehrerer AWS Compliance-Programme. Hierzu zählen unter anderem SOC, PCI, FedRAMP und andere.

Eine Liste der AWS Services, die in den Geltungsbereich bestimmter Compliance-Programme fallen, finden Sie unter [AWS Services im Umfang nach Compliance-Programmen](#). Allgemeine Informationen finden Sie unter [AWS -Compliance-Programme](#).

Sie können Prüfberichte von Drittanbietern herunterladen, indem Sie AWS Artifact Weitere Informationen finden Sie unter [Berichte in AWS Artifact herunterladen](#).

Ihre Verantwortung für die Einhaltung der Vorschriften bei der AWS Serverless Application Repository Nutzung von hängt von der Sensibilität Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- [Schnellstartanleitungen zu Sicherheit und Compliance](#) — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte zur Implementierung sicherheits- und Compliance-orientierter Basisumgebungen beschrieben. AWS
- [AWS Ressourcen zur Einhaltung](#) von Vorschriften — Diese Sammlung von Arbeitsmappen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.
- [AWS Config](#) — Dieser AWS Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- [AWS Security Hub](#) — Dieser AWS Service bietet einen umfassenden Überblick über Ihren Sicherheitsstatus und hilft Ihnen AWS, die Einhaltung der Sicherheitsstandards und bewährten Verfahren der Sicherheitsbranche zu überprüfen.

# Resilienz in der AWS Serverless Application Repository

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Availability Zones ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser hoch verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu AWS Regionen und Availability Zones finden Sie unter [AWS Globale Infrastruktur](#).

## Infrastruktursicherheit in der AWS Serverless Application Repository

Als verwalteter Dienst AWS Serverless Application Repository ist er durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter [AWS Cloud-Sicherheit](#). Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff AWS Serverless Application Repository über das Netzwerk. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit [AWS Security Token Service](#) (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

## Zugriff AWS Serverless Application Repository über einen Schnittstellenendpunkt (AWS PrivateLink)

Sie können AWS PrivateLink damit eine private Verbindung zwischen Ihrer VPC und AWS Serverless Application Repository herstellen. Sie können darauf zugreifen, AWS Serverless Application Repository als ob es in Ihrer VPC wäre, ohne ein Internet-Gateway, ein NAT-Gerät, eine VPN-Verbindung oder AWS Direct Connect eine Verbindung zu verwenden. Instances in Ihrer VPC benötigen für den Zugriff AWS Serverless Application Repository keine öffentlichen IP-Adressen.

Sie stellen diese private Verbindung her, indem Sie einen Schnittstellen-Endpunkt erstellen, der von AWS PrivateLink unterstützt wird. Wir erstellen eine Endpunkt-Netzwerkschnittstelle in jedem Subnetz, das Sie für den Schnittstellen-Endpunkt aktivieren. Hierbei handelt es sich um vom Anforderer verwaltete Netzwerkschnittstellen, die als Eingangspunkt für den Datenverkehr dienen, der für AWS Serverless Application Repository bestimmt ist.

Weitere Informationen finden Sie unter [Zugriff auf AWS-Services über AWS PrivateLink](#) im AWS PrivateLink -Leitfaden.

## Überlegungen zu AWS Serverless Application Repository

Bevor Sie einen Schnittstellen-Endpunkt für einrichten AWS Serverless Application Repository, lesen Sie die [Überlegungen](#) im AWS PrivateLink Handbuch.

AWS Serverless Application Repository unterstützt Aufrufe aller API-Aktionen über den Schnittstellenendpunkt.

## Erstellen Sie einen Schnittstellenendpunkt für AWS Serverless Application Repository

Sie können einen Schnittstellenendpunkt für die AWS Serverless Application Repository Verwendung entweder der Amazon VPC-Konsole oder der AWS Command Line Interface (AWS CLI) erstellen. Weitere Informationen finden Sie unter [Erstellen eines Schnittstellenendpunkts](#) im AWS PrivateLink -Leitfaden.

Erstellen Sie einen Schnittstellenendpunkt für die AWS Serverless Application Repository Verwendung des folgenden Servicenamens:

```
com.amazonaws.region.serverlessrepo
```

Wenn Sie `privates DNS` für den Schnittstellenendpunkt aktivieren, können Sie API-Anfragen an die AWS Serverless Application Repository Verwendung des standardmäßigen regionalen DNS-Namens stellen. Beispiel, `serverlessrepo.us-east-1.amazonaws.com`.

## Erstellen einer Endpunktrichtlinie für Ihren Schnittstellen-Endpunkt

Eine Endpunktrichtlinie ist eine IAM-Ressource, die Sie an einen Schnittstellen-Endpunkt anfügen können. Die standardmäßige Endpunktrichtlinie ermöglicht den vollen Zugriff AWS Serverless Application Repository über den Schnittstellenendpunkt. Um den Zugriff AWS Serverless Application Repository von Ihrer VPC aus zu kontrollieren, fügen Sie dem Schnittstellenendpunkt eine benutzerdefinierte Endpunktrichtlinie hinzu.

Eine Endpunktrichtlinie gibt die folgenden Informationen an:

- Die Prinzipale, die Aktionen ausführen können (AWS-Konten, IAM-Benutzer und IAM-Rollen).
- Aktionen, die ausgeführt werden können
- Die Ressourcen, auf denen die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie unter [Steuern des Zugriffs auf Services mit Endpunktrichtlinien](#) im AWS PrivateLink -Leitfaden.

Beispiel: VPC-Endpunktrichtlinie für Aktionen AWS Serverless Application Repository

Im Folgenden finden Sie ein Beispiel für eine benutzerdefinierte Endpunktrichtlinie. Wenn Sie diese Richtlinie an Ihren Schnittstellenendpunkt anhängen, gewährt sie allen Prinzipalen auf allen Ressourcen Zugriff auf die aufgelistete AWS Serverless Application Repository Aktion. Das folgende Beispiel gewährt allen Benutzern die Berechtigung, Anwendungen über den VPC-Endpunkt zu erstellen.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "serverlessrepo:CreateApplication"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

# AWS Serverless Application Repository Kontingente

Das AWS Serverless Application Repository hat ein Kontingent für die Anzahl der öffentlichen Anwendungen, die ein AWS Konto in jeder AWS Region haben kann. Dieses Kontingent gilt pro Region und kann erhöht werden. Um eine Erhöhung anzufordern, verwenden Sie die [Support-Center-Konsole](#).

Ressource	Standardkontingent
Öffentliche Anwendungen (pro AWS Konto pro AWS Region)	100

Die folgenden Kontingente gelten für Speicher, der für Codepakete und Anwendungsrichtlinien verfügbar ist. Sie können diese Kontingente nicht ändern.

Ressource	Kontingent
Kostenloser Amazon S3 S3-Speicherplatz für Code-Pakete (pro AWS Konto pro AWS Region)	5 GB
Länge der Anwendungsrichtlinie	6.144 Zeichen

# Problembehandlung bei AWS Serverless Application Repository

Wenn Sie die verwenden AWS Serverless Application Repository, können Probleme auftreten, wenn Sie Ihre Anwendungen erstellen, aktualisieren oder löschen. In diesem Abschnitt finden Sie Hilfe zur Behebung typischer Probleme. Sie können auch in den [AWS Serverless Application Repository - Foren](#) nach Antworten suchen und Fragen posten.

## Note

Anwendungen in der AWS Serverless Application Repository werden mithilfe von bereitgestellt AWS CloudFormation. Informationen zur AWS CloudFormation Problembhebung finden Sie im [Leitfaden zur AWS CloudFormation Problembehandlung](#).

## Themen

- [Sie können eine Anwendung nicht veröffentlichen](#)
- [Ein Kontingent wurde überschritten](#)
- [Eine aktualisierte Readme-Datei erscheint nicht sofort](#)
- [Sie können eine Anwendung infolge von unzureichenden IAM-Berechtigungen nicht bereitstellen](#)
- [Sie können eine Anwendung nicht zweimal bereitstellen](#)
- [Warum ist meine Anwendung nicht öffentlich verfügbar](#)
- [Den -Support kontaktieren](#)

## Sie können eine Anwendung nicht veröffentlichen

Wenn Sie Ihre Anwendung nicht veröffentlichen können, fehlt möglicherweise eine von der Open Source Initiative (OSI) genehmigte Lizenzdatei für Ihre Anwendung.

Zur Veröffentlichung Ihrer Anwendung benötigen Sie eine von OSI genehmigte Lizenzdatei und eine erfolgreich veröffentlichte Version der Anwendung mit einer Quellcode-URL für die Version. Sie können die Lizenz einer Anwendung nach der Erstellung der Anwendung nicht mehr aktualisieren.

Wenn Sie Ihre Anwendung wegen einer fehlenden Lizenzdatei nicht veröffentlichen können, löschen Sie die Anwendung und erstellen Sie eine neue mit dem gleichen Namen. Stellen Sie sicher, dass Sie

mindestens eine Open-Source-Lizenz dafür bereitstellen, die von der Open Source Initiative (OSI)-Organisation genehmigt wurde.

## Ein Kontingent wurde überschritten

Wenn eine Fehlermeldung angezeigt wird, die angibt, dass ein Kontingent überschritten wurde, überprüfen Sie, ob Sie ein Ressourcenkontingent erreicht haben. Informationen zu AWS Serverless Application Repository Kontingenten finden Sie unter [AWS Serverless Application Repository Kontingente](#).

## Eine aktualisierte Readme-Datei erscheint nicht sofort

Bei Veröffentlichung Ihrer Anwendung kann die Aktualisierung der Inhalte Ihrer Anwendung bis zu 24 Stunden dauern. Wenn es länger als 24 Stunden zu Verzögerungen kommt, wenden Sie sich an den AWS Support, um Hilfe zu erhalten. Details dazu finden im folgenden Abschnitt.

## Sie können eine Anwendung infolge von unzureichenden IAM-Berechtigungen nicht bereitstellen

Um eine AWS Serverless Application Repository Anwendung bereitzustellen, benötigen Sie Berechtigungen für AWS Serverless Application Repository Ressourcen und AWS CloudFormation Stacks. Möglicherweise benötigen Sie auch eine Berechtigung für die Verwendung der zugrunde liegenden Services in der Anwendung. Wenn Sie beispielsweise einen Amazon S3-Bucket oder eine Amazon DynamoDB-Tabelle erstellen, benötigen Sie Berechtigungen für Amazon S3 oder DynamoDB.

Wenn Sie auf ein solches Problem stoßen, überprüfen Sie Ihre AWS Identity and Access Management (IAM-) Richtlinie und stellen Sie sicher, dass Sie über die erforderlichen Berechtigungen verfügen. Weitere Informationen finden Sie unter [Zugriffskontrolle mit AWS Identity and Access Management](#).

## Sie können eine Anwendung nicht zweimal bereitstellen

Der von Ihnen angegebene Anwendungsname wird als Name des AWS CloudFormation Stacks verwendet. Wenn Sie Probleme bei der Bereitstellung einer Anwendung haben, stellen Sie sicher, dass Sie nicht über einen vorhandenen AWS CloudFormation Stack mit demselben Namen verfügen.

Falls doch, geben Sie einen anderen Anwendungsnamen an oder löschen Sie den vorhandenen Stack, um die Anwendung erneut mit dem gleichen Namen bereitzustellen.

## Warum ist meine Anwendung nicht öffentlich verfügbar

Standardmäßig werden Anwendungen als privat eingestuft. Führen Sie [diese](#) Schritte aus, um Ihre Anwendung öffentlich zu machen.

## Den -Support kontaktieren

In einigen Fällen können Sie möglicherweise in diesem Abschnitt oder über die [AWS Serverless Application Repository -Foren](#) keine Lösungen zur Fehlerbehebung finden. Wenn Sie AWS Premium-Support haben, können Sie unter Support einen Fall für technischen [AWS Support](#) erstellen.

Bevor Sie sich an den AWS Support wenden, stellen Sie sicher, dass Sie den Amazon-Ressourcennamen (ARN) für die Anwendung erhalten haben, zu der Sie Fragen haben. Sie finden den Anwendungs-ARN in der [AWS Serverless Application Repository -Konsole](#).

# Operationen

Die REST-API des AWS Serverless Application Repository umfasst die folgenden Operationen.

- [CreateApplication](#)

Erstellt eine Anwendung, optional einschließlich einer AWS SAM-Datei, um die erste Anwendungsversion im selben Aufruf zu erstellen.

- [CreateApplicationVersion](#)

Erstellt eine Anwendungsversion

- [CreateCloudFormationChangeSet](#)

Erstellt einen AWS CloudFormation Änderungssatz für die angegebene Anwendung.

- [CreateCloudFormationTemplate](#)

Erstellt eine AWS CloudFormation Vorlage.

- [DeleteApplication](#)

Löscht die angegebene Anwendung.

- [GetApplication](#)

Ruft die angegebene Anwendung ab

- [GetApplicationPolicy](#)

Ruft die Richtlinie für die Anwendung ab.

- [GetCloudFormationTemplate](#)

Ruft die angegebene AWS CloudFormation Vorlage ab.

- [ListApplicationDependencies](#)

Ruft die Liste der Anwendungen ab, die in der enthaltenden Anwendung verschachtelt sind.

- [ListApplications](#)

Listet Anwendungen im Besitz des Auftraggebers auf

- [ListApplicationVersions](#)

Listet die Versionen für die angegebene Anwendung auf.

- [PutApplicationPolicy](#)

Legt die Berechtigungsrichtlinie für eine Anwendung fest. Eine Liste der für diesen Vorgang unterstützten Aktionen finden Sie unter [Anwendungsberechtigungen](#).

- [UnshareApplication](#)

Macht die Freigabe einer Anwendung von einer AWS Organisation rückgängig.

Dieser Vorgang kann nur über das Verwaltungskonto der Organisation aufgerufen werden.

- [UpdateApplication](#)

Aktualisiert die angegebene Anwendung.

# Ressourcen

Die AWS Serverless Application Repository REST-API umfasst die folgenden Ressourcen.

## Themen

- [Applications](#)
- [applicationId](#)
- [Anwendungs-ID-Änderungssätze](#)
- [Applications applicationId Dependencies](#)
- [ApplicationID-Richtlinie für Anwendungen](#)
- [Applications applicationId Templates](#)
- [Applications applicationId Templates templateId](#)
- [Applications applicationId Unshare](#)
- [Anwendungs-ID-Versionen](#)
- [Anwendungen ApplicationID-Versionen SemanticVersion](#)

## Applications

### URI

/applications

### HTTP-Methoden

#### GET

Vorgangs-ID: ListApplications

Listet Anwendungen im Besitz des Auftraggebers auf

## Abfrageparameter

Name	Typ	Erforderlich	Beschreibung
maxItems	String	False	Die Gesamtzahl der zurückzugebenden Elemente.
nextToken	String	False	Ein Token für den Beginn der Seitennummerierung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">ApplicationPage</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## POST

### Vorgangs-ID: CreateApplication

Erstellt eine Anwendung, optional mit einer AWS SAM-Datei, damit die erste Anwendungsversion im selben Aufruf erstellt wird

## Antworten

Statuscode	Antwortmodell	Beschreibung
201	<a href="#">Application</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
409	<a href="#">ConflictException</a>	Die Ressource ist bereits vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Anfragetexte

#### POST-Schema

```
{
  "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"
}
```

## Antworttexte

### ApplicationPage-Schema

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

### Application-Schema

```
{
```

```
"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-Schema

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

### ForbiddenException-Schema

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

### NotFoundException-Schema

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

### ConflictException-Schema

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

### TooManyRequestsException-Schema

```
{
```

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### Application

Details zum Antrag.

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

name

Der Name der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Pattern: "[a-zA-Z0-9\-\-]+";

Typ: Zeichenfolge

Erforderlich: True

description

Die Beschreibung der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Typ: Zeichenfolge

Erforderlich: True

author

Der Name des Anwendungsgebers.

Minimale Länge = 1. Maximale Länge = 127.

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

Typ: Zeichenfolge

Erforderlich: True

isVerifiedAuthor

Gibt an, ob der Autor dieser Anwendung verifiziert wurde. Das bedeutet, AWS dass das Unternehmen als vernünftiger und umsichtiger Dienstleister die vom Antragsteller bereitgestellten Informationen nach bestem Wissen und Gewissen geprüft und bestätigt hat, dass die Identität des Antragstellers den Angaben entspricht.

Typ: Boolesch

Erforderlich: Falsch

verifiedAuthorUrl

Die URL zum öffentlichen Profil eines verifizierten Autors. Diese URL wurde vom Autor eingereicht.

Typ: Zeichenfolge

Erforderlich: Falsch

spdxLicenseId

Ein gültiger Bezeichner von <https://spdx.org/licenses/>.

Typ: Zeichenfolge

Erforderlich: Falsch

## licenseUrl

Ein Link zu einer Lizenzdatei der App, die dem sdxLicenseID-Wert Ihrer Anwendung entspricht.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## readmeUrl

Ein Link zur Readme-Datei in Markdown-Sprache, die eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## labels

Labels zur besseren Auffindbarkeit von Apps in Suchergebnissen.

Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl von Beschriftungen: 10

Pattern: "`^[a-zA-Z0-9+\\-_:\\V@]+`";

Typ: Array des Typs string

Erforderlich: Falsch

## creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: Falsch

## homePageUrl

Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.

Typ: Zeichenfolge

Erforderlich: Falsch

version

Versionsinformationen über die Anwendung.

Typ: [Version](#)

Erforderlich: Falsch

## ApplicationPage

Eine Liste mit Anwendungsdetails.

applications

Eine Reihe von Anwendungszusammenfassungen.

Typ: Array des Typs [ApplicationSummary](#)

Erforderlich: True

nextToken

Das Token, um die nächste Ergebnisseite anzufordern.

Typ: Zeichenfolge

Erforderlich: Falsch

## ApplicationSummary

Zusammenfassung der Einzelheiten zur Anwendung.

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

## name

Der Name der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Pattern: "[a-zA-Z0-9\\-]+";

Typ: Zeichenfolge

Erforderlich: True

## description

Die Beschreibung der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Typ: Zeichenfolge

Erforderlich: True

## author

Der Name des Anwendungsgebers.

Minimale Länge = 1. Maximale Länge = 127.

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

Typ: Zeichenfolge

Erforderlich: True

## spdxLicenseld

Ein gültiger Bezeichner von <https://spdx.org/licenses/>.

Typ: Zeichenfolge

Erforderlich: Falsch

## labels

Labels zur besseren Auffindbarkeit von Apps in Suchergebnissen.

Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl von Beschriftungen: 10

Pattern: "`^[a-zA-Z0-9+\\-_:\\@]+`";

Typ: Array des Typs string

Erforderlich: Falsch

creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: Falsch

homePageUrl

Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.

Typ: Zeichenfolge

Erforderlich: Falsch

## BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

## Capability

Werte, die angegeben werden müssen, um einige Anwendungen bereitzustellen.

CAPABILITY\_IAM  
CAPABILITY\_NAMED\_IAM  
CAPABILITY\_AUTO\_EXPAND  
CAPABILITY\_RESOURCE\_POLICY

## ConflictException

Die Ressource ist bereits vorhanden.

message

Die Ressource ist bereits vorhanden.

Typ: Zeichenfolge  
Erforderlich: Falsch

errorCode

409

Typ: Zeichenfolge  
Erforderlich: Falsch

## CreateApplicationInput

Erstellen einer Anwendungsanfrage.

name

Der Name der Anwendung an, die Sie veröffentlichen möchten.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Pattern: "[a-zA-Z0-9\-\-]+";

Typ: Zeichenfolge

Erforderlich: True

## description

Die Beschreibung der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Typ: Zeichenfolge

Erforderlich: True

## author

Der Name des Anwendungsgebers.

Minimale Länge = 1. Maximale Länge = 127.

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

Typ: Zeichenfolge

Erforderlich: True

## spdxLicenseId

Ein gültiger Bezeichner von <https://spdx.org/licenses/>.

Typ: Zeichenfolge

Erforderlich: Falsch

## licenseBody

Eine lokale Textdatei, die die Lizenz der App enthält, die dem `spdxLicenseId`-Wert Ihrer Anwendung entspricht. Die -Datei hat Format `file://<path>/<filename>`:

Maximale Größe 5 MB

Sie können nur einen Wert von `licenseBody` und angeben. `licenseUrl` Andernfalls tritt ein Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

### licenseUrl

Ein Link zum S3-Objekt, das die Lizenz der App enthält, die dem `spdxLicenseId`-Wert Ihrer Anwendung entspricht.

Maximale Größe 5 MB

Sie können nur einen Wert von `licenseBody` und angeben. `licenseUrl` Andernfalls tritt ein Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

### readmeBody

Eine lokale Text-Readme-Datei in Markdown-Sprache, die eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält. Die -Datei hat Format `file:///path/<filename>`:

Maximale Größe 5 MB

Sie können nur einen Wert von `readmeBody` und angeben. `readmeUrl` Andernfalls tritt ein Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

### readmeUrl

Ein Link zum S3-Objekt in Markdown-Sprache, der eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält.

Maximale Größe 5 MB

Sie können nur einen Wert von `readmeBody` und angeben. `readmeUrl` Andernfalls tritt ein Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## labels

Labels zur besseren Auffindbarkeit von Apps in Suchergebnissen.

Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl von Beschriftungen: 10

Pattern: "`^[a-zA-Z0-9+\\-_:\\@]+`";

Typ: Array des Typs string

Erforderlich: Falsch

## homePageUrl

Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.

Typ: Zeichenfolge

Erforderlich: Falsch

## semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: Falsch

## templateBody

Die lokale rohe, verpackte AWS SAM Vorlagendatei Ihrer Anwendung. Die -Datei hat Format `file://<path>/<filename>`:

Sie können nur eines von `templateBody` und `templateUrl` angeben, `templateUrl` andernfalls wird ein Fehler angezeigt.

Typ: Zeichenfolge

Erforderlich: Falsch

### templateUrl

Ein Link zum S3-Objekt, das die verpackte AWS SAM Vorlage Ihrer Anwendung enthält.

Sie können nur eines von `templateBody` und `templateUrl` angeben. Andernfalls wird ein Fehler angezeigt.

Typ: Zeichenfolge

Erforderlich: Falsch

### sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

### sourceCodeArchiveUrl

Ein Link zum S3-Objekt, das das ZIP-Archiv des Quellcodes für diese Version Ihrer Anwendung enthält.

Maximale Größe 50 MB

Typ: Zeichenfolge

Erforderlich: Falsch

### ForbiddenException

Der Client ist nicht authentifiziert.

### message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## ParameterDefinition

Von der Anwendung unterstützte Parameter.

name

Der Name des Parameters.

Typ: Zeichenfolge

Erforderlich: True

defaultValue

Gibt einen Wert des entsprechenden Typs an, den die Vorlage nutzen soll, wenn zum Zeitpunkt der Stack-Erstellung kein Wert angegeben wird. Wenn Sie Einschränkungen für den Parameter definieren, müssen Sie einen Wert angeben, der diese Einschränkungen einhält.

Typ: Zeichenfolge

Erforderlich: Falsch

description

Gibt eine Zeichenfolge mit maximal 4 000 Zeichen an, die den Parameter beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

type

Der Parametertyp.

Zulässige Werte: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

**String:** Gibt eine Literalzeichenfolge an.

Benutzer können beispielsweise angeben `"MyUserName"`.

**Number:** Eine Ganzzahl oder eine Fließkommazahl. AWS CloudFormation validiert den Parameterwert als Zahl. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die Ref intrinsische Funktion), wird der Parameterwert eine Zeichenfolge.

Benutzer könnten beispielsweise angeben `"8888"`

**List<Number>:** Gibt ein Array von Ganzzahlwert oder Ganzzahlwert an, jeweils durch Komma getrennt. AWS CloudFormation validiert den Parameterwert als Zahlen. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die Ref intrinsische Funktion), wird der Parameterwert zu einer Liste von Zeichenfolgen.

Benutzer könnten beispielsweise `„80,20“` angeben und dann das Ergebnis anzeigen. Ref `["80", "20"]`

**CommaDelimitedList:** Gibt ein Array von Literalzeichenfolgen an, jeweils durch Komma getrennt. Die Gesamtanzahl von Zeichenfolgen sollte um 1 höher sein als die Gesamtanzahl von Kommas. Jede Mitgliedszeichenfolge wird zudem gekürzt.

Benutzer könnten beispielsweise `„test, dev, prod“` angeben und dann die Ergebnisse anzeigen. Ref `["test", "dev", "prod"]`

Typ: Zeichenfolge

Erforderlich: Falsch

**noEcho**

Ob der Parameterwert maskiert werden soll, wenn jemand einen Aufruf tätigt, der den Stack beschreibt. Wenn Sie den Wert zuweisen, wird der Parameterwert mit Sternchen (`*****`) maskiert.

Typ: Boolesch

Erforderlich: Falsch

**allowedPattern**

Gibt einen regulären Ausdruck an, der die zulässigen Muster für `String`-Typen beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

### constraintDescription

Gibt eine Zeichenfolge als Beschreibung für eine Einschränkung an. Diese Beschreibung wird bei Verstößen gegen die Einschränkung angezeigt. Für einen Parameter mit einem zulässigen Muster `[A-Za-z0-9]+` beispielsweise wird ohne Einschränkungsbeschreibung die folgende Fehlermeldung angezeigt, wenn der Benutzer einen ungültigen Wert angibt:

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

Durch das Hinzufügen der Beschreibung einer Einschränkung, wie z. B. dürfen nur Groß- und Kleinbuchstaben und Zahlen enthalten, können Sie die folgende benutzerdefinierte Fehlermeldung anzeigen:

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

Typ: Zeichenfolge

Erforderlich: Falsch

### minValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Mindestwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

### maxValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Maximalwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

## minLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

## maxLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

## allowedValues

Gibt ein Array mit einer Liste der für den Parameter zulässigen Werte an.

Typ: Array des Typs `string`

Erforderlich: Falsch

## referencedByResources

Eine Liste von AWS SAM Ressourcen, die diesen Parameter verwenden.

Typ: Array des Typs `string`

Erforderlich: `True`

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

## message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Version

Details zur Anwendungsversion.

### applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

### semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

### sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

### sourceCodeArchiveUrl

Ein Link zum S3-Objekt, das das ZIP-Archiv des Quellcodes für diese Version Ihrer Anwendung enthält.

Maximale Größe 50 MB

Typ: Zeichenfolge

Erforderlich: Falsch

### templateUrl

Ein Link zur verpackten AWS SAM Vorlage Ihrer Anwendung.

Typ: Zeichenfolge

Erforderlich: True

### creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

### parameterDefinitions

Eine Reihe von Parametertypen, die von der Anwendung unterstützt werden.

Typ: Array des Typs [ParameterDefinition](#)

Erforderlich: True

### requiredCapabilities

Eine Liste der Werte, die Sie angeben müssen, bevor Sie bestimmte Anwendungen bereitstellen können. Einige Anwendungen enthalten möglicherweise Ressourcen, die die Berechtigungen in Ihrem AWS Konto beeinflussen, beispielsweise, wenn neue AWS Identity and Access Management (IAM) -Benutzer erstellt werden. Für diese Anwendungen müssen Sie ihre Funktionen ausdrücklich bestätigen, indem Sie diesen Parameter angeben.

Die einzig gültigen Werte sind

CAPABILITY\_IAM, CAPABILITY\_NAMED\_IAM, CAPABILITY\_RESOURCE\_POLICY,  
und CAPABILITY\_AUTO\_EXPAND.

Für die folgenden Ressourcen müssen Sie CAPABILITY\_IAM oder

angeben CAPABILITY\_NAMED\_IAM: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#),

und [AWS::IAM::Role](#). Wenn die Anwendung IAM-Ressourcen enthält, können Sie entweder `CAPABILITY_IAM` oder `CAPABILITY_NAMED_IAM` angeben. Beinhaltet die Anwendung IAM-Ressourcen mit benutzerdefinierten Namen, müssen Sie `CAPABILITY_NAMED_IAM` angeben.

Für die folgenden Ressourcen müssen Sie Folgendes angeben `CAPABILITY_RESOURCE_POLICY`: [AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#), [AWS::S3::BucketPolicy](#) und [AWS::SQS::QueuePolicy](#) [AWS::SNS::TopicPolicy](#)

Bei Anwendungen, die eine oder mehrere verschachtelte Anwendungen enthalten, müssen Sie `CAPABILITY_AUTO_EXPAND` angeben.

Wenn Ihre Anwendungsvorlage eine der oben genannten Ressourcen enthält, empfehlen wir Ihnen, vor der Bereitstellung alle mit der Anwendung verbundenen Berechtigungen zu überprüfen. Wenn Sie diesen Parameter nicht für eine Anwendung angeben, die Fähigkeiten benötigt, schlägt der Aufruf fehl.

Typ: Array des Typs [Capability](#)

Erforderlich: True

`resourcesSupported`

Ob alle in dieser Anwendung enthaltenen AWS Ressourcen in der Region unterstützt werden, in der sie abgerufen wird.

Typ: Boolesch

Erforderlich: True

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### ListApplications

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)

- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## CreateApplication

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## applicationId

### URI

/applications/*applicationId*

### HTTP-Methoden

#### GET

Vorgangs-ID: GetApplication

Ruft die angegebene Anwendung ab

## Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Abfrageparameter

Name	Typ	Erforderlich	Beschreibung
semanticVersion	String	False	Die semantische Version der Anwendung an, die Sie abrufen möchten.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">Application</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Statuscode	Antwortmodell	Beschreibung
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## DELETE

Vorgangs-ID: DeleteApplication

Löscht die angegebene Anwendung.

Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

Antworten

Statuscode	Antwortmodell	Beschreibung
204	Keine	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
409	<a href="#">ConflictException</a>	Die Ressource ist bereits vorhanden.

Statuscode	Antwortmodell	Beschreibung
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## PATCH

Vorgangs-ID: `UpdateApplication`

Aktualisiert die angegebene Anwendung.

## Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">Application</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
409	<a href="#">ConflictException</a>	Die Ressource ist bereits vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## Schemata

### Anfragetexte

## PATCH-Schema

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

## Antworttexte

### Application-Schema

```
{
  "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-Schema

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

### ForbiddenException-Schema

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

### NotFoundException-Schema

```
{
```

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

## ConflictException-Schema

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

## TooManyRequestsException-Schema

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### Application

Details zum Antrag.

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

## name

Der Name der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Pattern: "[a-zA-Z0-9\-\-]+";

Typ: Zeichenfolge

Erforderlich: True

## description

Die Beschreibung der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Typ: Zeichenfolge

Erforderlich: True

## author

Der Name des Anwendungsgebers.

Minimale Länge = 1. Maximale Länge = 127.

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

Typ: Zeichenfolge

Erforderlich: True

## isVerifiedAuthor

Gibt an, ob der Autor dieser Anwendung verifiziert wurde. Das bedeutet, AWS dass das Unternehmen als vernünftiger und umsichtiger Dienstleister die vom Antragsteller bereitgestellten Informationen nach bestem Wissen und Gewissen geprüft und bestätigt hat, dass die Identität des Antragstellers den Angaben entspricht.

Typ: Boolesch

Erforderlich: Falsch

## verifiedAuthorUrl

Die URL zum öffentlichen Profil eines verifizierten Autors. Diese URL wurde vom Autor eingereicht.

Typ: Zeichenfolge

Erforderlich: Falsch

## spdxLicenseId

Ein gültiger Bezeichner von <https://spdx.org/licenses/>.

Typ: Zeichenfolge

Erforderlich: Falsch

## licenseUrl

Ein Link zu einer Lizenzdatei der App, die dem spdxLicenseID-Wert Ihrer Anwendung entspricht.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## readmeUrl

Ein Link zur Readme-Datei in Markdown-Sprache, die eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## labels

Labels zur besseren Auffindbarkeit von Apps in Suchergebnissen.

Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl von Beschriftungen: 10

Pattern: `"^[a-zA-Z0-9+\\-_:\\V@]+$"`;

Typ: Array des Typs string

Erforderlich: Falsch

#### creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: Falsch

#### homePageUrl

Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.

Typ: Zeichenfolge

Erforderlich: Falsch

#### version

Versionsinformationen über die Anwendung.

Typ: [Version](#)

Erforderlich: Falsch

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

#### message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

#### errorCode

400

Typ: Zeichenfolge  
Erforderlich: Falsch

## Capability

Werte, die angegeben werden müssen, um einige Anwendungen bereitzustellen.

CAPABILITY\_IAM  
CAPABILITY\_NAMED\_IAM  
CAPABILITY\_AUTO\_EXPAND  
CAPABILITY\_RESOURCE\_POLICY

## ConflictException

Die Ressource ist bereits vorhanden.

message

Die Ressource ist bereits vorhanden.

Typ: Zeichenfolge  
Erforderlich: Falsch

errorCode

409

Typ: Zeichenfolge  
Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## ParameterDefinition

Von der Anwendung unterstützte Parameter.

name

Der Name des Parameters.

Typ: Zeichenfolge

Erforderlich: True

defaultValue

Gibt einen Wert des entsprechenden Typs an, den die Vorlage nutzen soll, wenn zum Zeitpunkt der Stack-Erstellung kein Wert angegeben wird. Wenn Sie Einschränkungen für den Parameter definieren, müssen Sie einen Wert angeben, der diese Einschränkungen einhält.

Typ: Zeichenfolge

Erforderlich: Falsch

description

Gibt eine Zeichenfolge mit maximal 4 000 Zeichen an, die den Parameter beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

type

Der Parametertyp.

Zulässige Werte: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

**String:** Gibt eine Literalzeichenfolge an.

Benutzer können beispielsweise angeben `"MyUserName"`.

**Number:** Eine Ganzzahl oder eine Fließkommazahl. AWS CloudFormation validiert den Parameterwert als Zahl. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die Ref intrinsische Funktion), wird der Parameterwert eine Zeichenfolge.

Benutzer könnten beispielsweise angeben `"8888"`

**List<Number>:** Gibt ein Array von Ganzzahlwert oder Ganzzahlwert an, jeweils durch Komma getrennt. AWS CloudFormation validiert den Parameterwert als Zahlen. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die Ref intrinsische Funktion), wird der Parameterwert zu einer Liste von Zeichenfolgen.

Benutzer könnten beispielsweise `„80,20“` angeben und dann das Ergebnis anzeigen. Ref `["80", "20"]`

**CommaDelimitedList:** Gibt ein Array von Literalzeichenfolgen an, jeweils durch Komma getrennt. Die Gesamtanzahl von Zeichenfolgen sollte um 1 höher sein als die Gesamtanzahl von Kommas. Jede Mitgliedszeichenfolge wird zudem gekürzt.

Benutzer könnten beispielsweise `„test, dev, prod“` angeben und dann die Ergebnisse anzeigen. Ref `["test", "dev", "prod"]`

Typ: Zeichenfolge

Erforderlich: Falsch

**noEcho**

Ob der Parameterwert maskiert werden soll, wenn jemand einen Aufruf tätigt, der den Stack beschreibt. Wenn Sie den Wert zuweisen, wird der Parameterwert mit Sternchen (`*****`) maskiert.

Typ: Boolesch

Erforderlich: Falsch

**allowedPattern**

Gibt einen regulären Ausdruck an, der die zulässigen Muster für `String`-Typen beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

### constraintDescription

Gibt eine Zeichenfolge als Beschreibung für eine Einschränkung an. Diese Beschreibung wird bei Verstößen gegen die Einschränkung angezeigt. Für einen Parameter mit einem zulässigen Muster `[A-Za-z0-9]+` beispielsweise wird ohne Einschränkungsbeschreibung die folgende Fehlermeldung angezeigt, wenn der Benutzer einen ungültigen Wert angibt:

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

Durch das Hinzufügen der Beschreibung einer Einschränkung, wie z. B. dürfen nur Groß- und Kleinbuchstaben und Zahlen enthalten, können Sie die folgende benutzerdefinierte Fehlermeldung anzeigen:

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

Typ: Zeichenfolge

Erforderlich: Falsch

### minValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Mindestwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

### maxValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Maximalwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

## minLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

## maxLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

## allowedValues

Gibt ein Array mit einer Liste der für den Parameter zulässigen Werte an.

Typ: Array des Typs `string`

Erforderlich: Falsch

## referencedByResources

Eine Liste von AWS SAM Ressourcen, die diesen Parameter verwenden.

Typ: Array des Typs `string`

Erforderlich: `True`

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

## message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## UpdateApplicationInput

Aktualisieren der Anwendungsanforderung.

description

Die Beschreibung der Anwendung.

Minimale Länge = 1. Höchstlänge = 1 024 Zeichen.

Typ: Zeichenfolge

Erforderlich: Falsch

author

Der Name des Anwendungsgebers.

Minimale Länge = 1. Maximale Länge = 127.

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

Typ: Zeichenfolge

Erforderlich: Falsch

readmeBody

Eine Text-Readme-Datei in Markdown-Sprache, die eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## readmeUrl

Ein Link zur Readme-Datei in Markdown-Sprache, die eine detailliertere Beschreibung der Anwendung und ihrer Funktionsweise enthält.

Maximale Größe 5 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## labels

Labels zur besseren Auffindbarkeit von Apps in Suchergebnissen.

Minimale Länge = 1. Maximale Länge = 127. Maximale Anzahl von Beschriftungen: 10

Pattern: `"^[a-zA-Z0-9+\\-_:\\@]+$"`;

Typ: Array des Typs string

Erforderlich: Falsch

## homePageUrl

Eine URL mit weiteren Informationen über die Anwendung, z. B. den Speicherort Ihres GitHub Repositorys für die Anwendung.

Typ: Zeichenfolge

Erforderlich: Falsch

## Version

Details zur Anwendungsversion.

## applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

### semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

### sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

### sourceCodeArchiveUrl

Ein Link zum S3-Objekt, das das ZIP-Archiv des Quellcodes für diese Version Ihrer Anwendung enthält.

Maximale Größe 50 MB

Typ: Zeichenfolge

Erforderlich: Falsch

### templateUrl

Ein Link zur verpackten AWS SAM Vorlage Ihrer Anwendung.

Typ: Zeichenfolge

Erforderlich: True

### creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

## parameterDefinitions

Eine Reihe von Parametertypen, die von der Anwendung unterstützt werden.

Typ: Array des Typs [ParameterDefinition](#)

Erforderlich: True

## requiredCapabilities

Eine Liste der Werte, die Sie angeben müssen, bevor Sie bestimmte Anwendungen bereitstellen können. Einige Anwendungen enthalten möglicherweise Ressourcen, die die Berechtigungen in Ihrem AWS Konto beeinflussen, beispielsweise, wenn neue AWS Identity and Access Management (IAM) -Benutzer erstellt werden. Für diese Anwendungen müssen Sie ihre Funktionen ausdrücklich bestätigen, indem Sie diesen Parameter angeben.

Die einzig gültigen Werte sind

CAPABILITY\_IAM, CAPABILITY\_NAMED\_IAM, CAPABILITY\_RESOURCE\_POLICY,  
und CAPABILITY\_AUTO\_EXPAND.

Für die folgenden Ressourcen müssen Sie CAPABILITY\_IAM oder angeben CAPABILITY\_NAMED\_IAM: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), und [AWS::IAM::Role](#). Wenn die Anwendung IAM-Ressourcen enthält, können Sie entweder CAPABILITY\_IAM oder CAPABILITY\_NAMED\_IAM angeben. Beinhaltet die Anwendung IAM-Ressourcen mit benutzerdefinierten Namen, müssen Sie CAPABILITY\_NAMED\_IAM angeben.

Für die folgenden Ressourcen müssen Sie Folgendes angeben CAPABILITY\_RESOURCE\_POLICY: [AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#), [AWS::S3::BucketPolicy](#) und [AWS::SQS::QueuePolicy](#), [AWS::SNS::TopicPolicy](#)

Bei Anwendungen, die eine oder mehrere verschachtelte Anwendungen enthalten, müssen Sie CAPABILITY\_AUTO\_EXPAND angeben.

Wenn Ihre Anwendungsvorlage eine der oben genannten Ressourcen enthält, empfehlen wir Ihnen, vor der Bereitstellung alle mit der Anwendung verbundenen Berechtigungen zu überprüfen. Wenn Sie diesen Parameter nicht für eine Anwendung angeben, die Fähigkeiten benötigt, schlägt der Aufruf fehl.

Typ: Array des Typs [Capability](#)

Erforderlich: True

resourcesSupported

Ob alle in dieser Anwendung enthaltenen AWS Ressourcen in der Region unterstützt werden, in der sie abgerufen wird.

Typ: Boolesch

Erforderlich: True

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### GetApplication

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

### DeleteApplication

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)

- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## UpdateApplication

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Anwendungs-ID-Änderungssätze

### URI

/applications/*applicationId*/changesets

### HTTP-Methoden

### POST

Vorgangs-ID: CreateCloudFormationChangeSet

Erstellt einen AWS CloudFormation Änderungsatz für die angegebene Anwendung.

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

#### Antworten

Statuscode	Antwortmodell	Beschreibung
201	<a href="#">ChangeSetDetails</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Anfragetexte

#### POST-Schema

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

## Antworttexte

### ChangeSetDetails-Schema

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

### BadRequestException-Schema

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

### ForbiddenException-Schema

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

### TooManyRequestsException-Schema

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

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

### ChangeSetDetails

Details zum Änderungssatz.

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

### semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

### changeSetId

Der Amazon-Ressourcenname (ARN) des Änderungssatzes.

Längenbeschränkungen: Minimale Länge von 1.

Pattern: ARN: [-a-zA-Z0-9:/] \*

Typ: Zeichenfolge

Erforderlich: True

### stackId

Die eindeutige ID des Stacks.

Typ: Zeichenfolge

Erforderlich: True

## CreateCloudFormationChangeSetInput

Erstellen Sie eine Anfrage für ein Anwendungs-Änderungssatz.

### stackName

Diese Eigenschaft entspricht dem AWS CloudFormation [CreateChangeSet](#) gleichnamigen Parameter für die API.

Typ: Zeichenfolge

Erforderlich: True

## semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: Falsch

## templateId

Die von zurückgegebene UUID. CreateCloudFormationTemplate

Muster: [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}

Typ: Zeichenfolge

Erforderlich: Falsch

## parameterOverrides

Eine Liste von Parameterwerten für die Parameter der Anwendung.

Typ: Array des Typs [ParameterValue](#)

Erforderlich: Falsch

## capabilities

Eine Liste der Werte, die Sie angeben müssen, bevor Sie bestimmte Anwendungen bereitstellen können. Einige Anwendungen enthalten möglicherweise Ressourcen, die die Berechtigungen in Ihrem AWS Konto beeinflussen, beispielsweise, wenn neue AWS Identity and Access Management (IAM) -Benutzer erstellt werden. Für diese Anwendungen müssen Sie ihre Funktionen ausdrücklich bestätigen, indem Sie diesen Parameter angeben.

Die einzig gültigen Werte sind

CAPABILITY\_IAM, CAPABILITY\_NAMED\_IAM, CAPABILITY\_RESOURCE\_POLICY,  
und CAPABILITY\_AUTO\_EXPAND.

Für die folgenden Ressourcen müssen Sie CAPABILITY\_IAM oder angeben CAPABILITY\_NAMED\_IAM: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#), und [AWS::IAM::Role](#). Wenn die Anwendung IAM-Ressourcen enthält, können Sie entweder

CAPABILITY\_IAM oder CAPABILITY\_NAMED\_IAM angeben. Beinhaltet die Anwendung IAM-Ressourcen mit benutzerdefinierten Namen, müssen Sie CAPABILITY\_NAMED\_IAM angeben.

Für die folgenden Ressourcen müssen Sie CAPABILITY\_RESOURCE\_POLICY:

[AWS::Lambda::Permission](#), [AWS::IAM::Policy](#),  
[AWS::ApplicationAutoScaling::ScalingPolicy](#), [AWS::S3::BucketPolicy](#), [AWS::SQS::QueuePolicy](#), und  
[AWS::SNS::TopicPolicy](#) angeben.

Bei Anwendungen, die eine oder mehrere verschachtelte Anwendungen enthalten, müssen Sie CAPABILITY\_AUTO\_EXPAND angeben.

Wenn Ihre Anwendungsvorlage eine der oben genannten Ressourcen enthält, empfehlen wir Ihnen, vor der Bereitstellung alle mit der Anwendung verbundenen Berechtigungen zu überprüfen. Wenn Sie diesen Parameter nicht für eine Anwendung angeben, die Fähigkeiten benötigt, schlägt der Aufruf fehl.

Typ: Array des Typs string

Erforderlich: Falsch

#### changeSetName

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Zeichenfolge

Erforderlich: Falsch

#### clientToken

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Zeichenfolge

Erforderlich: Falsch

#### description

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Zeichenfolge

Erforderlich: Falsch

#### notificationArns

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Array des Typs string

Erforderlich: Falsch

#### resourceTypes

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Array des Typs string

Erforderlich: Falsch

#### rollbackConfiguration

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: [RollbackConfiguration](#)

Erforderlich: Falsch

#### tags

Diese Eigenschaft entspricht dem gleichnamigen Parameter für die AWS CloudFormation [CreateChangeSetAPI](#).

Typ: Array des Typs [Tag](#)

Erforderlich: Falsch

#### ForbiddenException

Der Client ist nicht authentifiziert.

## message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## ParameterValue

Parameterwert der Anwendung.

## name

Der Schlüssel, der dem Parameter zugeordnet ist. Wenn Sie keinen Schlüssel und Wert für einen bestimmten Parameter angeben, AWS CloudFormation verwendet den Standardwert, der in Ihrer Vorlage angegeben ist.

Typ: Zeichenfolge

Erforderlich: True

## value

Der dem Parameter zugeordnete Eingabewert.

Typ: Zeichenfolge

Erforderlich: True

## RollbackConfiguration

Diese Eigenschaft entspricht dem AWS CloudFormation [RollbackConfiguration](#)Datentyp.

### rollbackTriggers

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [RollbackConfiguration](#)Datentyp.

Typ: Array des Typs [RollbackTrigger](#)

Erforderlich: Falsch

### monitoringTimeInMinutes

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [RollbackConfiguration](#)Datentyp.

Typ: Ganzzahl

Erforderlich: Falsch

## RollbackTrigger

Diese Eigenschaft entspricht dem AWS CloudFormation [RollbackTrigger](#)Datentyp.

## arn

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [RollbackTrigger](#)Datentyp.

Typ: Zeichenfolge

Erforderlich: True

## type

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [RollbackTrigger](#)Datentyp.

Typ: Zeichenfolge

Erforderlich: True

## Tag

Diese Eigenschaft entspricht dem AWS CloudFormation [Tag-Datentyp](#).

### key

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [Tag-Datentyp](#).

Typ: Zeichenfolge

Erforderlich: True

### value

Diese Eigenschaft entspricht dem gleichnamigen Inhalt für den AWS CloudFormation [Tag-Datentyp](#).

Typ: Zeichenfolge

Erforderlich: True

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

## message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### CreateCloudFormationChangeSet

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

# Applications applicationId Dependencies

## URI

/applications/*applicationId*/dependencies

## HTTP-Methoden

### GET

Vorgangs-ID: ListApplicationDependencies

Ruft die Liste der Anwendungen ab, die in der enthaltenden Anwendung verschachtelt sind.

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

### Abfrageparameter

Name	Typ	Erforderlich	Beschreibung
nextToken	String	False	Ein Token für den Beginn der Seitennummerierung.
maxItems	String	False	Die Gesamtzahl der zurückzugebenden Elemente.
semanticVersion	String	False	Die semantische Version der Anwendung an, die Sie abrufen möchten.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">ApplicationDependencyPage</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Antworttexte

#### ApplicationDependencyPage-Schema

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

#### BadRequestException-Schema

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

#### ForbiddenException-Schema

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

#### NotFoundException-Schema

```
{
```

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

### TooManyRequestsException-Schema

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

### InternalServerErrorException-Schema

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

## Eigenschaften

### ApplicationDependencyPage

Eine Liste von Anwendungszusammenfassungen, die in der Anwendung verschachtelt sind.

dependencies

Eine Reihe von Anwendungszusammenfassungen, die in der Anwendung verschachtelt sind.

Typ: Array des Typs [ApplicationDependencySummary](#)

Erforderlich: True

nextToken

Das Token, um die nächste Ergebnisseite anzufordern.

Typ: Zeichenfolge

Erforderlich: Falsch

## ApplicationDependencySummary

Eine Zusammenfassung verschachtelter Anwendungen.

applicationId

Der Amazon-Ressourcenname (ARN) der verschachtelten Anwendung.

Typ: Zeichenfolge

Erforderlich: True

semanticVersion

Die semantische Version der verschachtelten Anwendung.

Typ: Zeichenfolge

Erforderlich: True

## BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

## message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

## message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

## message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

## ListApplicationDependencies

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## ApplicationID-Richtlinie für Anwendungen

### URI

/applications/*applicationId*/policy

### HTTP-Methoden

#### GET

Vorgangs-ID: GetApplicationPolicy

Ruft die Richtlinie für die Anwendung ab.

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Resourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">ApplicationPolicy</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## PUT

Vorgangs-ID: PutApplicationPolicy

Legt die Berechtigungsrichtlinie für eine Anwendung fest. Eine Liste der für diesen Vorgang unterstützten Aktionen finden Sie unter [Anwendungsberechtigungen](#).

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">ApplicationPolicy</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Anfragetexte

#### PUT-Schema

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

### Antworttexte

#### ApplicationPolicy-Schema

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

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

### BadRequestException-Schema

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

### ForbiddenException-Schema

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

### NotFoundException-Schema

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

### TooManyRequestsException-Schema

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### ApplicationPolicy

Grundsatzserklärungen, die auf die Anwendung angewendet wurden.

statements

Eine Reihe von Richtlinienserklärungen, die auf die Anwendung angewendet wurden.

Typ: Array des Typs [ApplicationPolicyStatement](#)

Erforderlich: True

### ApplicationPolicyStatement

Für die Anwendung geltende Grundsatzserklärung.

statementId

Eine eindeutige ID für den Kontoauszug.

Typ: Zeichenfolge

Erforderlich: Falsch

principals

Eine Reihe von AWS Konten IDs , mit denen die Anwendung gemeinsam genutzt werden soll, oder \*, um die Anwendung öffentlich zu machen.

Typ: Array des Typs string

Erforderlich: True

## actions

Eine Liste der Aktionen, die für diesen Vorgang unterstützt werden, finden Sie unter [Anwendungsberechtigungen](#).

Typ: Array des Typs string

Erforderlich: True

## principalOrgIDs

Die AWS Organizations ID, mit der die Anwendung gemeinsam genutzt werden soll.

Typ: Array des Typs string

Erforderlich: Falsch

## BadRequestException

Einer der Parameter der Anfrage ist ungültig.

### message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

### errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

### message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### GetApplicationPolicy

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)

- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## PutApplicationPolicy

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Applications applicationId Templates

### URI

/applications/*applicationId*/templates

### HTTP-Methoden

#### POST

Vorgangs-ID: CreateCloudFormationTemplate

Erzeugt eine AWS CloudFormation Vorlage.

## Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

## Antworten

Statuscode	Antwortmodell	Beschreibung
201	<a href="#">TemplateDetails</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Anfragetexte

#### POST-Schema

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

### Antworttexte

#### TemplateDetails-Schema

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

```
}
```

### BadRequestException-Schema

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

### ForbiddenException-Schema

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

### NotFoundException-Schema

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

### TooManyRequestsException-Schema

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

### InternalServerErrorException-Schema

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

## Eigenschaften

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

### CreateCloudFormationTemplateInput

Erstellen Sie eine Vorlagenanfrage.

semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: Falsch

### ForbiddenException

Der Client ist nicht authentifiziert.

message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TemplateDetails

Details zur Vorlage.

templateId

Die von zurückgegebene UUID. CreateCloudFormationTemplate

Muster: [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}

Typ: Zeichenfolge

Erforderlich: True

templateUrl

Ein Link zur Vorlage, mit der die Anwendung bereitgestellt werden kann AWS CloudFormation.

Typ: Zeichenfolge

Erforderlich: True

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

status

Status des Workflows zur Vorlagenerstellung.

Der mögliche Wert ist: PREPARING | ACTIVE | EXPIRED

Typ: Zeichenfolge

Erforderlich: True

Werte: PREPARING | ACTIVE | EXPIRED

creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

expirationTime

Das Datum und Uhrzeit, zu denen diese Vorlage abläuft. Vorlagen laufen 1 Stunde nach ihrer Erstellung ab.

Typ: Zeichenfolge

Erforderlich: True

TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### CreateCloudFormationTemplate

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Applications applicationId Templates templateId

### URI

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

### HTTP-Methoden

#### GET

Vorgangs-ID: GetCloudFormationTemplate

Ruft die angegebene AWS CloudFormation Vorlage ab.

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.
<i>templateId</i>	String	True	Die von zurückgegebene UUID. CreateCloudFormationTemplate  Muster: [0-9a-fa-F]{8}\ - [0-9a-fa-F]{4}\ - [0-9a-fa-F]{4}\ - [0-9a-fa-F]{4}\ - [0-9a-fa-F]{4}\ - [0-9a-fa-F]{12}

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#">TemplateDetails</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.

Statuscode	Antwortmodell	Beschreibung
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.
<i>templateId</i>	String	True	Die von zurückgegebene UUID. CreateCloudFormationTemplate  Muster: [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}

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	200 Antworten

# Schemata

## Antworttexte

### TemplateDetails-Schema

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

### BadRequestException-Schema

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

### ForbiddenException-Schema

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

### NotFoundException-Schema

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

## TooManyRequestsException-Schema

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

### ForbiddenException

Der Client ist nicht authentifiziert.

message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TemplateDetails

Details zur Vorlage.

templateId

Die von zurückgegebene UUID. CreateCloudFormationTemplate

Muster: [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}

Typ: Zeichenfolge

Erforderlich: True

templateUrl

Ein Link zur Vorlage, mit der die Anwendung bereitgestellt werden kann AWS CloudFormation.

Typ: Zeichenfolge

Erforderlich: True

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

status

Status des Workflows zur Vorlagenerstellung.

Der mögliche Wert ist: PREPARING | ACTIVE | EXPIRED

Typ: Zeichenfolge

Erforderlich: True

Werte: PREPARING | ACTIVE | EXPIRED

creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

expirationTime

Das Datum und Uhrzeit, zu denen diese Vorlage abläuft. Vorlagen laufen 1 Stunde nach ihrer Erstellung ab.

Typ: Zeichenfolge

Erforderlich: True

TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### GetCloudFormationTemplate

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Applications applicationId Unshare

### URI

/applications/*applicationId*/unshare

### HTTP-Methoden

### POST

Vorgangs-ID: UnshareApplication

Macht die Freigabe einer Anwendung aus einer AWS Organisation rückgängig.

Dieser Vorgang kann nur über das Verwaltungskonto der Organisation aufgerufen werden.

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

#### Antworten

Statuscode	Antwortmodell	Beschreibung
204	Keine	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Anfragetexte

#### POST-Schema

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

### Antworttexte

#### BadRequestException-Schema

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

#### ForbiddenException-Schema

```
{
```

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

## NotFoundException-Schema

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

## TooManyRequestsException-Schema

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

### message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

### message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## UnshareApplicationInput

Anwendungsanfrage nicht mehr teilen.

organizationId

Die AWS Organizations ID, für die die Freigabe der Anwendung aufgehoben werden soll.

Typ: Zeichenfolge

Erforderlich: True

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

### UnshareApplication

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Anwendungs-ID-Versionen

### URI

`/applications/applicationId/versions`

## HTTP-Methoden

### GET

Vorgangs-ID: `ListApplicationVersions`

Listet die Versionen für die angegebene Anwendung auf.

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i><code>applicationId</code></i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

#### Abfrageparameter

Name	Typ	Erforderlich	Beschreibung
<code>maxItems</code>	String	False	Die Gesamtzahl der zurückzugebenden Elemente.
<code>nextToken</code>	String	False	Ein Token für den Beginn der Seitennummerierung.

#### Antworten

Statuscode	Antwortmodell	Beschreibung
200	<a href="#"><u>ApplicationVersionPage</u></a>	Herzlichen Glückwunsch
400	<a href="#"><u>BadRequestException</u></a>	Einer der Parameter der Anfrage ist ungültig.

Statuscode	Antwortmodell	Beschreibung
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
404	<a href="#">NotFoundException</a>	Die in der Anfrage angegebene Ressource (z. B. eine Anweisung zur Zugriffsrichtlinie) ist nicht vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

### Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort 200

## Schemata

### Antworttexte

## ApplicationVersionPage-Schema

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

## BadRequestException-Schema

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

## ForbiddenException-Schema

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

## NotFoundException-Schema

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

## TooManyRequestsException-Schema

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

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### ApplicationVersionPage

Eine Liste der Versionszusammenfassungen für die Anwendung.

versions

Eine Reihe von Versionszusammenfassungen für die Anwendung.

Typ: Array des Typs [VersionSummary](#)

Erforderlich: True

nextToken

Das Token, um die nächste Ergebnisseite anzufordern.

Typ: Zeichenfolge

Erforderlich: Falsch

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

### message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

### message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## NotFoundException

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

message

Die in der Anfrage angegebene Ressource (z. B. eine Erklärung zur Zugriffsrichtlinie) ist nicht vorhanden.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

404

Typ: Zeichenfolge

Erforderlich: Falsch

## TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## VersionSummary

Eine Zusammenfassung der Anwendungsversion.

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs sowie Referenzen finden Sie unter:

## ListApplicationVersions

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

## Anwendungen ApplicationID-Versionen SemanticVersion

### URI

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

### HTTP-Methoden

#### PUT

Vorgangs-ID: CreateApplicationVersion

Erstellt eine Anwendungsversion

#### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Resourcenname (ARN) der Anwendung.

Name	Typ	Erforderlich	Beschreibung
<i>semanticVersion</i>	String	True	Die semantische Version der neuen Version.

## Antworten

Statuscode	Antwortmodell	Beschreibung
201	<a href="#">Version</a>	Herzlichen Glückwunsch
400	<a href="#">BadRequestException</a>	Einer der Parameter der Anfrage ist ungültig.
403	<a href="#">ForbiddenException</a>	Der Client ist nicht authentifiziert.
409	<a href="#">ConflictException</a>	Die Ressource ist bereits vorhanden.
429	<a href="#">TooManyRequestsException</a>	Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.
500	<a href="#">InternalServerErrorException</a>	Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## OPTIONS

### Pfadparameter

Name	Typ	Erforderlich	Beschreibung
<i>applicationId</i>	String	True	Der Amazon-Ressourcenname (ARN) der Anwendung.

Name	Typ	Erforderlich	Beschreibung
<i>semanticVersion</i>	String	True	Die semantische Version der neuen Version.

## Antworten

Statuscode	Antwortmodell	Beschreibung
200	Keine	Antwort von 200

## Schemata

### Anfragetexte

#### PUT-Schema

```
{
  "templateBody": "string",
  "templateUrl": "string",
  "sourceCodeUrl": "string",
  "sourceCodeArchiveUrl": "string"
}
```

### Antworttexte

#### Version-Schema

```
{
  "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-Schema

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

### ForbiddenException-Schema

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

### ConflictException-Schema

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

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

## TooManyRequestsException-Schema

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

## InternalServerErrorException-Schema

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

## Eigenschaften

### BadRequestException

Einer der Parameter der Anfrage ist ungültig.

message

Einer der Parameter der Anfrage ist ungültig.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

400

Typ: Zeichenfolge

Erforderlich: Falsch

### Capability

Werte, die angegeben werden müssen, um einige Anwendungen bereitzustellen.

CAPABILITY\_IAM  
CAPABILITY\_NAMED\_IAM  
CAPABILITY\_AUTO\_EXPAND  
CAPABILITY\_RESOURCE\_POLICY

## ConflictException

Die Ressource ist bereits vorhanden.

message

Die Ressource ist bereits vorhanden.

Typ: Zeichenfolge  
Erforderlich: Falsch

errorCode

409

Typ: Zeichenfolge  
Erforderlich: Falsch

## CreateApplicationVersionInput

Erstellen Sie eine Versionsanfrage.

templateBody

Die AWS SAM unformatierte Paketvorlage Ihrer Anwendung.

Typ: Zeichenfolge  
Erforderlich: Falsch

templateUrl

Ein Link zur verpackten AWS SAM Vorlage Ihrer Anwendung.

Typ: Zeichenfolge  
Erforderlich: Falsch

## sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

## sourceCodeArchiveUrl

Ein Link zum S3-Objekt, das das ZIP-Archiv des Quellcodes für diese Version Ihrer Anwendung enthält.

Maximale Größe 50 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## ForbiddenException

Der Client ist nicht authentifiziert.

## message

Der Client ist nicht authentifiziert.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

403

Typ: Zeichenfolge

Erforderlich: Falsch

## InternalServerErrorException

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

## message

Beim AWS Serverless Application Repository Service trat ein interner Fehler auf.

Typ: Zeichenfolge

Erforderlich: Falsch

## errorCode

500

Typ: Zeichenfolge

Erforderlich: Falsch

## ParameterDefinition

Von der Anwendung unterstützte Parameter.

### name

Der Name des Parameters.

Typ: Zeichenfolge

Erforderlich: True

### defaultValue

Gibt einen Wert des entsprechenden Typs an, den die Vorlage nutzen soll, wenn zum Zeitpunkt der Stack-Erstellung kein Wert angegeben wird. Wenn Sie Einschränkungen für den Parameter definieren, müssen Sie einen Wert angeben, der diese Einschränkungen einhält.

Typ: Zeichenfolge

Erforderlich: Falsch

### description

Gibt eine Zeichenfolge mit maximal 4 000 Zeichen an, die den Parameter beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

## type

Der Parametertyp.

Zulässige Werte: `String` | `Number` | `List<Number>` | `CommaDelimitedList`

`String`: Gibt eine Literalzeichenfolge an.

Benutzer können beispielsweise angeben `"MyUserName"`.

`Number`: Eine Ganzzahl oder eine Fließkommazahl. AWS CloudFormation validiert den Parameterwert als Zahl. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die `Ref` intrinsische Funktion), wird der Parameterwert eine Zeichenfolge.

Benutzer könnten beispielsweise angeben `"8888"`

`List<Number>`: Gibt ein Array von Ganzzahlwert oder Ganzzahlwert an, jeweils durch Komma getrennt. AWS CloudFormation validiert den Parameterwert als Zahlen. Wenn Sie den Parameter jedoch an anderer Stelle in Ihrer Vorlage verwenden (zum Beispiel über die `Ref` intrinsische Funktion), wird der Parameterwert zu einer Liste von Zeichenfolgen.

Benutzer könnten beispielsweise `„80,20“` angeben und dann das Ergebnis anzeigen. `Ref` `["80", "20"]`

`CommaDelimitedList`: Gibt ein Array von Literalzeichenfolgen an, jeweils durch Komma getrennt. Die Gesamtanzahl von Zeichenfolgen sollte um 1 höher sein als die Gesamtanzahl von Kommas. Jede Mitgliedszeichenfolge wird zudem gekürzt.

Benutzer könnten beispielsweise `„test, dev, prod“` angeben und dann die Ergebnisse angeben. `Ref` `["test", "dev", "prod"]`

Typ: Zeichenfolge

Erforderlich: Falsch

## noEcho

Ob der Parameterwert maskiert werden soll, wenn jemand einen Aufruf tätigt, der den Stack beschreibt. Wenn Sie den Wert zuweisen, wird der Parameterwert mit Sternchen (`****`) maskiert.

Typ: Boolesch

Erforderlich: Falsch

## allowedPattern

Gibt einen regulären Ausdruck an, der die zulässigen Muster für `String`-Typen beschreibt.

Typ: Zeichenfolge

Erforderlich: Falsch

## constraintDescription

Gibt eine Zeichenfolge als Beschreibung für eine Einschränkung an. Diese Beschreibung wird bei Verstößen gegen die Einschränkung angezeigt. Für einen Parameter mit einem zulässigen Muster `[A-Za-z0-9]+` beispielsweise wird ohne Einschränkungsbeschreibung die folgende Fehlermeldung angezeigt, wenn der Benutzer einen ungültigen Wert angibt:

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

Durch das Hinzufügen der Beschreibung einer Einschränkung, wie z. B. dürfen nur Groß- und Kleinbuchstaben und Zahlen enthalten, können Sie die folgende benutzerdefinierte Fehlermeldung anzeigen:

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

Typ: Zeichenfolge

Erforderlich: Falsch

## minValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Mindestwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

## maxValue

Gibt einen numerischen Wert an, der den zulässigen numerischen Maximalwert für `Number`-Typen definiert.

Typ: Ganzzahl

Erforderlich: Falsch

### minLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

### maxLength

Gibt einen Ganzzahlwert an, der festlegt, wie viele Zeichen maximal für `String`-Typen zulässig sind.

Typ: Ganzzahl

Erforderlich: Falsch

### allowedValues

Gibt ein Array mit einer Liste der für den Parameter zulässigen Werte an.

Typ: Array des Typs `string`

Erforderlich: Falsch

### referencedByResources

Eine Liste von AWS SAM Ressourcen, die diesen Parameter verwenden.

Typ: Array des Typs `string`

Erforderlich: `True`

### TooManyRequestsException

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

### message

Der Client sendet mehr als die zulässige Anzahl von Anfragen pro Zeiteinheit.

Typ: Zeichenfolge

Erforderlich: Falsch

errorCode

429

Typ: Zeichenfolge

Erforderlich: Falsch

## Version

Details zur Anwendungsversion

applicationId

Der Amazon-Ressourcenname (ARN) der -Anwendung

Typ: Zeichenfolge

Erforderlich: True

semanticVersion

Die semantische Version der Anwendung:

<https://semver.org/>

Typ: Zeichenfolge

Erforderlich: True

sourceCodeUrl

Ein Link zu einem öffentlichen Repository für den Quellcode Ihrer Anwendung, z. B. die URL eines bestimmten GitHub Commits.

Typ: Zeichenfolge

Erforderlich: Falsch

## sourceCodeArchiveUrl

Ein Link zum S3-Objekt, das das ZIP-Archiv des Quellcodes für diese Version Ihrer Anwendung enthält.

Maximale Größe 50 MB

Typ: Zeichenfolge

Erforderlich: Falsch

## templateUrl

Ein Link zur verpackten AWS SAM Vorlage Ihrer Anwendung.

Typ: Zeichenfolge

Erforderlich: True

## creationTime

Das Datum und Uhrzeit, zu denen diese Ressource erstellt wurde.

Typ: Zeichenfolge

Erforderlich: True

## parameterDefinitions

Eine Reihe von Parametertypen, die von der Anwendung unterstützt werden.

Typ: Array des Typs [ParameterDefinition](#)

Erforderlich: True

## requiredCapabilities

Eine Liste der Werte, die Sie angeben müssen, bevor Sie bestimmte Anwendungen bereitstellen können. Einige Anwendungen enthalten möglicherweise Ressourcen, die die Berechtigungen in Ihrem AWS Konto beeinflussen, beispielsweise, wenn neue AWS Identity and Access Management (IAM) -Benutzer erstellt werden. Für diese Anwendungen müssen Sie ihre Funktionen ausdrücklich bestätigen, indem Sie diesen Parameter angeben.

Die einzig gültigen Werte sind

CAPABILITY\_IAM, CAPABILITY\_NAMED\_IAM, CAPABILITY\_RESOURCE\_POLICY,  
und CAPABILITY\_AUTO\_EXPAND.

Für die folgenden Ressourcen müssen Sie CAPABILITY\_IAM oder  
angeben CAPABILITY\_NAMED\_IAM: [AWS::IAM::Group](#), [AWS::IAM::InstanceProfile](#), [AWS::IAM::Policy](#),  
und [AWS::IAM::Role](#). Wenn die Anwendung IAM-Ressourcen enthält, können Sie entweder  
CAPABILITY\_IAM oder CAPABILITY\_NAMED\_IAM angeben. Beinhaltet die Anwendung IAM-  
Ressourcen mit benutzerdefinierten Namen, müssen Sie CAPABILITY\_NAMED\_IAM angeben.

Für die folgenden Ressourcen müssen Sie Folgendes angeben CAPABILITY\_RESOURCE\_POLICY:  
[AWS::Lambda::Permission](#), [AWS::IAM::Policy](#), [AWS::ApplicationAutoScaling::ScalingPolicy](#),  
[AWS::S3::BucketPolicy](#) und [AWS::SQS::QueuePolicy](#), [AWS::SNS::TopicPolicy](#)

Bei Anwendungen, die eine oder mehrere verschachtelte Anwendungen enthalten, müssen Sie  
CAPABILITY\_AUTO\_EXPAND angeben.

Wenn Ihre Anwendungsvorlage eine der oben genannten Ressourcen enthält, empfehlen wir Ihnen,  
vor der Bereitstellung alle mit der Anwendung verbundenen Berechtigungen zu überprüfen. Wenn  
Sie diesen Parameter nicht für eine Anwendung angeben, die Fähigkeiten benötigt, schlägt der Aufruf  
fehl.

Typ: Array des Typs [Capability](#)

Erforderlich: True

resourcesSupported

Ob alle in dieser Anwendung enthaltenen AWS Ressourcen in der Region unterstützt werden, in der  
sie abgerufen wird.

Typ: Boolesch

Erforderlich: True

## Weitere Informationen finden Sie auch unter

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen API AWS SDKs  
sowie Referenzen finden Sie unter:

## CreateApplicationVersion

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS SDK für .NET](#)
- [AWS SDK für C++](#)
- [AWS-SDK SDK for Go v2](#)
- [AWS SDK für Java V2](#)
- [AWS-SDK für JavaScript Version 3](#)
- [AWS-SDK für Kotlin](#)
- [AWS SDK für PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK für Ruby V3](#)

# Dokumentverlauf

- API-Version: aktuelle
- Letzte Aktualisierung der Dokumentation: 10. März 2020

In der folgenden Tabelle werden die wichtigen Änderungen in den einzelnen Versionen des AWS Serverless Application Repository Developer Guide beschrieben. Um Benachrichtigungen über Aktualisierungen dieser Dokumentation zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
<a href="#">Aktualisierungen zur gemeinsamen Nutzung und Beschränkung des Zugriffs auf Anwendungen</a>	Unterstützung für die gemeinsame Nutzung von Anwendungen für Konten in einer AWS Organisation und die Beschränkung des Zugriffs auf öffentliche Anwendungen für AWS Konten und AWS Organisationen hinzugefügt. Weitere Beispiele für die gemeinsame Nutzung von Anwendungen für Benutzer in einer Organisation finden Sie unter <a href="#">Beispiele für AWS Serverless Application Repository Anwendungsrichtlinien</a> . Beispiele für die Beschränkung des Zugriffs auf öffentliche Anwendungen finden Sie unter <a href="#">Identitätsbasierte AWS Serverless Application Repository -Richtlinienbeispiele</a> .	10. März 2020
<a href="#">Neue unterstützte Ressourcen</a>	Eine Reihe zusätzlicher Ressourcen wird nun unterstützt	17. Januar 2020

zt. Die vollständige Liste der unterstützten Ressourcen finden Sie unter [Liste der unterstützten AWS Ressourcen](#).

### [Regionen in China](#)

Das AWS Serverless Application Repository ist jetzt in den Regionen China, Peking und Ningxia erhältlich. Weitere Informationen zu AWS Serverless Application Repository Regionen und Endpunkten finden Sie unter [Regionen und Endpunkte](#) in der. Allgemeine AWS-Referenz

15. Januar 2020

### [Der Abschnitt Sicherheit wurde aus Gründen der Konsistenz mit anderen AWS Diensten aktualisiert.](#)

Weitere Informationen finden Sie unter [Sicherheit](#).

2. Januar 2020

## [Vereinfachter Prozess für die Veröffentlichung von Anwendungen](#)

Der neue `publish` Befehl in der AWS SAM CLI vereinfacht das Veröffentlichen von serverlosen Anwendungen in der AWS Serverless Application Repository. Ein end-to-end Tutorial zum Herunterladen und Veröffentlichen einer Beispielanwendung finden Sie unter [Schnellstart: Veröffentlichen von Anwendungen](#). Anweisungen zum Veröffentlichen einer Anwendung, die Sie bereits in der AWS Cloud entwickelt und getestet haben, finden Sie unter [Veröffentlichen einer Anwendung über die AWS SAM CLI](#).

21. Dezember 2018

## [Unterstützung für verschachtelte Anwendungen und Ebenen](#)

Support für verschachtelte Anwendungen und Ebenen hinzugefügt. Dazu gehören Aktualisierungen der [unterstützten AWS Ressourcen](#) und die [Bestätigung der Anwendungsfunktionen](#).

29. November 2018

[Veröffentlichen von Anwendungen mit benutzerdefinierten IAM-Rollen und Ressourcenrichtlinien](#)

Support für das Veröffentlichen von Anwendungen mit benutzerdefinierten IAM-Rollen und Ressourcenrichtlinien hinzugefügt. Dazu gehören Aktualisierungen der Workflows „[Anwendungen nutzen](#)“ und „[Anwendungen veröffentlichen](#)“ sowie Aktualisierungen der [unterstützten AWS Ressourcen](#) und der [API-Referenz](#) im AWS Serverless Application Repository Entwicklerhandbuch.

16. November 2018

[Aktualisierungen der Richtlinienvorlagen](#)

Aktualisierungen der unterstützten [Richtlinienvorlagen](#) im AWS Serverless Application Repository Entwicklerhandbuch.

26. September 2018

[Aktualisierungen der Dokumentation](#)

Das Thema Authentifizierung und Zugriffskontrolle wurde dem AWS Serverless Application Repository Entwicklerhandbuch hinzugefügt.

2. Juli 2018

## Öffentliche Veröffentlichung

Öffentliche Version von AWS Serverless Application Repository, die jetzt in 14 AWS Regionen verfügbar ist. Weitere Informationen zu den AWS Regionen, in denen der verfügbar AWS Serverless Application Repository ist, und zu den AWS Serverless Application Repository Endpunkten finden Sie unter [Regionen und Endpunkte](#) in der. Allgemeine AWS-Referenz

20. Februar 2018

## Neues Handbuch

Dies ist die erste Vorabversion des AWS Serverless Application Repository Developer Guide.

30. November 2017

# AWS Glossar

Die neueste AWS Terminologie finden Sie im [AWS Glossar](#) in der AWS-Glossar Referenz.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.