Benutzerhandbuch

Entwicklertools-Konsole

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

Entwicklertools-Konsole: Benutzerhandbuch

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 die Entwicklertools-Konsole? ... 1
Verwenden Sie zum ersten Mal? ... 3
Funktionen der Entwicklertools-Konsole ... 3
Was sind Benachrichtigungen? ... 4
Wofür kann ich Benachrichtigungen verwenden? ... 4
Wie funktionieren Benachrichtigungen? ... 4
Wie kann ich in die Verwendung von Benachrichtigungen einsteigen? ... 5
Benachrichtigungskonzepte ... 5
Einrichtung ... 14
Einstieg in die Verwendung von Benachrichtigungen ... 21
Arbeiten mit Benachrichtigungsregeln ... 29
Arbeiten mit Benachrichtigungsregelzielen ... 43
Konfigurieren der Integration zwischen Benachrichtigungen und AWS Chatbot ... 53
API-Aufrufe für AWS CodeStar Benachrichtigungen protokollieren mit AWS CloudTrail ... 59
Fehlerbehebung ... 62
Kontingente ... 66
Was sind Verbindungen? ... 66
Was kann ich mit Verbindungen tun? ... 67
Für welche Drittanbieter kann ich Verbindungen erstellen? ... 67
Was AWS-Services lässt sich in Verbindungen integrieren? ... 68
Wie funktionieren Verbindungen? ... 68
Globale Ressourcen in AWS CodeConnections ... 75
Was sind die ersten Schritte mit Verbindungen? ... 76
Verbindungen – Konzepte ... 76
AWS CodeConnections unterstützte Anbieter und Versionen ... 77
Produkt- und Serviceintegrationen mit AWS CodeConnections ... 78
Einrichten von Verbindungen ... 81
Erste Schritte mit Verbindungen ... 85
Arbeiten mit Verbindungen ... 92
Arbeiten mit Hosts ... 155
Arbeiten mit Synchronisierungskonfigurationen für verknüpfte Repositorys ... 168
Protokollieren von Verbindungen API-Aufrufe mit CloudTrail ... 178
VPC-Endpunkte (AWS PrivateLink) ... 219
Fehlerbehebung bei Verbindungen ... 222

Kontingente ... 237
IP-Adressen, die Sie Ihrer Zulassungsliste hinzufügen möchten ... 238
Sicherheit ... 240
Grundlagen zu Benachrichtigungsinhalten und -sicherheit ... 241
Datenschutz ... 242
Identity and Access Management ... 243
Zielgruppe ... 244
Authentifizierung mit Identitäten ... 245
Verwalten des Zugriffs mit Richtlinien ... 248
Funktionsweise von Funktionen in der Entwicklertools-Konsole mit IAM ... 249
AWS CodeConnections Referenz zu Berechtigungen ... 256
Beispiele für identitätsbasierte Richtlinien ... 272
Verwenden von Tags zur Steuerung des Zugriffs auf AWS CodeConnections - Ressourcen ... 285
Verwenden der Konsole ... 287
Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer ... 288
Fehlerbehebung ... 289
Verwenden von dienstbezogenen Rollen für Benachrichtigungen AWS CodeStar ... 292
Verwenden von serviceverknüpften Rollen für AWS CodeConnections ... 296
AWS verwaltete Richtlinien ... 299
Compliance-Validierung ... 302
Ausfallsicherheit ... 303
Sicherheit der Infrastruktur ... 303
Datenverkehr zwischen Ressourcen von AWS CodeConnections über Regionen hinweg ... 304
Verbindungen umbenennen — Zusammenfassung der Änderungen ... 305
Dienstpräfix umbenannt ... 305
Aktionen in IAM wurden umbenannt ... 306
Neue Ressource ARN ... 306
Betroffene Richtlinien für Servicerollen ... 4
Neue CloudFormation Ressource ... 5
Dokumentverlauf ... 308
AWS Glossar ... 317
... cccxviii

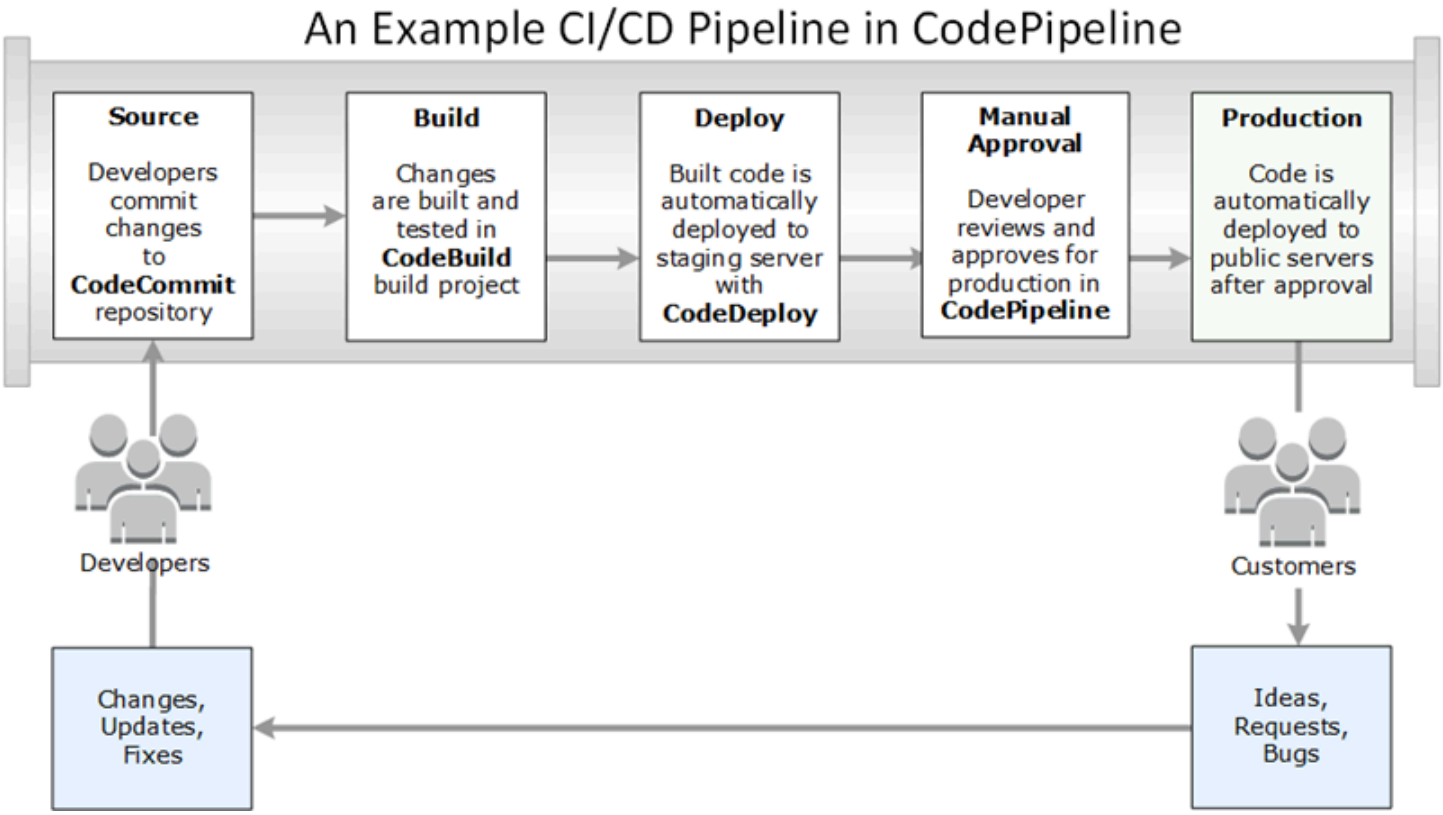
Was ist die Entwicklertools-Konsole?

Die Entwicklertools-Konsole umfasst eine Reihe von Services und Funktionen, die Sie einzeln oder gemeinsam bei der Entwicklung von Software unterstützen, ob allein oder im Team. Mit den Entwicklertools können Sie Ihre Software sicher speichern, erstellen, testen und bereitstellen. Diese Tools werden einzeln oder gemeinsam verwendet und bieten Unterstützung für DevOps Continuous Integration und Continuous Delivery (CI/CD).

Die Entwicklertools-Konsole umfasst die folgenden Services:

- AWS CodeCommit ist ein vollständig verwalteter Service für Quellcodekontrolle, der private Git-Repositorys hostet. Sie können Repositorys verwenden, um Komponenten (wie Dokumente, Quellcode und Binärdateien) privat in der AWS Cloud zu speichern und zu verwalten. In den Repositorys wird der gesamte Projektverlauf – vom ersten Commit bis zur letzten Änderung – gespeichert. Sie können Code in Repositorys gemeinsam bearbeiten, indem Sie Code kommentieren und Pull-Anforderungen erstellen, um die Codequalität sicherzustellen.
- AWS CodeBuild ist ein vollständig verwalteter Build-Service. Sie können damit Ihren Quellcode kompilieren, Einheitentests ausführen und bereitstellbare Artefakte generieren. Es bietet vorgefertigte Build-Umgebungen für gängige Programmiersprachen und Build-Tools wie Apache Maven, Gradle und mehr. Sie können Build-Umgebungen auch so anpassen CodeBuild , dass Sie Ihre eigenen Build-Tools verwenden.
- AWS CodeDeployist ein vollständig verwalteter Bereitstellungsservice, der Softwarebereitstellungen für Rechendienste wie Amazon EC2 und Ihre lokalen Server automatisiert. AWS Lambda Der Service kann Ihnen helfen, neue Funktionen schnell zu veröffentlichen, Ausfallzeiten während der Anwendungsbereitstellung zu vermeiden und die Komplexität der Aktualisierung Ihrer Anwendungen zu bewältigen.
- AWS CodePipeline ist ein Service für kontinuierliche Integration und kontinuierliche Bereitstellung, mit dem Sie die für die Freigabe Ihrer Software erforderlichen Schritte entwickeln, visualisieren und automatisieren können. Sie können die verschiedenen Phasen eines Prozesses für die Veröffentlichung von Software schnell modellieren und konfigurieren. Sie können den Code jedes Mal erstellen, testen und bereitstellen, wenn eine Code-Änderung vorgenommen wurde, und zwar nach den von Ihnen definierten Freigabeprozessmodellen.

Im Folgenden sehen Sie an einem Beispiel, wie Sie die Services in der Entwicklertools-Konsole gemeinsam zur Unterstützung der Softwareentwicklung nutzen können.



In diesem Beispiel erstellen Entwickler ein Repository in CodeCommit und verwenden es, um ihren Code zu entwickeln und gemeinsam daran zu arbeiten. Sie erstellen ein Build-Projekt, CodeBuild um ihren Code zu erstellen und zu testen, und verwenden es, CodeDeploy um ihren Code in Test- und Produktionsumgebungen bereitzustellen. Sie möchten schnell iterieren und erstellen daher eine Pipeline, CodePipeline um die Änderungen im CodeCommit Repository zu erkennen. Diese Änderungen werden erstellt, es werden Tests ausgeführt und der erfolgreich erstellte und getestete Code wird auf dem Testserver bereitgestellt. Das Team fügt der Pipeline Testphasen hinzu, um weitere Tests auf dem Staging-Server auszuführen, beispielsweise Integrations- oder Auslastungstests. Nach erfolgreichem Abschluss dieser Tests überprüft ein Teammitglied die Ergebnisse und genehmigt, wenn sie zufrieden sind, die Änderungen manuell für die Produktion. CodePipeline stellt den getesteten und genehmigten Code auf Produktionsinstanzen bereit.

Dies ist nur ein einfaches Beispiel dafür, wie Sie einen oder mehrere der in der Entwicklertools-Konsole verfügbaren Services zur Unterstützung bei der Softwareentwicklung nutzen können. Jeder dieser Services kann Ihren Anforderungen entsprechend angepasst werden. Sie bieten viele Integrationen mit anderen Produkten und Diensten, sowohl in AWS als auch mit anderen Tools von Drittanbietern. Weitere Informationen finden Sie unter den folgenden Themen:

- CodeCommit: Produkt- und Serviceintegrationen

- CodeBuild: CodeBuild Mit Jenkins verwenden
- CodeDeploy: Produkt- und Serviceintegrationen
- CodePipeline: Produkt- und Serviceintegrationen

Verwenden Sie zum ersten Mal?

Wenn Sie einen oder mehrere Services, die in der Entwicklertools-Konsole verfügbar sind, zum ersten Mal verwenden, empfehlen wir Ihnen, zunächst die folgenden Themen zu lesen:

- Erste Schritte mit CodeCommit
- Erste Schritte mit CodeBuild, Concepts
- Erste Schritte mit CodeDeploy, Hauptkomponenten
- Erste Schritte mit CodePipeline, Concepts

Funktionen der Entwicklertools-Konsole

Die Entwicklertools-Konsole umfasst die folgenden Funktionen:

- Die Developer Tools-Konsole enthält eine Benachrichtigungs-Manager-Funktion, mit der Sie Ereignisse in AWS CodeBuild, AWS CodeCommit AWS CodeDeploy, und abonnieren können AWS CodePipeline. Diese Funktion verfügt über eine eigene API, AWS CodeStar Benachrichtigungen. Sie können die Benachrichtigungsfunktion verwenden, um Benutzer schnell über Ereignisse in den Repositorys, Build-Projekten, Bereitstellungsanwendungen und Pipelines zu informieren, die für ihre Arbeit am wichtigsten sind. Ein Benachrichtigungsmanager hilft, Benutzer auf Ereignisse in Repositorys, Builds, Bereitstellungen oder Pipelines aufmerksam zu machen, sodass sie schnell Maßnahmen ergreifen, also beispielsweise Änderungen genehmigen oder Fehler beheben können. Weitere Informationen finden Sie unter Was sind Benachrichtigungen?.
- Die Entwicklertools-Konsole enthält eine Verbindungsfunktion, mit der Sie Ihre AWS -Ressourcen mit Quellcode von Drittanbietern verknüpfen können. Diese Funktion hat eine eigene API, AWS CodeConnections. Sie können die Verbindungsfunktion verwenden, um eine autorisierte Verbindung mit einem Drittanbieter einzurichten und die Verbindungsressource mit anderen AWS Diensten zu verwenden. Weitere Informationen finden Sie unter Was sind Verbindungen?.

Was sind Benachrichtigungen?

Die Benachrichtigungsfunktion in der Developer Tools-Konsole ist ein Benachrichtigungsmanager für das Abonnieren von Ereignissen in AWS CodeBuild AWS CodeCommit, AWS CodeDeploy und AWS CodePipeline. Es hat seine eigene API, AWS CodeStar Benachrichtigungen. Sie können die Benachrichtigungsfunktion verwenden, um Benutzer schnell über Ereignisse in den Repositorys, Build-Projekten, Bereitstellungsanwendungen und Pipelines zu informieren, die für ihre Arbeit am wichtigsten sind. Ein Benachrichtigungsmanager hilft, Benutzer auf Ereignisse in Repositorys, Builds, Bereitstellungen oder Pipelines aufmerksam zu machen, sodass sie schnell Maßnahmen ergreifen, also beispielsweise Änderungen genehmigen oder Fehler beheben können.

Wofür kann ich Benachrichtigungen verwenden?

Sie können die Benachrichtigungsfunktion verwenden, um Benachrichtigungsregeln zu erstellen und zu verwalten, mit denen Sie Benutzer über wichtige Änderungen an ihren Ressourcen informieren können. Hierzu gehören u. a.:

- Bauen Sie Erfolge und Misserfolge in CodeBuild Build-Projekten auf.
- Erfolge und Misserfolge bei der Bereitstellung CodeDeploy von Anwendungen.
- Erstellung und Aktualisierungen von Pull-Anforderungen, einschließlich Kommentaren zu Code, in CodeCommit-Repositorys.
- Manuelle Genehmigungsstatus und Pipeline laufen in CodePipeline.

Sie können Benachrichtigungen so einrichten, dass sie an die E-Mail-Adressen der Benutzer gesendet werden, die ein Amazon-SNS-Thema abonniert haben. Sie können diese Funktion auch in Verbindung mit AWS -Chatbot verwenden und Benachrichtigungen an Slack- oder Microsoft Teams-Kanäle oder Amazon-Chime-Chatrooms senden lassen.

Wie funktionieren Benachrichtigungen?

Wenn Sie eine Benachrichtigungsregel für eine unterstützte Ressource wie ein Repository, ein Build-Projekt, eine Anwendung oder eine Pipeline konfigurieren, erstellt die Benachrichtigungsfunktion eine EventBridge Amazon-Regel, die die von Ihnen angegebenen Ereignisse überwacht. Wenn ein Ereignis dieses Typs eintritt, sendet die Benachrichtigungsregel Benachrichtigungen an die Amazon–SNS-Themen, die als Ziele für diese Regel angegeben sind. Abonnenten dieser Ziele erhalten Benachrichtigungen über diese Ereignisse.

Wie kann ich in die Verwendung von Benachrichtigungen einsteigen?

Hier finden Sie einige nützliche Themen für den Einstieg:

- Erfahren Sie mehr über die Konzepte von Benachrichtigungen.
- Richten Sie die benötigten Ressourcen ein, um mit Benachrichtigungen zu arbeiten.
- Beginnen Sie mit Ihren ersten Benachrichtigungsregeln, und erhalten Sie Ihre ersten Benachrichtigungen.

Benachrichtigungskonzepte

Das Einrichten und Verwenden von Benachrichtigungen ist einfacher, wenn Sie die Konzepte und Begriffe verstehen. Im Folgenden finden Sie einige Konzepte, die Sie kennen sollten, wenn Sie Benachrichtigungen verwenden.

Themen

- Benachrichtigungen
- Benachrichtigungsregeln
- --Ereignisse
- Detailtypen
- Targets (Ziele)
- Benachrichtigungen und AWS CodeStar Benachrichtigungen
- Ereignisse für Benachrichtigungsregeln für Repositorys
- Ereignisse für Benachrichtigungsregeln für Build-Projekte
- Ereignisse für Benachrichtigungsregeln für Bereitstellungsanwendungen
- Ereignisse für Benachrichtigungsregeln für Pipelines

Benachrichtigungen

Eine Benachrichtigung ist eine Nachricht, die Informationen zu Ereignissen enthält, die in den von Ihnen und Ihren Entwicklern verwendeten Ressourcen auftreten. Sie können Benachrichtigungen einrichten, damit Benutzer einer Ressource, wie z. B. eines Build-Projekts, eines Repositorys, einer Bereitstellungsanwendung oder einer Pipeline, E-Mail-Nachrichten zu den Ereignistypen erhalten, die Sie entsprechend der von Ihnen erstellten Benachrichtigungsregel angeben.

Benachrichtigungen für AWS CodeCommit können mithilfe von Sitzungs-Tags Informationen zur Benutzeridentität enthalten, z. B. einen Anzeigenamen oder eine E-Mail-Adresse. CodeCommit unterstützt die Verwendung von Sitzungs-Tags. Dabei handelt es sich um Schlüssel-Wert-Paarattribute, die Sie übergeben, wenn Sie eine IAM-Rolle übernehmen, temporäre Anmeldeinformationen verwenden oder einen Benutzer in () verbinden. AWS Security Token Service AWS STS Sie können einem IAM-Benutzer auch Tags zuordnen. CodeCommit schließt die Werte für `displayName` und `emailAddress` im Inhalt von Benachrichtigungen ein, sofern diese Tags vorhanden sind. Weitere Informationen finden Sie unter Verwenden von Tags zur Bereitstellung zusätzlicher Identitätsinformationen unter CodeCommit.

> **Important**
>
> Benachrichtigungen enthalten projektspezifische Informationen wie Build-Status, Bereitstellungsstatus, Codezeilen mit Kommentaren und Pipeline-Genehmigungen. Der Inhalt einer Benachrichtigung kann sich ändern, wenn neue Funktionen hinzugefügt werden. Sie sollten eine bewährte Methode für die Sicherheit anwenden und die Ziele der Benachrichtigungsregeln sowie die Amazon–SNS-Themenabonnenten regelmäßig überprüfen. Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

Benachrichtigungsregeln

Eine Benachrichtigungsregel ist eine AWS Ressource, die Sie erstellen, um anzugeben, wann und wohin Benachrichtigungen gesendet werden. Sie definiert Folgendes:

- Die Bedingungen, unter denen eine Benachrichtigung erstellt wird. Diese Bedingungen basieren auf Ereignissen, die Sie auswählen, und die spezifisch für den Ressourcentyp sind. Zu den unterstützten Ressourcentypen gehören Build-Projekte in AWS CodeBuild, Deployment-Anwendungen in AWS CodeDeploy, Pipelines in AWS CodePipeline und Repositorys in. AWS CodeCommit
- Die Ziele, an die die Benachrichtigung gesendet wird. Sie können bis zu 10 Ziele für eine Benachrichtigungsregel angeben.

Benachrichtigungsregeln gelten für einzelne Build-Projekte, Bereitstellungsanwendungen, Pipelines und Repositorys. Benachrichtigungsregeln haben sowohl benutzerdefinierte Anzeigenamen als auch Amazon-Ressourcennamen (ARNs). Benachrichtigungsregeln müssen in derselben AWS Region

erstellt werden, in der die Ressource existiert. Wenn sich Ihr Build-Projekt beispielsweise in der Region USA Ost (Ohio) befindet, muss Ihre Benachrichtigungsregel ebenfalls in der Region USA Ost (Ohio) erstellt werden.

Sie können bis zu 10 Benachrichtigungsregeln für eine Ressource definieren.

--Ereignisse

Ein Ereignis ist eine Statusänderung für eine Ressource, die Sie überwachen möchten. Jede Ressource verfügt über eine Liste von Ereignistypen, die Sie auswählen können. Wenn Sie eine Benachrichtigungsregel für eine Ressource einrichten, geben Sie die Ereignisse an, die dazu führen, dass Benachrichtigungen gesendet werden. Wenn du beispielsweise Benachrichtigungen für ein Repository in CodeCommit einrichtest und sowohl für Pull Request als auch für Branches und Tags die Option Erstellt auswählst, wird jedes Mal eine Benachrichtigung gesendet, wenn ein Benutzer in diesem Repository eine Pull-Anfrage, einen Branch oder ein Git-Tag erstellt.

Detailtypen

Beim Erstellen einer Benachrichtigungsregel können Sie die Detailgenauigkeit (Level of Detail) oder den Detailtyp (Detail Type) wählen, die/der in den Benachrichtigungen enthalten ist (Full (Vollständig) oder Basic (Einfach)). Die Einstellung Full (Vollständig) (die Standardeinstellung) umfasst alle für das Ereignis in der Benachrichtigung verfügbaren Informationen, einschließlich aller erweiterten Informationen, die von Services für bestimmte Ereignisse bereitgestellt werden. Die Einstellung Basic (Einfach) enthält nur eine Teilmenge der verfügbaren Informationen.

In der folgenden Tabelle werden die für bestimmte Ereignistypen verfügbaren erweiterten Informationen aufgeführt und die Unterschiede zwischen den Detailtypen beschrieben.

Service	Ereignis	Full (Vollständig) enthält	Basic (Einfach) enthält nicht
CodeCommit	Kommentare zu Commits Kommentare zu Pull-Anforderungen	alle Ereignisdetails und der Inhalt des Kommentars, einschließlich aller Antworten oder Kommentar-Threads. Enthalten sind auch die Zeilennummer	den Inhalt des Kommentars, die Zeilennummer, Codezeile oder irgendwelche Kommentar-Threads.

Service	Ereignis	Full (Vollständig) enthält	Basic (Einfach) enthält nicht
		und die Codezeile, zu der der Kommentar abgegeben wurde.	
CodeCommit	Pull-Anforderung erstellt	alle Ereignisdetails und die Anzahl der Dateien, die in der Pull-Anforderung in Bezug auf den Ziel-Branch hinzugefügt, geändert oder gelöscht wurden.	eine Liste von Dateien oder Details darüber, ob in der Pull-Anforderung in Bezug auf den Quell-Branch Dateien hinzugefügt, geändert oder gelöscht wurden.
CodePipeline	Manuelle Genehmigung erforderlich	alle Ereignisdetails und benutzerdefinierten Daten (falls konfiguriert). Die Benachrichtigung enthält auch einen Link zur erforderlichen Genehmigung in der Pipeline.	keine benutzerdefinierten Daten oder Links.
CodePipeline	Aktionsausführung fehlgeschlagen Pipelineausführung fehlgeschlagen Phasenausführung fehlgeschlagen	alle Ereignisdetails und den Inhalt der Fehlermeldung für den Ausfall.	keinen Inhalt der Fehlermeldung.

Targets (Ziele)

Ein Ziel ist ein Speicherort für den Empfang von Benachrichtigungen von Benachrichtigungsregeln. Die zulässigen Zieltypen sind Amazon SNS SNS-Themen und AWS Chatbot-Clients, die für Slack- oder Microsoft Teams-Kanäle konfiguriert sind. Jeder Benutzer, der das Zielthema abonniert hat, erhält Benachrichtigungen über die von Ihnen in der Benachrichtigungsregel angegebenen Ereignisse.

Wenn Sie die Reichweite von Benachrichtigungen vergrößern möchten, können Sie die Integration zwischen Benachrichtigungen und AWS Chatbot manuell konfigurieren, sodass Benachrichtigungen an Amazon Chime-Chatrooms gesendet werden. Anschließend können Sie das Amazon SNS SNS-Thema, das für diesen AWS Chatbot-Client konfiguriert ist, als Ziel für die Benachrichtigungsregel auswählen. Weitere Informationen finden Sie unter Um Benachrichtigungen mit AWS Chatbot und Amazon Chime zu integrieren.

Wenn Sie einen AWS Chatbot-Client als Ziel verwenden möchten, müssen Sie diesen Client zuerst in AWS Chatbot erstellen. Wenn Sie einen AWS Chatbot-Client als Ziel für eine Benachrichtigungsregel auswählen, wird für diesen AWS Chatbot-Client ein Amazon SNS SNS-Thema mit allen Richtlinien konfiguriert, die für das Senden von Benachrichtigungen an den Slack- oder Microsoft Teams-Kanal erforderlich sind. Sie müssen keine vorhandenen Amazon SNS SNS-Themen für den AWS Chatbot-Client konfigurieren.

Sie können beim Erstellen einer Benachrichtigungsregel ein Amazon-SNS-Thema als Ziel erstellen (empfohlen). Sie können auch ein vorhandenes Amazon SNS SNS-Thema in derselben AWS Region wie die Benachrichtigungsregel auswählen, müssen es jedoch mit der erforderlichen Richtlinie konfigurieren. Das Amazon SNS SNS-Thema, das Sie für ein Ziel verwenden, muss in Ihrem AWS Konto enthalten sein. Es muss sich außerdem in derselben AWS Region befinden wie die Benachrichtigungsregel und die AWS Ressource, für die die Regel erstellt wurde.

Wenn Sie beispielsweise eine Benachrichtigungsregel für ein Repository in der Region USA Ost (Ohio) erstellen, muss das Amazon-SNS-Thema auch in dieser Region vorhanden sein. Wenn Sie ein Amazon-SNS-Thema im Rahmen des Erstellens einer Benachrichtigungsregel erstellen, wird das Thema mit der erforderlichen Richtlinie konfiguriert, um die Veröffentlichung von Ereignissen für das Thema zu ermöglichen. Dies ist die beste Methode, um mit Zielen und Benachrichtigungsregeln zu arbeiten. Wenn Sie ein bereits vorhandenes Thema verwenden oder ein Thema manuell erstellen möchten, müssen Sie es mit den erforderlichen Berechtigungen konfigurieren. Erst dann erhalten Benutzer Benachrichtigungen. Weitere Informationen finden Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

Benachrichtigungen und AWS CodeStar Benachrichtigungen

Benachrichtigungen sind zwar eine Funktion der Developer Tools-Konsole, verfügen aber über eine eigene API, AWS CodeStar Benachrichtigungen. Darüber hinaus verfügt sie über einen eigenen AWS -Ressourcentyp (Benachrichtigungsregeln), Berechtigungen und Ereignisse. Ereignisse für Benachrichtigungsregeln werden in AWS CloudTrail protokolliert. API-Aktionen können über IAM-Richtlinien erlaubt oder verweigert werden.

Ereignisse für Benachrichtigungsregeln für Repositorys

Kategorie	--Ereignisse	Ereignis IDs

Kommentare	Auf Commits
	
	Auf Pull-Anforderungen
		`codecommit-repository-comments-on-commits`
		
		`codecommit-repository-comments-on-pull-requests`

Kategorie	--Ereignisse	Ereignis IDs

Genehmigungen	Status geändert
	
	Regelüberschreibung
		`codecommit-repository-approvals-status-changed`
		
		`codecommit-repository-approvals-rule-override`

Pull-Anforderung	Erstellt
	
	Quelle aktualisiert
	
	Status geändert
	
	Zusammengeführt
		`codecommit-repository-pull-request-created`
		
		`codecommit-repository-pull-request-source-updated`
		
		`codecommit-repository-pull-request-status-changed`
		
		`codecommit-repository-pull-request-merged`

Zweige und Tags	Erstellt
	
	Gelöscht
	
	Aktualisiert
		`codecommit-repository-branches-and-tags-created`
		
		`codecommit-repository-branches-and-tags-deleted`
		`codecommit-repository-branches-and-tags-updated`

Ereignisse für Benachrichtigungsregeln für Build-Projekte

Kategorie	--Ereignisse	Ereignis IDs

Status des Builds	Fehlgeschlagen
	Erfolgreich
	In Bearbeitung
	Angehalten
		`codebuild-project-build-state-failed`
		`codebuild-project-build-state-succeeded`
		`codebuild-project-build-state-in-progress`
		`codebuild-project-build-state-stopped`

Build-Phase	Fehler Herzlichen Glückwunsch	`codebuild-project-build-phase-failure` `codebuild-project-build-phase-success`

Ereignisse für Benachrichtigungsregeln für Bereitstellungsanwendungen

Kategorie	--Ereignisse	Ereignis IDs

Bereitstellung	Fehlgeschlagen
	Erfolgreich
	Started
		`codedeploy-application-deployment-failed`
		`codedeploy-application-deployment-succeeded`
		`codedeploy-application-deployment-started`

Ereignisse für Benachrichtigungsregeln für Pipelines

Kategorie	--Ereignisse	Ereignis IDs

Aktionsausführung	Erfolgreich
	Fehlgeschlagen
	Canceled
	Started
		`codepipeline-pipeline-action-execution-succeeded`
		`codepipeline-pipeline-action-execution-failed`
		`codepipeline-pipeline-action-execution-canceled`
		`codepipeline-pipeline-action-execution-started`

Phasenausführung	Started
	Erfolgreich
	Fortgesetzt
	Canceled
	Fehlgeschlagen
		`codepipeline-pipeline-stage-execution-started`
		`codepipeline-pipeline-stage-execution-succeeded`
		`codepipeline-pipeline-stage-execution-resumed`
		`codepipeline-pipeline-stage-execution-canceled`
		`codepipeline-pipeline-stage-execution-failed`

Pipeline-Ausführung	Fehlgeschlagen
	Canceled
	Started
	Fortgesetzt
	Erfolgreich
		`codepipeline-pipeline-pipeline-execution-failed`
		`codepipeline-pipeline-pipeline-execution-canceled`
		`codepipeline-pipeline-pipeline-execution-started`

Kategorie	--Ereignisse	Ereignis IDs

	Ersetzt	`codepipeline-pipeline-pipeline-execution-resumed`
		
		`codepipeline-pipeline-pipeline-execution-succeeded`
		
		`codepipeline-pipeline-pipeline-execution-superseded`

Manuelle Genehmigung	Fehlgeschlagen
	
	Erforderlich
	
	Erfolgreich
		`codepipeline-pipeline-manual-approval-failed`
		
		`codepipeline-pipeline-manual-approval-needed`
		`codepipeline-pipeline-manual-approval-succeeded`

Einrichtung

Wenn Sie eine verwaltete Richtlinie für AWS CodeBuild, AWS CodeCommit AWS CodeDeploy, oder AWS CodePipeline auf Ihren IAM-Benutzer oder Ihre IAM-Rolle angewendet haben, verfügen Sie über die erforderlichen Berechtigungen, um mit Benachrichtigungen innerhalb der Einschränkungen der Rollen und Berechtigungen zu arbeiten, die in der Richtlinie vorgesehen sind. Benutzer beispielsweise, für die verwaltete Richtlinie `AWSCodeBuildAdminAccess`, `AWSCodeCommitFullAccess`, `AWSCodeDeployFullAccess` oder `AWSCodePipeline_FullAccess` angewendet wurde, verfügen über vollständigen Administratorzugriff auf Benachrichtigungen.

Weitere Informationen hierzu und auch zu Beispielrichtlinien finden Sie unter Identitätsbasierte Richtlinien.

Wenn Sie eine dieser Richtlinien auf Ihren IAM-Benutzer oder Ihre IAM-Rolle und ein Build-Projekt, ein Repository CodeBuild, eine Bereitstellungsanwendung oder eine Pipeline angewendet haben CodeDeploy, sind Sie bereit CodePipeline, Ihre erste Benachrichtigungsregel zu erstellen. CodeCommit Fahren Sie fort mit Einstieg in die Verwendung von Benachrichtigungen. Wenn dies nicht der Fall ist, beachten Sie die folgenden Themen:

- CodeBuild: Erste Schritte mit CodeBuild
- CodeCommit: Erste Schritte mit CodeCommit
- CodeDeploy: Tutorials
- CodePipeline: Erste Schritte mit CodePipeline

Wenn Sie Administratorberechtigungen für Benachrichtigungen für IAM-Benutzer, -Gruppen oder -Rollen selbst verwalten möchten, führen Sie die in diesem Thema beschriebenen Schritte aus, um die Berechtigungen und Ressourcen einzurichten, die Sie für die Verwendung des Service benötigen.

Wenn Sie zuvor erstellte Amazon-SNS-Themen für Benachrichtigungen verwenden möchten, anstatt speziell Themen für Benachrichtigungen zu erstellen, müssen Sie ein Amazon-SNS-Thema konfigurieren, das als Ziel für eine Benachrichtigungsregel verwendet werden soll, indem Sie eine Richtlinie anwenden, die die Veröffentlichung von Ereignissen in diesem Thema zulässt.

> **Note**
>
> Um die folgenden Verfahren ausführen zu können, müssen Sie mit einem Konto angemeldet sein, das über Administratorberechtigungen verfügt. Weitere Informationen finden Sie unter Erstellen Ihres ersten IAM-Administratorbenutzers und Ihrer ersten IAM-Administratorgruppe.

Themen

- Erstellen und Anwenden einer Richtlinie für Administratorzugriff auf Benachrichtigungen
- Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen
- Abonnieren von Amazon-SNS-Themen, die als Ziele fungieren, für Benutzer

Erstellen und Anwenden einer Richtlinie für Administratorzugriff auf Benachrichtigungen

Sie können Benachrichtigungen verwalten, indem Sie sich mit einem IAM-Benutzer anmelden oder eine Rolle verwenden, die über Zugriffsberechtigungen für den Dienst und die Dienste (AWS CodeBuild, AWS CodeCommit AWS CodeDeploy, oder AWS CodePipeline) verfügt, für die Sie Benachrichtigungen erstellen möchten. Sie können auch eigene Richtlinien erstellen und auf Benutzer oder Gruppen anwenden.

Das folgende Verfahren zeigt, wie Sie eine IAM-Gruppe mit Berechtigungen zum Verwalten von Benachrichtigungen und Hinzufügen von IAM-Benutzern konfigurieren. Wenn Sie keine Gruppe einrichten möchten, können Sie diese Richtlinie direkt auf IAM-Benutzer oder auf eine IAM-Rolle anwenden, die von Benutzern übernommen werden kann. Sie können auch die verwalteten Richtlinien für CodeBuild,, oder verwenden CodeCommit CodeDeploy CodePipeline, die je nach Geltungsbereich der Richtlinie einen richtliniengerechten Zugriff auf Benachrichtigungsfunktionen beinhalten.

Geben Sie für die folgende Richtlinie einen Namen (z. B. `AWSCodeStarNotificationsFullAccess`) sowie eine optionale Beschreibung für diese Richtlinie ein. Anhand der Beschreibung ist zu erkennen, welchen Zweck die Richtlinie verfolgt (z. B. **`This policy provides full access to AWS CodeStar Notifications.`**

So verwenden Sie den JSON-Richtlinieneditor zum Erstellen einer Richtlinie

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die IAM-Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich auf der linken Seite Policies (Richtlinien).

   Wenn Sie zum ersten Mal Policies (Richtlinien) auswählen, erscheint die Seite Welcome to Managed Policies (Willkommen bei verwalteten Richtlinien). Wählen Sie Get Started.
3. Wählen Sie oben auf der Seite Create policy (Richtlinie erstellen) aus.
4. Wählen Sie im Bereich Policy editor (Richtlinien-Editor) die Option JSON aus.
5. Geben Sie folgendes JSON-Richtliniendokument ein:

```
{
   "Version": "2012-10-17",
   "Statement": [
     {
       "Sid": "AWSCodeStarNotificationsFullAccess",
       "Effect": "Allow",
       "Action": [
           "codestar-notifications:CreateNotificationRule",
           "codestar-notifications:DeleteNotificationRule",
           "codestar-notifications:DescribeNotificationRule",
           "codestar-notifications:ListNotificationRules",
           "codestar-notifications:UpdateNotificationRule",
           "codestar-notifications:Subscribe",
           "codestar-notifications:Unsubscribe",
           "codestar-notifications:DeleteTarget",
```

```
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource",
            "codestar-notifications:TagResource",
            "codestar-notifications:UntagResource"
        ],
        "Resource": "*"
    }
  ]
}
```

6. Wählen Sie Weiter.

> **Note**
>
> Sie können jederzeit zwischen den Editoroptionen Visual und JSON wechseln. Wenn Sie jedoch Änderungen vornehmen oder im Visual-Editor Weiter wählen, strukturiert IAM Ihre Richtlinie möglicherweise um, um sie für den visuellen Editor zu optimieren. Weitere Informationen finden Sie unter Richtlinienrestrukturierung im IAM-Benutzerhandbuch.

7. Geben Sie auf der Seite Prüfen und erstellen unter Richtlinienname einen Namen und unter Beschreibung (optional) eine Beschreibung für die Richtlinie ein, die Sie erstellen. Überprüfen Sie Permissions defined in this policy (In dieser Richtlinie definierte Berechtigungen), um die Berechtigungen einzusehen, die von Ihrer Richtlinie gewährt werden.
8. Wählen Sie Create policy (Richtlinie erstellen) aus, um Ihre neue Richtlinie zu speichern.

Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen

Die einfachste Möglichkeit, Benachrichtigungen einzurichten, besteht darin, beim Erstellen einer Benachrichtigungsregel ein Amazon-SNS-Thema zu erstellen. Sie können ein vorhandenes Amazon-SNS-Thema verwenden, wenn es die folgenden Anforderungen erfüllt:

- Sie wurde in derselben Weise erstellt AWS-Region wie die Ressource (Build-Projekt, Bereitstellungsanwendung, Repository oder Pipeline), für die Sie Benachrichtigungsregeln erstellen möchten.
- Es wurde nicht für das Senden von Benachrichtigungen CodeCommit vor dem 5. November 2019 verwendet. Wenn dies der Fall ist, enthält es Richtlinienanweisungen, die diese Funktionalität aktiviert haben. Sie können dieses Thema verwenden, aber Sie müssen die zusätzliche Richtlinie wie im Verfahren angegeben hinzufügen. Sie sollten die vorhandene Richtlinienanweisung nicht

entfernen, wenn ein oder mehrere Repositorys noch für Benachrichtigungen vor dem 5. November 2019 konfiguriert sind.

- Es gibt eine Richtlinie, die es AWS CodeStar Notifications ermöglicht, Benachrichtigungen zu diesem Thema zu veröffentlichen.

So konfigurieren Sie ein Amazon SNS SNS-Thema zur Verwendung als Ziel für AWS CodeStar Benachrichtigungsregeln

1. Melden Sie sich bei https://console.aws.amazon.com/sns/v3/home an AWS Management Console und öffnen Sie die Amazon SNS SNS-Konsole.
2. Wählen Sie in der Navigationsleiste die Option Topics (Themen) und dann das Thema aus, das Sie konfigurieren möchten. Wählen Sie anschließend Edit (Bearbeiten) aus.
3. Klappen Sie Access policy (Zugriffsrichtlinie) aus und wählen Sie danach Advanced (Erweitert) aus.
4. Geben Sie im JSON-Editor die folgende Anweisung für die Richtlinie ein. Geben Sie den Themen-ARN AWS-Region, die AWS-Konto ID und den Themennamen an.

```
 {
      "Sid": "AWSCodeStarNotifications_publish",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "codestar-notifications.amazonaws.com"
        ]
      },
      "Action": "SNS:Publish",
      "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules"
    }
```

Die Richtlinienanweisung sollte so aussehen.

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__default_statement_ID",
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "SNS:GetTopicAttributes",
        "SNS:SetTopicAttributes",
        "SNS:AddPermission",
        "SNS:RemovePermission",
        "SNS:DeleteTopic",
        "SNS:Subscribe",
        "SNS:ListSubscriptionsByTopic",
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules",
      "Condition": {
        "StringEquals": {
          "AWS:SourceOwner": "123456789012"
        }
      }
    },
 {
      "Sid": "AWSCodeStarNotifications_publish",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "codestar-notifications.amazonaws.com"
        ]
      },
      "Action": "SNS:Publish",
      "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules"
    }
  ]
}
```

5. Wählen Sie Änderungen speichern.
6. Wenn Sie ein AWS KMS-verschlüsseltes Amazon SNS SNS-Thema zum Senden von Benachrichtigungen verwenden möchten, müssen Sie auch die Kompatibilität zwischen der Ereignisquelle (AWS CodeStar Benachrichtigungen) und dem verschlüsselten Thema aktivieren, indem Sie die folgende Erklärung zur Richtlinie von hinzufügen. AWS KMS key Ersetzen Sie

AWS-Region (in diesem Beispiel us-east-2) durch den Ort, AWS-Region an dem der Schlüssel erstellt wurde.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "codestar-notifications.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey*",
                "kms:Decrypt"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "sns.us-east-2.amazonaws.com"
                }
            }
        }
    ]
}
```

Weitere Informationen finden Sie unter Verschlüsselung im Ruhezustand und Verwenden von Richtlinienbedingungen mit AWS KMS im AWS Key Management Service -Entwicklerhandbuch.

Abonnieren von Amazon-SNS-Themen, die als Ziele fungieren, für Benutzer

Bevor Benutzer Benachrichtigungen erhalten können, müssen sie das Amazon-SNS-Thema abonnieren, das Ziel der Benachrichtigungsregel ist. Wenn das Abonnement für die Benutzer über die E-Mail-Adresse erfolgt, müssen sie ihr Abonnement bestätigen, bevor sie Benachrichtigungen erhalten. Informationen zum Senden von Benachrichtigungen an Benutzer in Slack- oder Microsoft Teams-Kanälen oder Amazon-Chime-Chatrooms finden Sie unter Konfigurieren der Integration zwischen Benachrichtigungen und AWS Chatbot.

Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren

1. Melden Sie sich bei https://console.aws.amazon.com/sns/v3/home an AWS Management Console und öffnen Sie die Amazon SNS SNS-Konsole.
2. Wählen Sie im Navigationsbereich Topics (Themen) und anschließend das Thema aus, das Sie für die Benutzer abonnieren möchten.
3. Wählen Sie unter Subscriptions (Abonnements) die Option Create subscription (Abonnement erstellen) aus.
4. Wählen Sie unter Protocol (Protokoll) die Option Email (E-Mail) aus. Geben Sie unter Endpoint (Endpunkt) die E-Mail-Adresse ein und wählen Sie dann Create subscription (Abonnement erstellen) aus.

Einstieg in die Verwendung von Benachrichtigungen

Der einfachste Weg, um mit der Verwendung von Benachrichtigungen zu beginnen, besteht darin, eine Benachrichtigungsregel für eine(s) Ihrer Build-Projekte, Bereitstellungsanwendungen, Pipelines oder Repositorys einzurichten.

> **Note**
>
> Wenn Sie zum ersten Mal eine Benachrichtigungsregel erstellen, wird in Ihrem Konto eine serviceverknüpfte Rolle erstellt. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Benachrichtigungen AWS CodeStar .

Themen

- Voraussetzungen
- Erstellen einer Benachrichtigungsregel für ein Repository
- Erstellen einer Benachrichtigungsregel für ein Build-Projekt
- Erstellen einer Benachrichtigungsregel für eine Bereitstellungsanwendung
- Erstellen einer Benachrichtigungsregel für eine Pipeline

Voraussetzungen

Führen Sie die Schritte unter Einrichtung aus. Außerdem benötigen Sie eine Ressource, für die Sie eine Benachrichtigungsregel erstellen.

- Erstellen Sie ein Build-Projekt in CodeBuild oder verwenden Sie ein vorhandenes.
- Erstellen Sie eine Anwendung oder verwenden Sie eine vorhandene Bereitstellungsanwendung.
- Erstellen Sie eine Pipeline in CodePipeline oder verwenden Sie eine bestehende.
- Erstellen Sie ein AWS CodeCommit Repository oder verwenden Sie ein vorhandenes.

Erstellen einer Benachrichtigungsregel für ein Repository

Sie können Benachrichtigungsregeln erstellen, um Benachrichtigungen zu Repository-Ereignissen zu senden, die für Sie wichtig sind. Die folgenden Schritte zeigen, wie Sie eine Benachrichtigungsregel für ein einzelnes Repository-Ereignis einrichten. Diese Schritte wurden unter der Annahme geschrieben, dass Sie in Ihrem AWS Konto ein Repository konfiguriert haben.

> **Important**
>
> Wenn Sie Benachrichtigungen CodeCommit vor dem 5. November 2019 eingerichtet haben, enthalten die Amazon SNS SNS-Themen, die für diese Benachrichtigungen verwendet werden, eine Richtlinie, die es ermöglicht, auf Amazon CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

1. Öffnen Sie die CodeCommit Konsole unter https://console.aws.amazon.com/codecommit/.
2. Wählen Sie ein Repository aus der Liste aus und öffnen Sie es.
3. Wählen Sie Notify (Benachrichtigung) und dann Create notification rule (Benachrichtigungsregel erstellen) aus. Sie können auch Settings (Einstellungen), Notifications (Benachrichtigungen) und dann Create notification rule (Benachrichtigungsregel erstellen) auswählen.
4. Geben Sie unter Notification name (Benachrichtigungsname) einen Namen für die Regel ein.
5. Wählen Sie unter Detailtyp die Option Basic aus, wenn Sie möchten, dass nur die Informationen, die Amazon zur Verfügung gestellt wurden, in der Benachrichtigung EventBridge enthalten sind. Wählen Sie Vollständig, wenn Sie Informationen, die Amazon zur Verfügung gestellt

wurden, EventBridge und Informationen, die möglicherweise vom Ressourcenservice oder vom Notification Manager bereitgestellt wurden, einbeziehen möchten.

Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

6. Wählen Sie im Bereich Events that trigger notifications (Ereignisse, die Benachrichtigungen auslösen) unter Branches and tags (Zweige und Tags) die Option Created (Erstellt)aus.
7. Wählen Sie unter Targets (Ziele) die Option Create SNS topic (SNS-Thema erstellen) aus.

> **Note**
>
> Wenn Sie das Thema im Rahmen der Erstellung der Benachrichtigungsregel erstellen, wird die Richtlinie, die das Veröffentlichen von Ereignissen zu dem Thema ermöglicht CodeCommit , für Sie angewendet. Durch die Verwendung eines Themas, das für Benachrichtigungsregeln erstellt wurde, kann sichergestellt werden, dass Sie das Thema nur für die Benutzer abonnieren, die Benachrichtigungen zu diesem Repository erhalten sollen.

Geben Sie hinter dem Präfix codestar-notifications- einen Namen für das Thema ein und wählen Sie anschließend Submit (Absenden) aus.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter

Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

8. Wählen Sie Submit (Absenden) und überprüfen Sie die Benachrichtigungsregel.
9. Abonnieren Sie das soeben erstellte Amazon-SNS-Thema mit Ihrer E-Mail-Adresse. Weitere Informationen finden Sie unter Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren.
10. Navigieren Sie zu Ihrem Repository und erstellen Sie einen Testzweig aus dem Standard-Zweig.
11. Nachdem Sie den Zweig erstellt haben, sendet die Benachrichtigungsregel an alle Themenabonnenten eine Benachrichtigung mit Informationen zu diesem Ereignis.

Erstellen einer Benachrichtigungsregel für ein Build-Projekt

Sie können Benachrichtigungsregeln erstellen, um Benachrichtigungen über die Ereignisse in Ihrem Build-Projekt zu senden, die für Sie wichtig sind. Die folgenden Schritte zeigen, wie Sie eine Benachrichtigungsregel für ein einzelnes Build-Projektereignis einrichten. Diese Schritte wurden unter der Annahme geschrieben, dass Sie in Ihrem AWS Konto ein Build-Projekt konfiguriert haben.

1. Öffnen Sie die CodeBuild Konsole unter https://console.aws.amazon.com/codebuild/.
2. Wählen Sie ein Build-Projekt aus der Liste aus und öffnen Sie es.
3. Wählen Sie Notify (Benachrichtigung) und dann Create notification rule (Benachrichtigungsregel erstellen) aus. Sie können auch Settings (Einstellungen) und dann Create notification rule (Benachrichtigungsregel erstellen) auswählen.
4. Geben Sie unter Notification name (Benachrichtigungsname) einen Namen für die Regel ein.
5. Wählen Sie unter Detailtyp die Option Basic aus, wenn Sie möchten, dass nur die Informationen, die Amazon zur Verfügung gestellt wurden, in der Benachrichtigung EventBridge enthalten sind. Wählen Sie Vollständig, wenn Sie Informationen, die Amazon zur Verfügung gestellt wurden, EventBridge und Informationen, die möglicherweise vom Ressourcenservice oder vom Notification Manager bereitgestellt wurden, einbeziehen möchten.

   Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.
6. Wählen Sie im Bereich Events that trigger notifications (Ereignisse, die Benachrichtigungen auslösen) unter Build phase (Build-Phase) die Option Success (Erfolg) aus.

7. Wählen Sie unter Targets (Ziele) die Option Create SNS topic (SNS-Thema erstellen) aus.

> **Note**
>
> Wenn Sie das Thema im Rahmen der Erstellung der Benachrichtigungsregel erstellen, wird die Richtlinie, die das Veröffentlichen von Ereignissen zu dem Thema ermöglicht CodeBuild , für Sie angewendet. Durch die Verwendung eines Themas, das für Benachrichtigungsregeln erstellt wurde, kann sichergestellt werden, dass Sie das Thema nur für die Benutzer abonnieren, die Benachrichtigungen zu diesem Build-Projekt erhalten sollen.

Geben Sie hinter dem Präfix codestar-notifications- einen Namen für das Thema ein und wählen Sie anschließend Submit (Absenden) aus.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

8. Wählen Sie Submit (Absenden) und überprüfen Sie die Benachrichtigungsregel.
9. Abonnieren Sie das soeben erstellte Amazon-SNS-Thema mit Ihrer E-Mail-Adresse. Weitere Informationen finden Sie unter Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren.
10. Navigieren Sie zu Ihrem Build-Projekt, und starten Sie einen Build.

11. Nach erfolgreichem Abschluss der Build-Phase sendet die Benachrichtigungsregel an alle Themenabonnenten eine Benachrichtigung mit Informationen zu diesem Ereignis.

Erstellen einer Benachrichtigungsregel für eine Bereitstellungsanwendung

Sie können Benachrichtigungsregeln erstellen, um Benachrichtigungen zu den Ereignissen in Ihre Bereitstellungsanwendung zu senden, die für Sie wichtig sind. Die folgenden Schritte zeigen, wie Sie eine Benachrichtigungsregel für ein einzelnes Build-Projektereignis einrichten. Bei diesen Schritten wird davon ausgegangen, dass Sie eine Bereitstellungsanwendung in Ihrem AWS -Konto konfiguriert haben.

1. Öffnen Sie die CodeDeploy Konsole unter https://console.aws.amazon.com/codedeploy/.
2. Wählen Sie eine Anwendung aus der Liste aus und öffnen Sie sie.
3. Wählen Sie Notify (Benachrichtigung) und dann Create notification rule (Benachrichtigungsregel erstellen) aus. Sie können auch Settings (Einstellungen) und dann Create notification rule (Benachrichtigungsregel erstellen) auswählen.
4. Geben Sie unter Notification name (Benachrichtigungsname) einen Namen für die Regel ein.
5. Wählen Sie unter Detailtyp die Option Basic aus, wenn Sie möchten, dass nur die Informationen, die Amazon zur Verfügung gestellt wurden, in der Benachrichtigung EventBridge enthalten sind. Wählen Sie Vollständig, wenn Sie Informationen, die Amazon zur Verfügung gestellt wurden, EventBridge und Informationen, die möglicherweise vom Ressourcenservice oder vom Notification Manager bereitgestellt wurden, einbeziehen möchten.

   Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

6. Wählen Sie im Bereich Events that trigger notifications (Ereignisse, die Benachrichtigungen auslösen) unter Deployment (Bereitstellung) die Option Succeeded (Erfolgreich) aus.
7. Wählen Sie unter Targets (Ziele) die Option Create SNS topic (SNS-Thema erstellen) aus.

> **Note**
>
> Wenn Sie das Thema im Rahmen der Erstellung der Benachrichtigungsregel erstellen, wird die Richtlinie, die das Veröffentlichen von Ereignissen zu dem Thema ermöglicht CodeDeploy , für Sie angewendet. Durch die Verwendung eines Themas, das für Benachrichtigungsregeln erstellt wurde, kann sichergestellt werden, dass Sie

das Thema nur für die Benutzer abonnieren, die Benachrichtigungen zu dieser Bereitstellungsanwendung erhalten sollen.

Geben Sie hinter dem Präfix codestar-notifications- einen Namen für das Thema ein und wählen Sie anschließend Submit (Absenden) aus.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

8. Wählen Sie Submit (Absenden) und überprüfen Sie die Benachrichtigungsregel.
9. Abonnieren Sie das soeben erstellte Amazon-SNS-Thema mit Ihrer E-Mail-Adresse. Weitere Informationen finden Sie unter Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren.
10. Navigieren Sie zu Ihrer Bereitstellungsanwendung und starten Sie eine Bereitstellung.
11. Nach erfolgreicher Bereitstellung sendet die Benachrichtigungsregel an alle Themenabonnenten eine Benachrichtigung mit Informationen zu dem Ereignis.

Erstellen einer Benachrichtigungsregel für eine Pipeline

Sie können Benachrichtigungsregeln erstellen, um Benachrichtigungen über die Ereignisse in Ihrer Pipeline zu senden, die für Sie wichtig sind. Die folgenden Schritte zeigen, wie Sie eine

Benachrichtigungsregel für ein einzelnes Pipeline-Ereignis einrichten. Diese Schritte wurden unter der Annahme geschrieben, dass Sie in Ihrem AWS Konto eine Pipeline konfiguriert haben.

1. Öffnen Sie die CodePipeline Konsole unter https://console.aws.amazon.com/codepipeline/.
2. Wählen Sie eine Pipeline aus der Liste aus und öffnen Sie sie.
3. Wählen Sie Notify (Benachrichtigung) und dann Create notification rule (Benachrichtigungsregel erstellen) aus. Sie können auch Settings (Einstellungen) und dann Create notification rule (Benachrichtigungsregel erstellen) auswählen.
4. Geben Sie unter Notification name (Benachrichtigungsname) einen Namen für die Regel ein.
5. Wählen Sie unter Detailtyp die Option Basic aus, wenn Sie möchten, dass nur die Informationen, die Amazon zur Verfügung gestellt wurden, in der Benachrichtigung EventBridge enthalten sind. Wählen Sie Vollständig, wenn Sie Informationen, die Amazon zur Verfügung gestellt wurden, EventBridge und Informationen, die möglicherweise vom Ressourcenservice oder vom Notification Manager bereitgestellt wurden, einbeziehen möchten.

   Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.
6. Wählen Sie im Bereich Events that trigger notifications (Ereignisse, die Benachrichtigungen auslösen) unter Action execution (Aktionsausführung) die Option Started (Gestartet) aus.
7. Wählen Sie unter Targets (Ziele) die Option Create SNS topic (SNS-Thema erstellen) aus.

   > **Note**
   >
   > Wenn Sie das Thema im Rahmen der Erstellung der Benachrichtigungsregel erstellen, wird die Richtlinie, die das Veröffentlichen von Ereignissen zu dem Thema ermöglicht CodePipeline , für Sie angewendet. Durch die Verwendung eines Themas, das für Benachrichtigungsregeln erstellt wurde, kann sichergestellt werden, dass Sie das Thema nur für die Benutzer abonnieren, die Benachrichtigungen zu dieser Pipeline erhalten sollen.

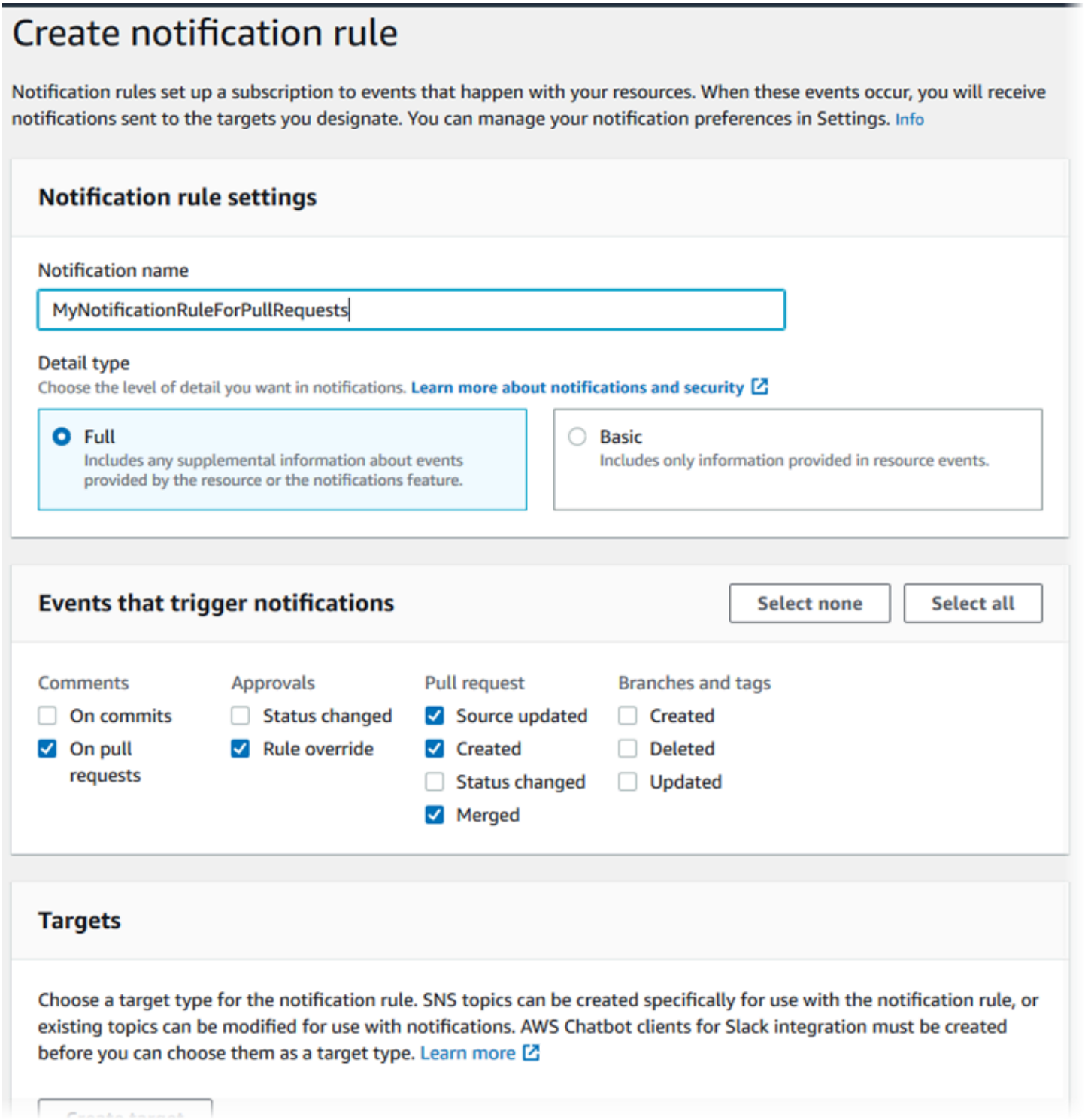
   Geben Sie hinter dem Präfix codestar-notifications- einen Namen für das Thema ein und wählen Sie anschließend Submit (Absenden) aus.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

8. Wählen Sie Submit (Absenden) und überprüfen Sie die Benachrichtigungsregel.
9. Abonnieren Sie das soeben erstellte Amazon-SNS-Thema mit Ihrer E-Mail-Adresse. Weitere Informationen finden Sie unter Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren.
10. Navigieren Sie zu Ihrer Pipeline, und wählen Sie dann Release change (Versionswechsel)aus.
11. Wenn die Aktion gestartet wird, sendet die Benachrichtigungsregel an alle Themenabonnenten eine Benachrichtigung mit Informationen zu dem Ereignis.

Arbeiten mit Benachrichtigungsregeln

In einer Benachrichtigungsregel konfigurieren Sie die Ereignisse, zu denen Benutzer Benachrichtigungen erhalten sollen, und geben die Ziele an, die diese Benachrichtigungen erhalten sollen. Sie können Benachrichtigungen direkt über Amazon SNS oder über AWS Chatbot-Clients, die für Slack- oder Microsoft Teams-Kanäle konfiguriert sind, an Benutzer senden. Wenn Sie die Reichweite von Benachrichtigungen vergrößern möchten, können Sie die Integration zwischen Benachrichtigungen und AWS Chatbot manuell konfigurieren, sodass Benachrichtigungen an Amazon Chime-Chatrooms gesendet werden. Weitere Informationen erhalten Sie unter Targets (Ziele) und Um Benachrichtigungen mit AWS Chatbot und Amazon Chime zu integrieren.



Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um Benachrichtigungsregeln zu erstellen und zu verwalten.

Themen

- Erstellen einer Benachrichtigungsregel

- Benachrichtigungsregeln anzeigen
- Bearbeiten einer Benachrichtigungsregel
- Aktivieren oder Deaktivieren von Benachrichtigungen für eine Benachrichtigungsregel
- Löschen einer Benachrichtigungsregel

Erstellen einer Benachrichtigungsregel

Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um Benachrichtigungsregeln zu erstellen. Sie können beim Erstellen einer Regel ein Amazon-SNS-Thema als Ziel der Benachrichtigungsregel erstellen (empfohlen). Wenn Sie einen AWS Chatbot-Client als Ziel verwenden möchten, müssen Sie diesen Client erstellen, bevor Sie die Regel erstellen können. Weitere Informationen finden Sie unter Einen AWS Chatbot-Client für einen Slack-Kanal konfigurieren.

So erstellen Sie eine Benachrichtigungsregel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Verwenden Sie die Navigationsleiste, um zu der Ressource zu navigieren.
   - Wählen Sie für CodeBuild Build, dann Build projects und wählen Sie ein Build-Projekt aus.
   - Wählen Sie für „Quelle“ CodeCommit, dann „Repositories“ und anschließend ein Repository aus.
   - Wählen Sie für CodeDeploy Anwendungen und wählen Sie eine Anwendung aus.
   - Wählen Sie für CodePipeline Pipeline, dann Pipelines und anschließend eine Pipeline aus.
3. Wählen Sie auf der Ressourcenseite Notify (Benachrichtigung) und dann Create notification rule (Benachrichtigungsregel erstellen) aus. Sie können auch die Seite Settings (Einstellungen) für die Ressource aufrufen, zu Notifications (Benachrichtigungen) oder Notification rules (Benachrichtigungsregeln) wechseln und dann Create notification rule (Benachrichtigungsregel erstellen) auswählen.
4. Geben Sie unter Notification name (Benachrichtigungsname) einen Namen für die Regel ein.
5. Wählen Sie unter Detailtyp die Option Basic aus, wenn Sie möchten, dass nur die Informationen, die Amazon zur Verfügung gestellt wurden, in der Benachrichtigung EventBridge enthalten sind. Wählen Sie Vollständig, wenn Sie Informationen, die Amazon zur Verfügung gestellt

wurden, EventBridge und Informationen, die möglicherweise vom Ressourcenservice oder vom Notification Manager bereitgestellt wurden, einbeziehen möchten.

Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

6. Wählen Sie unter Events that trigger notifications (Ereignisse, die Benachrichtigungen auslösen) die Ereignisse aus, für die Sie Benachrichtigungen senden möchten. Informationen zu den Ereignistypen für eine Ressource finden Sie in folgenden Abschnitten:

   - CodeBuild: Ereignisse für Benachrichtigungsregeln für Build-Projekte
   - CodeCommit: Ereignisse für Benachrichtigungsregeln für Repositorys
   - CodeDeploy: Ereignisse für Benachrichtigungsregeln für Bereitstellungsanwendungen
   - CodePipeline: Ereignisse für Benachrichtigungsregeln für Pipelines

7. Führen Sie unter Targets (Ziele) einen der folgenden Schritte aus:

   - Wenn Sie bereits eine Ressource zur Verwendung mit Benachrichtigungen konfiguriert haben, wählen Sie unter Choose target type (Zieltyp auswählen) entweder AWS Chatbot (Slack), AWS Chatbot (Microsoft Teams) oder SNS topic (SNS-Thema) aus. Wählen Sie unter Ziel auswählen den Namen des Clients (für einen in AWS Chatbot konfigurierten Slack- oder Microsoft Teams-Client) oder den Amazon-Ressourcennamen (ARN) des Amazon SNS-Themas (für Amazon SNS SNS-Themen, die bereits mit der für Benachrichtigungen erforderlichen Richtlinie konfiguriert wurden).
   - Wenn Sie keine Ressource für die Verwendung mit Benachrichtigungen konfiguriert haben, wählen Sie Create target (Ziel erstellen) und dann SNS topic (SNS-Thema) aus. Geben Sie nach codestar-notifications- einen Namen für das Thema an und wählen Sie dann Create (Erstellen).

> **Note**
>
> - Wenn Sie das Amazon-SNS-Thema im Rahmen des Erstellens der Benachrichtigungsregel erstellen, wird die Richtlinie, die es ermöglicht, Ereignisse in dem Thema zu veröffentlichen, für Sie angewendet. Durch die Verwendung eines Themas, das für Benachrichtigungsregeln erstellt wurde, kann sichergestellt werden, dass Sie das Thema nur für die Benutzer abonnieren, die Benachrichtigungen zu dieser Ressource erhalten sollen.

- Sie können im Rahmen der Erstellung einer Benachrichtigungsregel keinen AWS Chatbot-Client erstellen. Wenn du AWS Chatbot (Slack) oder AWS Chatbot (Microsoft Teams) wählst, siehst du eine Schaltfläche, die dich anweist, einen Client in Chatbot zu konfigurieren. AWS Wenn Sie diese Option wählen, wird die AWS Chatbot-Konsole geöffnet. Weitere Informationen finden Sie unter Einen AWS Chatbot-Client für einen Slack-Kanal konfigurieren.
- Wenn Sie ein vorhandenes Amazon SNS SNS-Thema als Ziel verwenden möchten, müssen Sie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen zusätzlich zu allen anderen Richtlinien hinzufügen, die möglicherweise für dieses Thema existieren. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

8. Wählen Sie Submit (Absenden) und überprüfen Sie die Benachrichtigungsregel.

> **Note**
>
> Benutzer müssen das Amazon-SNS-Thema abonnieren und das Abonnement bestätigen, das Sie als Ziel der Regel angegeben haben, bevor sie Benachrichtigungen erhalten. Weitere Informationen finden Sie unter Ein Amazon-SNS-Thema, das für Benachrichtigungen verwendet wird, für Benutzer abonnieren.

So erstellen Sie eine Benachrichtigungsregel (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den Befehl create-notification rule aus, um das JSON-Skelett zu generieren:

```
aws codestar-notifications create-notification-rule --generate-cli-skeleton
 > rule.json
```

   Sie können die Datei beliebig benennen. In diesem Beispiel heißt die Datei `rule.json`.

2. Öffnen Sie die JSON-Datei in einem Texteditor und bearbeiten Sie sie so, dass sie die Ressource, die Ereignistypen und das gewünschte Amazon-SNS-Ziel für die Regel enthält.

Das folgende Beispiel zeigt eine Benachrichtigungsregel, die **`MyNotificationRule`** nach einem Repository benannt ist, das *`MyDemoRepo`* in einem AWS Konto mit der ID *`123456789012`* benannt ist. Benachrichtigungen mit dem vollständigen Detailtyp werden an ein Amazon SNS SNS-Thema gesendet, das benannt wird*`MyNotificationTopic`*, wenn Branches und Tags erstellt werden.

```
{
    "Name": "MyNotificationRule",
    "EventTypeIds": [
        "codecommit-repository-branches-and-tags-created"
    ],
    "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
    "Targets": [
        {
            "TargetType": "SNS",
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic"
        }
    ],
    "Status": "ENABLED",
    "DetailType": "FULL"
}
```

Speichern Sie die Datei.

3. Führen Sie unter Verwendung der soeben bearbeiteten Datei im Terminal oder in der Befehlszeile erneut den Befehl create-notification-rule aus, um die Benachrichtigungsregel zu erstellen:

```
aws codestar-notifications create-notification-rule --cli-input-json
 file://rule.json
```

4. Bei Erfolg gibt der Befehl den ARN der Benachrichtigungsregel zurück, der ähnlich wie nachfolgend gezeigt aussieht:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

So listen Sie Ereignistypen für Benachrichtigungsregeln auf (AWS CLI):

1. Führen Sie in einem Terminal oder an einer Eingabeaufforderung den Befehl list-event-types aus. Sie können die `--filters`-Option verwenden, um die Antwort auf einen bestimmten Ressourcentyp oder ein anderes Attribut zu beschränken. Im Folgenden wird beispielsweise eine Liste von Ereignistypen für CodeDeploy Anwendungen zurückgegeben.

```
aws codestar-notifications list-event-types --filters
 Name=SERVICE_NAME,Value=CodeDeploy
```

2. Die Ausgabe dieses Befehls sieht etwa wie folgt aus.

```
{
    "EventTypes": [
        {
            "EventTypeId": "codedeploy-application-deployment-succeeded",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Succeeded",
            "ResourceType": "Application"
        },
        {
            "EventTypeId": "codedeploy-application-deployment-failed",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Failed",
            "ResourceType": "Application"
        },
        {
            "EventTypeId": "codedeploy-application-deployment-started",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Started",
            "ResourceType": "Application"
        }
    ]
}
```

So fügen Sie einer Benachrichtigungsregel ein Tag hinzu (AWS CLI):

1. Führen Sie in einem Terminal oder an einer Eingabeaufforderung den Befehl tag-resource aus. Verwenden Sie beispielsweise den folgenden Befehl, um ein Tag-Schlüssel-Wert-Paar hinzuzufügen, das den Namen *`Team`* und den Wert enthält. *`Li_Juan`*

```
aws codestar-notifications tag-resource --arn arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/fe1efd35-EXAMPLE --tags Team=Li_Juan
```

2. Die Ausgabe dieses Befehls sieht etwa wie folgt aus.

```
{
    "Tags": {
        "Team": "Li_Juan"
    }
}
```

Benachrichtigungsregeln anzeigen

Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um alle Benachrichtigungsregeln für alle Ressourcen in einer AWS Region einzusehen. Sie können auch die Details der einzelnen Benachrichtigungsregeln anzeigen. Anders als beim Erstellen einer Benachrichtigungsregel müssen Sie die Ressourcenseite für die Ressource nicht aufrufen.

So zeigen Sie Benachrichtigungsregeln an (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Sehen Sie sich unter Benachrichtigungsregeln die Liste der Regeln an, die für Ihre Ressourcen in Ihrem Land konfiguriert sind, AWS-Konto in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Um die Details einer Benachrichtigungsregel anzuzeigen, wählen Sie sie in der Liste aus und wählen Sie dann View details (Details anzeigen) aus. Sie können auch einfach ihren Namen in der Liste auswählen.

So zeigen Sie eine Liste der Benachrichtigungsregeln an (AWS CLI):

1. Führen Sie den Befehl in einem Terminal oder an einer list-notification-rules Befehlszeile aus, um alle Benachrichtigungsregeln für die angegebene AWS Region anzuzeigen.

```
aws codestar-notifications list-notification-rules --region us-east-1
```

2. Bei Erfolg gibt dieser Befehl die ID und den ARN für jede Benachrichtigungsregel in der AWS Region zurück, ähnlich wie im Folgenden.

```
{
    "NotificationRules": [
        {
            "Id": "dc82df7a-EXAMPLE",
            "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/dc82df7a-EXAMPLE"
        },
        {
            "Id": "8d1f0983-EXAMPLE",
            "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/8d1f0983-EXAMPLE"
        }
    ]
}
```

So zeigen Sie Details zu einer Benachrichtigungsregel an (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den Befehl describe-notification-rule aus und geben Sie dabei den ARN der Benachrichtigungsregel an.

```
aws codestar-notifications describe-notification-rule --arn arn:aws:codestar-
notifications:us-east-1:123456789012:notificationrule/dc82df7a-EXAMPLE
```

2. Bei erfolgreicher Ausführung gibt der Befehl eine Ausgabe zurück, die in etwa wie folgt aussieht:

```
{
    "LastModifiedTimestamp": 1569199844.857,
    "EventTypes": [
        {
            "ServiceName": "CodeCommit",
            "EventTypeName": "Branches and tags: Created",
            "ResourceType": "Repository",
            "EventTypeId": "codecommit-repository-branches-and-tags-created"
        }
    ],
```

```
    "Status": "ENABLED",
    "DetailType": "FULL",
    "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE",
    "Targets": [
        {
            "TargetStatus": "ACTIVE",
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic",
            "TargetType": "SNS"
        }
    ],
    "Name": "MyNotificationRule",
    "CreatedTimestamp": 1569199844.857,
    "CreatedBy": "arn:aws:iam::123456789012:user/Mary_Major"
}
```

So zeigen Sie eine Liste von Tags für eine Benachrichtigungsregel an (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den Befehl list-tags-for-resource aus, um alle Tags für einen angegebenen Benachrichtigungsregel-ARN anzuzeigen:

```
aws codestar-notifications list-tags-for-resource --arn arn:aws:codestar-
notifications:us-east-1:123456789012:notificationrule/fe1efd35-EXAMPLE
```

2. Ist der Befehl erfolgreich, wird eine Ausgabe zurückgegeben, die wie folgt aussehen sollte.

```
{
    "Tags": {
        "Team": "Li_Juan"
    }
}
```

Bearbeiten einer Benachrichtigungsregel

Sie können eine Benachrichtigungsregel bearbeiten, um ihren Namen, die Ereignisse, für die Benachrichtigungen gesendet werden, den Detailtyp oder das Ziel bzw. die Ziele, an das/die Benachrichtigungen gesendet werden, zu ändern. Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um eine Benachrichtigungsregel zu bearbeiten.

So bearbeiten Sie eine Benachrichtigungsregel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Überprüfen Sie unter Benachrichtigungsregeln die Regeln, die für Ressourcen in Ihrem AWS Konto konfiguriert sind, in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Wählen Sie die Regel aus der Liste aus und wählen Sie dann Edit (Bearbeiten) aus. Nehmen Sie die gewünschten Änderungen vor und wählen Sie dann Submit (Absenden) aus.

So bearbeiten Sie eine Benachrichtigungsregel (AWS CLI):

1. Führen Sie den Befehl in einem Terminal oder an einer describe-notification-ruleBefehlszeile aus, um die Struktur der Benachrichtigungsregel anzuzeigen.
2. Führen Sie den Befehl update-notification rule aus, um das JSON-Skelett zu generieren und anschließend in einer Datei zu speichern.

```
aws codestar-notifications update-notification-rule --generate-cli-skeleton
 > update.json
```

   Sie können die Datei beliebig benennen. In diesem Beispiel ist die Datei*update.json*.

3. Öffnen Sie die JSON-Datei in einem Texteditor und nehmen Sie Änderungen an der Regel vor.

   Das folgende Beispiel zeigt eine Benachrichtigungsregel, die **MyNotificationRule** nach einem Repository benannt *MyDemoRepo* ist, das nach einem AWS Konto mit der ID benannt ist*123456789012*. Benachrichtigungen werden an ein Amazon SNS SNS-Thema mit dem Namen gesendet*MyNotificationTopic*, wenn Branches und Tags erstellt werden. Der Regelname wurde geändert in*MyNewNotificationRule*.

```
{
    "Name": "MyNewNotificationRule",
    "EventTypeIds": [
        "codecommit-repository-branches-and-tags-created"
    ],
    "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
```

```
    "Targets": [
        {
            "TargetType": "SNS",
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic"
        }
    ],
    "Status": "ENABLED",
    "DetailType": "FULL"
}
```

Speichern Sie die Datei.

4. Führen Sie unter Verwendung der soeben bearbeiteten Datei im Terminal oder auf der Befehlszeile erneut den Befehl update-notification-rule aus, um die Benachrichtigungsregel zu aktualisieren.

```
aws codestar-notifications update-notification-rule --cli-input-json
 file://update.json
```

5. Bei Erfolg gibt der Befehl den Amazon-Ressourcennamen (ARN) der Benachrichtigungsregel zurück, der ähnlich wie nachfolgend gezeigt aussieht.

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

So entfernen Sie ein Tag aus einer Benachrichtigungsregel (AWS CLI):

1. Führen Sie in einem Terminal oder an einer Eingabeaufforderung den Befehl untag-resource aus. Mit dem folgenden Befehl wird beispielsweise ein Tag mit dem Namen entfernt*Team*.

```
aws codestar-notifications untag-resource --arn arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/fe1efd35-EXAMPLE --tag-keys Team
```

2. Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Weitere Informationen finden Sie auch unter

- Hinzufügen oder Entfernen eines Ziels für eine Benachrichtigungsregel
- Aktivieren oder Deaktivieren von Benachrichtigungen für eine Benachrichtigungsregel
- --Ereignisse

Aktivieren oder Deaktivieren von Benachrichtigungen für eine Benachrichtigungsregel

Wenn Sie eine Benachrichtigungsregel erstellen, werden Benachrichtigungen standardmäßig aktiviert. Sie müssen die Regel nicht löschen, um zu verhindern, dass Benachrichtigungen gesendet werden. Sie können einfach den Benachrichtigungsstatus ändern.

So ändern Sie den Benachrichtigungsstatus für eine Benachrichtigungsregel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Überprüfen Sie unter Benachrichtigungsregeln die Regeln, die für Ressourcen in Ihrem AWS Konto konfiguriert sind, in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Suchen Sie die Benachrichtigungsregel, die Sie aktivieren oder deaktivieren möchten, und wählen Sie sie aus, um Details zu der Regel anzuzeigen.
5. Wählen Sie unter Notification status (Benachrichtigungsstatus) den Schieberegler aus, um den Status der Regel zu ändern:
   - Sending notifications (Benachrichtigungen senden): Dies ist die Standardeinstellung.
   - Notification paused (Benachrichtigungen angehalten): Es werden keine Benachrichtigungen an die angegebenen Ziele gesendet.

So ändern Sie den Benachrichtigungsstatus für eine Benachrichtigungsregel (AWS CLI):

1. Befolgen Sie die Schritte unter So bearbeiten Sie eine Benachrichtigungsregel (AWS CLI):, um die JSON für die Benachrichtigungsregel zu erhalten.

2. Bearbeiten Sie das Feld `Status` und ändern Sie es in `ENABLED` (Standard) oder `DISABLED` (keine Benachrichtigungen). Führen Sie anschließend den Befehl update-notification-rule aus, um den Status zu ändern.

```
"Status": "ENABLED"
```

Löschen einer Benachrichtigungsregel

Es können nur 10 Benachrichtigungsregeln für eine Ressource konfiguriert werden. Daher sollten Sie Regeln, die Sie nicht mehr benötigen, löschen. Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um eine Benachrichtigungsregel zu löschen.

> **Note**
>
> Das Löschen einer Benachrichtigungsregel kann nicht rückgängig gemacht werden, Sie können die Benachrichtigungsregel aber neu erstellen. Durch das Löschen einer Benachrichtigungsregel wird das Ziel nicht gelöscht.

So löschen Sie eine Benachrichtigungsregel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Überprüfen Sie unter Benachrichtigungsregeln die Regeln, die für Ressourcen in Ihrem AWS Konto konfiguriert sind, in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Wählen Sie die Benachrichtigungsregel und dann Delete (Löschen) aus.
5. Geben Sie **`delete`** ein und wählen Sie dann Delete (Löschen) aus.

So löschen Sie eine Benachrichtigungsregel (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den Befehl delete-notification-rule aus und geben Sie dabei den ARN der Benachrichtigungsregel an.

```
aws codestar-notifications delete-notification-rule --arn arn:aws:codestar-
notifications:us-east-1:123456789012:notificationrule/dc82df7a-EXAMPLE
```

2. Bei Erfolg gibt der Befehl den ARN der gelöschten Benachrichtigungsregel zurück, ähnlich wie nachfolgend gezeigt.

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Arbeiten mit Benachrichtigungsregelzielen

Ein Benachrichtigungsregelziel ist ein Ziel, das definiert, wohin Benachrichtigungen gesendet werden sollen, wenn die Ereignisbedingungen einer Benachrichtigungsregel erfüllt sind. Sie können zwischen Amazon SNS SNS-Themen und AWS Chatbot-Clients wählen, die für Slack- oder Microsoft Teams-Kanäle konfiguriert sind. Sie können beim Erstellen einer Benachrichtigungsregel ein Amazon-SNS-Thema als Ziel erstellen (empfohlen). Sie können auch ein vorhandenes Amazon SNS SNS-Thema in derselben AWS Region wie die Benachrichtigungsregel auswählen, müssen es jedoch mit der erforderlichen Richtlinie konfigurieren. Wenn Sie einen AWS Chatbot-Client als Ziel verwenden möchten, müssen Sie diesen Client zuerst in AWS Chatbot erstellen.

Wenn Sie die Reichweite von Benachrichtigungen vergrößern möchten, können Sie die Integration zwischen Benachrichtigungen und AWS Chatbot manuell konfigurieren, sodass Benachrichtigungen an Amazon Chime-Chatrooms gesendet werden. Anschließend können Sie das für diesen AWS Chatbot-Client konfigurierte Amazon SNS SNS-Thema als Ziel für die Benachrichtigungsregel auswählen. Weitere Informationen finden Sie unter Um Benachrichtigungen mit AWS Chatbot und Amazon Chime zu integrieren.

Sie können die Developer Tools-Konsole oder die verwenden, um AWS CLI Benachrichtigungsziele zu verwalten. Sie können die Konsole oder die verwenden, AWS CLI um Amazon SNS SNS-Themen und AWS Chatbot-Clients als Ziele zu erstellen und zu konfigurieren. Sie können auch die Integration zwischen den Amazon SNS SNS-Themen, die Sie als Ziele konfigurieren, und AWS Chatbot konfigurieren. Dies ermöglicht es Ihnen, Benachrichtigungen an Amazon-Chime-Chatrooms zu senden. Weitere Informationen finden Sie unter Konfigurieren der Integration zwischen Benachrichtigungen und AWS Chatbot.

Themen

- Erstellen oder Konfigurieren eines Benachrichtigungsregelziels
- Anzeigen von Benachrichtigungsregelzielen
- Hinzufügen oder Entfernen eines Ziels für eine Benachrichtigungsregel
- Löschen eines Benachrichtigungsregelziels

Erstellen oder Konfigurieren eines Benachrichtigungsregelziels

Ziele für Benachrichtigungsregeln sind Amazon SNS SNS-Themen oder AWS Chatbot-Clients, die für Slack- oder Microsoft Teams-Kanäle konfiguriert sind.

Ein AWS Chatbot-Client muss erstellt werden, bevor Sie einen Kunden als Ziel auswählen können. Wenn Sie einen AWS Chatbot-Client als Ziel für eine Benachrichtigungsregel auswählen, wird für diesen AWS Chatbot-Client ein Amazon SNS SNS-Thema mit allen Richtlinien konfiguriert, die für das Senden von Benachrichtigungen an den Slack- oder Microsoft Teams-Kanal erforderlich sind. Sie müssen keine vorhandenen Amazon-SNS-Themen für den AWS -Chatbot-Client konfigurieren.

Sie können in der Entwicklertools-Konsole beim Erstellen einer Benachrichtigungsregel auch Amazon-SNS-Benachrichtigungsregelziele erstellen. Die Richtlinie, die das Senden von Benachrichtigungen an dieses Thema ermöglicht, wird für Sie angewendet. Dies ist die einfachste Möglichkeit, ein Ziel für eine Benachrichtigungsregel zu erstellen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel.

Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden, müssen Sie es mit einer Zugriffsrichtlinie konfigurieren, die es der Ressource ermöglicht, Benachrichtigungen an dieses Thema zu senden. Ein Beispiel finden Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen.

> **Note**
>
> Wenn Sie ein vorhandenes Amazon-SNS-Thema verwenden möchten, anstatt ein neues zu erstellen, wählen Sie unter Targets (Ziele), dessen ARN aus. Stellen Sie sicher, dass das Thema über die entsprechende Zugriffsrichtlinie verfügt und Abonnentenliste nur die Benutzer enthält, denen Informationen zur Ressource angezeigt werden dürfen. Wenn es sich bei dem Amazon SNS-Thema um ein Thema handelt, das vor dem 5. November 2019 für CodeCommit Benachrichtigungen verwendet wurde, enthält es eine Richtlinie, die es ermöglicht, dort CodeCommit zu veröffentlichen, die andere Berechtigungen enthält als die, die für AWS CodeStar Benachrichtigungen erforderlich sind. Das Verwenden dieser Themen

wird nicht empfohlen. Wenn Sie eine für dieses Erlebnis erstellte Richtlinie verwenden möchten, müssen Sie zusätzlich zu der bereits vorhandenen Richtlinie die erforderliche Richtlinie für AWS CodeStar Benachrichtigungen hinzufügen. Weitere Informationen erhalten Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen und Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

Wenn Sie die Reichweite von Benachrichtigungen vergrößern möchten, können Sie die Integration zwischen Benachrichtigungen und AWS Chatbot manuell konfigurieren, sodass Benachrichtigungen an Amazon Chime-Chatrooms gesendet werden. Weitere Informationen erhalten Sie unter Targets (Ziele) und Um Benachrichtigungen mit AWS Chatbot und Amazon Chime zu integrieren.

Ein vorhandenes Amazon-SNS-Thema zur Verwendung als Benachrichtigungsregelziel konfigurieren (Konsole)

1. Melden Sie sich bei https://console.aws.amazon.com/sns/v3/home an AWS Management Console und öffnen Sie die Amazon SNS SNS-Konsole.
2. Wählen Sie in der Navigationsleiste Topics aus. Wählen Sie das Thema und anschließend Edit (Bearbeiten) aus.
3. Klappen Sie Access policy (Zugriffsrichtlinie) aus und wählen Sie danach Advanced (Erweitert) aus.
4. Geben Sie im JSON-Editor die folgende Anweisung für die Richtlinie ein. Geben Sie den Themen-ARN AWS-Region, die AWS-Konto ID und den Themennamen an.

```
{
     "Sid": "AWSCodeStarNotifications_publish",
     "Effect": "Allow",
     "Principal": {
       "Service": [
         "codestar-notifications.amazonaws.com"
       ]
     },
     "Action": "SNS:Publish",
     "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules"
    }
```

Die Richtlinienanweisung sollte so aussehen.

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__default_statement_ID",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "SNS:GetTopicAttributes",
        "SNS:SetTopicAttributes",
        "SNS:AddPermission",
        "SNS:RemovePermission",
        "SNS:DeleteTopic",
        "SNS:Subscribe",
        "SNS:ListSubscriptionsByTopic",
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules",
      "Condition": {
        "StringEquals": {
          "AWS:SourceOwner": "123456789012"
        }
      }
    },
 {
      "Sid": "AWSCodeStarNotifications_publish",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "codestar-notifications.amazonaws.com"
        ]
      },
      "Action": "SNS:Publish",
      "Resource": "arn:aws:sns:us-east-2:123456789012:codestar-notifications-
MyTopicForNotificationRules"
    }
  ]
}
```

5. Wählen Sie Änderungen speichern.
6. Überprüfen Sie unter Subscriptions (Abonnements) die Liste der Abonnenten für das Thema. Fügen Sie Abonnenten entsprechend diesem Benachrichtigungsregelziel hinzu, bearbeiten oder löschen Sie sie. Stellen Sie sicher, dass die Abonnentenliste nur die Benutzer enthält, die Informationen über die Ressource anzeigen dürfen. Weitere Informationen finden Sie unter Grundlagen zu Benachrichtigungsinhalten und -sicherheit.

Um einen AWS Chatbot-Client mit Slack zu erstellen, der als Ziel verwendet werden kann

1. Folgen Sie den Anweisungen unter Einrichten von AWS Chatbot mit Slack im AWS -Chatbot-Administratorhandbuch. Berücksichtigen Sie dabei die folgenden Optionen für eine optimale Integration mit Benachrichtigungen:
   - Wenn Sie eine IAM-Rolle erstellen, sollten Sie einen Rollennamen auswählen, der die Identifizierung des Zwecks dieser Rolle erleichtert (z. B. **`AWSCodeStarNotifications-Chatbot-Slack-Role`**). Dies kann Ihnen helfen, den Zweck der Rolle in der Zukunft zu identifizieren.
   - Bei SNS-Themen musst du weder ein Thema noch eine Region auswählen. AWS Wenn Sie den AWS Chatbot-Client als Ziel wählen, wird im Rahmen der Erstellung der Benachrichtigungsregeln ein Amazon SNS SNS-Thema mit allen erforderlichen Berechtigungen für den AWS Chatbot-Client erstellt und konfiguriert.
2. Schließen Sie den Client-Erstellungsprozess ab. Dieser Client steht Ihnen dann zur Verfügung, wenn Sie Benachrichtigungsregeln erstellen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel.

> **Note**
>
> Entfernen Sie das Amazon SNS SNS-Thema nicht aus dem AWS Chatbot-Client, nachdem es für Sie konfiguriert wurde. Andernfalls wird verhindert, dass Benachrichtigungen an Slack gesendet werden.

Um einen AWS Chatbot-Client mit Microsoft Teams zu erstellen, der als Ziel verwendet werden kann

1. Folgen Sie den Anweisungen unter Einrichten von AWS Chatbot mit Microsoft Teams im AWS -Chatbot-Administratorhandbuch. Berücksichtigen Sie dabei die folgenden Optionen für eine optimale Integration mit Benachrichtigungen:

- Wenn Sie eine IAM-Rolle erstellen, sollten Sie einen Rollennamen auswählen, der die Identifizierung des Zwecks dieser Rolle erleichtert (z. B. **`AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role`**). Dies kann Ihnen helfen, den Zweck der Rolle in der Zukunft zu identifizieren.
- Bei SNS-Themen müssen Sie weder ein Thema noch eine Region auswählen. AWS Wenn Sie den AWS Chatbot-Client als Ziel wählen, wird im Rahmen der Erstellung der Benachrichtigungsregeln ein Amazon SNS SNS-Thema mit allen erforderlichen Berechtigungen für den AWS Chatbot-Client erstellt und konfiguriert.

2. Schließen Sie den Client-Erstellungsprozess ab. Dieser Client steht Ihnen dann zur Verfügung, wenn Sie Benachrichtigungsregeln erstellen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel.

> **Note**
>
> Entfernen Sie das Amazon SNS SNS-Thema nicht aus dem AWS Chatbot-Client, nachdem es für Sie konfiguriert wurde. Andernfalls wird verhindert, dass Benachrichtigungen an Microsoft Teams gesendet werden.

Anzeigen von Benachrichtigungsregelzielen

Sie können die Developer Tools-Konsole und nicht die Amazon SNS SNS-Konsole verwenden, um alle Benachrichtigungsregelziele für alle Ressourcen in einer AWS Region anzuzeigen. Sie können auch die Details zu einem Benachrichtigungsregelziel anzeigen.

So zeigen Sie Benachrichtigungsregelziele an (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Sehen Sie sich unter Ziele für Benachrichtigungsregeln die Liste der Ziele an, die von den Benachrichtigungsregeln in Ihrem Land verwendet werden, AWS-Konto in AWS-Region dem Sie gerade angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern. Wenn der Zielstatus als Unreachable (Nicht erreichbar), angezeigt wird, müssen Sie möglicherweise nach der Ursache dafür suchen. Weitere Informationen finden Sie unter Fehlerbehebung .

So zeigen Sie eine Liste der Benachrichtigungsregelziele an (AWS CLI):

1. Führen Sie in einem Terminal oder bei einer Eingabeaufforderung den list-targets-Befehl aus, um eine Liste aller Benachrichtigungsregelziele für die angegebene AWS -Region anzuzeigen:

```
aws codestar-notifications list-targets --region us-east-2
```

2. Bei Erfolg gibt dieser Befehl die ID und den ARN für jede Benachrichtigungsregel in der AWS Region zurück, ähnlich wie im Folgenden:

```
{
    "Targets": [
        {
            "TargetAddress": "arn:aws:sns:us-
east-2:123456789012:MySNSTopicForNotificationRules",
            "TargetType": "SNS",
            "TargetStatus": "ACTIVE"
        },
        {
            "TargetAddress": "arn:aws:chatbot::123456789012:chat-configuration/
slack-channel/MySlackChannelClientForMyDevTeam",
            "TargetStatus": "ACTIVE",
            "TargetType": "AWSChatbotSlack"
        },
        {
            "TargetAddress": "arn:aws:sns:us-
east-2:123456789012:MySNSTopicForNotificationsAboutMyDemoRepo",
            "TargetType": "SNS",
            "TargetStatus": "ACTIVE"
        }
    ]
}
```

Hinzufügen oder Entfernen eines Ziels für eine Benachrichtigungsregel

Sie können eine Benachrichtigungsregel bearbeiten, um das Ziel bzw. die Ziele zu ändern, an das/die Benachrichtigungen gesendet werden. Sie können die Developer Tools-Konsole oder die verwenden AWS CLI , um die Ziele einer Benachrichtigungsregel zu ändern.

So ändern Sie die Ziele für eine Benachrichtigungsregel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Sehen Sie sich unter Benachrichtigungsregeln die Liste der Regeln an, die für Ihre Ressourcen in Ihrem AWS Konto konfiguriert sind, in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Wählen Sie die Regel und anschließend Edit (Bearbeiten) aus.
5. Führen Sie unter Targets (Ziele) einen der folgenden Schritte aus:
   - Um ein weiteres Ziel hinzuzufügen, wählen Sie Ziel hinzufügen und wählen Sie dann das Amazon SNS SNS-Thema oder den AWS Chatbot- (Slack) - oder AWS Chatbot- (Microsoft Teams) -Client aus, den Sie aus der Liste hinzufügen möchten. Sie können auch Create SNS topic (SNS-Thema erstellen) auswählen, um ein Thema zu erstellen und als Ziel hinzuzufügen. Eine Benachrichtigungsregel kann bis zu 10 Ziele haben.
   - Um ein Ziel zu entfernen, wählen Sie Remove target (Ziel entfernen) neben dem Ziel aus, das Sie entfernen möchten.
6. Wählen Sie Submit (Absenden) aus.

So fügen Sie einer Benachrichtigungsregel ein Ziel hinzu (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den subscribe-Befehl aus, um ein Ziel hinzuzufügen. Mit dem folgenden Befehl wird beispielsweise ein Amazon-SNS-Thema als Ziel für eine Benachrichtigungsregel hinzugefügt.

```
aws codestar-notifications subscribe --arn arn:aws:codestar-
notifications:us-east-1:123456789012:notificationrule/dc82df7a-
EXAMPLE --target TargetType=SNS,TargetAddress=arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic
```

2. Bei Erfolg gibt der Befehl den ARN der aktualisierten Benachrichtigungsregel zurück, ähnlich wie nachfolgend gezeigt:

```
{
```

```
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

So entfernen Sie ein Ziel aus einer Benachrichtigungsregel (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den unsubscribe-Befehl aus, um ein Ziel zu entfernen. Mit dem folgenden Befehl wird beispielsweise ein Amazon-SNS-Thema als Ziel für eine Benachrichtigungsregel entfernt.

```
aws codestar-notifications unsubscribe --arn arn:aws:codestar-
notifications:us-east-1:123456789012:notificationrule/dc82df7a-
EXAMPLE --target TargetType=SNS,TargetAddress=arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic
```

2. Bei Erfolg gibt der Befehl den ARN der aktualisierten Benachrichtigungsregel und Informationen über das entfernte Ziel zurück, ähnlich wie nachfolgend gezeigt:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
    "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"
}
```

Weitere Informationen finden Sie auch unter

- Bearbeiten einer Benachrichtigungsregel
- Aktivieren oder Deaktivieren von Benachrichtigungen für eine Benachrichtigungsregel

Löschen eines Benachrichtigungsregelziels

Sie können ein Ziel löschen, wenn es nicht mehr benötigt wird. Für eine Ressource können nur zehn (10) Benachrichtigungsregelziele konfiguriert werden. Das Löschen nicht benötigter Ziele kann daher helfen, Platz für andere Ziele zu schaffen, die Sie dieser Benachrichtigungsregel hinzufügen möchten.

> **Note**
>
> Durch das Löschen eines Benachrichtigungsregelziels wird das Ziel aus allen Benachrichtigungsregeln entfernt, die für die Verwendung als Ziel konfiguriert sind. Das Ziel selbst wird jedoch nicht gelöscht.

So löschen Sie ein Benachrichtigungsregelziel (Konsole):

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.
2. Erweitern Sie in der Navigationsleiste Settings (Einstellungen), und wählen Sie dann Notification rules (Benachrichtigungsregeln) aus.
3. Sehen Sie sich unter Ziele für Benachrichtigungsregeln die Liste der Ziele an, die für Ihre Ressourcen in Ihrem AWS Konto konfiguriert sind, in AWS-Region dem Sie derzeit angemeldet sind. Verwenden Sie den Selektor, um die AWS-Region zu ändern.
4. Wählen Sie das Benachrichtigungsregelziel und anschließend Delete (Löschen) aus.
5. Geben Sie **`delete`** ein und wählen Sie dann Delete (Löschen) aus.

So löschen Sie ein Benachrichtigungsregelziel (AWS CLI):

1. Führen Sie in einem Terminal oder einer Eingabeaufforderung den Befehl delete-target aus. und geben Sie dabei den ARN des Ziels an. Mit dem folgenden Befehl wird beispielsweise ein Ziel gelöscht, das ein Amazon-SNS-Thema verwendet.

   ```
   aws codestar-notifications delete-target --target-address arn:aws:sns:us-
   east-1:123456789012:MyNotificationTopic
   ```

2. Wenn der Befehl erfolgreich ist, wird nichts zurückgegeben. Bei einem Fehlschlag gibt der Befehl einen Fehler zurück. Der häufigste Fehler besteht darin, dass das Thema Ziel für eine oder mehrere Benachrichtigungsregeln ist.

   ```
   An error occurred (ValidationException) when calling the DeleteTarget operation:
    Unsubscribe target before deleting.
   ```

Sie können den `--force-unsubscribe-all`-Parameter verwenden, um das Ziel aus allen Benachrichtigungsregeln zu entfernen, die für die Verwendung dieses Ziels als Ziel konfiguriert sind, und dann das Ziel löschen.

```
aws codestar-notifications delete-target --target-address arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic --force-unsubscribe-all
```

Konfigurieren der Integration zwischen Benachrichtigungen und AWS Chatbot

AWS Chatbot ist ein AWS Service, der es Softwareentwicklungsteams ermöglicht, Amazon Chime Chime-Chatrooms, Slack-Kanäle und Microsoft-Team-Kanäle zu nutzen, um betriebliche Ereignisse in der zu überwachen und darauf zu reagieren. DevOps AWS Cloud Sie können die Integration zwischen Benachrichtigungsregelzielen und AWS Chatbot so konfigurieren, dass Benachrichtigungen über Ereignisse im Amazon Chime Chime-Raum, Slack-Kanal oder Microsoft Teams-Kanal Ihrer Wahl angezeigt werden. Weitere Informationen finden Sie in der AWS -Chatbot-Dokumentation.

Bevor Sie die Integration mit AWS Chatbot konfigurieren, müssen Sie eine Benachrichtigungsregel und ein Regelziel konfigurieren. Weitere Informationen erhalten Sie unter Einrichtung und Erstellen einer Benachrichtigungsregel. Außerdem müssen Sie einen Slack-Kanal, Microsoft Teams-Kanal oder einen Amazon-Chime-Chatroom in AWS Chatbot konfigurieren. Weitere Informationen finden Sie in der Dokumentation zu diesen Services.

Themen

- Einen AWS Chatbot-Client für einen Slack-Kanal konfigurieren
- Einen AWS Chatbot-Client für einen Microsoft Teams-Kanal konfigurieren
- Manuelles Konfigurieren von Clients für Slack oder Amazon Chime

Einen AWS Chatbot-Client für einen Slack-Kanal konfigurieren

Sie können Benachrichtigungsregeln erstellen, die einen AWS Chatbot-Client als Ziel verwenden. Wenn Sie einen Client für einen Slack-Kanal erstellen, können Sie diesen Client direkt als Ziel im Workflow zum Erstellen einer Benachrichtigungsregel verwenden. Dies ist der einfachste Weg, um Benachrichtigungen einzurichten, die in Slack-Kanälen angezeigt werden.

Um einen AWS Chatbot-Client mit Slack zu erstellen, der als Ziel verwendet werden kann

1. Folgen Sie den Anweisungen unter Einrichten von AWS Chatbot mit Slack im AWS -Chatbot-Administratorhandbuch. Berücksichtigen Sie dabei die folgenden Optionen für eine optimale Integration mit Benachrichtigungen:

   - Wenn Sie eine IAM-Rolle erstellen, sollten Sie einen Rollennamen auswählen, der die Identifizierung des Zwecks dieser Rolle erleichtert (z. B. **`AWSCodeStarNotifications-Chatbot-Slack-Role`**). Dies kann Ihnen helfen, den Zweck der Rolle in der Zukunft zu identifizieren.
   - Bei SNS-Themen musst du weder ein Thema noch eine Region auswählen. AWS Wenn Sie den AWS Chatbot-Client als Ziel wählen, wird im Rahmen der Erstellung der Benachrichtigungsregeln ein Amazon SNS SNS-Thema mit allen erforderlichen Berechtigungen für den AWS Chatbot-Client erstellt und konfiguriert.

2. Schließen Sie den Client-Erstellungsprozess ab. Dieser Client steht Ihnen dann zur Verfügung, wenn Sie Benachrichtigungsregeln erstellen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel.

   > **Note**
   >
   > Entfernen Sie das Amazon SNS SNS-Thema nicht aus dem AWS Chatbot-Client, nachdem es für Sie konfiguriert wurde. Andernfalls wird verhindert, dass Benachrichtigungen an Slack gesendet werden.

Einen AWS Chatbot-Client für einen Microsoft Teams-Kanal konfigurieren

Sie können Benachrichtigungsregeln erstellen, die einen AWS Chatbot-Client als Ziel verwenden. Wenn Sie einen Client für einen Microsoft Teams-Kanal erstellen, können Sie diesen Client direkt als Ziel im Workflow zum Erstellen einer Benachrichtigungsregel verwenden. Dies ist der einfachste Weg, um Benachrichtigungen einzurichten, die in Microsoft Teams-Kanälen angezeigt werden.

Um einen AWS Chatbot-Client mit Microsoft Teams zu erstellen, der als Ziel verwendet werden kann

1. Folgen Sie den Anweisungen unter Einrichten von AWS Chatbot mit Microsoft Teams im AWS -Chatbot-Administratorhandbuch. Berücksichtigen Sie dabei die folgenden Optionen für eine optimale Integration mit Benachrichtigungen:

- Wenn Sie eine IAM-Rolle erstellen, sollten Sie einen Rollennamen auswählen, der die Identifizierung des Zwecks dieser Rolle erleichtert (z. B. **`AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role`**). Dies kann Ihnen helfen, den Zweck der Rolle in der Zukunft zu identifizieren.
- Bei SNS-Themen müssen Sie weder ein Thema noch eine Region auswählen. AWS Wenn Sie den AWS Chatbot-Client als Ziel wählen, wird im Rahmen der Erstellung der Benachrichtigungsregeln ein Amazon SNS SNS-Thema mit allen erforderlichen Berechtigungen für den AWS Chatbot-Client erstellt und konfiguriert.

2. Schließen Sie den Client-Erstellungsprozess ab. Dieser Client steht Ihnen dann zur Verfügung, wenn Sie Benachrichtigungsregeln erstellen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel.

> **Note**
>
> Entfernen Sie das Amazon SNS SNS-Thema nicht aus dem AWS Chatbot-Client, nachdem es für Sie konfiguriert wurde. Andernfalls wird verhindert, dass Benachrichtigungen an Microsoft Teams gesendet werden.

Manuelles Konfigurieren von Clients für Slack oder Amazon Chime

Sie können die Integration zwischen Benachrichtigungen und Slack oder Amazon Chime direkt erstellen. Dies ist die einzige Möglichkeit für die Konfiguration von Benachrichtigungen für Amazon-Chime-Chatrooms. Wenn Sie diese Integration manuell konfigurieren, erstellen Sie einen AWS Chatbot-Client, der ein Amazon SNS SNS-Thema verwendet, das Sie zuvor als Ziel für eine Benachrichtigungsregel konfiguriert haben.

Um Benachrichtigungen manuell mit AWS Chatbot und Slack zu integrieren

1. Öffne die AWS Developer Tools-Konsole unter Einstellungen/Benachrichtigungenhttps://console.aws.amazon.com/codesuite/.
2. Wählen Sie Settings (Einstellungen) und dann Notification rules (Benachrichtigungsregeln) aus.
3. Suchen Sie das Ziel unter Notification rule targets (Benachrichtigungsregelziele) und kopieren Sie es.

> **Note**
>
> Sie können mehr als eine Benachrichtigungsregel so konfigurieren, dass dasselbe Amazon-SNS-Thema als Ziel verwendet wird. Dies kann Ihnen helfen, das Messaging zu konsolidieren, kann aber unbeabsichtigte Folgen haben, wenn die Abonnementliste für eine einzelne Benachrichtigungsregel oder Ressource vorgesehen ist.

4. Öffnen Sie die AWS Chatbot-Konsole unter https://console.aws.amazon.com/chatbot/.
5. Wählen Sie Configure new client (Neuen Client konfigurieren) und dann Slack.
6. Wählen Sie Konfigurieren aus.
7. Melden Sie sich bei Ihrem Slack-Workspace an.
8. Wenn Sie aufgefordert werden, Ihre Auswahl zu bestätigen, wählen Sie Allow (Zulassen) aus.
9. Wählen Sie Configure new channel (Neuen Kanal konfigurieren) aus.
10. Geben Sie unter Configuration details (Konfigurationsdetails) in Configuration name (Konfigurationsname) einen Namen für Ihren Client ein. Dies ist der Name, der in der Liste der verfügbaren Ziele für den Zieltyp AWS Chatbot (Slack) angezeigt wird, wenn Sie Benachrichtigungsregeln erstellen.
11. Wählen Sie unter Configure Slack Channel (Slack-Channel konfigurieren) unter Channel type (Channel-Typ) die Option Public (Öffentlich) oder Private (Privat) aus, je nachdem, welchen Channel Sie integrieren möchten.
    - Wählen Sie unter Public channel (Öffentlicher Kanal), den Namen des Slack-Kanals aus der Liste aus.
    - Geben Sie unter Private Channel ID (ID des privaten Kanals), den Kanalcode oder die URL ein.
12. Gehen Sie zu IAM permissions (IAM-Berechtigungen) und wählen Sie unter Role (Rolle) die Option Create an IAM role using a template (IAM-Rolle mit einer Vorlage erstellen) aus. Wählen Sie in Policy templates (Richtlinienvorlagen) die Option Notification permissions (Benachrichtigungsberechtigungen) aus. Geben Sie unter Role name (Rollenname) einen Namen für diese Rolle ein (z. B. **`AWSCodeStarNotifications-Chatbot-Slack-Role`**). Wählen Sie in Policy templates (Richtlinienvorlagen) die Option Notification permissions (Benachrichtigungsberechtigungen) aus.
13. Wählen Sie unter SNS-Themen unter SNS-Region den Ort aus, an AWS-Region dem Sie das Ziel der Benachrichtigungsregel erstellt haben. Wählen Sie unter SNS topics (SNS-Themen)

den Namen des Amazon-SNS-Themas aus, das Sie als Benachrichtigungsregelziel konfiguriert haben.

> **Note**
>
> Dieser Schritt ist nicht erforderlich, wenn Sie eine Benachrichtigungsregel mit diesem Client als Ziel erstellen.

14. Wählen Sie Konfigurieren aus.

> **Note**
>
> Wenn Sie die Integration mit einem privaten Kanal konfiguriert haben, müssen Sie AWS Chatbot zum Kanal einladen, bevor in diesem Kanal Benachrichtigungen angezeigt werden. Weitere Informationen finden Sie in der AWS -Chatbot-Dokumentation.

15. (Optional) Um die Integration zu testen, nehmen Sie eine Änderung an der Ressource vor, die einem Ereignistyp für eine Benachrichtigungsregel entspricht, die für die Verwendung des Amazon-SNS-Themas als Ziel konfiguriert ist. Wenn Sie beispielsweise eine Benachrichtigungsregel so konfiguriert haben, dass Benachrichtigungen gesendet werden, wenn Kommentare zu einer Pull-Anforderung gemacht werden, kommentieren Sie eine Pull-Anforderung und beobachten Sie dann den Slack-Kanal im Browser, um zu sehen, wann die Benachrichtigung erscheint.

Um Benachrichtigungen mit AWS Chatbot und Amazon Chime zu integrieren

1. Öffnen Sie die AWS Developer Tools-Konsole unter Einstellungen/Benachrichtigungenhttps://console.aws.amazon.com/codesuite/.
2. Wählen Sie Settings (Einstellungen) und dann Notification rules (Benachrichtigungsregeln) aus.
3. Suchen Sie das Ziel unter Notification rule targets (Benachrichtigungsregelziele) und kopieren Sie es.

> **Note**
>
> Sie können mehr als eine Benachrichtigungsregel so konfigurieren, dass dasselbe Amazon-SNS-Thema als Ziel verwendet wird. Dies kann Ihnen helfen, das Messaging zu

konsolidieren, kann aber unbeabsichtigte Folgen haben, wenn die Abonnementliste für eine einzelne Benachrichtigungsregel oder Ressource sein soll.

4. Öffnen Sie in Amazon Chime den Chatroom, den Sie für die Integration konfigurieren möchten.
5. Wählen Sie das Zahnradsymbol rechts oben und anschließend Manage webhooks (Webhooks verwalten) aus.
6. Wählen Sie im Dialogfeld Manage webhooks (Webhooks verwalten) New (Neu) aus, geben Sie einen Namen für den Webhook ein und wählen Sie anschließend Create (Erstellen) aus.
7. Überprüfen Sie, ob der Webhook angezeigt wird, und wählen Sie dann Copy webhook URL (Webhook-URL kopieren) aus.
8. Öffnen Sie die AWS Chatbot-Konsole unter https://console.aws.amazon.com/chatbot/.
9. Wählen Sie Configure new client (Neuen Client konfigurieren) und dann Amazon Chime.
10. Geben Sie unter Configuration details (Konfigurationsdetails) in Configuration name (Konfigurationsname) einen Namen für Ihren Client ein.
11. Fügen Sie in Webhook URL die URL ein. Geben Sie in Webhook description (Webhook-Beschreibung) eine optionale Beschreibung an.
12. Gehen Sie zu IAM permissions (IAM-Berechtigungen) und wählen Sie unter Role (Rolle) die Option Create an IAM role using a template (IAM-Rolle mit einer Vorlage erstellen) aus. Wählen Sie in Policy templates (Richtlinienvorlagen) die Option Notification permissions (Benachrichtigungsberechtigungen) aus. Geben Sie unter Role name (Rollenname) einen Namen für diese Rolle ein (z. B. **`AWSCodeStarNotifications-Chatbot-Chime-Role`**).
13. Wählen Sie unter SNS-Themen unter SNS-Region den Ort aus, an AWS-Region dem Sie das Ziel der Benachrichtigungsregel erstellt haben. Wählen Sie unter SNS topics (SNS-Themen) den Namen des Amazon-SNS-Themas aus, das Sie als Benachrichtigungsregelziel konfiguriert haben.
14. Wählen Sie Konfigurieren aus.
15. (Optional) Um die Integration zu testen, nehmen Sie eine Änderung an der Ressource vor, die einem Ereignistyp für eine Benachrichtigungsregel entspricht, die für die Verwendung des Amazon-SNS-Themas als Ziel konfiguriert ist. Wenn Sie beispielsweise eine Benachrichtigungsregel so konfiguriert haben, dass Benachrichtigungen gesendet werden, wenn Kommentare zu einer Pull-Anforderung gemacht werden, kommentieren Sie eine Pull-Anforderung und beobachten Sie dann den Amazon-Chime-Chatroom, um zu sehen, wann die Benachrichtigung erscheint.

API-Aufrufe für AWS CodeStar Benachrichtigungen protokollieren mit AWS CloudTrail

AWS CodeStar Notifications ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der von einem Benutzer, einer Rolle oder einem AWS Dienst ausgeführten Aktionen bereitstellt. CloudTrail erfasst alle API-Aufrufe für Benachrichtigungen als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Developer Tools-Konsole und Codeaufrufen für die AWS CodeStar Notifications API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für Benachrichtigungen. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von CloudTrail gesammelten Informationen können Sie die Anfrage an AWS CodeStar Benachrichtigungen, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde und andere Details ermitteln.

Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.

AWS CodeStar Benachrichtigungen, Informationen in CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn in den AWS CodeStar Benachrichtigungen eine Aktivität auftritt, wird diese Aktivität zusammen mit anderen CloudTrail AWS Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem anzeigen, suchen und herunterladen AWS-Konto. Weitere Informationen finden Sie unter Ereignisse mit CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem Konto AWS-Konto, einschließlich der Ereignisse für AWS CodeStar Benachrichtigungen, 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 anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen 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 CodeStar Benachrichtigungsaktionen werden von protokolliert CloudTrail und sind in der dokumentiert*AWS CodeStar Notifications API Reference*. Zum Beispiel werden durch Aufrufe der Aktionen `CreateNotificationRule`, `Subscribe` und `ListEventTypes` Einträge in den CloudTrail-Protokolldateien generiert.

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 -Element userIdentity.

Grundlagen zu -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 Erstellung einer Benachrichtigungsregel demonstriert, einschließlich der `Subscribe` Aktionen `CreateNotificationRule` und.

> **Note**
>
> Einige der Ereignisse in den Benachrichtigungsprotokoll-Dateieinträgen stammen möglicherweise von der serviceverknüpften Rolle AWSServiceRoleForCodeStarNotifications.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/Mary_Major",
        "accountId":"123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName":"Mary_Major"
    },
    "eventTime": "2019-10-07T21:34:41Z",
    "eventSource": "events.amazonaws.com",
    "eventName": "CreateNotificationRule",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "codestar-notifications.amazonaws.com",
    "userAgent": "codestar-notifications.amazonaws.com",
    "requestParameters": {
        "description": "This rule is used to route CodeBuild, CodeCommit, CodePipeline,
 and other Developer Tools notifications to AWS CodeStar Notifications",
        "name": "awscodestarnotifications-rule",
        "eventPattern": "{\"source\":[\"aws.codebuild\",\"aws.codecommit\",
\"aws.codepipeline\"]}"
    },
    "responseElements": {
        "ruleArn": "arn:aws:events:us-east-1:123456789012:rule/
awscodestarnotifications-rule"
    },
    "requestID": "ff1f309a-EXAMPLE",
    "eventID": "93c82b07-EXAMPLE",
    "eventType": "AwsApiCall",
    "apiVersion": "2015-10-07",
    "recipientAccountId": "123456789012"
}
```

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/Mary_Major",
        "accountId":"123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName":"Mary_Major"
```

```
    },
    "eventTime": "2019-10-07T21:34:41Z",
    "eventSource": "events.amazonaws.com",
    "eventName": "Subscribe",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "codestar-notifications.amazonaws.com",
    "userAgent": "codestar-notifications.amazonaws.com",
    "requestParameters": {
        "targets": [
            {
                "arn": "arn:aws:codestar-notifications:us-east-1:::",
                "id": "codestar-notifications-events-target"
            }
        ],
        "rule": "awscodestarnotifications-rule"
    },
    "responseElements": {
        "failedEntryCount": 0,
        "failedEntries": []
    },
    "requestID": "9466cbda-EXAMPLE",
    "eventID": "2f79fdad-EXAMPLE",
    "eventType": "AwsApiCall",
    "apiVersion": "2015-10-07",
    "recipientAccountId": "123456789012"
}
```

Fehlerbehebung

Die folgenden Informationen können Ihnen bei der Behebung von häufigen Problemen mit Benachrichtigungen helfen.

Themen

- Ich erhalte einen Berechtigungsfehler, wenn ich versuche, eine Benachrichtigungsregel für eine Ressource zu erstellen
- Ich kann keine Benachrichtigungsregeln sehen.
- Ich kann keine Benachrichtigungsregeln erstellen.
- Ich erhalte Benachrichtigungen für eine Ressource, auf die ich nicht zugreifen kann
- Ich erhalte keine Amazon-SNS-Benachrichtigungen
- Ich erhalte doppelte Benachrichtigungen zu Ereignissen

- Ich möchte verstehen, warum ein Benachrichtigungszielstatus als „Unreachable (Nicht erreichbar)“ angezeigt wird
- Ich möchte meine Kontingente für Benachrichtigungen und Ressourcen erhöhen

Ich erhalte einen Berechtigungsfehler, wenn ich versuche, eine Benachrichtigungsregel für eine Ressource zu erstellen

Stellen Sie sicher, dass Sie über ausreichende Berechtigungen verfügen. Weitere Informationen finden Sie unter Beispiele für identitätsbasierte Richtlinien.

Ich kann keine Benachrichtigungsregeln sehen.

Problem: Wenn Sie in der Entwicklertools-Konsole unter Notifications (Einstellungen) die Option Notifications (Benachrichtigungen) wählen, wird ein Berechtigungsfehler angezeigt.

Mögliche Lösungen: Möglicherweise verfügen Sie nicht über die erforderlichen Berechtigungen zum Anzeigen von Benachrichtigungen. Während die meisten verwalteten Richtlinien für AWS Developer Tools-Dienste, wie z. B. CodeCommit und CodePipeline, Berechtigungen für Benachrichtigungen enthalten, enthalten Dienste, die derzeit keine Benachrichtigungen unterstützen, keine Berechtigungen zum Anzeigen dieser Benachrichtigungen. Es könnte aber auch eine benutzerdefinierte Richtlinie für Ihren IAM-Benutzer oder Ihre IAM-Rolle aktiv sein, die das Anzeigen von Benachrichtigungen nicht zulässt. Weitere Informationen finden Sie unter Beispiele für identitätsbasierte Richtlinien.

Ich kann keine Benachrichtigungsregeln erstellen.

Möglicherweise verfügen Sie nicht über die erforderlichen Berechtigungen zum Erstellen einer Benachrichtigungsregel. Weitere Informationen finden Sie unter Beispiele für identitätsbasierte Richtlinien.

Ich erhalte Benachrichtigungen für eine Ressource, auf die ich nicht zugreifen kann

Wenn Sie eine Benachrichtigungsregel erstellen und ein Ziel hinzufügen, überprüft die Benachrichtigungsfunktion nicht, ob der Empfänger Zugriff auf die Ressource hat. Es ist möglich, dass Sie Benachrichtigungen zu einer Ressource erhalten, auf die Sie nicht zugreifen können. Bitten Sie, aus der Abonnementliste für das Ziel entfernt zu werden, wenn Sie sich nicht selbst entfernen können.

Ich erhalte keine Amazon-SNS-Benachrichtigungen

Um Probleme mit dem Amazon-SNS-Thema zu beheben, überprüfen Sie Folgendes:

- Stellen Sie sicher, dass das Amazon SNS SNS-Thema in derselben AWS Region wie die Benachrichtigungsregel erstellt wurde.
- Stellen Sie sicher, dass Ihr E-Mail-Alias das richtige Thema abonniert hat und dass Sie das Abonnement bestätigt haben. Weitere Informationen finden Sie unter Abonnieren eines Endpunkts für ein Amazon-SNS-Thema.
- Vergewissern Sie sich, dass die Themenrichtlinie so bearbeitet wurde, dass AWS CodeStar Benachrichtigungen Push-Benachrichtigungen zu diesem Thema senden können. Die Themarichtlinie sollte eine Anweisung ähnlich der folgenden enthalten:

```
{
    "Sid": "AWSCodeStarNotifications_publish",
    "Effect": "Allow",
    "Principal": {
        "Service": [
            "codestar-notifications.amazonaws.com"
        ]
    },
    "Action": "SNS:Publish",
    "Resource": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopicName",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "123456789012"
        }
    }
}
```

  Weitere Informationen finden Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen.

Ich erhalte doppelte Benachrichtigungen zu Ereignissen

Die häufigsten Gründe für den Erhalt mehrerer Benachrichtigungen:

- Für eine Ressource wurden mehrere Benachrichtigungsregeln konfiguriert, die denselben Ereignistyp enthalten, und Sie haben die Amazon-SNS-Themen abonniert, die Ziele für diese

Regeln sind. Um dieses Problem zu lösen, melden Sie sich entweder von einem der Themen ab ,oder bearbeiten Sie die Benachrichtigungsregeln, um eine Duplizierung zu vermeiden.

- Ein oder mehrere Benachrichtigungsregelziele sind in AWS Chatbot integriert und Sie erhalten Benachrichtigungen in Ihrem E-Mail-Posteingang und einem Slack-Kanal, Microsoft Teams-Kanal oder Amazon Chime Chime-Chatroom. Um dieses Problem zu lösen, sollten Sie Ihre E-Mail-Adresse von dem Amazon-SNS-Thema abmelden, das Ziel für die Regel ist, und den Slack- oder Microsoft Teams-Kanal oder Amazon-Chime-Chatroom verwenden, um Benachrichtigungen anzuzeigen.

Ich möchte verstehen, warum ein Benachrichtigungszielstatus als „Unreachable (Nicht erreichbar)“ angezeigt wird

Ziele haben zwei mögliche Status: Active (Aktiv) und Unreachable (Nicht erreichbar). Unreachable (Nicht erreichbar) gibt an, dass Benachrichtigungen an ein Ziel gesendet wurden und die Zustellung nicht erfolgreich war. Benachrichtigungen werden weiterhin an dieses Ziel gesendet und bei Erfolg wird der Status auf Active (Aktiv) zurückgesetzt.

Das Ziel für eine Benachrichtigungsregel ist möglicherweise aus einem der folgenden Gründe nicht verfügbar:

- Die Ressource (Amazon SNS SNS-Thema oder AWS Chatbot-Client) wurde gelöscht. Wählen Sie ein anderes Ziel für die Benachrichtigungsregel aus.
- Das Amazon SNS SNS-Thema ist verschlüsselt, und entweder fehlt die erforderliche Richtlinie für verschlüsselte Themen, oder der AWS KMS Schlüssel wurde gelöscht. Weitere Informationen finden Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen.
- Das Amazon-SNS-Thema verfügt nicht über die erforderliche Richtlinie für Benachrichtigungen. Benachrichtigungen können nur an ein Amazon-SNS-Thema gesendet werden, wenn es über die Richtlinie verfügt. Weitere Informationen finden Sie unter Konfigurieren von Amazon-SNS-Themen für Benachrichtigungen.
- Beim unterstützenden Dienst für das Ziel (Amazon SNS oder AWS Chatbot) treten möglicherweise Probleme auf.

Ich möchte meine Kontingente für Benachrichtigungen und Ressourcen erhöhen

Derzeit können Sie keine Kontingente ändern. Siehe Kontingente für Benachrichtigungen.

Kontingente für Benachrichtigungen

In der folgenden Tabelle sind die Kontingente (auch als Limits bezeichnet) für Benachrichtigungen in der Entwicklertools-Konsole ausgeführt. Informationen zu Limits, für die Änderungen möglich sind, finden Sie unter AWS Service Quotas.

Ressource	Standardlimit
Maximale Anzahl von Benachrichtigungsregeln in einem AWS Konto	1000
Maximale Anzahl von Zielen für eine Benachrichtigungsregel	10
Maximale Anzahl von Benachrichtigungsregeln für eine Ressource	10

Was sind Verbindungen?

Sie können die Verbindungsfunktion in der Developer Tools-Konsole verwenden, um AWS Ressourcen zu verbinden, z. B. AWS CodePipeline mit externen Code-Repositorys. Diese Funktion hat eine eigene API, die AWS CodeConnectionsAPI-Referenz. Jede Verbindung ist eine Ressource, die du AWS Diensten zur Verfügung stellen kannst, um eine Verbindung zu einem Repository eines Drittanbieters wie Bitbucket herzustellen. Du kannst die Verbindung beispielsweise CodePipeline so hinzufügen, dass sie deine Pipeline auslöst, wenn eine Codeänderung an deinem Code-Repository eines Drittanbieters vorgenommen wird. Jede Verbindung wird benannt und mit einem eindeutigen Amazon-Ressourcenname (ARN) verknüpft, der auf die Verbindung verweist.

> **Important**
>
> Der Servicename AWS CodeStar Connections wurde umbenannt. Ressourcen, die mit dem vorherigen Namespace codestar-connections erstellt wurden, werden weiterhin unterstützt.

Was kann ich mit Verbindungen tun?

Sie können Verbindungen verwenden, um Ressourcen von Drittanbietern in Ihre AWS -Ressourcen in Entwicklertools zu integrieren. Beispiele:

- Stelle eine Connect zu einem Drittanbieter wie Bitbucket her und verwende die Drittanbieterverbindung als Quellintegration mit deinen AWS Ressourcen, z. B. CodePipeline
- Verwalte den Zugriff auf deine Verbindung über deine Ressourcen hinweg einheitlich, indem du Projekte, CodeDeploy Anwendungen und Pipelines CodePipeline für deinen Drittanbieter CodeBuild erstellst.
- Verwenden Sie einen Verbindungs-ARN in Ihren Stack-Vorlagen, CodeBuild um Projekte, CodeDeploy Anwendungen und Pipelines zu erstellen CodePipeline, ohne auf gespeicherte Geheimnisse oder Parameter verweisen zu müssen.

Für welche Drittanbieter kann ich Verbindungen erstellen?

Verbindungen können Ihre AWS Ressourcen mit den folgenden Repositorys von Drittanbietern verknüpfen:

- Bitbucket Cloud
- GitHub.com
- GitHub Cloud für Unternehmen
- GitHub Server für Unternehmen
- GitLab.com

> **Important**
>
> Die Unterstützung von Verbindungen für GitLab umfasst Version 15.x und höher.

- GitLab selbstverwaltete Installation (für Enterprise Edition oder Community Edition)

Eine Übersicht über den Workflow bei Verbindungen finden Sie unter Workflow zum Erstellen oder Ändern von Verbindungen.

Die Schritte zum Erstellen von Verbindungen für einen Cloud-Anbietertyp, z. B. GitHub, unterscheiden sich von den Schritten für einen installierten Anbietertyp, z. B. GitHub Enterprise

Server. Weitere Hinweise zur allgemeinen Vorgehensweise beim Erstellen einer Verbindung nach Anbietertyp finden Sie unter Arbeiten mit Verbindungen.

> **Note**
>
> Um Verbindungen in Europa (Mailand) nutzen zu können AWS-Region, müssen Sie:
>
> 1. Regionsspezifische App installieren
> 2. Region aktivieren
>
> Diese regionsspezifische App unterstützt Verbindungen in der Region Europa (Mailand). Sie ist auf der Website des Drittanbieters veröffentlicht und von der bestehenden App getrennt, die Verbindungen für andere Regionen unterstützt. Durch die Installation dieser App autorisieren Sie Drittanbieter, Ihre Daten nur für diese Region an den Dienst weiterzugeben, und Sie können diese Autorisierung jederzeit widerrufen, indem Sie die App deinstallieren. Der Dienst verarbeitet oder speichert Ihre Daten nicht, es sei denn, Sie aktivieren die Region. Durch die Aktivierung dieser Region gewähren Sie unserem Dienst die Erlaubnis, Ihre Daten zu verarbeiten und zu speichern.
> Auch wenn die Region nicht aktiviert ist, können Drittanbieter Ihre Daten trotzdem mit unserem Dienst teilen, wenn die regionsspezifische App installiert bleibt. Achten Sie also darauf, die App zu deinstallieren, sobald Sie die Region deaktivieren. Weitere Informationen finden Sie unter Aktivieren einer Region.

Was AWS-Services lässt sich in Verbindungen integrieren?

Sie können mit Verbindungen Ihr Drittanbieter-Repository in andere AWS-Services-Services integrieren. Informationen zu den Serviceintegrationen für Verbindungen finden Sie unterProdukt- und Service-Integrationen mit AWS CodeConnections.

Wie funktionieren Verbindungen?

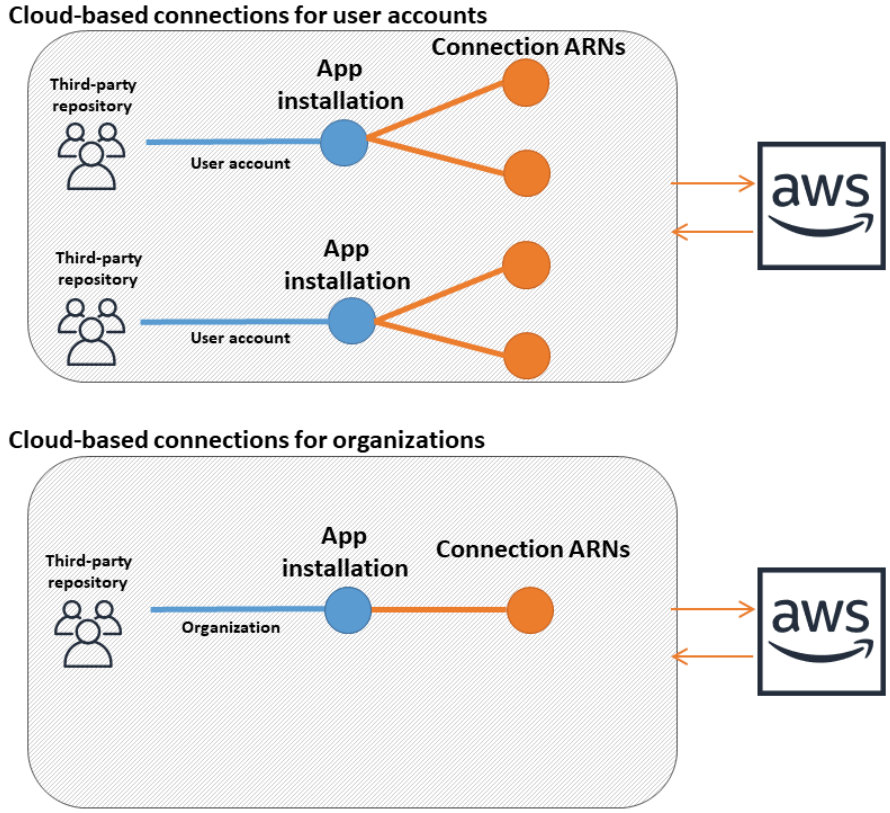
Bevor Sie eine Verbindung erstellen können, müssen Sie zunächst die AWS -Authentifizierungs-App in Ihrem Drittanbieterkonto installieren oder den Zugriff darauf bereitstellen. Nachdem eine Verbindung installiert wurde, kann sie so eingestellt werden, dass sie diese Installation verwendet. Wenn Sie eine Verbindung erstellen, erteilen Sie Zugriff auf die AWS -Ressource in Ihrem Drittanbieterkonto. Dadurch kann die Verbindung im Namen Ihrer AWS Ressourcen auf Inhalte wie

Quell-Repositorys im Drittanbieter-Konto zugreifen. Sie können diese Verbindung dann mit anderen teilen AWS-Services , um sichere OAuth Verbindungen zwischen den Ressourcen herzustellen.

Cloud-basierte Verbindungen werden wie folgt konfiguriert, wobei die Unterschiede zwischen Benutzerkonten oder Organisationen angegeben werden.

- Benutzerkonten: Für jedes cloudbasierte Benutzerkonto eines Drittanbieters ist eine Connector-App installiert. Der App-Installation können mehrere Verbindungen zugeordnet werden.
- Organizations: Jede cloudbasierte Drittanbieterorganisation hat eine Connector-App-Installation. Bei Verbindungen in Organisationen ist Ihre Verbindungszuordnung zu jedem Organisationskonto in der Organisation 1:1. Der App-Installation können nicht mehrere Verbindungen zugeordnet werden. Weitere Informationen darüber, wie Organisationen mit Verbindungen arbeiten, finden Sie unterWie AWS CodeConnections funktionieren Verbindungen mit Organisationen.

Das folgende Diagramm zeigt, wie cloudbasierte Verbindungen mit Benutzerkonten oder Organisationen funktionieren.



Verbindungen gehören demjenigen AWS-Konto , der sie erstellt hat. Verbindungen werden durch einen ARN identifiziert, der eine Verbindungs-ID enthält. Die Verbindungs-ID ist eine UUID, die nicht geändert oder neu zugeordnet werden kann. Durch das Löschen und Wiedererstellen einer

Verbindung entsteht eine neue Verbindungs-ID und damit ein neuer Verbindungs-ARN. Das bedeutet, dass Verbindungen ARNs niemals wiederverwendet werden.

Eine neu erstellte Verbindung befindet sich im Zustand `Pending`. Ein Handshake-Prozess (OAuth Flow) eines Drittanbieters ist erforderlich, um die Einrichtung der Verbindung abzuschließen und sie von einem `Available` bestimmten Status `Pending` zu ändern. Sobald dieser Vorgang abgeschlossen ist, ist `Available` und kann eine Verbindung mit AWS Diensten verwendet werden, wie CodePipeline z.

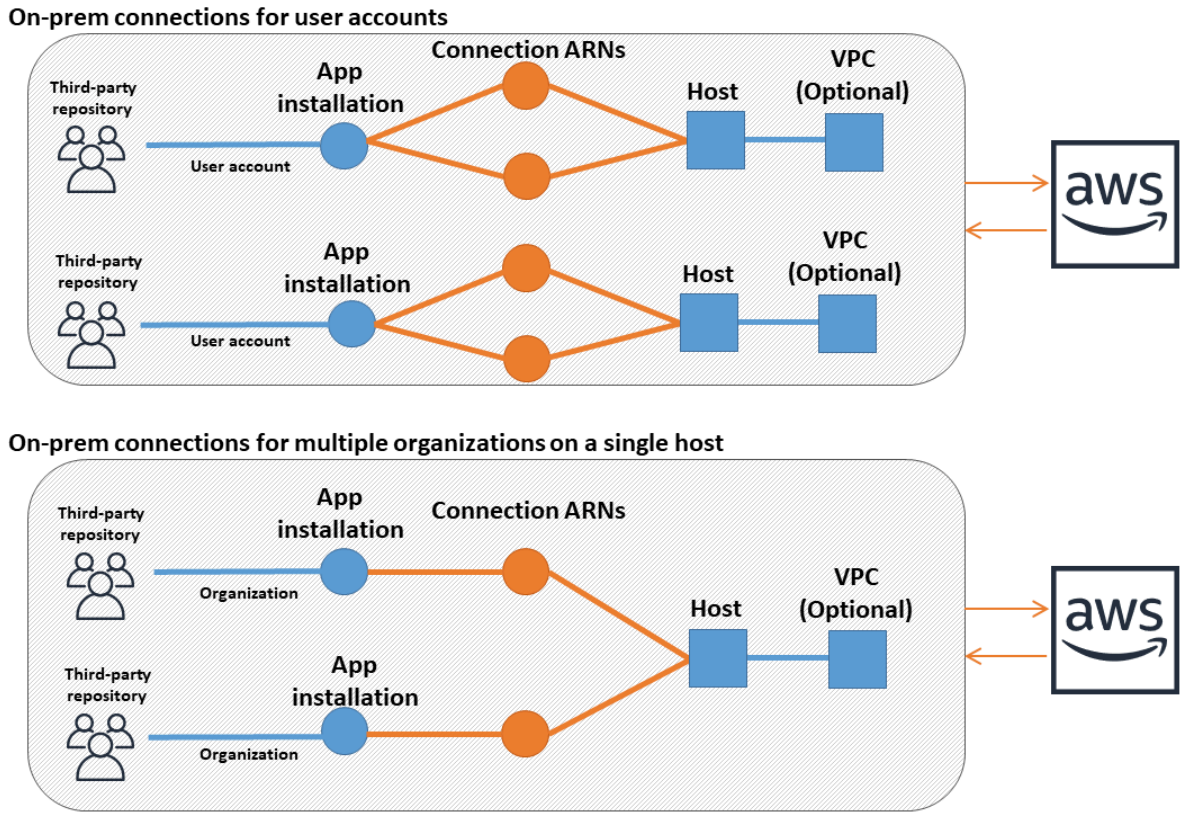
Wenn Sie eine Verbindung zu einem installierten Anbietertyp (lokal) herstellen möchten, z. B. GitHub Enterprise Server oder GitLab selbstverwaltet, verwenden Sie für Ihre Verbindung eine Hostressource.

Lokale Verbindungen werden wie folgt konfiguriert, wobei die Unterschiede zwischen Benutzerkonten oder Organisationen angegeben werden.

- Benutzerkonten: Für jedes lokale Drittanbieter-Benutzerkonto ist eine Connector-App installiert. Einem Host können mehrere Verbindungen für einen lokalen Anbieter zugeordnet werden.
- Organizations: Jede lokale Drittanbieterorganisation hat eine Connector-App-Installation. Für lokale Verbindungen in Organisationen wie GitHub Organizations for GitHub Enterprise Server erstellen Sie für jede Verbindung in Ihrer Organisation einen neuen Host und achten darauf, dieselben Informationen in die Netzwerkfelder (VPC, Subnetz IDs und Sicherheitsgruppe IDs) für den Host einzugeben. Weitere Informationen darüber, wie Organisationen mit Verbindungen arbeiten, finden Sie unter. Wie AWS CodeConnections funktionieren Verbindungen mit Organisationen
- Alle: Für jede lokale Verbindung kann jede VPC jeweils nur einem Host zugeordnet werden.

In allen Fällen müssen Sie die URL für Ihren lokalen Server angeben. Wenn sich der Server in einer privaten VPC befindet (d. h. nicht über das Internet zugänglich ist), müssen Sie außerdem VPC-Informationen zusammen mit optionalen TLS-Zertifikatsinformationen angeben. Diese Konfigurationen ermöglichen CodeConnections die Kommunikation mit der Instanz und werden von allen Verbindungen gemeinsam genutzt, die für diesen Host erstellt wurden. Für eine einzelne GitHub Enterprise Server-Instanz würden Sie beispielsweise eine einzelne App erstellen, die durch einen Host repräsentiert wird. Für die Konfiguration des Benutzerkontos könnten Sie dann mehrere Verbindungen für diesen Host erstellen, die Ihrer App-Installation entsprechen, wie in der folgenden Abbildung dargestellt. Andernfalls erstellen Sie für eine Organisation eine einzige App-Installation und Verbindung für diesen Host.

Das folgende Diagramm zeigt, wie lokale Verbindungen mit Benutzerkonten oder Organisationen funktionieren.



Ein neu erstellter Host befindet sich im Zustand `Pending` (Ausstehend). Ein Drittanbieter-Registrierungs-Prozess ist erforderlich, um die Einrichtung des Hosts abzuschließen und vom Zustand `Pending` (Ausstehend) zum Zustand `Available` (Verfügbar) zu gelangen. Nachdem das abgeschlossen ist, ist ein Host `Available` (Verfügbar) und kann für Verbindungen mit installierten Anbietertypen verwendet werden.

Eine Übersicht über den Workflow bei Verbindungen finden Sie unter Workflow zum Erstellen oder Ändern von Verbindungen. Eine Übersicht über den Workflow zur Host-Erstellung für installierte Anbieter finden Sie unter Workflow zum Erstellen oder Aktualisieren eines Hosts. Weitere Hinweise zur allgemeinen Vorgehensweise beim Erstellen einer Verbindung nach Anbietertyp finden Sie unter Arbeiten mit Verbindungen.

Wie AWS CodeConnections funktionieren Verbindungen mit Organisationen

Für Organizations mit einem Anbieter, wie z. B. GitHub Organizations, können Sie eine GitHub App nicht in mehreren GitHub Organisationen installieren. Bei einer Verbindung wird mithilfe der Github-Connector-App eine 1:1 -Zuordnung zu einer Organisation hergestellt. Die Connector-App sollte

für jede Organisation in unserem GitHub GitHub Enterprise Server separat sein und ihr sollte eine Verbindung zugeordnet sein.

Um beispielsweise mit mehreren Organisationen auf demselben GitHub Server zu arbeiten, müssen Sie separate Verbindungen für jede Organisation erstellen und separate GitHub Apps für diese Organisationen installieren. Das Zielkonto auf der Github-Seite kann jedoch dasselbe sein.

Workflow zum Erstellen oder Ändern von Verbindungen

Wenn Sie eine Verbindung herstellen, erstellen oder verwenden Sie auch eine vorhandene Connector-App-Installation für den Auth-Handshake mit dem Drittanbieter.

Verbindungen können die folgenden Status haben:

- `Pending` (Ausstehend) – Eine `pending` (ausstehende) Verbindung ist eine Verbindung, die abgeschlossen werden muss (d. h. in den Zustand `available` (verfügbar) gelangen), bevor sie verwendet werden kann.
- `Available` (Verfügbar) – Sie können eine `available` (verfügbare) Verbindung in Ihrem Konto verwenden oder an andere Ressourcen und Benutzern übergeben.
- `Error` (Fehler) – Ist eine Verbindung im Zustand `error` (Fehler), wird die Erstellung automatisch erneut versucht. Sie kann nicht verwendet werden, bis sie im Status `available` (verfügbar) ist.

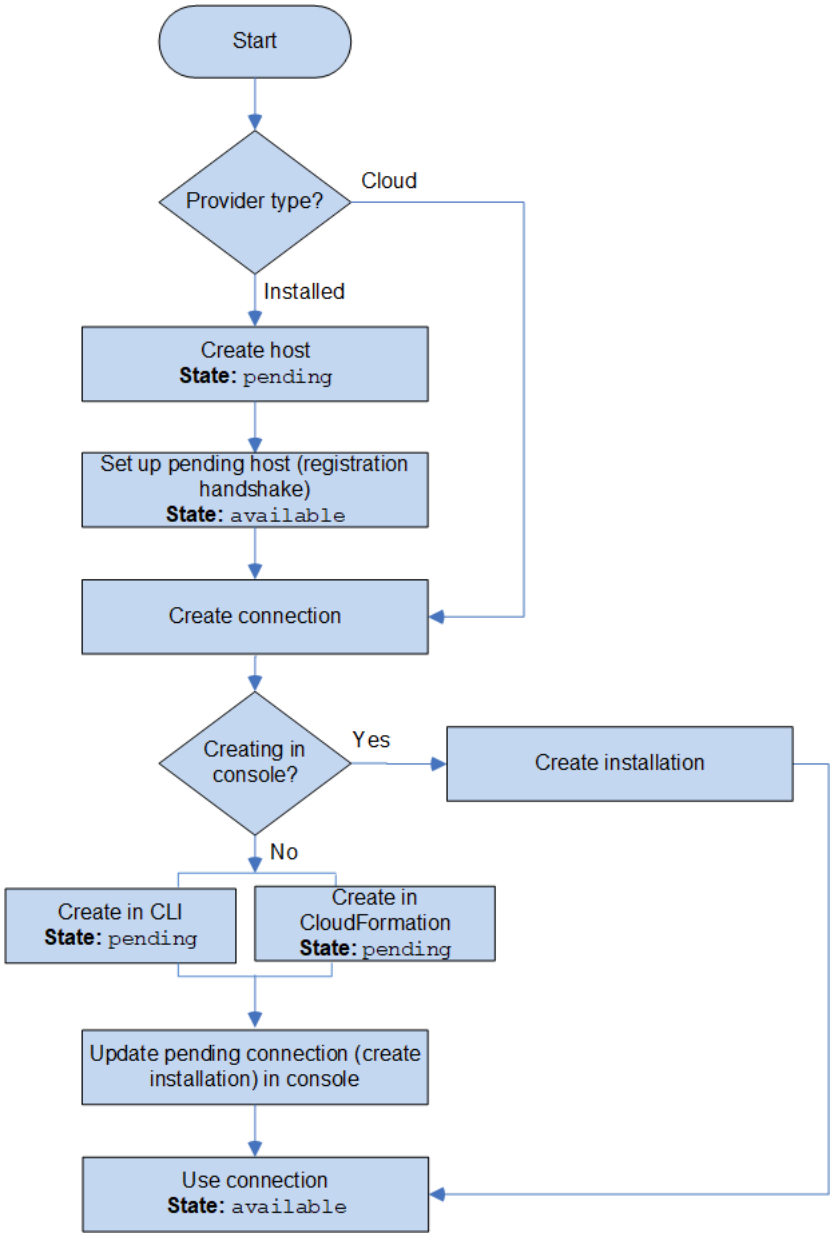
Workflow: Erstellen oder Ändern einer Verbindung mit CLI, SDK oder AWS CloudFormation

Sie verwenden die CreateConnectionAPI, um eine Verbindung mithilfe von AWS Command Line Interface (AWS CLI), SDK oder herzustellen. AWS CloudFormation Nachdem sie erstellt wurde, befindet sich die Verbindung im Zustand `pending` (ausstehend). Sie schließen den Vorgang mithilfe der Option Set up pending connection (Ausstehende Verbindung einrichten) mit der Konsole ab. Die Konsole fordert Sie auf, eine Installation zu erstellen oder eine vorhandene Installation für die Verbindung zu verwenden. Sie verwenden dann die Konsole, um den Handshake abzuschließen und die Verbindung in den Zustand `available` (verfügbar) zu bringen, indem Sie Complete connection (Verbindung abschließen) in der Konsole auswählen.

Workflow: Erstellen oder Ändern einer Verbindung über die Konsole

Wenn Sie eine Verbindung zu einem installierten Anbietertyp wie GitHub Enterprise Server herstellen, erstellen Sie zunächst einen Host. Wenn Sie eine Verbindung zu einem Cloud-Anbietertyp herstellen, z. B. Bitbucket, können Sie das Erstellen des Hosts überspringen und direkt die Verbindung erstellen.

Um mit der Konsole eine Verbindung herzustellen oder zu aktualisieren, verwenden Sie die Aktionsseite CodePipeline „Aktion bearbeiten“ in der Konsole, um Ihren Drittanbieter auszuwählen. Die Konsole fordert Sie auf, eine Installation zu erstellen oder eine vorhandene Installation für die Verbindung zu verwenden. Danach erstellen Sie mit der Konsole die Verbindung. Die Konsole schließt den Handshake ab und ändert den Zustand der Verbindung automatisch von `pending` (ausstehend) zu `available` (verfügbar).



Workflow zum Erstellen oder Aktualisieren eines Hosts

Wenn Sie eine Verbindung für einen installierten Anbieter (vor Ort) herstellen, verwenden Sie eine Hostressource.

> **Note**
>
> Bei Organisationen, die GitHub Enterprise Server nutzen oder GitLab selbst verwaltet werden, geben Sie keinen verfügbaren Host weiter. Sie erstellen für jede Verbindung in Ihrer Organisation einen neuen Host und müssen sicherstellen, dass Sie dieselben Informationen in die Netzwerkfelder (VPC-ID, Subnetz IDs und Sicherheitsgruppe IDs) für den Host eingeben. Weitere Informationen finden Sie unter Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen.

Hosts können die folgenden Status haben:

- `Pending` – Ein Host mit dem Status `pending` ist ein Host, der erstellt wurde und eingerichtet (in den Status `available` versetzt) werden muss, bevor er verwendet werden kann.
- `Available` – Sie können einen Host mit dem Status `available` verwenden oder an Ihre Verbindung übergeben.

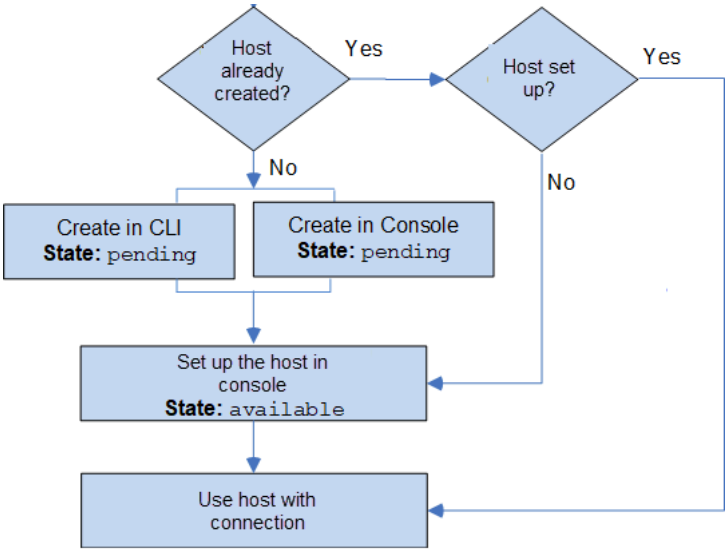
Workflow: Erstellen oder Aktualisieren eines Hosts über die CLI, das SDK oder AWS CloudFormation

Sie verwenden die CreateHostAPI, um einen Host mithilfe von AWS Command Line Interface (AWS CLI), SDK oder zu erstellen. AWS CloudFormation Nachdem der Host erstellt wurde, befindet er sich im Status `pending`. Sie schließen den Vorgang mithilfe der Option Einrichten in der Konsole ab.

Workflow: Erstellen oder Aktualisieren eines Hosts über die Konsole

Wenn Sie eine Verbindung zu einem installierten Anbietertyp herstellen, z. B. GitHub Enterprise Server oder GitLab selbstverwaltet, erstellen Sie einen Host oder verwenden einen vorhandenen Host. Wenn Sie eine Verbindung zu einem Cloud-Anbietertyp herstellen, z. B. Bitbucket, können Sie das Erstellen des Hosts überspringen und direkt die Verbindung erstellen.

Verwenden Sie die Konsole, um den Host einzurichten und seinen Status von `pending` in `available` zu ändern.



Globale Ressourcen in AWS CodeConnections

Verbindungen sind globale Ressourcen. Das bedeutet, dass die Ressource in allen AWS-Regionen repliziert wird.

Obwohl das Format des Verbindungs-ARNs den Namen der Region widerspiegelt, in der sie erstellt wurde, ist die Ressource nicht auf eine Region beschränkt. Die Region, in der die Verbindungsressource erstellt wurde, ist die Region, in der Änderungen an den Daten der Verbindungsressourcen gesteuert werden. Beispiele für API-Vorgänge, die Änderungen an den Daten der Verbindungsressourcen steuern, sind: das Erstellen einer Verbindung, das Aktualisieren einer Installation, das Löschen einer Verbindung oder das Markieren einer Verbindung.

Hostressourcen für Verbindungen sind keine global verfügbaren Ressourcen. Sie verwenden Hostressourcen nur in der Region, in der sie erstellt wurden.

- Sie müssen nur einmal eine Verbindung erstellen und können diese dann in jeder AWS-Region verwenden.
- Wenn in der Region, in der die Verbindung hergestellt wurde, Probleme auftreten, wirkt sich dies auf APIs die Verbindungsressourcendaten aus. Sie können die Verbindung jedoch weiterhin erfolgreich in jeder anderen Region verwenden.
- Wenn Sie Verbindungsressourcen in der Konsole oder der CLI auflisten, werden in der Liste alle Verbindungsressourcen angezeigt, die mit Ihrem Konto in sämtlichen Regionen verknüpft sind.
- Wenn Sie Hostressourcen in der Konsole oder der CLI auflisten, werden in der Liste alle Hostressourcen angezeigt, die mit Ihrem Konto verknüpft sind, jedoch nur in den ausgewählten Regionen.

- Wenn eine Verbindung mit einer zugeordneten Hostressource mit der CLI aufgelistet oder angezeigt wird, gibt die Ausgabe den Host-ARN zurück, unabhängig von der konfigurierten CLI-Region.

Was sind die ersten Schritte mit Verbindungen?

Hier finden Sie einige nützliche Themen für den Einstieg:

- Erfahren Sie mehr über die Konzepte für Verbindungen.
- Richten Sie die benötigten Ressourcen für Verbindungen ein.
- Gehen Sie erste Schritte und erstellen Sie Ihre erste Verbindungen und verbinden Sie sie mit einer Ressource.

Verbindungen – Konzepte

Das Einrichten und Verwenden von Verbindungen ist einfacher, wenn Sie die Konzepte und Begriffe verstehen. Im Folgenden finden Sie einige Konzepte, die Sie kennen sollten, wenn Sie Verbindungen in der Entwicklertools-Konsole verwenden:

installation (Installation)

Eine Instanz der AWS App auf einem Drittanbieter-Konto. Die Installation der AWS CodeStar Connector-App ermöglicht AWS den Zugriff auf Ressourcen innerhalb des Drittanbieterkontos. Eine Installation kann nur auf der Website des Drittanbieters bearbeitet werden.

connection (Verbindung)

Eine AWS Ressource, die verwendet wird, um Quell-Repositorys von Drittanbietern mit anderen AWS Diensten zu verbinden.

third-party repository (Drittanbieter-Repository)

Ein Repository, das von einem Service oder Unternehmen bereitgestellt wird, das nicht Teil von AWS ist. Ein Bitbucket-Repository ist beispielsweise ein Drittanbieter-Repository.

provider type (Anbietertyp)

Ein Service oder Unternehmen, das das Quell-Repository eines Drittanbieters bereitstellt, mit dem Sie eine Verbindung erstellen möchten. Du verbindest deine AWS Ressourcen mit externen

Anbietertypen. Ein Anbietertyp, bei dem das Quell-Repository im Netzwerk und in der Infrastruktur installiert ist, ist ein installierter Anbietertyp. GitHub Enterprise Server ist beispielsweise ein installierter Anbietertyp.

Host

Eine Ressource, die die Infrastruktur darstellt, in der ein Drittanbieter installiert ist. Verbindungen verwenden den Host, um den Server darzustellen, auf dem Ihr Drittanbieter installiert ist, z. B. GitHub Enterprise Server. Sie erstellen einen Host für alle Verbindungen mit diesem Anbietertyp.

> **Note**
>
> Wenn Sie die Konsole verwenden, um eine Verbindung zu GitHub Enterprise Server herzustellen, erstellt die Konsole im Rahmen des Prozesses eine Hostressource für Sie.

AWS CodeConnections unterstützte Anbieter und Versionen

Dieses Kapitel enthält Informationen zu den Anbietern und Versionen, die AWS CodeConnections unterstützt werden.

Themen

- Unterstützter Anbietertyp für Bitbucket
- Unterstützter Anbietertyp für GitHub und Enterprise Cloud GitHub
- Unterstützter Anbietertyp und unterstützte Versionen für GitHub Enterprise Server
- Unterstützter Anbietertyp für .com GitLab
- Unterstützter Anbietertyp für Selbstverwaltung GitLab

Unterstützter Anbietertyp für Bitbucket

Du kannst die Verbindungs-App mit Atlassian Bitbucket Cloud verwenden.

Installierte Bitbucket-Anbietertypen wie Bitbucket Server werden nicht unterstützt.

Unterstützter Anbietertyp für GitHub und Enterprise Cloud GitHub

Sie können die Verbindungs-App mit einer GitHub GitHub Enterprise Cloud verwenden.

Unterstützter Anbietertyp und unterstützte Versionen für GitHub Enterprise Server

Sie können die Verbindungs-App mit unterstützten Versionen von GitHub Enterprise Server verwenden. Eine Liste der unterstützten Versionen finden Sie unter https://enterprise.github.com/releases/.

> **Important**
>
> AWS CodeConnections unterstützt keine veralteten GitHub Enterprise Server-Versionen. AWS CodeConnections Unterstützt beispielsweise GitHub Enterprise Server Version 2.22.0 aufgrund eines bekannten Problems in der Version nicht. Um eine Verbindung zu erstellen, aktualisieren Sie auf Version 2.22.1 bzw. die neueste verfügbare Version.

Unterstützter Anbietertyp für .com GitLab

Sie können Verbindungen mit GitLab .com verwenden. Weitere Informationen finden Sie unter Stellen Sie eine Verbindung her zu GitLab.

> **Important**
>
> Die Verbindungsunterstützung für GitLab umfasst Version 15.x und höher.

Unterstützter Anbietertyp für Selbstverwaltung GitLab

Sie können Verbindungen mit GitLab selbstverwalteter Installation (für Enterprise Edition oder Community Edition) verwenden. Weitere Informationen finden Sie unter Stellen Sie eine Verbindung zu GitLab Selfmanaged her.

Produkt- und Service-Integrationen mit AWS CodeConnections

AWS CodeConnections ist in eine Reihe von AWS Diensten sowie Partnerprodukten und -dienstleistungen integriert. Mithilfe der Informationen in den folgenden Abschnitten können Sie Verbindungen für die Integration in die von Ihnen verwendeten Produkte und Services konfigurieren.

Die folgenden verwandten Ressourcen bieten Ihnen nützliche Informationen für die Arbeit mit diesem Service.

Themen

- CodeGuru Amazon-Rezensent
- Amazon Q Developer
- Amazon SageMaker
- AWS App Runner
- AWS CloudFormation
- AWS CodeBuild
- AWS CodePipeline
- Servicekatalog
- AWS Proton

CodeGuru Amazon-Rezensent

CodeGuru Reviewer ist ein Service zur Überwachung Ihres Repository-Codes. Sie können Verbindungen verwenden, um das Drittanbieter-Repository zu verknüpfen, das den zu überprüfenden Code enthält. Ein Tutorial, in dem Sie erfahren, wie Sie CodeGuru Reviewer so konfigurieren, dass es den Quellcode in einem GitHub Repository überwacht, sodass Empfehlungen zur Verbesserung des Codes erstellt werden können, finden Sie unter Tutorial: Quellcode in einem GitHub Repository überwachen im Amazon CodeGuru Reviewer-Benutzerhandbuch.

Amazon Q Developer

Amazon Q Developer ist ein generativer KI-gestützter Konversationsassistent, der Ihnen helfen kann, Anwendungen zu verstehen, zu erstellen, zu erweitern und zu betreiben AWS . Weitere Informationen finden Sie unter Was ist Amazon Q Developer? im Benutzerhandbuch zu Amazon Q Developer.

Amazon SageMaker

Amazon SageMaker ist ein Service zum Erstellen, Trainieren und Bereitstellen von Sprachmodellen für maschinelles Lernen. Ein Tutorial, in dem Sie eine Verbindung zu Ihrem GitHub Repository konfigurieren, finden Sie unter SageMaker MLOps Project Walkthrough Using Third-Party Git Repos im Amazon SageMaker Developer Guide.

AWS App Runner

AWS App Runner ist ein Service, der eine schnelle, einfache und kostengünstige Möglichkeit der direkten Bereitstellung aus dem Quellcode oder einem Container-Image in einer skalierbaren

und sicheren Webanwendung in der AWS Cloud bietet. Sie können Anwendungscode aus Ihrem Repository mit einer automatischen Integrations- und Bereitstellungspipeline von App Runner bereitstellen. Sie können Verbindungen verwenden, um Ihren Quellcode aus einem privaten GitHub Repository für einen App Runner-Service bereitzustellen. Weitere Informationen finden Sie unter Anbieter des Quellcode-Repository im AWS App Runner -Entwicklerhandbuch.

AWS CloudFormation

AWS CloudFormationist ein Service, der Sie bei der Modellierung und Einrichtung Ihrer AWS Ressourcen unterstützt, sodass Sie weniger Zeit mit der Verwaltung dieser Ressourcen verbringen und sich mehr auf Ihre Anwendungen konzentrieren können, die in ausgeführt AWS werden. Sie erstellen eine Vorlage, die alle gewünschten AWS Ressourcen beschreibt (wie EC2 Amazon-Instances oder Amazon RDS-DB-Instances), und CloudFormation kümmert sich um die Bereitstellung und Konfiguration dieser Ressourcen für Sie.

Sie verwenden Verbindungen mit Git Sync CloudFormation , um eine Synchronisierungskonfiguration zu erstellen, die Ihr Git-Repository überwacht. Ein Tutorial, das dich durch die Verwendung von Git Sync für Stack-Bereitstellungen führt, findest du unter Arbeiten mit CloudFormation Git Sync im AWS CloudFormation Benutzerhandbuch.

Weitere Informationen CloudFormation dazu findest du unter Registrierung deines Kontos für die Veröffentlichung von CloudFormation Erweiterungen im Benutzerhandbuch für die CloudFormation Befehlszeilenschnittstelle.

AWS CodeBuild

AWS CodeBuildist ein Dienst zum Erstellen und Testen Ihres Codes. CodeBuild macht die Bereitstellung, Verwaltung und Skalierung Ihrer eigenen Build-Server überflüssig und bietet vorgefertigte Build-Umgebungen für beliebte Programmiersprachen und Build-Tools. Weitere Informationen zur Verwendung CodeBuild mit Verbindungen zu GitLab finden Sie unter GitLabVerbindungen im AWS CodeBuild Benutzerhandbuch.

AWS CodePipeline

CodePipeline ist ein kontinuierlicher Bereitstellungsservice, mit dem Sie die für die Freigabe Ihrer Software erforderlichen Schritte entwickeln, visualisieren und automatisieren können. Sie können Verbindungen verwenden, um ein Drittanbieter-Repository für CodePipeline Quellaktionen zu konfigurieren.

Weitere Informationen:

- Informationen zur CodePipeline Aktion finden Sie auf der Referenzseite zur `SourceConnections` Aktionskonfiguration. Die Konfigurationsparameter und ein Beispiel für ein JSON/YAML-Snippet finden Sie im Benutzerhandbuch. CodeStarSourceConnectionAWS CodePipeline
- Wenn Sie sich ein Tutorial für den Einstieg ansehen möchten, in dem eine Pipeline mit einem Quell-Repository eines Drittanbieters erstellt wird, beachten Sie Erste Schritte mit Verbindungen.

Servicekatalog

Service Catalog ermöglicht es Unternehmen, Kataloge mit Produkten zu erstellen und zu verwalten, die für die Verwendung auf AWS zugelassen sind.

Wenn du eine Verbindung zwischen dir AWS-Konto und einem externen Repository-Anbieter wie GitHub GitHub Enterprise oder Bitbucket autorisierst, ermöglicht dir die Verbindung, Service Catalog-Produkte mit Vorlagendateien zu synchronisieren, die über Repositorys von Drittanbietern verwaltet werden.

Weitere Informationen findest du unter Synchronisieren von Service Catalog-Produkten mit Vorlagendateien von GitHub GitHub Enterprise oder Bitbucket im Service Catalog-Benutzerhandbuch.

AWS Proton

AWS Proton ist ein Cloud-basierter Service für die Bereitstellung in der Cloud-Infrastruktur. Sie können Verbindungen verwenden, um einen Link zu Ihren Drittanbieter-Repositorys für die Ressourcen in Ihren Vorlagen für AWS Proton zu erstellen. Weitere Informationen finden Sie unter Erstellen eines Link zu Ihrem Repository im AWS Proton -Benutzerhandbuch.

Einrichten von Verbindungen

Führen Sie die Aufgaben in diesem Abschnitt aus, um mit der Einrichtung und Verwendung der Verbindungsfunktion in der Entwicklertools-Konsole loslegen zu können.

Themen

- Melde dich an für AWS
- Eine Richtlinie mit Berechtigungen zum Erstellen von Verbindungen erstellen und anwenden

Melde dich an für AWS

Melde dich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Ein Teil des Anmeldevorgangs umfasst den Empfang eines Telefonanrufs oder einer Textnachricht und die Eingabe eines Bestätigungscodes auf der Telefontastatur.

   Wenn Sie sich für eine anmelden AWS-Konto, wird eine Root-Benutzer des AWS-Kontoserstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu https://aws.amazon.com/gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

Anweisungen finden Sie unter Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer (Konsole) im IAM-Benutzerhandbuch.

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.

2. Gewähren Sie einem Administratorbenutzer im IAM Identity Center Benutzerzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie IAM-Identity-Center-Verzeichnis im Benutzerhandbuch unter Benutzerzugriff mit der Standardeinstellung konfigurieren.AWS IAM Identity Center

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

  Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Access-Portal.

Weiteren Benutzern Zugriff zuweisen

1. Erstellen Sie im IAM-Identity-Center einen Berechtigungssatz, der den bewährten Vorgehensweisen für die Anwendung von geringsten Berechtigungen folgt.

   Anweisungen hierzu finden Sie unter Berechtigungssatz erstellen im AWS IAM Identity Center Benutzerhandbuch.

2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Eine genaue Anleitung finden Sie unter Gruppen hinzufügen im AWS IAM Identity Center Benutzerhandbuch.

Eine Richtlinie mit Berechtigungen zum Erstellen von Verbindungen erstellen und anwenden

So verwenden Sie den JSON-Richtlinieneditor zum Erstellen einer Richtlinie

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die IAM-Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich auf der linken Seite Policies (Richtlinien).

   Wenn Sie zum ersten Mal Policies (Richtlinien) auswählen, erscheint die Seite Welcome to Managed Policies (Willkommen bei verwalteten Richtlinien). Wählen Sie Get Started.
3. Wählen Sie oben auf der Seite Create policy (Richtlinie erstellen) aus.
4. Wählen Sie im Bereich Policy editor (Richtlinien-Editor) die Option JSON aus.
5. Geben Sie folgendes JSON-Richtliniendokument ein:

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "codeconnections:CreateConnection",
               "codeconnections:DeleteConnection",
               "codeconnections:GetConnection",
               "codeconnections:ListConnections",
               "codeconnections:GetInstallationUrl",
               "codeconnections:GetIndividualAccessToken",
               "codeconnections:ListInstallationTargets",
               "codeconnections:StartOAuthHandshake",
               "codeconnections:UpdateConnectionInstallation",
               "codeconnections:UseConnection"
           ],
           "Resource": [
               "*"
           ]
       }
   ]
}
```

6. Wählen Sie Weiter aus.

> **Note**
>
> Sie können jederzeit zwischen den Editoroptionen Visual und JSON wechseln. Wenn Sie jedoch Änderungen vornehmen oder im Visual-Editor Weiter wählen, strukturiert IAM Ihre Richtlinie möglicherweise um, um sie für den visuellen Editor zu optimieren. Weitere Informationen finden Sie unter Richtlinienrestrukturierung im IAM-Benutzerhandbuch.

7. Geben Sie auf der Seite Prüfen und erstellen unter Richtlinienname einen Namen und unter Beschreibung (optional) eine Beschreibung für die Richtlinie ein, die Sie erstellen. Überprüfen Sie Permissions defined in this policy (In dieser Richtlinie definierte Berechtigungen), um die Berechtigungen einzusehen, die von Ihrer Richtlinie gewährt werden.
8. Wählen Sie Create policy (Richtlinie erstellen) aus, um Ihre neue Richtlinie zu speichern.

Erste Schritte mit Verbindungen

Der einfachste Weg, mit Verbindungen zu beginnen, besteht darin, eine Verbindung einzurichten, die Ihr Quell-Repository eines Drittanbieters mit Ihren AWS Ressourcen verknüpft. Wenn Sie Ihre Pipeline beispielsweise mit einer AWS Quelle verbinden möchten CodeCommit, würden Sie als Quellaktion eine Verbindung zu ihr herstellen. Wenn Sie allerdings ein externes Repository haben und das Repository Ihrer Pipeline zuordnen möchten, brauchen Sie eine Verbindung. In diesem Tutorial richten Sie eine Verbindung mit einem Bitbucket-Repository und einer Pipeline ein.

In diesem Abschnitt verwenden Sie Verbindungen mit:

- AWS CodePipeline: In diesen Schritten erstellen Sie eine Pipeline mit Ihrem Bitbucket-Repository als Pipeline-Quelle.
- Amazon CodeGuru Reviewer: Als Nächstes verbindest du dein Bitbucket-Repository mit deinen Feedback- und Analysetools in CodeGuru Reviewer.

Themen

- Voraussetzungen
- Schritt 1: Bearbeiten der Quelldatei
- Schritt 2: Erstellen der Pipeline
- Schritt 3: Verknüpfen Sie Ihr Repository mit CodeGuru Reviewer

Voraussetzungen

Bevor Sie beginnen, führen Sie die Schritte in Einrichtung aus. Du benötigst außerdem ein Quell-Repository eines Drittanbieters, das du mit deinen AWS Diensten verbinden möchtest und das die Verbindung zur Verwaltung der Authentifizierung für dich nutzen möchtest. Beispielsweise möchtest du vielleicht ein Bitbucket-Repository mit deinen AWS Diensten verbinden, die in Quell-Repositorys integriert sind.

- Erstellen Sie ein Bitbucket-Repository mit Ihrem Bitbucket-Konto.
- Halten Sie die Anmeldeinformationen für Bitbucket bereit. Wenn du den verwendest AWS Management Console , um eine Verbindung einzurichten, wirst du aufgefordert, dich mit deinen Bitbucket-Anmeldedaten anzumelden.

Schritt 1: Bearbeiten der Quelldatei

Beim Erstellen eines Bitbucket-Repositorys gibt es auch eine Standard-Datei (`README.md`), die Sie bearbeiten müssen.

1. Melden Sie sich bei Ihrem Bitbucket-Repository an und wählen Sie Source (Quelle) aus.
2. Wählen Sie die Datei `README.md` aus und dann Edit (Bearbeiten) oben auf der Seite. Löschen Sie den vorhandenen Text und geben Sie den folgenden Text ein.

   ```
   This is a Bitbucket repository!
   ```

3. Wählen Sie Commit (Übergeben).

   Die Datei `README.md` muss sich im Stammverzeichnis Ihres Repositorys befinden.

Schritt 2: Erstellen der Pipeline

In diesem Abschnitt erstellen Sie eine Pipeline mit den folgenden Aktionen:

- Eine Quellphase mit einer Verbindung mit dem Bitbucket-Repository und der Aktion.
- Eine Build-Phase mit einer AWS CodeBuild Build-Aktion.

So erstellen Sie mit dem Assistenten eine Pipeline

1. Melden Sie sich bei der CodePipeline Konsole an unter https://console.aws.amazon.com/codepipeline/.
2. Wählen Sie auf der Seite Welcome (Willkommen) die Option Getting started (Erste Schritte) oder auf der Seite Pipelines die Option Create pipeline (Pipeline erstellen).
3. Geben Sie unter Step 1: Choose pipeline settings (Schritt 1: Auswahl der Pipeline-Einstellungen) unter Pipeline name (Pipeline-Name) **`MyBitbucketPipeline`** ein.
4. Wählen Sie unter Service role (Servicerolle) die Option New service role (Neue Servicerolle).

   > **Note**
   >
   > Wenn Sie stattdessen Ihre bestehende CodePipeline Servicerolle verwenden möchten, stellen Sie sicher, dass Sie die `codeconnections:UseConnection` IAM-Berechtigung zu Ihrer Servicerollenrichtlinie hinzugefügt haben. Anweisungen für die CodePipeline Servicerolle finden Sie unter Hinzufügen von Berechtigungen zur CodePipeline Servicerolle.

5. Lassen Sie die Standardwerte bei Erweiterte Einstellungen unverändert. Wählen Sie unter Artifact store (Artefaktspeicher) die Option Default location (Standardstandort) aus, um den Standard-Artefakt-Speicherort für die Pipeline in der entsprechenden Region zu verwenden, beispielsweise mit dem Amazon S3-Artefakt-Bucket als Standard.

   > **Note**
   >
   > Dabei handelt es sich nicht um den Quell-Bucket für Ihren Quellcode, sondern um den Artefaktspeicher für Ihre Pipeline. Für jede Pipeline benötigen Sie einen separaten Artefaktspeicher, z. B. einen S3 Bucket.

   Wählen Sie Next (Weiter).

6. Fügen Sie auf der Seite Step 2: Add source stage (Schritt 2: Quell-Stufe hinzufügen) eine Quellphase hinzu:

   a. Wählen Sie unter Source provider (Quellanbieter) die Option Bitbucket aus.

   b. Wählen Sie unter Connection (Verbindung) die Option Connect to Bitbucket (Verbindung mit Bitbucket erstellen) aus.

c. Geben Sie auf der Seite Connect to Bitbucket (Verbindung mit Bitbucket erstellen) unter Connection name (Verbindungsname) den Namen für die Verbindung ein, die Sie erstellen möchten. Der Name hilft Ihnen, diese Verbindung später zu identifizieren.

   Wählen Sie unter Bitbucket apps (Bitbucket-Apps) die Option Install a new app (Neue App installieren) aus.

d. Auf der App-Installationsseite wird eine Meldung angezeigt, dass die AWS CodeStar App versucht, eine Verbindung zu deinem Bitbucket-Konto herzustellen. Wählen Sie Grant access (Zugriff gewähren). Nachdem du die Verbindung autorisiert hast, werden deine Repositorys auf Bitbucket erkannt und du kannst wählen, ob du eines mit deiner Ressource verknüpfen möchtest. AWS

e. Die Verbindungs-ID für die neue Installation wird angezeigt. Wählen Sie Complete connection (Verbindung abschließen). Du wirst zur Konsole zurückgeleitet. CodePipeline

f. Wählen Sie im Feld Repository name (Repositoryname) den Namen Ihres Bitbucket-Repositorys aus.

g. Wählen Sie unter Branch name (Verzweigungsname) die Verzweigung für Ihr Repository aus.

h. Stellen Sie sicher, dass die Option Starten der Pipeline bei Änderung des Quellcodes ausgewählt ist.

i. Wählen Sie unter Ausgabeartefaktformat eine der folgenden Optionen aus: CodePipeline Standard.

   - Wählen Sie CodePipeline Standard, um das Standard-ZIP-Format für Artefakte in der Pipeline zu verwenden.
   - Wählen Sie Vollständiger Klon aus, um Git-Metadaten über das Repository für Artefakte in die Pipeline aufzunehmen. Dies wird nur für CodeBuild Aktionen unterstützt.

Wählen Sie Weiter.

7. Fügen Sie unter Add build stage (Build-Phase hinzufügen) eine Build-Phase hinzu:

   a. Wählen Sie unter Build provider (Build-Anbieter) die Option AWS CodeBuild aus. Belassen Sie unter Region als Standardeinstellung die Pipeline-Region.

   b. Wählen Sie Create project (Projekt erstellen) aus.

   c. Geben Sie unter Project name (Projektname) einen Namen für dieses Build-Projekt ein.

d. Wählen Sie für Environment image (Umgebungs-Image) die Option Managed image (Verwaltetes Image) aus. Wählen Sie für Operating system (Betriebssystem) die Option Ubuntu aus.

e. Wählen Sie unter Runtime (Laufzeit) die Option Standard aus. Wählen Sie für Image: 5.0aws/codebuild/standard.

f. Wählen Sie unter Service role (Servicerolle) die Option New service role (Neue Servicerolle) aus.

g. Wählen Sie unter BuildSpec bei Build specifications (Build-Spezifikationen) die Option Insert build commands (Build-Befehle einfügen) aus. Wählen Sie Switch to editor (Zum Editor wechseln) aus und fügen Sie Folgendes unter Build commands (Build-Befehle) ein:

```
version: 0.2

phases:
  install:
    #If you use the Ubuntu standard image 2.0 or later, you must specify
 runtime-versions.
    #If you specify runtime-versions and use an image other than Ubuntu
 standard image 2.0, the build fails.
    runtime-versions:
      nodejs: 12
      # name: version
    #commands:
      # - command
      # - command
  pre_build:
    commands:
      - ls -lt
      - cat README.md
  # build:
    #commands:
      # - command
      # - command
  #post_build:
    #commands:
      # - command
      # - command
#artifacts:
  #files:
    # - location
    # - location
```

```
  #name: $(date +%Y-%m-%d)
  #discard-paths: yes
  #base-directory: location
#cache:
  #paths:
    # - paths
```

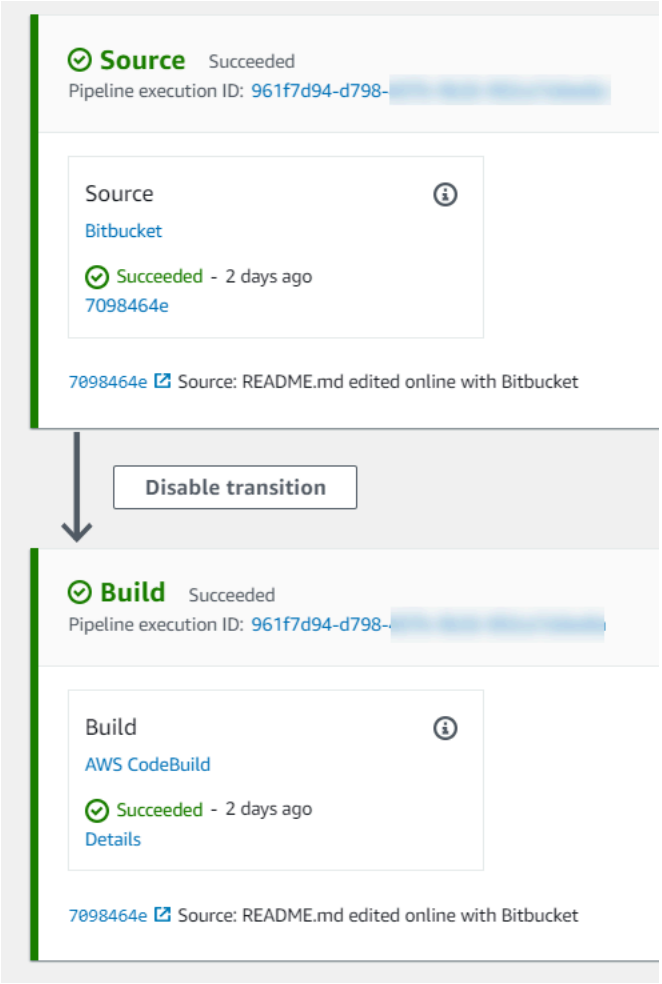
h. Wählen Sie Weiter zu. CodePipeline Dadurch kehren Sie zur CodePipeline Konsole zurück und erstellen ein CodeBuild Projekt, das Ihre Build-Befehle für die Konfiguration verwendet. Das Build-Projekt verwendet eine Servicerolle zur Verwaltung von AWS Dienstberechtigungen. Dieser Vorgang kann einige Minuten dauern.

i. Wählen Sie Weiter.

8. Wählen Sie auf der Seite Step 4: Add deploy stage (Schritt 4: Bereitstellungsstufe hinzufügen) die Option Skip deploy stage (Bereitstellungsstufe überspringen) aus und akzeptieren Sie anschließend die Warnmeldung, indem Sie erneut Skip (Überspringen) auswählen. Wählen Sie Weiter.

9. Wählen Sie unter Step 5: Review (Schritt 5: Überprüfen) die Option Create pipeline (Pipeline erstellen) aus.

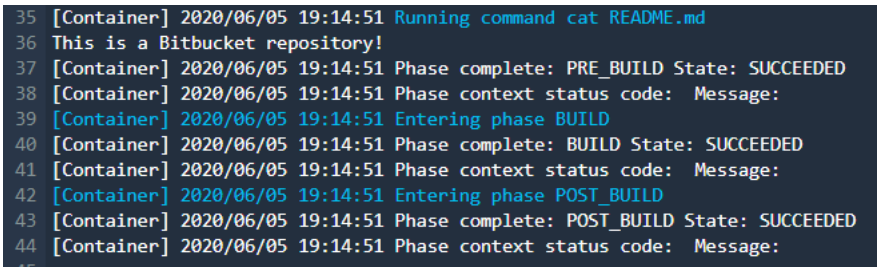
10. Wenn die Pipeline erfolgreich erstellt wurde, wird eine Pipelineausführung gestartet.



11. Wählen Sie auf der erfolgreichen Build-Phase Details aus.

    Sehen Sie sich unter Ausführungsdetails die CodeBuild Build-Ausgabe an. Die Befehle geben den Inhalt der Datei `README.md` von wie folgt aus:

```
This is a Bitbucket repository!
```



Schritt 3: Verknüpfen Sie Ihr Repository mit CodeGuru Reviewer

Nachdem Sie eine Verbindung hergestellt haben, können Sie diese Verbindung für alle Ihre AWS Ressourcen in demselben Konto verwenden. Du kannst beispielsweise dieselbe Bitbucket-Verbindung für eine CodePipeline Quellaktion in einer Pipeline und deine Repository-Commit-Analyse in CodeGuru Reviewer verwenden.

1. Melde dich bei der CodeGuru Reviewer-Konsole an.
2. Wählen Sie unter CodeGuru Reviewer die Option Associate repository aus.

   Der Assistent (eine Seite) wird geöffnet.
3. Wählen Sie unter Select source provider (Quellanbieter auswählen) die Option Bitbucket aus.
4. Wähle unter Connect to Bitbucket (with AWS CodeConnections) die Verbindung aus, die du für deine Pipeline erstellt hast.
5. Wählen Sie unter Repository location (Repository-Speicherort) den Namen Ihres Bitbucket-Repositorys und Associate (zuordnen) aus.

   Sie können weitere Codeüberprüfungen einrichten. Weitere Informationen findest du unter Verbindung zu Bitbucket herstellen, um ein Repository mit CodeGuru Reviewer zu verknüpfen im Amazon CodeGuru Reviewer-Benutzerhandbuch.

Arbeiten mit Verbindungen

Verbindungen sind Konfigurationen, mit denen Sie AWS -Ressourcen mit externen Code-Repositorys verbinden. Jede Verbindung ist eine Ressource, die an Dienste weitergegeben werden kann, z. B. AWS CodePipeline um eine Verbindung zu einem Repository eines Drittanbieters wie Bitbucket herzustellen. Du kannst die Verbindung beispielsweise CodePipeline so hinzufügen, dass sie deine Pipeline auslöst, wenn eine Codeänderung an deinem Code-Repository eines Drittanbieters vorgenommen wird. Sie können Ihre AWS Ressourcen auch mit einem installierten Anbietertyp wie GitHub Enterprise Server verbinden.

> **Note**
>
> Für Organizations in GitHub oder GitHub Enterprise Server können Sie eine GitHub App nicht in mehreren GitHub Organisationen installieren. Bei der Zuordnung zwischen App und GitHub Organisation handelt es sich um eine 1:1 -Zuordnung. Eine Organisation kann jeweils nur über eine App verfügen. Sie können jedoch mehrere Verbindungen haben, die auf dieselbe

App verweisen. Weitere Details erhalten Sie unter Wie AWS CodeConnections funktionieren Verbindungen mit Organisationen.

Wenn Sie eine Verbindung zu einem installierten Anbietertyp wie GitHub Enterprise Server herstellen möchten, erstellt die Konsole einen Host für Sie. Ein Host ist eine Ressource, die den Server darstellt, auf dem Ihr Anbieter installiert ist. Weitere Informationen finden Sie unter Arbeiten mit Hosts.

Wenn Sie eine Verbindung herstellen, verwenden Sie einen Assistenten in der Konsole, um die Verbindungs-App mit Ihrem Drittanbieter zu installieren und sie einer neuen Verbindung zuzuordnen. Wenn Sie die -App bereits installiert haben, können Sie sie verwenden.

Note

Um Verbindungen in Europa (Mailand) nutzen zu können AWS-Region, müssen Sie:

1. Regionsspezifische App installieren
2. Region aktivieren

Diese regionsspezifische App unterstützt Verbindungen in der Region Europa (Mailand). Sie ist auf der Website des Drittanbieters veröffentlicht und von der bestehenden App getrennt, die Verbindungen für andere Regionen unterstützt. Durch die Installation dieser App autorisieren Sie Drittanbieter, Ihre Daten nur für diese Region an den Dienst weiterzugeben, und Sie können diese Autorisierung jederzeit widerrufen, indem Sie die App deinstallieren. Der Dienst verarbeitet oder speichert Ihre Daten nicht, es sei denn, Sie aktivieren die Region. Durch die Aktivierung dieser Region gewähren Sie unserem Dienst die Erlaubnis, Ihre Daten zu verarbeiten und zu speichern.
Auch wenn die Region nicht aktiviert ist, können Drittanbieter Ihre Daten trotzdem mit unserem Dienst teilen, wenn die regionsspezifische App installiert bleibt. Achten Sie also darauf, die App zu deinstallieren, sobald Sie die Region deaktivieren. Weitere Informationen finden Sie unter Aktivieren einer Region.

Weitere Informationen zu Verbindungen finden Sie in der AWS CodeConnections API-Referenz. Weitere Informationen zur CodePipeline Quellaktion für Bitbucket findest du CodestarConnectionSourceim AWS CodePipeline Benutzerhandbuch.

Informationen zum Erstellen oder Anhängen einer Richtlinie für deinen AWS Identity and Access Management (IAM-) Benutzer oder deine Rolle mit den für die Verwendung von Verbindungen erforderlichen Berechtigungen findest du unter. AWS CodeConnections Referenz zu Berechtigungen Je nachdem, wann Ihre CodePipeline Servicerolle erstellt wurde, müssen Sie möglicherweise ihre Berechtigungen auf Support AWS CodeConnections aktualisieren. Eine genaue Anleitung finden Sie unter Update the service role (Ändern der Servicerolle) im AWS CodePipeline -Benutzerhandbuch.

Themen

- Eine Verbindung erstellen
- Erstellen einer Verbindung mit Bitbucket
- Stellen Sie eine Verbindung her zu GitHub
- Stellen Sie eine Verbindung zu GitHub Enterprise Server her
- Stellen Sie eine Verbindung her zu GitLab
- Stellen Sie eine Verbindung zu GitLab Selfmanaged her
- Aktualisieren einer ausstehenden Verbindung
- Auflisten von Verbindungen
- Eine Verbindung löschen
- Ressourcen für Tag-Verbindungen
- Anzeigen von Verbindungsdetails
- Verbindungen teilen mit AWS-Konten

Eine Verbindung erstellen

Sie können Verbindungen mit den folgenden Drittanbietertypen erstellen:

- Informationen zum Erstellen einer Verbindung mit Bitbucket finden Sie unter Erstellen einer Verbindung mit Bitbucket.
- Informationen zum Herstellen einer Verbindung zur GitHub GitHub Enterprise Cloud finden Sie unterStellen Sie eine Verbindung her zu GitHub.
- Informationen zum Herstellen einer Verbindung zu GitHub Enterprise Server, einschließlich der Erstellung Ihrer Host-Ressource, finden Sie unterStellen Sie eine Verbindung zu GitHub Enterprise Server her.
- Informationen zum Herstellen einer Verbindung zu GitLab finden Sie unterStellen Sie eine Verbindung her zu GitLab.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Erstellen einer Verbindung mit Bitbucket

Du kannst das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu einem auf bitbucket.org gehosteten Repository herzustellen.

Bevor Sie beginnen:

- Sie benötigen ein Konto bei Bitbucket.
- Sie müssen bereits ein Code-Repository auf bitbucket.org haben.

> **Note**
>
> Sie können Verbindungen mit einem Bitbucket-Cloud-Repository erstellen. Installierte Bitbucket-Anbietertypen wie Bitbucket Server werden nicht unterstützt. Siehe AWS CodeConnections unterstützte Anbieter und Versionen.

> **Note**
>
> Verbindungen bieten nur Zugriff auf Repositorys, die dem Konto gehören, das zum Erstellen der Verbindung verwendet wurde.
> Wenn die Anwendung in einem Bitbucket-Arbeitsbereich installiert wird, benötigen Sie die Berechtigungen „Workspace verwalten“. Andernfalls wird die Option zum Installieren der App nicht angezeigt.

Themen

- Erstellen einer Verbindung mit Bitbucket (Konsole)
- Erstellen einer Verbindung mit Bitbucket (CLI)

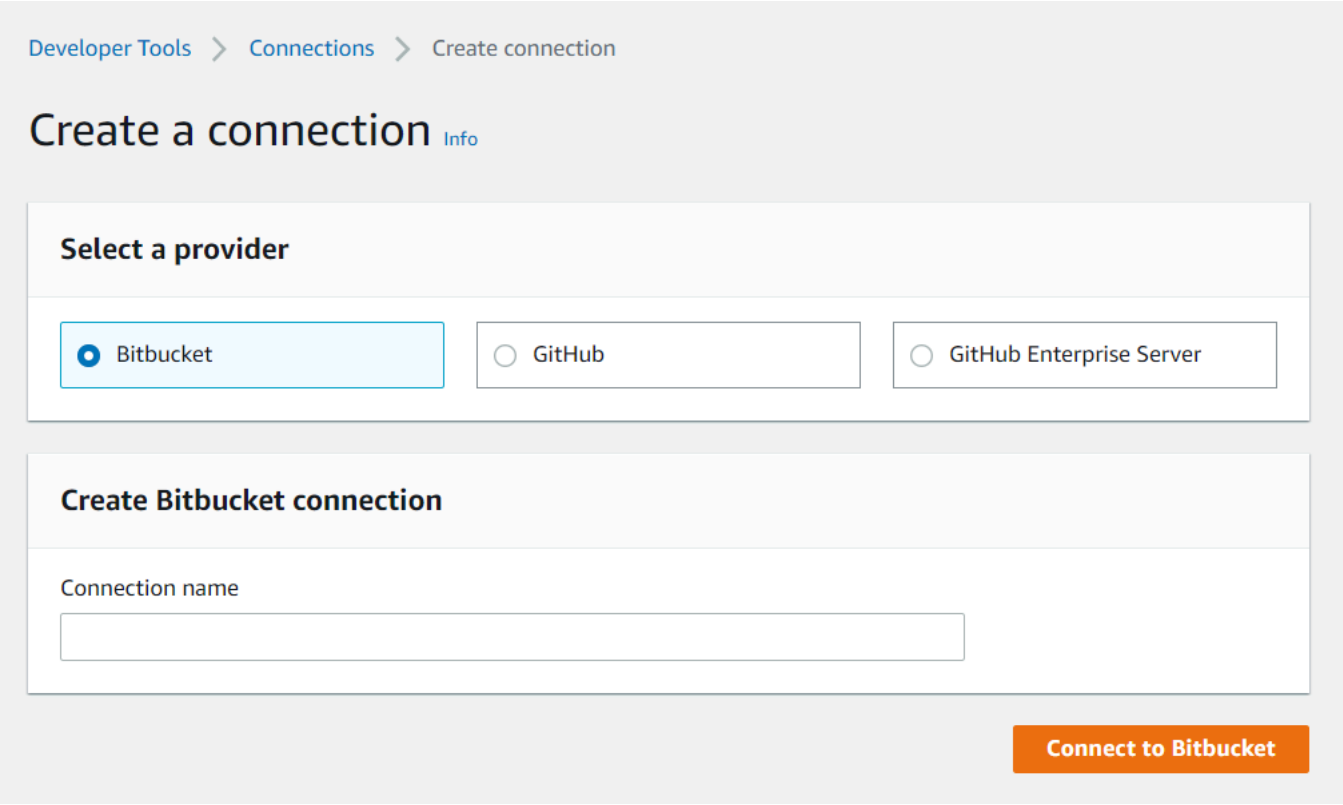
Erstellen einer Verbindung mit Bitbucket (Konsole)

Du kannst die Konsole verwenden, um eine Verbindung zu Bitbucket herzustellen.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Schritt 1: Erstellen einer Verbindung

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die AWS Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) und danach Create connection (Verbindung erstellen).
3. Um eine Verbindung zu einem Bitbucket-Repository zu erstellen, wählen Sie unter Select a provider (Anbieter auswählen) die Option Bitbucket aus. Geben Sie unter Connection name (Verbindungsname) den Namen für die Verbindung ein, die Sie erstellen möchten. Wählen Sie Connect to Bitbucket (Verbindung mit Bitbucket erstellen) und fahren Sie mit Schritt 2 fort.

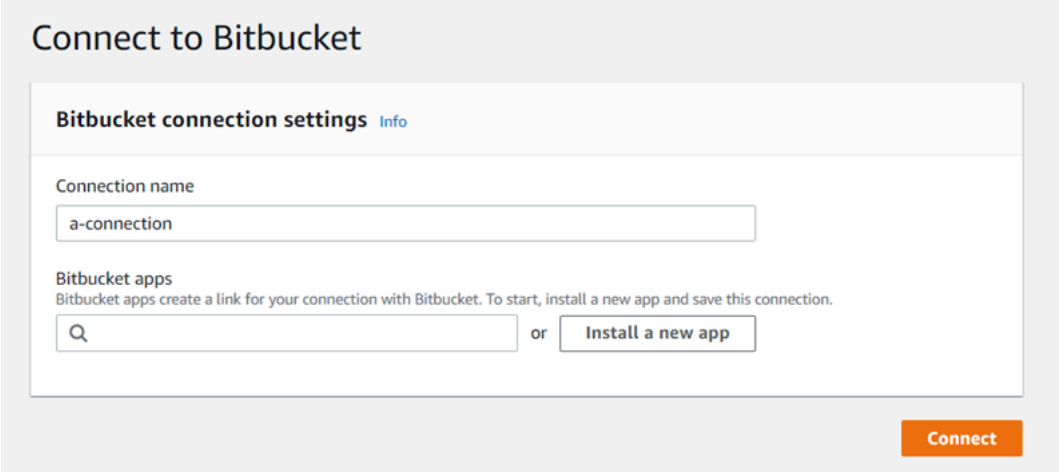


Schritt 2: Eine Verbindung mit Bitbucket erstellen

1. Auf der Einstellungsseite Connect to Bitbucket (Verbindung mit Bitbucket erstellen) wird der Verbindungsname angezeigt.

   Wählen Sie unter Bitbucket apps (Bitbucket-Apps) eine App-Installation aus oder wählen Sie Install a new app (Neue App installieren), um eine App zu erstellen.

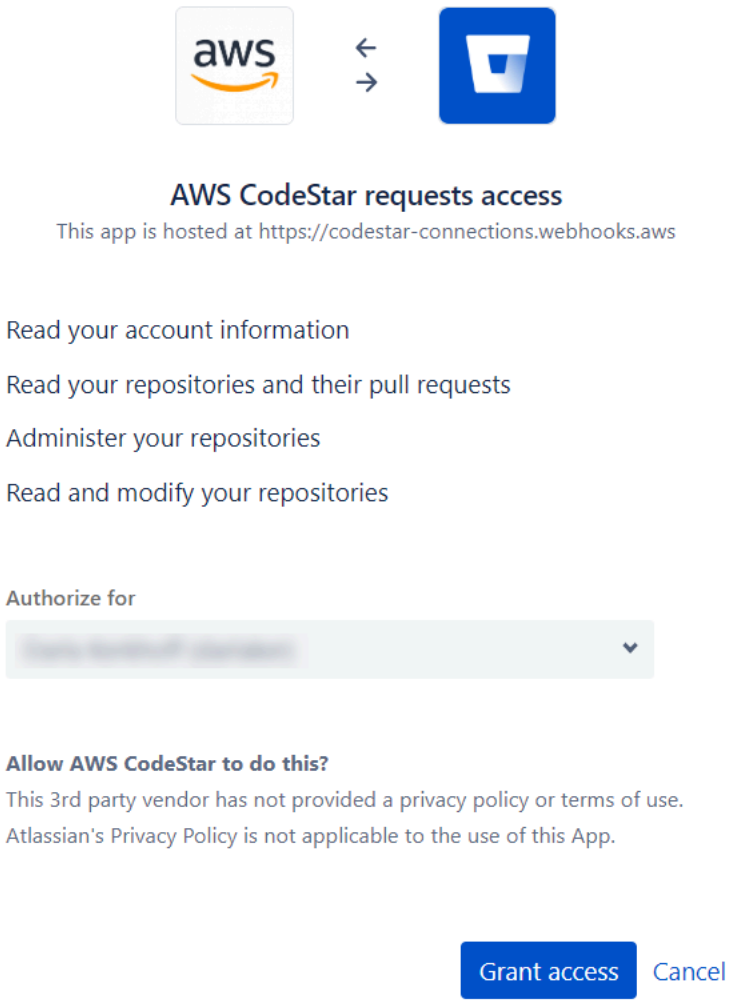
   > **Note**
   >
   > Sie installieren die App pro Bitbucket-Workspace oder nur einmal. Wenn Sie die Bitbucket-App bereits installiert haben, wählen Sie sie aus und wechseln Sie zum letzten Schritt in diesem Abschnitt.



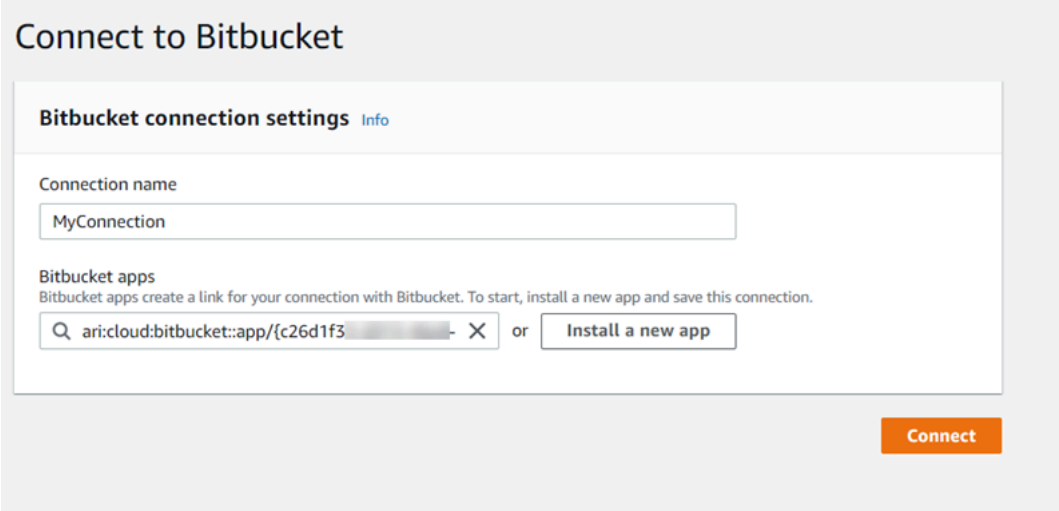
2. Wenn die Anmeldeseite für Bitbucket angezeigt wird, melden Sie sich mit Ihren Anmeldeinformationen an und fahren Sie fort.
3. Auf der App-Installationsseite wird eine Meldung angezeigt, dass die AWS CodeStar App versucht, eine Verbindung zu deinem Bitbucket-Konto herzustellen.

   Wenn Sie einen Bitbucket-Workspace verwenden, ändern Sie die Option Authorize for (Autorisieren für) auf den Workspace. Es werden nur Workspaces angezeigt, für die Sie über den Administratorzugriff verfügen.

   Wählen Sie Grant access (Zugriff gewähren).



4. In der Bitbucket-App wird die Verbindungs-ID für die neue Installation angezeigt. Wählen Sie Connect aus. Die erstellte Verbindung wird in der Verbindungsliste angezeigt.



Erstellen einer Verbindung mit Bitbucket (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung herzustellen.

Verwenden Sie dazu den Befehl create-connection.

> **Important**
>
> Eine Verbindung, die über AWS CLI oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit der CLI hergestellt haben oder verwenden Sie die Konsole AWS CloudFormation, um die Verbindung so zu bearbeiten, dass sie ihren Status festlegt`AVAILABLE`.

Eine Verbindung mit Bitbucket erstellen

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-connection Befehl auszuführen, und geben Sie dabei das `--provider-type` und `--connection-name` für Ihre Verbindung an. In diesem Beispiel lautet der Name des Drittanbieters `Bitbucket` und der angegebene Verbindungsname `MyConnection`.

   ```
   aws codeconnections create-connection --provider-type Bitbucket --connection-name
    MyConnection
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

   ```
   {
       "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
   aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"
   }
   ```

2. Verwenden Sie die Konsole, um die Verbindung fertigzustellen. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Stellen Sie eine Verbindung her zu GitHub

Sie können das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu herzustellen GitHub.

Bevor Sie beginnen:

- Sie müssen bereits ein Konto bei erstellt haben GitHub.
- Sie benötigen ein Code-Repository eines Drittanbieters.

> **Note**
>
> Um die Verbindung herzustellen, müssen Sie der Eigentümer der GitHub Organisation sein. Bei Repositorys, die keiner Organisation angehören, müssen Sie der Repository-Besitzer sein.

Themen

- Stellen Sie eine Verbindung zu GitHub (Konsole) her
- Verbindung herstellen zu GitHub (CLI)

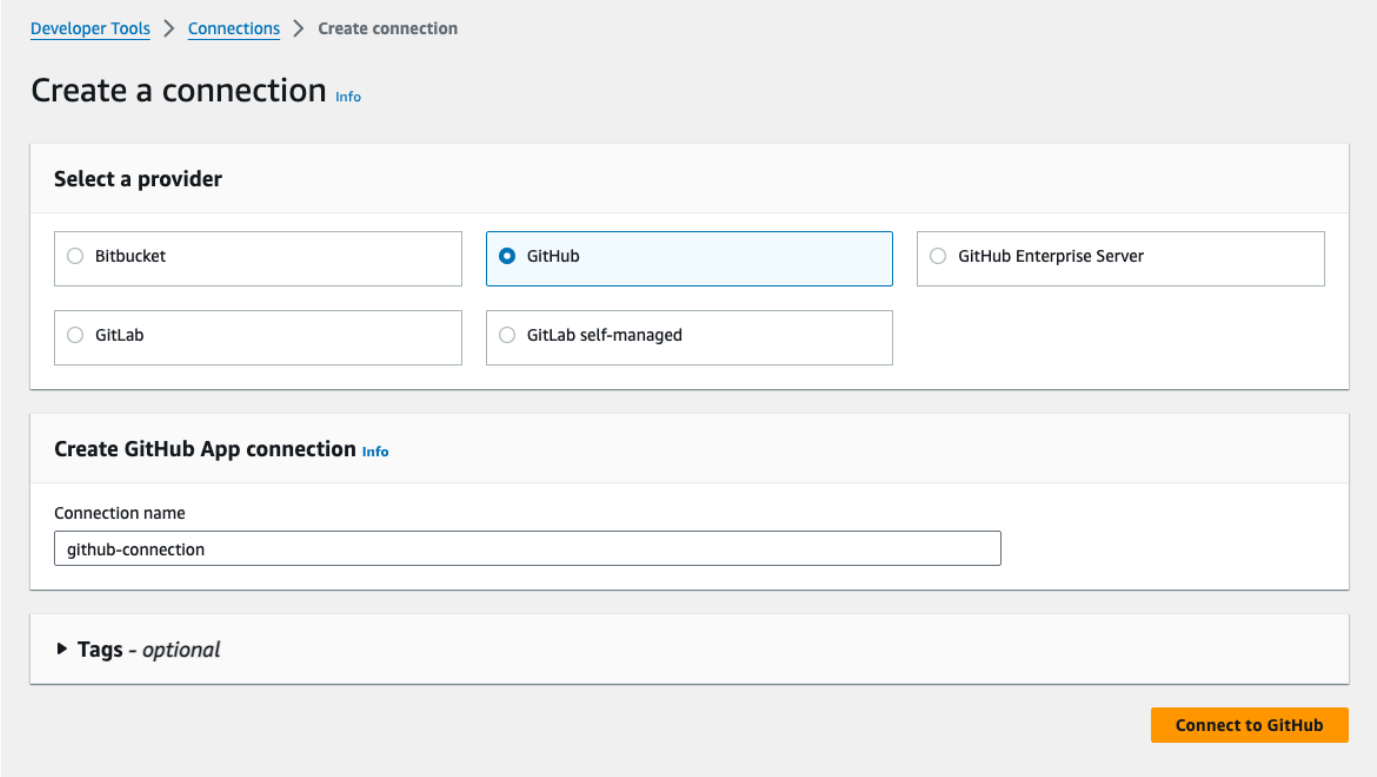
Stellen Sie eine Verbindung zu GitHub (Konsole) her

Sie können die Konsole verwenden, um eine Verbindung zu herzustellen GitHub.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

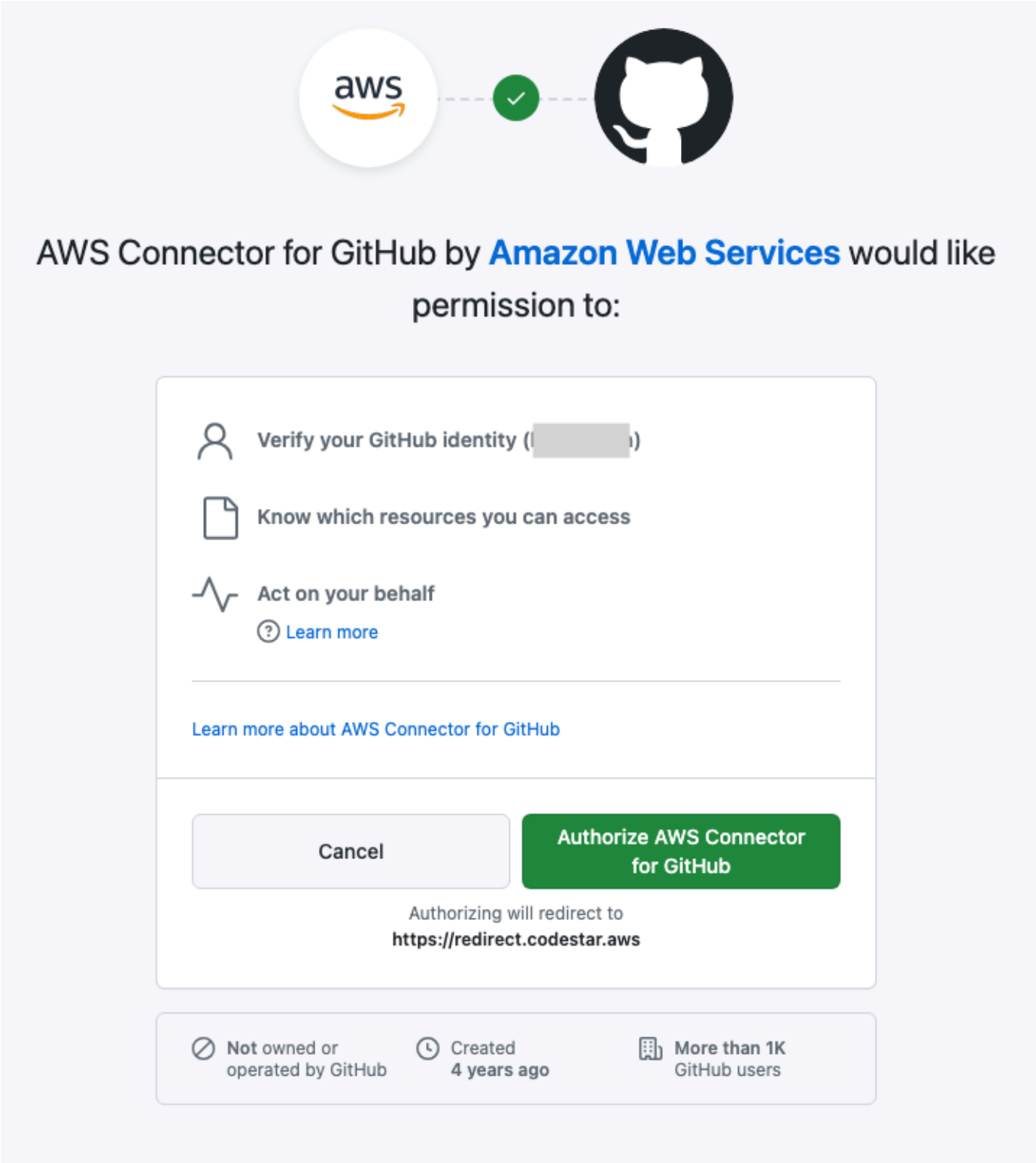
1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) und danach Create connection (Verbindung erstellen).
3. Um eine Verbindung zu einem GitHub oder GitHub Enterprise Cloud-Repository herzustellen, wählen Sie unter Anbieter auswählen die Option GitHub. Geben Sie unter Connection name

(Verbindungsname) den Namen für die Verbindung ein, die Sie erstellen möchten. Wählen Sie GitHub Connect mit und fahren Sie mit Schritt 2 fort.

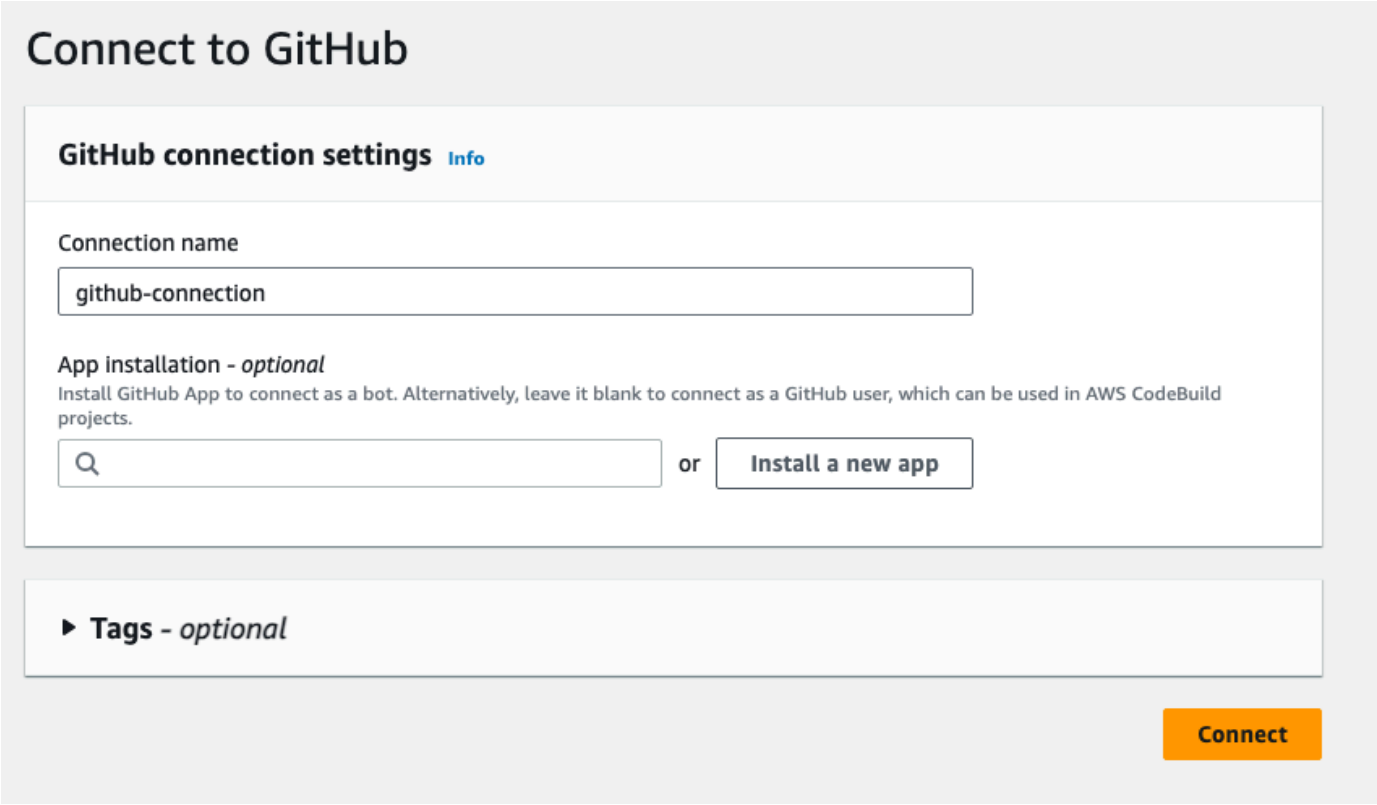


Um eine Verbindung herzustellen zu GitHub

1. Unter GitHub Verbindungseinstellungen wird Ihr Verbindungsname unter Verbindungsname angezeigt. Wählen Sie Connect to GitHub. Die Seite für die Zugriffsanforderung wird angezeigt.



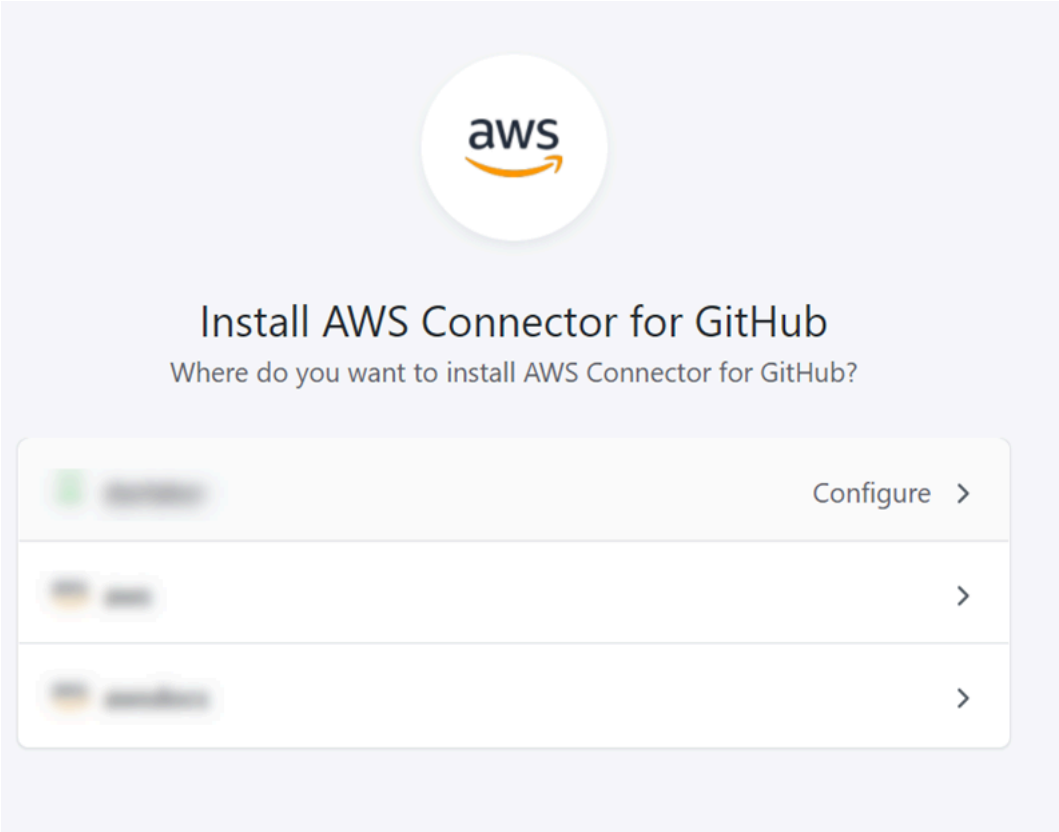
2. Wählen Sie AWS Connector autorisieren für GitHub. Auf der Verbindungsseite wird das Feld GitHub Apps angezeigt und angezeigt.



3. Wählen Sie unter GitHub Apps eine App-Installation aus oder wählen Sie Neue App installieren, um eine zu erstellen.

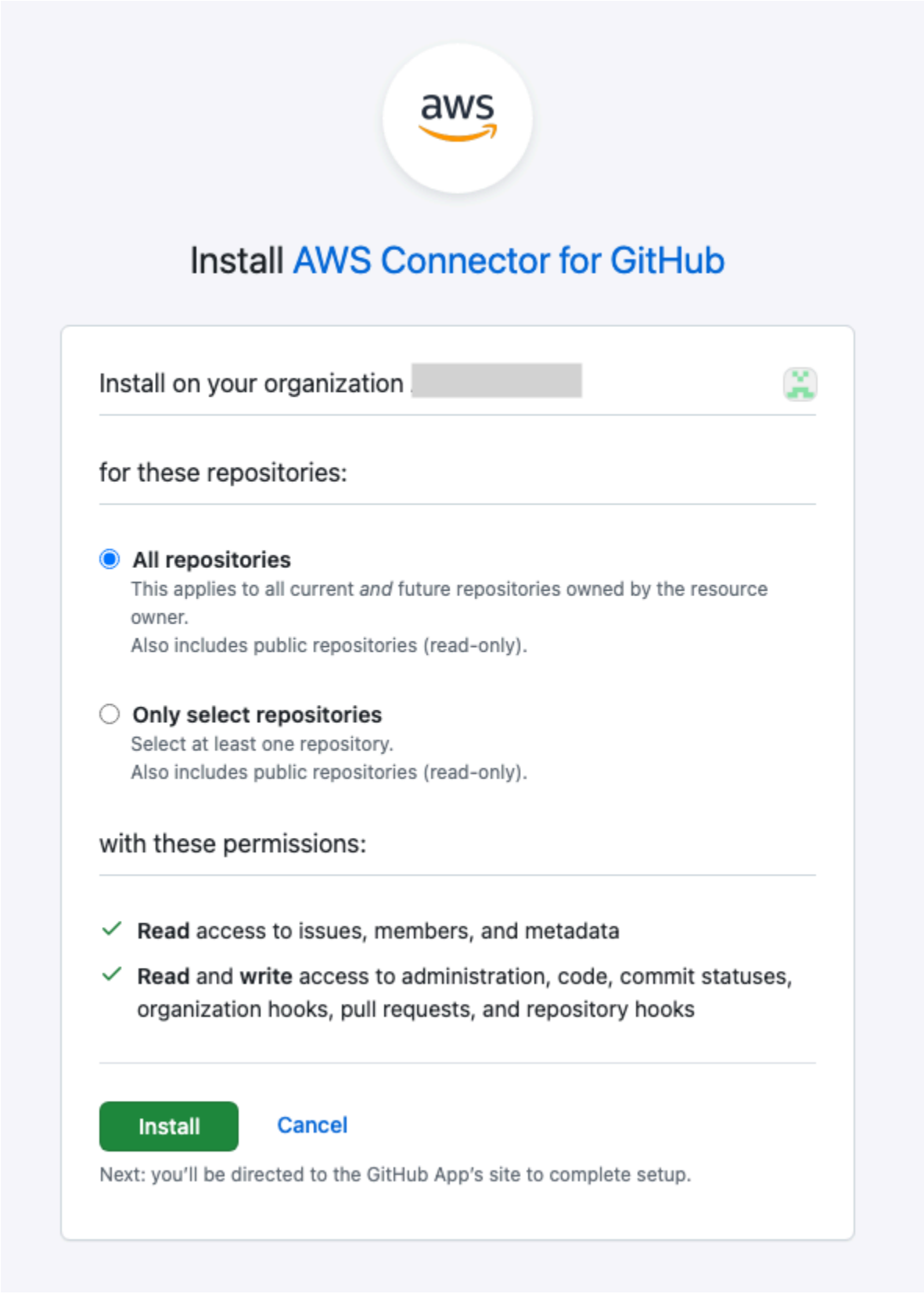
> **Note**
>
> Sie installieren eine App für alle Verbindungen mit einem bestimmten Anbieter. Wenn Sie den AWS Connector für GitHub App bereits installiert haben, wählen Sie ihn aus und überspringen Sie diesen Schritt.

4. Wählen Sie auf der GitHub Seite AWS Connector installieren für das Konto aus, in dem Sie die App installieren möchten.



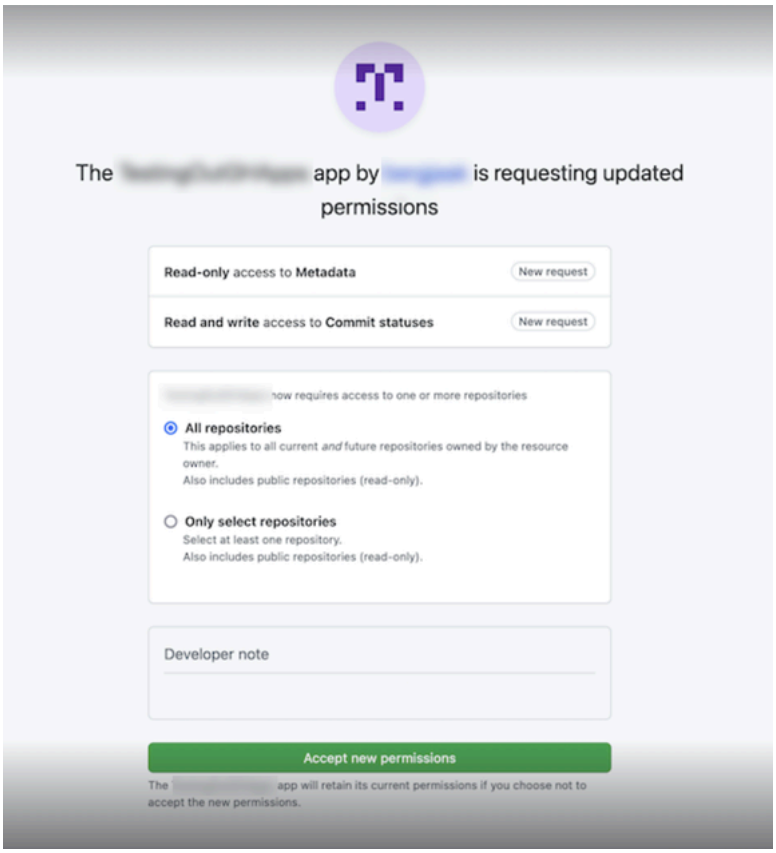
> **Note**
>
> Sie installieren die App nur einmal für jedes GitHub Konto. Wenn Sie die App schon einmal installiert haben, können Sie Configure (Konfiguration) wählen und mit einer Änderungsseite für die App-Installation fortfahren. Alternativ kommen Sie über die Schaltfläche „Back“ (Zurück) zur Konsole zurück.

5. Behalten Sie auf der GitHub Seite „ AWS Connector installieren für“ die Standardeinstellungen bei und wählen Sie Installieren aus.


aws
Install AWS Connector for GitHub
Install on your organization
for these repositories:
All repositories
This applies to all current and future repositories owned by the resource owner.
Also includes public repositories (read-only).
Only select repositories
Select at least one repository.
Also includes public repositories (read-only).
with these permissions:
Read access to issues, members, and metadata
Read and write access to administration, code, commit statuses, organization hooks, pull requests, and repository hooks
Install
Cancel
Next: you'll be directed to the GitHub App's site to complete setup.

Nach diesem Schritt wird möglicherweise eine aktualisierte Berechtigungsseite unter angezeigt GitHub.

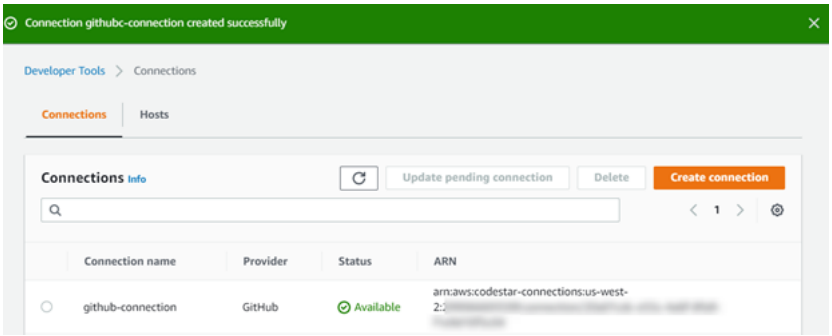
6. Wenn eine Seite mit aktualisierten Berechtigungen für den AWS Connector für die GitHub App angezeigt wird, wählen Sie Neue Berechtigungen akzeptieren aus.



7. Sie kehren zur GitHub Seite Connect to zurück. Die Verbindungs-ID für Ihre neue Installation wird in GitHubApps angezeigt. Wählen Sie Connect aus.

Anzeigen der erstellten Verbindung

- Die erstellte Verbindung wird in der Verbindungsliste angezeigt.



Verbindung herstellen zu GitHub (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu herzustellen GitHub.

Verwenden Sie dazu den Befehl create-connection.

> **Important**
>
> Eine Verbindung, die über AWS CLI oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit der CLI hergestellt haben oder verwenden Sie die Konsole AWS CloudFormation, um die Verbindung so zu bearbeiten, dass sie ihren Status festlegt`AVAILABLE`.

Um eine Verbindung herzustellen zu GitHub

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-connection Befehl auszuführen, und geben Sie dabei `--provider-type` und `--connection-name` für Ihre Verbindung an. In diesem Beispiel lautet der Name des Drittanbieters `GitHub` und der angegebene Verbindungsname `MyConnection`.

   ```
   aws codeconnections create-connection --provider-type GitHub --connection-name
    MyConnection
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

   ```
   {
       "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
   aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"
   }
   ```

2. Verwenden Sie die Konsole, um die Verbindung fertigzustellen. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Stellen Sie eine Verbindung zu GitHub Enterprise Server her

Sie verwenden Verbindungen, um Ihre AWS Ressourcen mit einem Drittanbieter-Repository zu verknüpfen. Sie können das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu GitHub Enterprise Server herzustellen.

Verbindungen bieten nur Zugriff auf Repositorys, die dem GitHub Enterprise Server-Konto gehören, das bei der Verbindungserstellung verwendet wird, um die Installation der GitHub App zu autorisieren.

Bevor Sie beginnen:

- Sie müssen bereits über eine GitHub Enterprise Server-Instanz und ein Repository verfügen.
- Sie müssen Administrator der GitHub Enterprise Server-Instanz sein, um GitHub Apps zu erstellen und eine Host-Ressource zu erstellen, wie in diesem Abschnitt gezeigt.

> **Important**
>
> Wenn Sie Ihren Host für GitHub Enterprise Server einrichten, wird ein VPC-Endpunkt für Webhooks-Ereignisdaten für Sie erstellt. Wenn Sie Ihren Host vor dem 24. November 2020 erstellt haben und PrivateLink VPC-Webhook-Endpunkte verwenden möchten, müssen Sie zuerst Ihren Host löschen und dann einen neuen Host erstellen.

> **Note**
>
> Für Organisationen, die GitHub Enterprise Server nutzen oder GitLab selbst verwaltet werden, geben Sie keinen verfügbaren Host weiter. Sie erstellen für jede Verbindung in Ihrer Organisation einen neuen Host und müssen sicherstellen, dass Sie dieselben Informationen in die Netzwerkfelder (VPC-ID, Subnetz IDs und Sicherheitsgruppe IDs) für den Host eingeben. Weitere Informationen finden Sie unter Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen.

Themen

- Stellen Sie eine Verbindung zum GitHub Enterprise Server (Konsole) her
- Verbindung zu GitHub Enterprise Server (CLI) herstellen

Stellen Sie eine Verbindung zum GitHub Enterprise Server (Konsole) her

Um eine GitHub Enterprise Server-Verbindung herzustellen, geben Sie Informationen darüber an, wo Ihr GitHub Enterprise Server installiert ist, und autorisieren die Verbindungsherstellung mit Ihren GitHub Enterprise-Anmeldeinformationen.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Themen

- Erstellen Sie Ihre GitHub Enterprise Server-Verbindung (Konsole)

Erstellen Sie Ihre GitHub Enterprise Server-Verbindung (Konsole)

Halten Sie Ihre Server-URL und Ihre GitHub Enterprise-Anmeldeinformationen bereit, um eine Verbindung zu GitHub Enterprise Server herzustellen.

Einen Host erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die AWS Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie auf der Registerkarte Hosts die Option Create Host (Host erstellen) aus.
3. Geben Sie unter Host name (Host-Name) den gewünschten Namen für Ihren Host ein.
4. Wählen Sie unter Anbieter auswählen eine der folgenden Optionen aus:
   - GitHub Enterprise Server
   - GitLab selbst verwaltet
5. Geben Sie unter URL den Endpunkt für die Infrastruktur ein, auf der der Anbieter installiert ist.
6. Wenn Ihr Server in einer Amazon VPC konfiguriert ist und Sie eine Verbindung mit Ihrer VPC erstellen möchten, wählen Sie Use a VPC (VPC verwenden) aus. Wählen Sie andernfalls No VPC (Keine VPC) aus.
7. Wenn Sie Ihre Instance in einer Amazon VPC gestartet haben und eine Verbindung mit Ihrer VPC erstellen möchten, wählen Sie Use a VPC (Verwenden einer VPC) aus und geben Sie Folgendes ein.

a. Wählen Sie unter VPC ID Ihre VPC-ID aus. Stellen Sie sicher, dass Sie die VPC für die Infrastruktur wählen, in der Ihre Instance installiert ist, oder eine VPC, die über VPN oder Direct Connect Zugriff auf Ihre Instance hat.

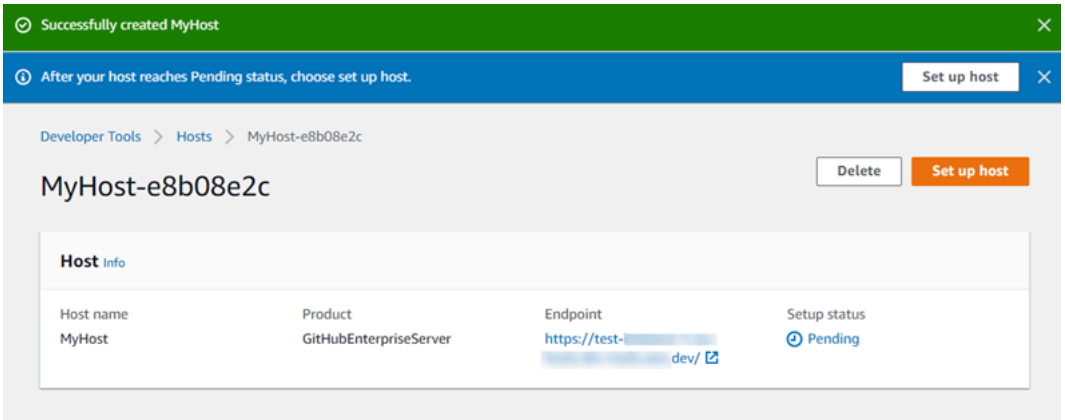
b. Wenn Sie eine private VPC konfiguriert haben und Ihre Instance so konfiguriert haben, dass eine TLS-Validierung bei einer nicht öffentlichen Zertifizierungsstelle durchgeführt wird, geben Sie unter TLS-Zertifikat Ihre Zertifikat-ID ein. Der Wert des TLS-Zertifikats ist der öffentliche Schlüssel des Zertifikats.

8. Wählen Sie Create hoste (Host erstellen) aus.

9. Sobald die Seite mit den Host-Details angezeigt wird, ändert sich der Status des erstellten Hosts.

> **Note**
>
> Wenn Ihr Host-Setup eine VPC-Konfiguration enthält, können Sie mehrere Minuten für die Bereitstellung von Hostnetzwerkkomponenten einplanen.

Warten Sie, bis Ihr Host in den Status Pending (Ausstehend) wechseln und schließen Sie das Setup ab. Weitere Informationen finden Sie unter Einrichten eines ausstehenden Hosts.



Schritt 2: Stellen Sie Ihre Verbindung zum GitHub Enterprise Server (Konsole) her

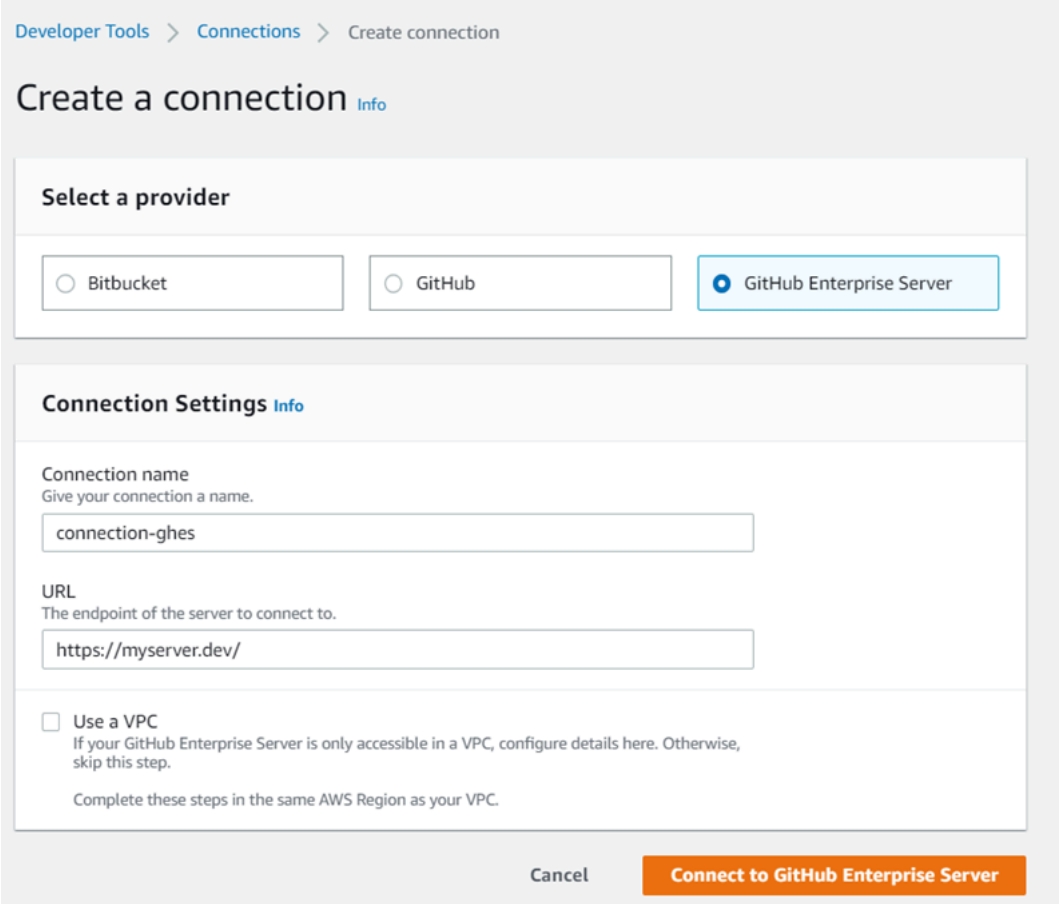
1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.

2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) und danach Create connection (Verbindung erstellen).

3. Um eine Verbindung zu einem installierten GitHub Enterprise Server-Repository herzustellen, wählen Sie GitHub Enterprise Server.

Connect zum GitHub Enterprise Server her

1. Geben Sie unter Connection Name (Verbindungsname) den Namen für die Verbindung ein.



2. Geben Sie unter URL den Endpunkt für Ihren Server ein.

> **Note**
>
> Wenn die angegebene URL bereits verwendet wurde, um einen GitHub Enterprise Server für eine Verbindung einzurichten, werden Sie aufgefordert, den Host-Ressourcen-ARN auszuwählen, der zuvor für diesen Endpunkt erstellt wurde.

3. (Optional) Wenn Sie Ihren Server in einer Amazon VPC gestartet haben und eine Verbindung mit Ihrer VPC herstellen möchten, wählen Sie VPC verwenden aus und führen Sie folgende Schritte aus.

> **Note**
>
> Für Organisationen, die GitHub Enterprise Server nutzen oder GitLab selbst verwaltet werden, geben Sie keinen verfügbaren Host weiter. Sie erstellen für jede Verbindung in Ihrer Organisation einen neuen Host und müssen sicherstellen, dass Sie dieselben Informationen in die Netzwerkfelder (VPC-ID, Subnetz IDs und Sicherheitsgruppe IDs) für den Host eingeben. Weitere Informationen finden Sie unter Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen.

a. Wählen Sie unter VPC ID Ihre VPC-ID aus. Stellen Sie sicher, dass Sie die VPC für die Infrastruktur wählen, in der Ihre GitHub Enterprise Server-Instance installiert ist, oder eine VPC mit Zugriff auf Ihre GitHub Enterprise Server-Instance über VPN oder Direct Connect.

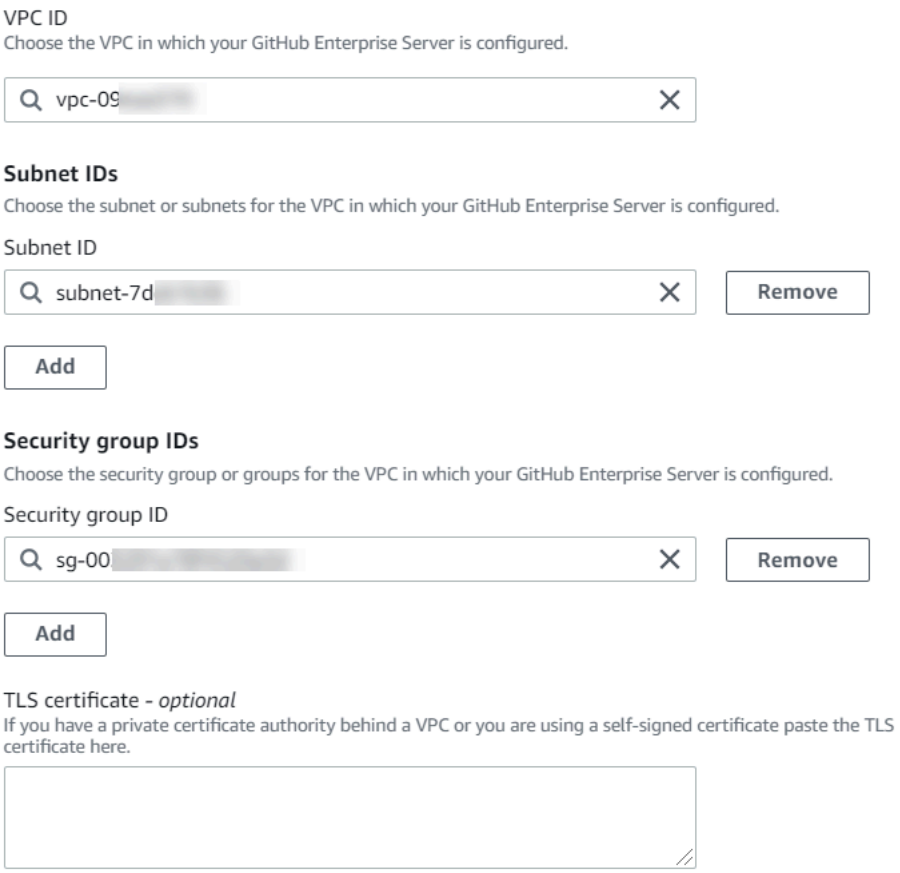
b. Wählen Sie unter Subnetz-ID (Subnetz-ID) die Option Add (Hinzufügen) aus. Wählen Sie im Feld die Subnetz-ID aus, die Sie für Ihren Host verwenden möchten. Sie können bis zu 10 Subnetze wählen.

   Stellen Sie sicher, dass Sie das Subnetz für die Infrastruktur auswählen, in der Ihre GitHub Enterprise Server-Instanz installiert ist, oder ein Subnetz mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.

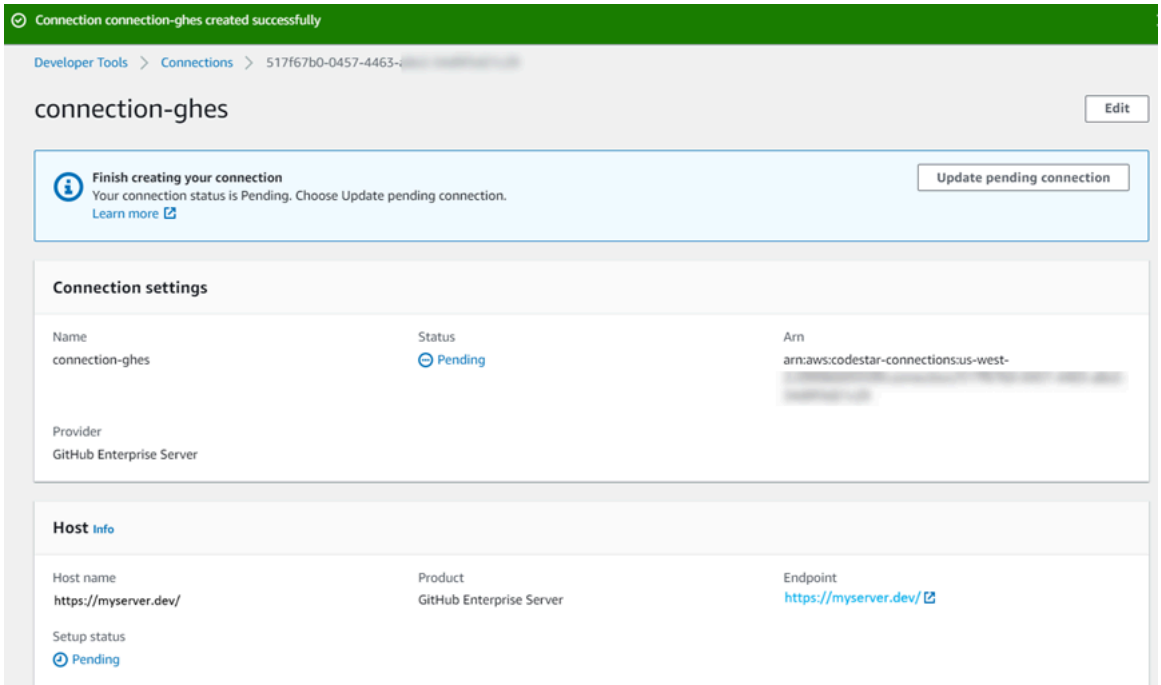
c. Wählen Sie unter Sicherheitsgruppe die Option IDs Hinzufügen aus. Wählen Sie im Feld die Sicherheitsgruppe aus, die Sie für Ihren Host verwenden möchten. Sie können bis zu 10 Sicherheitsgruppen auswählen.

   Stellen Sie sicher, dass Sie die Sicherheitsgruppe für die Infrastruktur auswählen, in der Ihre GitHub Enterprise Server-Instanz installiert ist, oder eine Sicherheitsgruppe mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.

d. Wenn Sie eine private VPC konfiguriert haben und Ihre GitHub Enterprise Server-Instanz so konfiguriert haben, dass sie die TLS-Validierung mithilfe einer nicht öffentlichen Zertifizierungsstelle durchführt, geben Sie im Feld TLS-Zertifikat Ihre Zertifikat-ID ein. Der TLS-Zertifikatwert sollte der öffentliche Schlüssel des Zertifikats sein.

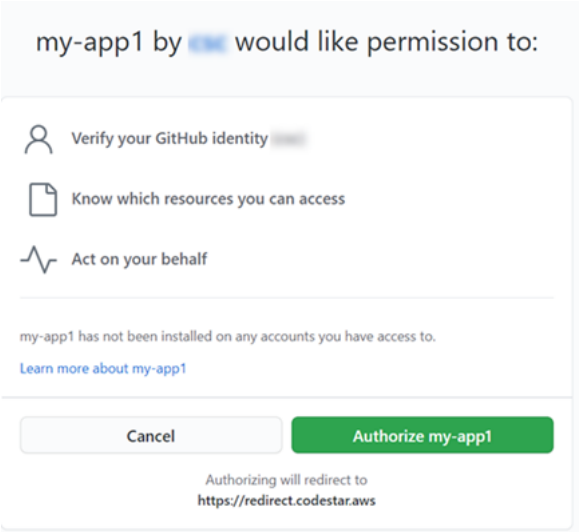


4. Wählen Sie Connect to GitHub Enterprise Server. Die erzeugte Verbindung wird mit dem Status Pending (Ausstehend) angezeigt. Für die Verbindung mit den von Ihnen angegebenen Serverinformationen wird eine Hostressource erstellt. Für den Hostnamen wird die URL verwendet.
5. Wählen Sie Update pending connection (Ausstehende aktualisieren) aus.



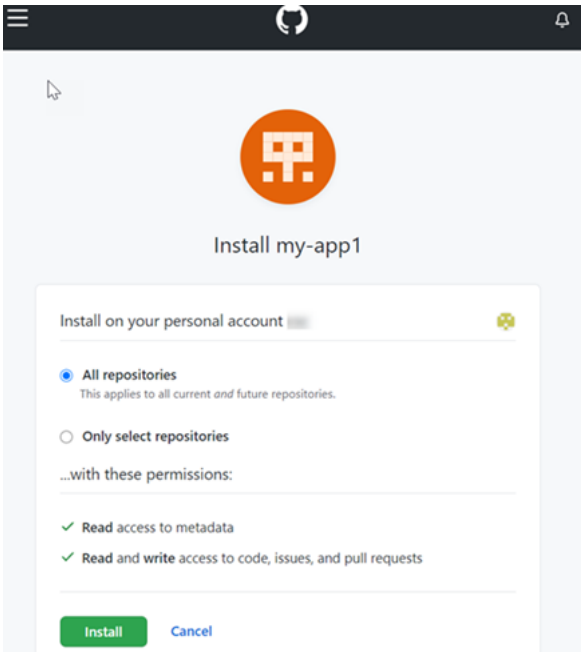
6. Wenn Sie dazu aufgefordert werden, melden Sie sich auf der GitHub Enterprise-Anmeldeseite mit Ihren GitHub Enterprise-Anmeldeinformationen an.
7. Wählen Sie auf der Seite „ GitHub App erstellen“ einen Namen für Ihre App aus.

8. <app-name>Wählen Sie auf der GitHub Autorisierungsseite die Option Autorisieren aus.



9. Auf der App-Installationsseite wird in einer Meldung angezeigt, dass die Connector-App zur Installation bereit ist. Wenn Sie mehrere Organisationen haben, werden Sie möglicherweise aufgefordert, die Organisation auszuwählen, in der Sie die App installieren möchten.

   Wählen Sie in die Repository-Einstellungen aus, wo Sie die App installieren möchten. Wählen Sie Installieren aus.



10. Die Verbindungsseite zeigt die erstellte Verbindung im Status Available (Verfügbar).

Verbindung zu GitHub Enterprise Server (CLI) herstellen

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung herzustellen.

Verwenden Sie hierfür die Befehle create-host und create-connection.

> **Important**
>
> Eine Verbindung, die über AWS CLI oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit der CLI hergestellt haben oder verwenden Sie die Konsole AWS CloudFormation, um die Verbindung so zu bearbeiten, dass sie ihren Status festlegt`AVAILABLE`.

Schritt 1: So erstellen Sie einen Host für GitHub Enterprise Server (CLI)

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den create-host Befehl auszuführen, und geben Sie dabei `--name``--provider-type`, und `--provider-endpoint` für Ihre Verbindung an. In diesem Beispiel lautet der Name des Drittanbieters `GitHubEnterpriseServer` und der Endpunkt `my-instance.dev`.

   ```
   aws codeconnections create-host --name MyHost --provider-type
    GitHubEnterpriseServer --provider-endpoint "https://my-instance.dev"
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die Amazon-Ressourcenname (ARN)-Informationen zum Host ähnlich der folgenden zurück.

   ```
   {
       "HostArn": "arn:aws:codeconnections:us-west-2:account_id:host/My-Host-28aef605"
   }
   ```

   Nach diesem Schritt befindet sich der Host im Status `PENDING` (Ausstehend).

2. Schließen Sie das Host-Setup mit der Konsole ab und ändern Sie den Host-Status zu `Available` (Verfügbar). Weitere Informationen finden Sie unter Einrichten eines ausstehenden Hosts.

Schritt 2: Einrichten eines ausstehenden Hosts in der Konsole

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Schließen Sie das Host-Setup mit der Konsole ab und ändern Sie den Host-Status zu `Available` (Verfügbar). Siehe Einrichten eines ausstehenden Hosts.

Schritt 3: So erstellen Sie eine Verbindung für GitHub Enterprise Server (CLI)

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-connection Befehl auszuführen, und geben Sie dabei `--host-arn` und `--connection-name` für Ihre Verbindung an.

   ```
   aws codeconnections create-connection --host-arn arn:aws:codeconnections:us-
   west-2:account_id:host/MyHost-234EXAMPLE --connection-name MyConnection
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

   ```
   {
       "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
   aEXAMPLE-8aad"
   }
   ```

2. Verwenden Sie die Konsole, um die ausstehende Verbindung einzurichten. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Schritt 4: So stellen Sie eine Verbindung für GitHub Enterprise Server in der Konsole her

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Verwenden Sie die Konsole, um die ausstehende Verbindung einzurichten und in den Status `Available` zu versetzen. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Stellen Sie eine Verbindung her zu GitLab

Du kannst das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu einem auf gitlab.com gehosteten Repository herzustellen.

> **Note**
>
> Indem du diese Verbindungsinstallation autorisierst GitLab, gewährst du unserem Service die Erlaubnis, deine Daten zu verarbeiten, und du kannst die Berechtigungen jederzeit widerrufen, indem du die Anwendung deinstallierst.

Bevor Sie beginnen:

- Sie müssen bereits ein Konto bei erstellt haben. GitLab

> **Note**
>
> Verbindungen bieten nur Zugriff für das Konto, das zum Erstellen und Autorisieren der Verbindung verwendet wurde.

> **Note**
>
> Sie können Verbindungen erstellen GitLab, in denen Sie die Rolle des Besitzers haben, und dann kann die Verbindung mit dem Repository mit Ressourcen wie verwendet werden CodePipeline. Bei Repositorys in Gruppen müssen Sie nicht der Gruppenbesitzer sein.

Themen

- Stellen Sie eine Verbindung zu GitLab (Konsole) her
- Verbindung herstellen zu GitLab (CLI)

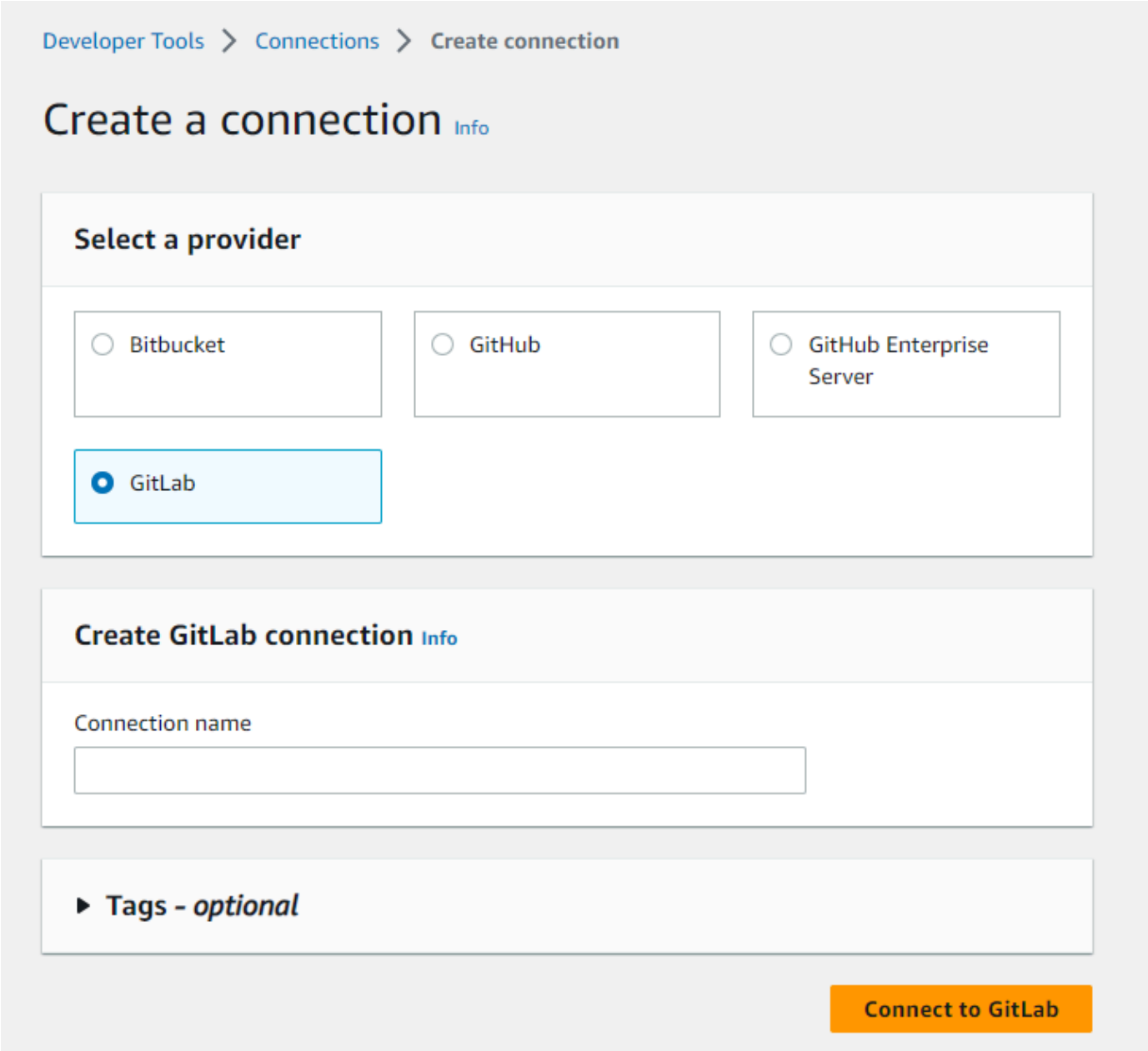
Stellen Sie eine Verbindung zu GitLab (Konsole) her

Sie können die Konsole verwenden, um eine Verbindung herzustellen.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

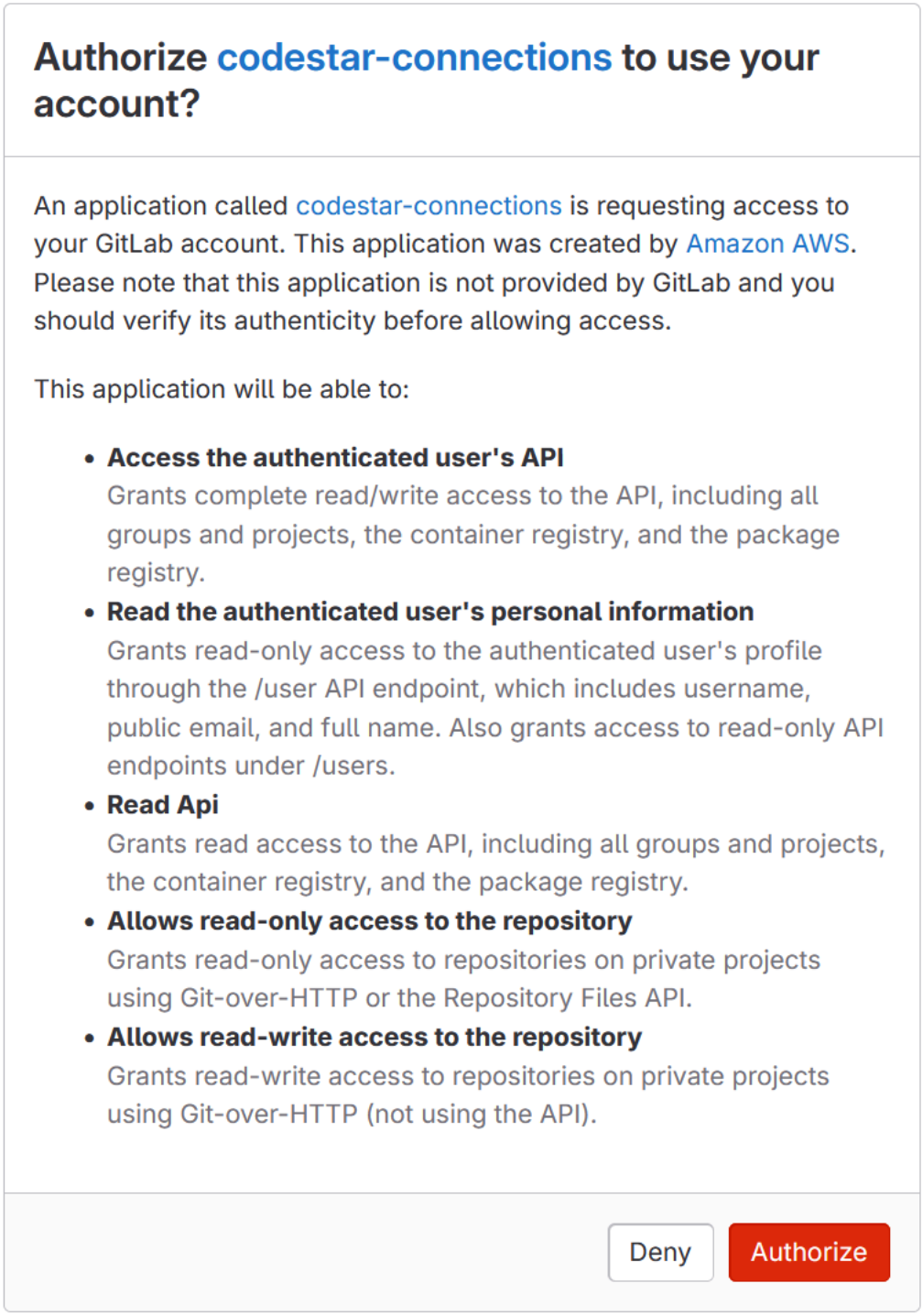
Schritt 1: Erstellen einer Verbindung

1. Melden Sie sich bei der an AWS Management Console, und öffnen Sie dann die AWS Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Einstellungen und dann die Registerkarte Verbindungen aus. Wählen Sie Create Connection (Verbindung erstellen) aus.
3. Um eine Verbindung zu einem GitLab Repository herzustellen, wählen Sie unter Anbieter auswählen die Option GitLab. Geben Sie unter Connection name (Verbindungsname) den Namen für die Verbindung ein, die Sie erstellen möchten. Wählen Sie Connect GitLab.



4. Wenn die Anmeldeseite für GitLab angezeigt wird, melden Sie sich mit Ihren Anmeldeinformationen an und wählen Sie dann Anmelden aus.
5. Es wird eine Autorisierungsseite mit einer Meldung angezeigt, in der Sie aufgefordert werden, die Verbindung für den Zugriff auf Ihr GitLab Konto zu autorisieren.

   Klicken Sie auf Authorize.



6. Der Browser kehrt zur Seite der Verbindungskonsole zurück. Unter GitLab Verbindung erstellen wird die neue Verbindung unter Verbindungsname angezeigt.
7. Wählen Sie Connect GitLab.

Nachdem die Verbindung erfolgreich hergestellt wurde, wird ein Erfolgsbanner angezeigt. Die Verbindungsdetails werden auf der Seite Verbindungseinstellungen angezeigt.

Verbindung herstellen zu GitLab (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung herzustellen.

Verwenden Sie dazu den Befehl create-connection.

> **Important**
>
> Eine Verbindung, die über AWS CLI oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit der CLI hergestellt haben oder verwenden Sie die Konsole AWS CloudFormation, um die Verbindung so zu bearbeiten, dass sie ihren Status festlegt`AVAILABLE`.

Um eine Verbindung herzustellen zu GitLab

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-connection Befehl auszuführen, und geben Sie dabei `--provider-type` und `--connection-name` für Ihre Verbindung an. In diesem Beispiel lautet der Name des Drittanbieters `GitLab` und der angegebene Verbindungsname `MyConnection`.

   ```
   aws codeconnections create-connection --provider-type GitLab --connection-name
    MyConnection
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

   ```
   {
       "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
   aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"
   }
   ```

2. Verwenden Sie die Konsole, um die Verbindung fertigzustellen. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Stellen Sie eine Verbindung zu GitLab Selfmanaged her

Mit einer selbstverwalteten Installation können Sie Verbindungen für die GitLab Enterprise Edition oder GitLab Community Edition herstellen.

Sie können das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung und einen Host für die GitLab Selbstverwaltung herzustellen.

> **Note**
>
> Indem Sie diese Verbindungsanwendung im GitLab selbstverwalteten Modus autorisieren, gewähren Sie unserem Service die Erlaubnis, Ihre Daten zu verarbeiten. Sie können diese Berechtigungen jederzeit widerrufen, indem Sie die Anwendung deinstallieren.

Bevor Sie eine Verbindung zu GitLab Self-Managed herstellen, müssen Sie einen Host erstellen, der für die Verbindung verwendet werden soll, wie in diesen Schritten beschrieben. Eine Übersicht über den Workflow zur Host-Erstellung für installierte Anbieter finden Sie unter Workflow zum Erstellen oder Aktualisieren eines Hosts.

Sie können Ihren Host optional mit einer VPC konfigurieren. Weitere Informationen zur Netzwerk- und VPC-Konfiguration für Ihre Host-Ressource finden Sie in den VPC-Voraussetzungen unter (Optional) Voraussetzungen: Netzwerk- oder Amazon-VPC-Konfiguration für Ihre Verbindung und Fehlerbehebung bei der VPC-Konfiguration für Ihren Host.

Bevor Sie beginnen:

- Sie müssen bereits ein Konto bei einer selbstverwalteten Installation erstellt haben GitLab und über die GitLab Enterprise Edition oder GitLab Community Edition verfügen. Weitere Informationen finden Sie unter https://docs.gitlab.com/ee/subscriptions/self_managed/.

  > **Note**
  >
  > Verbindungen bieten nur Zugriff für das Konto, das zum Erstellen und Autorisieren der Verbindung verwendet wurde.

> **Note**
>
> Sie können Verbindungen zu einem Repository erstellen, in dem Sie die Rolle des Besitzers haben GitLab, und dann kann die Verbindung mit Ressourcen wie verwendet werden. CodePipeline Bei Repositorys in Gruppen müssen Sie nicht der Gruppenbesitzer sein.

- Sie müssen bereits ein GitLab persönliches Zugriffstoken (PAT) mit nur den folgenden eingeschränkten Berechtigungen erstellt haben:,. `api` `admin_mode` Weitere Informationen finden Sie unter _access_tokens.html. https://docs.gitlab.com/ee/ user/profile/personal Sie müssen Administrator sein, um das PAT erstellen und verwenden zu können.

> **Note**
>
> Ihr PAT wird zur Autorisierung des Hosts verwendet und wird nicht anderweitig gespeichert oder von Verbindungen verwendet. Um einen Host einzurichten, können Sie ein temporäres PAT erstellen. Nachdem Sie den Host eingerichtet haben, können Sie das PAT löschen.

> **Note**
>
> Für Organisationen, die GitHub Enterprise Server nutzen oder GitLab selbst verwaltet werden, geben Sie keinen verfügbaren Host weiter. Sie erstellen für jede Verbindung in Ihrer Organisation einen neuen Host und müssen sicherstellen, dass Sie dieselben Informationen in die Netzwerkfelder (VPC-ID, Subnetz IDs und Sicherheitsgruppe IDs) für den Host eingeben. Weitere Informationen finden Sie unter Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen.

Themen

- Stellen Sie eine Verbindung zur GitLab Selbstverwaltung (Konsole) her
- Verbindung zu GitLab Self-Managed (CLI) herstellen

Stellen Sie eine Verbindung zur GitLab Selbstverwaltung (Konsole) her

Gehen Sie wie folgt vor, um in der Konsole einen Host und eine Verbindung zu GitLab Self-Managed herzustellen. Überlegungen zum Einrichten eines Hosts in einer VPC finden Sie unter (Optional) Voraussetzungen: Netzwerk- oder Amazon-VPC-Konfiguration für Ihre Verbindung.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

> **Note**
>
> Sie erstellen einen Host für eine einzelne GitLab selbstverwaltete Installation und können dann eine oder mehrere GitLab selbstverwaltete Verbindungen zu diesem Host verwalten.

Schritt 1: Erstellen Ihres Hosts

1. Melden Sie sich bei der an AWS Management Console, und öffnen Sie dann die AWS Developer Tools-Konsole unter. https://console.aws.amazon.com/codesuite/settings/connections
2. Wählen Sie auf der Registerkarte Hosts die Option Create Host (Host erstellen) aus.
3. Geben Sie unter Host name (Host-Name) den gewünschten Namen für Ihren Host ein.
4. Wählen Sie unter Anbieter auswählen die Option GitLabSelbstverwaltet aus.
5. Geben Sie unter URL den Endpunkt für die Infrastruktur ein, auf der der Anbieter installiert ist.
6. Wenn Ihr Server in einer Amazon VPC konfiguriert ist und Sie eine Verbindung mit Ihrer VPC erstellen möchten, wählen Sie Use a VPC (VPC verwenden) aus. Wählen Sie andernfalls No VPC (Keine VPC) aus.
7. (Optional) Wenn Sie Ihren Host in einer Amazon VPC gestartet haben und eine Verbindung mit Ihrer VPC herstellen möchten, wählen Sie VPC verwenden aus und führen Sie folgende Schritte aus.

   > **Note**
   >
   > Für Organisationen, die GitHub Enterprise Server nutzen oder GitLab selbst verwaltet werden, geben Sie keinen verfügbaren Host weiter. Sie erstellen für jede Verbindung

in Ihrer Organisation einen neuen Host und müssen sicherstellen, dass Sie dieselben Informationen in die Netzwerkfelder (VPC-ID, Subnetz IDs und Sicherheitsgruppe IDs) für den Host eingeben. Weitere Informationen finden Sie unter Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen.

a. Wählen Sie unter VPC ID Ihre VPC-ID aus. Stellen Sie sicher, dass Sie die VPC für die Infrastruktur wählen, in der Ihr Host installiert ist, oder eine VPC, die über VPN oder Direct Connect Zugriff auf Ihre Instance hat.

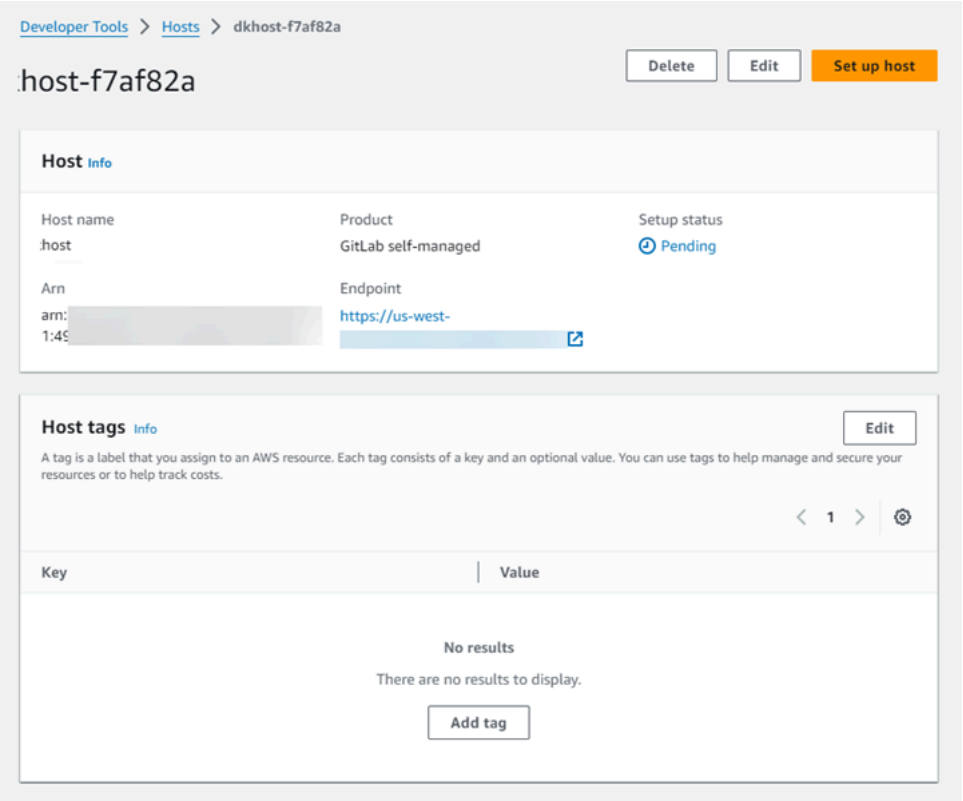
b. Wenn Sie eine private VPC konfiguriert haben und Ihren Host so konfiguriert haben, dass eine TLS-Validierung bei einer nicht öffentlichen Zertifizierungsstelle durchgeführt wird, geben Sie unter TLS-Zertifikat Ihre Zertifikat-ID ein. Der Wert des TLS-Zertifikats ist der öffentliche Schlüssel des Zertifikats.

8. Wählen Sie Create hoste (Host erstellen) aus.

9. Sobald die Seite mit den Host-Details angezeigt wird, ändert sich der Status des erstellten Hosts.

> **Note**
>
> Wenn Ihr Host-Setup eine VPC-Konfiguration enthält, können Sie mehrere Minuten für die Bereitstellung von Hostnetzwerkkomponenten einplanen.

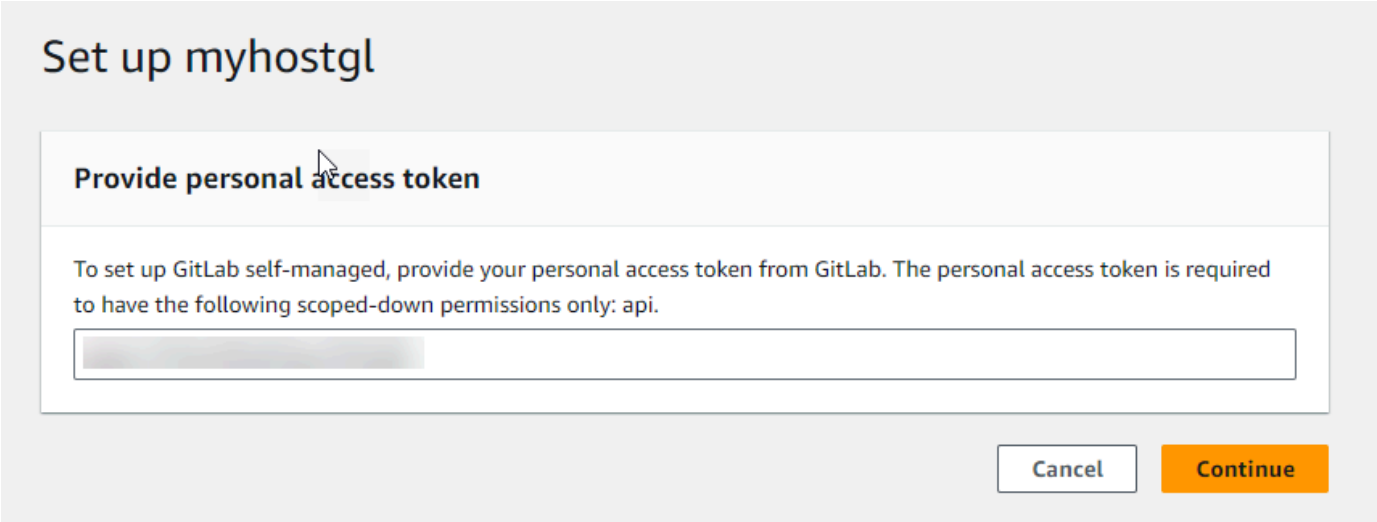
Warten Sie, bis Ihr Host in den Status Pending (Ausstehend) wechseln und schließen Sie das Setup ab. Weitere Informationen finden Sie unter Einrichten eines ausstehenden Hosts.



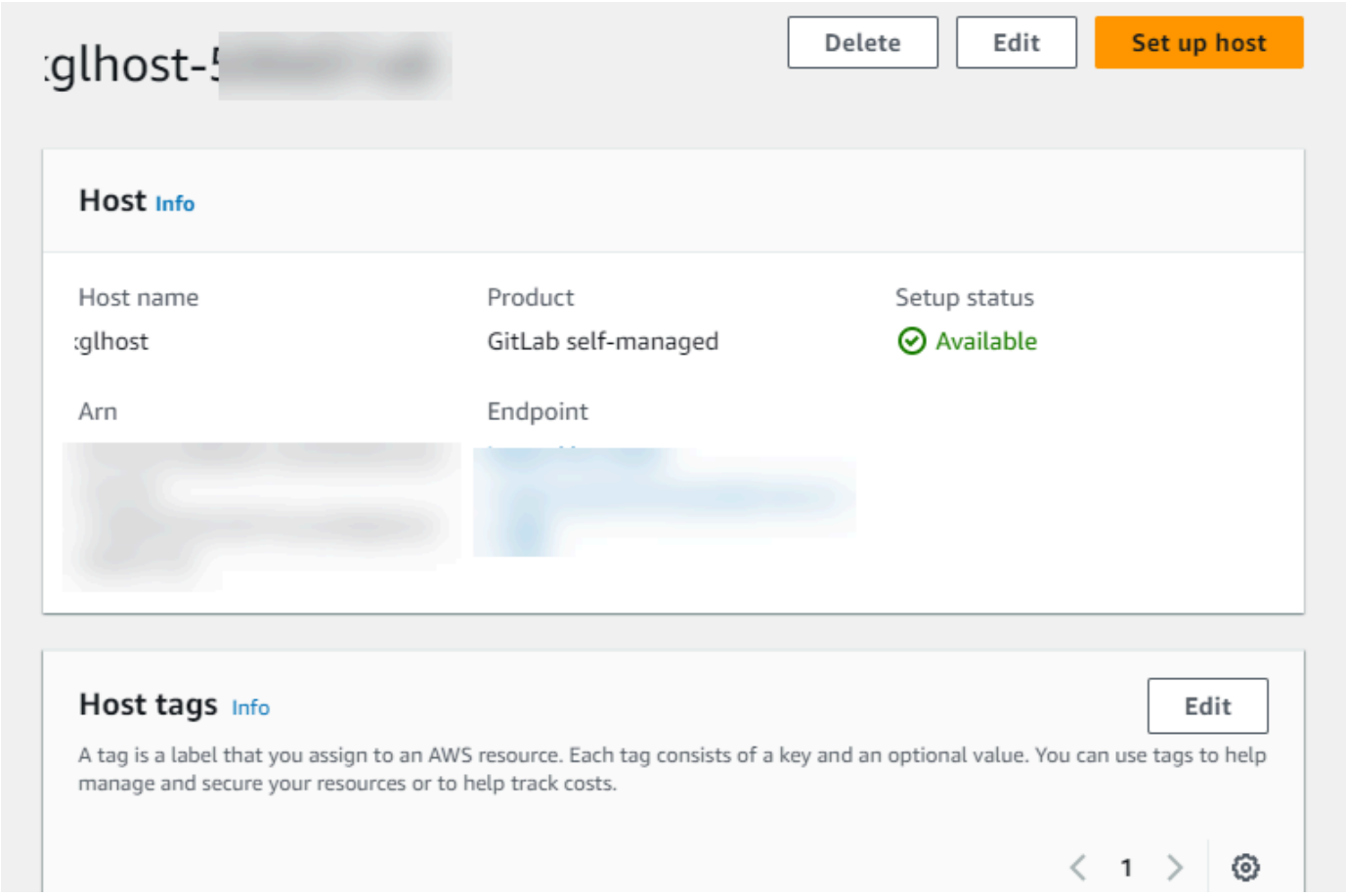
Schritt 2: Einrichten Ihres ausstehenden Hosts

1. Wählen Sie Host einrichten aus.
2. Eine ***host_name***Einrichtungsseite wird angezeigt. Geben Sie unter Persönliches Zugriffstoken bereitstellen Ihrem GitLab PAT nur die folgenden abgegrenzten Berechtigungen: und. `api` `admin_mode`

> **Note**
>
> Nur ein Administrator kann das PAT erstellen und verwenden.

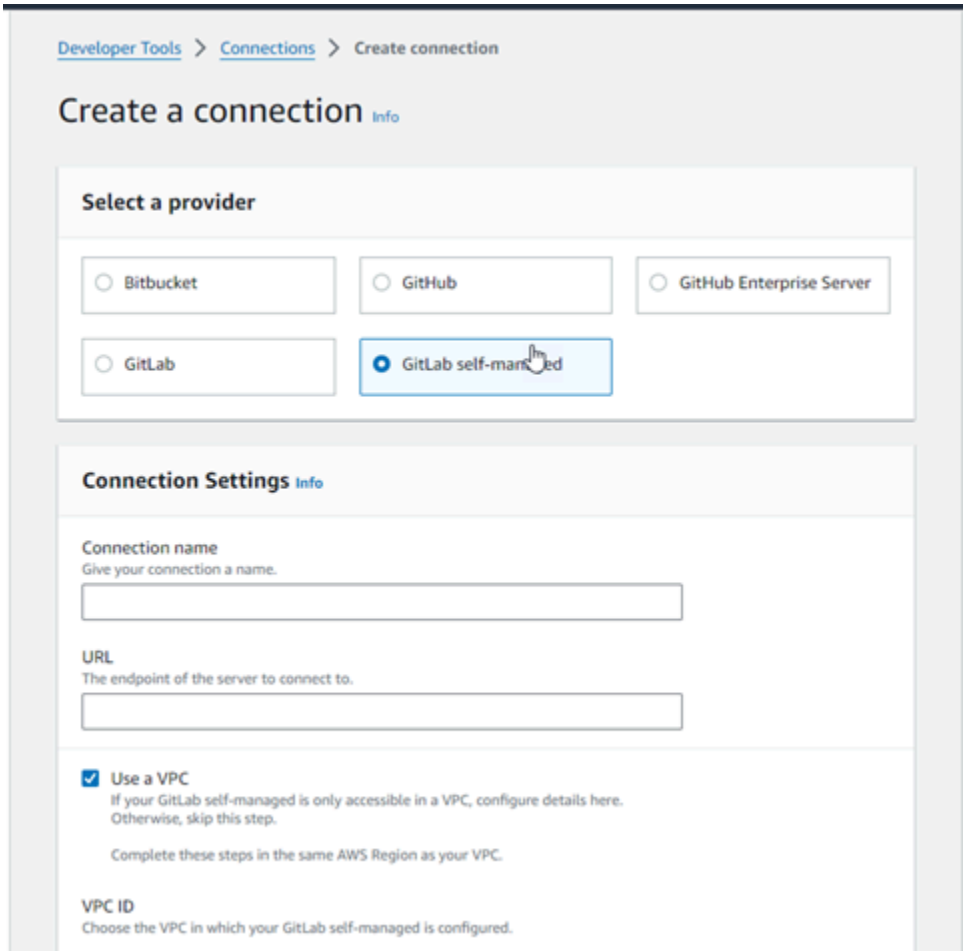


3. Nachdem Ihr Host erfolgreich registriert wurde, erscheint die Host-Detailseite und zeit den HoststatusAvailable (Verfügbar) an.



Schritt 3: Erstellen einer Verbindung

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie dann die AWS Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Einstellungen und dann die Registerkarte Verbindungen aus. Wählen Sie Create Connection (Verbindung erstellen) aus.
3. Um eine Verbindung zu einem GitLab Repository herzustellen, wählen Sie unter Anbieter auswählen die Option GitLab Selbstverwaltet aus. Geben Sie unter Connection name (Verbindungsname) den Namen für die Verbindung ein, die Sie erstellen möchten.



4. Geben Sie unter URL den Endpunkt für Ihren Server ein.
5. Wenn Sie Ihren Server in einer Amazon VPC gestartet haben und eine Verbindung mit Ihrer VPC erstellen möchten, wählen Sie Use a VPC (Verwenden einer VPC) aus und geben Sie Folgendes ein.

a. Wählen Sie unter VPC ID Ihre VPC-ID aus. Stellen Sie sicher, dass Sie die VPC für die Infrastruktur wählen, in der Ihr Host installiert ist, oder eine VPC, die über VPN oder Direct Connect Zugriff auf Ihren Host hat.

b. Wählen Sie unter Subnetz-ID (Subnetz-ID) die Option Add (Hinzufügen) aus. Wählen Sie im Feld die Subnetz-ID aus, die Sie für Ihren Host verwenden möchten. Sie können bis zu 10 Subnetze wählen.

   Stellen Sie sicher, dass Sie das Subnetz für die Infrastruktur wählen, in der Ihr Host installiert ist, oder ein Subnetz, das über VPN oder Direct Connect Zugriff auf Ihren installierten Host hat.

c. Wählen Sie unter Sicherheitsgruppe IDs die Option Hinzufügen aus. Wählen Sie im Feld die Sicherheitsgruppe aus, die Sie für Ihren Host verwenden möchten. Sie können bis zu 10 Sicherheitsgruppen auswählen.

   Stellen Sie sicher, dass Sie die Sicherheitsgruppe für die Infrastruktur wählen, in der Ihr Host installiert ist, oder eine Sicherheitsgruppe, die über VPN oder Direct Connect Zugriff auf Ihren installierten Host hat.

d. Wenn Sie eine private VPC konfiguriert haben und Ihren Host so konfiguriert haben, dass eine TLS-Validierung bei einer nicht öffentlichen Zertifizierungsstelle durchgeführt wird, geben Sie unter TLS-Zertifikat Ihre Zertifikat-ID ein. Der TLS-Zertifikatwert sollte der öffentliche Schlüssel des Zertifikats sein.

6. Wählen Sie Connect to GitLab self-managed aus. Die erzeugte Verbindung wird mit dem Status Pending (Ausstehend) angezeigt. Für die Verbindung mit den von Ihnen angegebenen Serverinformationen wird eine Hostressource erstellt. Für den Hostnamen wird die URL verwendet.
7. Wählen Sie Update pending connection (Ausstehende aktualisieren) aus.
8. Wenn die Anmeldeseite für GitLab angezeigt wird, melden Sie sich mit Ihren Anmeldeinformationen an und wählen Sie dann Anmelden aus.
9. Es wird eine Autorisierungsseite mit einer Meldung angezeigt, in der Sie aufgefordert werden, die Verbindung für den Zugriff auf Ihr GitLab Konto zu autorisieren.

   Klicken Sie auf Authorize.
10. Der Browser kehrt zur Seite der Verbindungskonsole zurück. Unter GitLab Verbindung erstellen wird die neue Verbindung im Feld Verbindungsname angezeigt.
11. Wählen Sie Connect to GitLab self-managed aus.

Nachdem die Verbindung erfolgreich hergestellt wurde, wird ein Erfolgsbanner angezeigt. Die Verbindungsdetails werden auf der Seite Verbindungseinstellungen angezeigt.

Verbindung zu GitLab Self-Managed (CLI) herstellen

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um einen Host und eine Verbindung für GitLab Self-Managed zu erstellen.

Verwenden Sie hierfür die Befehle create-host und create-connection.

> **Important**
>
> Eine Verbindung, die über AWS CLI oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit der CLI hergestellt haben oder verwenden Sie die Konsole AWS CloudFormation, um die Verbindung so zu bearbeiten, dass sie ihren Status festlegt`AVAILABLE`.

Schritt 1: So erstellen Sie einen Host für GitLab Selbstverwaltung (CLI)

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-host Befehl auszuführen, und geben Sie dabei die `--name``--provider-type`, und `--provider-endpoint` für Ihre Verbindung an. In diesem Beispiel lautet der Name des Drittanbieters `GitLabSelfManaged` und der Endpunkt `my-instance.dev`.

   ```
   aws codeconnections create-host --name MyHost --provider-type GitLabSelfManaged --
   provider-endpoint "https://my-instance.dev"
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die Amazon-Ressourcenname (ARN)-Informationen zum Host ähnlich der folgenden zurück.

   ```
   {
       "HostArn": "arn:aws:codeconnections:us-west-2:account_id:host/My-Host-28aef605"
   }
   ```

   Nach diesem Schritt befindet sich der Host im Status `PENDING` (Ausstehend).

2. Schließen Sie die Host-Einrichtung über die Konsole ab und versetzen Sie den Host im nächsten Schritt in den Status `Available`.

Schritt 2: Einrichten eines ausstehenden Hosts in der Konsole

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Schließen Sie das Host-Setup mit der Konsole ab und ändern Sie den Host-Status zu `Available` (Verfügbar). Siehe Einrichten eines ausstehenden Hosts.

Schritt 3: So erstellen Sie eine Verbindung für GitLab Self-Managed (CLI)

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-connection Befehl auszuführen, und geben Sie dabei `--host-arn` und `--connection-name` für Ihre Verbindung an.

```
aws codeconnections create-connection --host-arn arn:aws:codeconnections:us-
west-2:account_id:host/MyHost-234EXAMPLE --connection-name MyConnection
```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

```
{
    "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
aEXAMPLE-8aad"
}
```

2. Verwenden Sie die Konsole, um die ausstehende Verbindung im folgenden Schritt einzurichten.

Schritt 4: So stellen Sie eine Verbindung zur GitLab Selbstverwaltung in der Konsole her

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Verwenden Sie die Konsole, um die ausstehende Verbindung einzurichten und in den Status `Available` zu versetzen. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Aktualisieren einer ausstehenden Verbindung

Eine Verbindung, die über AWS Command Line Interface (AWS CLI) oder AWS CloudFormation erstellt wurde, hat standardmäßig `PENDING` den Status. Nachdem Sie eine Verbindung mit dem AWS CLI oder hergestellt haben AWS CloudFormation, aktualisieren Sie die Verbindung mithilfe der Konsole auf ihren Status`AVAILABLE`.

> **Note**
>
> Um eine ausstehende Verbindung zu aktualisieren, müssen Sie die Konsole verwenden. Sie können eine ausstehende Verbindung nicht mithilfe der AWS CLI aktualisieren.

Wenn Sie die Konsole zum ersten Mal verwenden, um eine neue Verbindung zu einem Drittanbieter hinzuzufügen, müssen Sie den OAuth Handshake mit dem Drittanbieter mithilfe der mit Ihrer Verbindung verknüpften Installation abschließen.

Sie können die Entwickler-Tools-Konsole verwenden, um eine ausstehende Verbindung fertigzustellen.

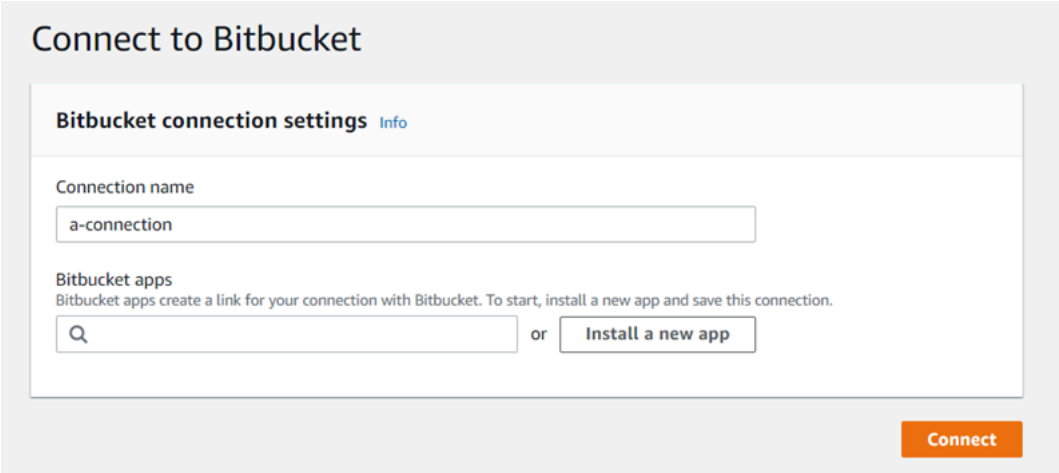
So stellen Sie eine Verbindung fertig

1. Öffnen Sie die AWS Developer Tools-Konsole unterhttps://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) aus.

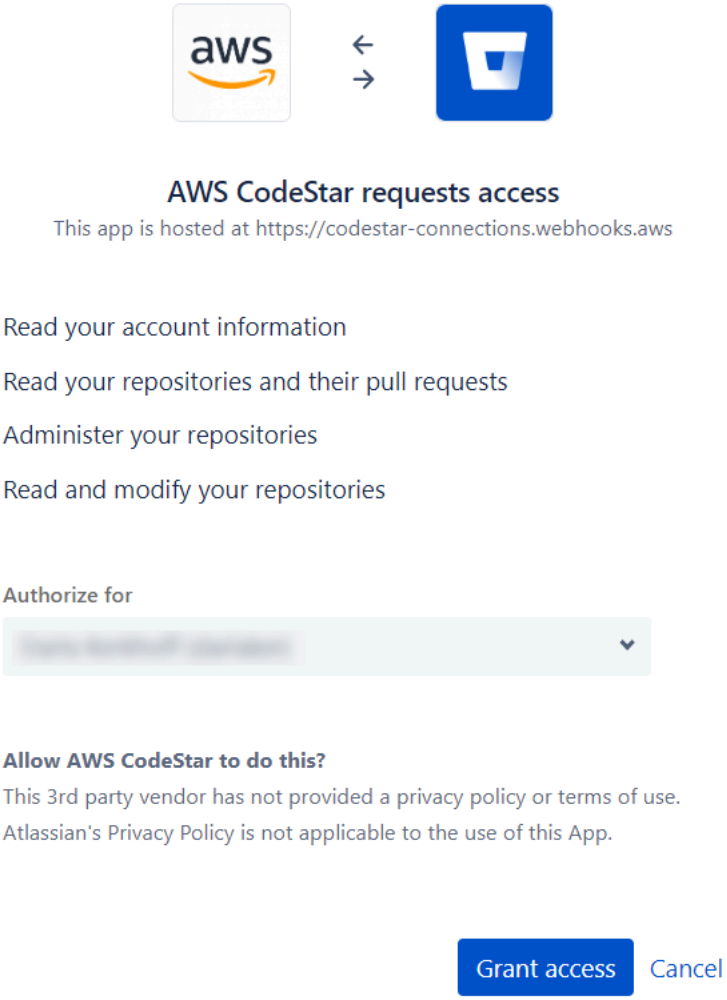
   Die Namen aller Verbindungen, die mit Ihrem AWS Konto verknüpft sind, werden angezeigt.
3. Wählen Sie unter Name den Namen der ausstehenden Verbindung aus, die Sie aktualisieren möchten.

   Der Befehl Update pending connection (Ausstehende Verbindung aktualisieren) ist verfügbar, wenn Sie eine Verbindung mit dem Status Pending (Ausstehend) auswählen.
4. Wählen Sie Update a pending connection (Eine ausstehende aktualisieren) aus.
5. Überprüfen Sie auf der Seite Connect to Bitbucket (Mit Bitbucket verbinden) unter Connection name (Verbindungsname) den Namen Ihrer Verbindung.

   Wählen Sie unter Bitbucket apps (Bitbucket-Apps) eine App-Installation aus oder wählen Sie Install a new app (Neue App installieren), um eine App zu erstellen.



6. Auf der App-Installationsseite wird eine Meldung angezeigt, dass die AWS CodeStar App versucht, eine Verbindung zu deinem Bitbucket-Konto herzustellen. Wählen Sie Grant access (Zugriff gewähren).



7. Die Verbindungs-ID für die neue Installation wird angezeigt. Wählen Sie Complete connection (Verbindung abschließen).

Auflisten von Verbindungen

Sie können die Entwickler-Tools-Konsole oder den Befehl list-connections in AWS Command Line Interface (AWS CLI) verwenden, um eine Liste der Verbindungen in Ihrem Konto anzuzeigen.

Auflisten von Verbindungen (Konsole)

Auflisten von Verbindungen

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) aus.
3. Zeigen Sie den Namen, den Status und den ARN für Ihre Verbindungen an.

Auflisten von Verbindungen (CLI)

Sie können die verwenden, um Ihre Verbindungen AWS CLI zu Code-Repositorys von Drittanbietern aufzulisten. Für eine Verbindung, die mit einer Hostressource verknüpft ist, z. B. Verbindungen zu GitHub Enterprise Server, gibt die Ausgabe zusätzlich den Host-ARN zurück.

Verwenden Sie dazu den Befehl list-connections.

Auflisten von Verbindungen

- Öffnen Sie ein Terminal (Linux, macOS oder Unix) oder eine Befehlszeile (Windows) und verwenden Sie die, AWS CLI um den list-connections Befehl auszuführen.

```
aws codeconnections list-connections --provider-type Bitbucket
--max-results 5 --next-token: next-token
```

Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
     "Connections": [
         {
             "ConnectionName": "my-connection",
```

```
            "ProviderType": "Bitbucket",
            "Status": "PENDING",
            "ARN": "arn:aws:codeconnections:us-west-2:account_id:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
            "OwnerAccountId": "account_id"
        },
        {
            "ConnectionName": "my-other-connection",
            "ProviderType": "Bitbucket",
            "Status": "AVAILABLE",
            "ARN": "arn:aws:codeconnections:us-west-2:account_id:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
            "OwnerAccountId": "account_id"
         },
      ],
     "NextToken": "next-token"
}
```

Eine Verbindung löschen

Sie können die Verbindung mit der Entwicklertools-Konsole oder dem Befehl delete-connection in der AWS Command Line Interface (AWS CLI, Befehlszeilenschnittstelle) löschen.

Themen

- Löschen einer Verbindung (Konsole)
- Löschen einer Verbindung (CLI)

Löschen einer Verbindung (Konsole)

So löschen Sie eine Verbindung

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) aus.
3. Wählen Sie unter Connection name (Verbindungsname) den Namen der Verbindung aus, die Sie löschen möchten.
4. Wählen Sie Löschen aus.

5. Geben Sie zur Bestätigung **`delete`** in das Feld ein. Wählen Sie anschließend Delete (Löschen) aus.

> **Important**
>
> Diese Aktion kann nicht rückgängig gemacht werden.

Löschen einer Verbindung (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu löschen.

Verwenden Sie dazu den Befehl delete-connection.

> **Important**
>
> Nachdem Sie den Befehl ausgeführt haben, wird die Verbindung gelöscht. Es wird kein Bestätigungsdialogfeld angezeigt. Sie können eine neue Verbindung erstellen. Der Amazon-Ressourcenname (ARN) wird jedoch niemals wiederverwendet.

So löschen Sie eine Verbindung

- Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den delete-connection Befehl auszuführen, und geben Sie dabei den ARN der Verbindung an, die Sie löschen möchten.

  ```
  aws codeconnections delete-connection --connection-arn arn:aws:codeconnections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
  ```

  Mit diesem Befehl wird kein Inhalt zurückgegeben.

Ressourcen für Tag-Verbindungen

Ein Tag ist eine benutzerdefinierte Attributbezeichnung, die Sie einer AWS Ressource AWS zuweisen oder zuweisen. Jedes AWS Tag besteht aus zwei Teilen:

- einem Tag-Schlüssel (z. B. `CostCenter`, `Environment` oder `Project`). Bei Tag-Schlüsseln wird zwischen Groß- und Kleinschreibung unterschieden.
- einem optionalen Feld, das als Tag-Wert bezeichnet wird (z. B. `111122223333`, `Production` oder ein Team-Name). Ein nicht angegebener Tag-Wert entspricht einer leeren Zeichenfolge. Wie bei Tag-Schlüsseln wird auch bei Tag-Werten zwischen Groß- und Kleinschreibung unterschieden.

Dies werden als Schlüssel-Wert-Paare bezeichnet.

Sie können Ressourcen über die Konsole oder CLI mit Tags markieren.

Sie können die folgenden Ressourcentypen in AWS taggen CodeConnections:

- Verbindungen
- Hosts

Bei diesen Schritten wird davon ausgegangen, dass Sie bereits eine aktuelle Version von installiert AWS CLI oder auf die aktuelle Version aktualisiert haben. Weitere Informationen finden Sie unter Installieren der AWS CLI im AWS Command Line Interface -Benutzerhandbuch.

Neben der Identifizierung, Organisation und Nachverfolgung Ihrer Ressource mithilfe von Tags können Sie mithilfe von Tags in AWS Identity and Access Management (IAM-) Richtlinien steuern, wer Ihre Ressource aufrufen und mit ihnen interagieren kann. Beispiele für Tag-basierte Zugriffsrichtlinien finden Sie unter Verwenden von Tags zur Steuerung des Zugriffs auf AWS CodeConnections -Ressourcen.

Themen

- Markieren von Ressourcen (Konsole)
- Markieren von Ressourcen mit Tags (CLI)

Markieren von Ressourcen (Konsole)

Sie können mit der Konsole Tags auf einer Verbindungsressource hinzufügen, ändern oder entfernen.

Themen

- Hinzufügen von Tags zu einer Verbindungsressource (Konsole)
- Anzeigen von Tags für eine Verbindungsressource (Konsole)

- Bearbeiten von Tags für eine Verbindungsressource (Konsole)
- Entfernen von Tags aus einer Verbindungsressource (Konsole)

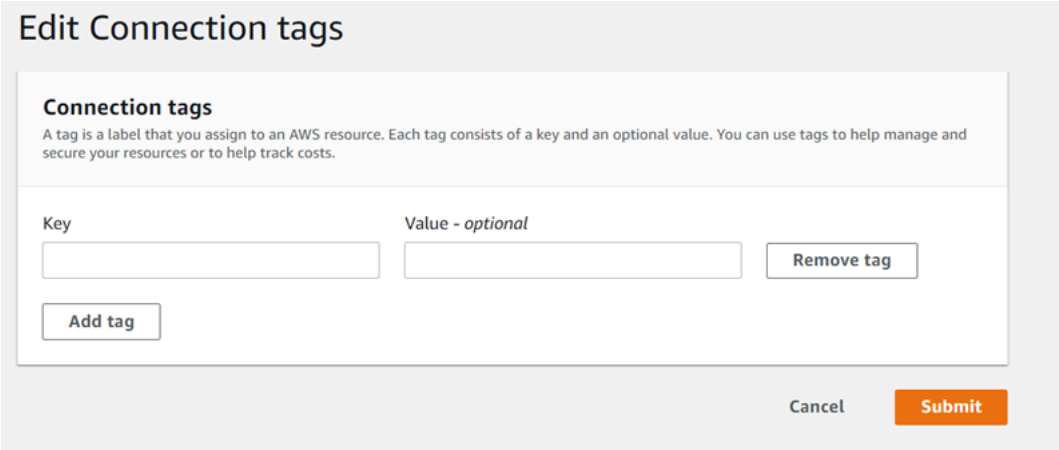
Hinzufügen von Tags zu einer Verbindungsressource (Konsole)

Sie können mit der Konsole Tags zu einer vorhandenen Verbindung/einem vorhandenen Host hinzufügen.

> **Note**
>
> Wenn Sie eine Verbindung für einen installierten Anbieter wie GitHub Enterprise Server herstellen und auch eine Hostressource für Sie erstellt wird, werden die Tags bei der Erstellung nur der Verbindung hinzugefügt. Auf diese Weise können Sie einen Host separat markieren, wenn Sie ihn für eine neue Verbindung wiederverwenden möchten. Gehen Sie wie folgt vor, wenn Sie den Host mit Tags versehen möchten.

Tags für eine Verbindung hinzufügen

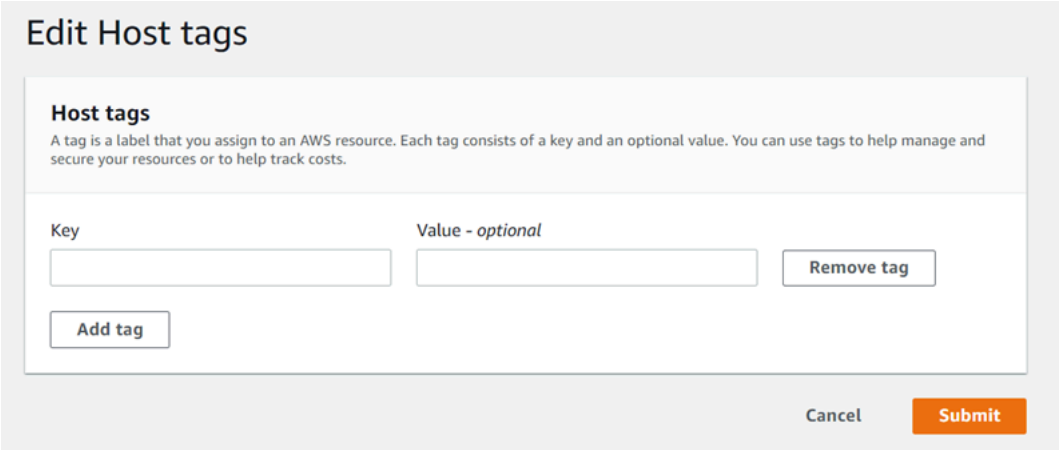
1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Connections (Verbindungen) aus.
3. Wählen Sie die Verbindung aus, die Sie bearbeiten möchten. Die Seite mit dem Verbindungseinstellungen wird angezeigt.
4. Wählen Sie unter Connection tags (Verbindungs-Tags)die Option Edit (Bearbeiten) aus. Die Seite Edit Connection tags (Verbindungs-Tags bearbeiten) erscheint.
5. Geben Sie in die Felder Key (Schlüssel) und Value (Wert) ein Schlüsselpaar für jeden hinzuzufügenden Tag-Satz ein. (Das Feld Value (Wert) ist optional.) Geben Sie beispielsweise für Key (Schlüssel) **`Project`** ein. Geben Sie unter Value (Wert) **`ProjectA`** ein.



6. (Optional) Wählen Sie Add tag (Tag hinzufügen) aus, um weitere Zeilen hinzuzufügen und weitere Tags einzugeben.
7. Wählen Sie Submit (Absenden) aus. Die Tags werden unter den Verbindungs-Einstellungen aufgeführt.

Tags für einen Host hinzufügen

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Hosts.
3. Wählen Sie den Host aus, den Sie bearbeiten möchten. Die Seite mit den Hosteinstellungen erscheint.
4. Wählen Sie unter Host tags (Host-Tags), die Option Edit (Bearbeiten) aus. Die Seite Host tags (Host-Tags) erscheint.
5. Geben Sie in die Felder Key (Schlüssel) und Value (Wert) ein Schlüsselpaar für jeden hinzuzufügenden Tag-Satz ein. (Das Feld Value (Wert) ist optional.) Geben Sie beispielsweise für Key (Schlüssel) **`Project`** ein. Geben Sie unter Value (Wert) **`ProjectA`** ein.



6. (Optional) Wählen Sie Add tag (Tag hinzufügen) aus, um mehr Zeilen hinzuzufügen und weitere Host-Tags einzugeben.
7. Wählen Sie Absenden aus. Die Tags werden unter den Host-Einstellungen aufgeführt.

Anzeigen von Tags für eine Verbindungsressource (Konsole)

Sie können mit der Konsole die Tags für vorhandene Ressourcen anzeigen.

Tags für eine Verbindung anzeigen

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Connections (Verbindungen) aus.
3. Wählen Sie die Verbindung aus, die Sie anzeigen möchten. Die Seite mit dem Verbindungseinstellungen wird angezeigt.
4. Sehen Sie unter Connection tags (Verbindungs-Tags) die Tags für die Verbindung in den Spalten Key (Schlüssel) und Value (Wert) an.

Tags für einen Host anzeigen

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Hosts.
3. Wählen Sie den Host aus, den Sie anzeigen möchten.

4. Sehen Sie unter Host tags (Host-Tags) die Tags für den Host in den Spalten Key (Schlüssel) und Value (Wert) an.

Bearbeiten von Tags für eine Verbindungsressource (Konsole)

Sie können über die Konsole Tags bearbeiten, die Verbindungsressourcen hinzugefügt wurden.

Tags für eine Verbindung bearbeiten

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Connections (Verbindungen) aus.
3. Wählen Sie die Verbindung aus, die Sie bearbeiten möchten. Die Seite mit dem Verbindungseinstellungen wird angezeigt.
4. Wählen Sie unter Connection tags (Verbindungs-Tags)die Option Edit (Bearbeiten) aus. Die Seite mit den Verbindungs-Tags wird angezeigt.
5. Aktualisieren Sie in den Feldern Key (Schlüssel) und Value (Wert) die Werte nach Bedarf. Ändern Sie beispielsweise für den Schlüssel **`Project`** unter Value (Wert) die Angabe **`ProjectA`** in **`ProjectB`**.
6. Wählen Sie Absenden aus.

Tags für einen Host bearbeiten

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Hosts.
3. Wählen Sie den Host aus, den Sie bearbeiten möchten. Die Seite mit den Hosteinstellungen erscheint.
4. Wählen Sie unter Host tags (Host-Tags), die Option Edit (Bearbeiten) aus. Die Seite Host tags (Host-Tags) erscheint.
5. Aktualisieren Sie in den Feldern Key (Schlüssel) und Value (Wert) die Werte nach Bedarf. Ändern Sie beispielsweise für den Schlüssel **`Project`** unter Value (Wert) die Angabe **`ProjectA`** in **`ProjectB`**.

6. Wählen Sie Absenden aus.

Entfernen von Tags aus einer Verbindungsressource (Konsole)

Sie können mit der Konsole Tags aus Verbindungsressourcen entfernen. Wenn Sie Tags aus der zugehörigen Ressource entfernen, werden die Tags gelöscht.

Tags für eine Verbindung entfernen

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Connections (Verbindungen) aus.
3. Wählen Sie die Verbindung aus, die Sie bearbeiten möchten. Die Seite mit dem Verbindungseinstellungen wird angezeigt.
4. Wählen Sie unter Connection tags (Verbindungs-Tags)die Option Edit (Bearbeiten) aus. Die Seite mit den Verbindungs-Tags wird angezeigt.
5. Wählen Sie neben dem Schlüssel und Wert für jedes zu löschende Tag Remove tag (Tag entfernen) aus.
6. Wählen Sie Absenden aus.

Tags für einen Host entfernen

1. Melden Sie sich an der Konsole an. Wählen Sie im Navigationsbereich Settings (Einstellungen) aus.
2. Wählen Sie unter Einstellungen (Settings) die Option Connections (Verbindungen) aus. Wählen Sie die Registerkarte Hosts.
3. Wählen Sie den Host aus, den Sie bearbeiten möchten. Die Seite mit den Hosteinstellungen erscheint.
4. Wählen Sie unter Host tags (Host-Tags), die Option Edit (Bearbeiten) aus. Die Seite Host tags (Host-Tags) erscheint.
5. Wählen Sie neben dem Schlüssel und Wert für jedes zu löschende Tag Remove tag (Tag entfernen) aus.
6. Wählen Sie Absenden aus.

Markieren von Ressourcen mit Tags (CLI)

Sie können mit der CLI Tags auf einer Verbindungsressource hinzufügen, ändern oder entfernen.

Themen

- Hinzufügen von Tags zu einer Verbindungsressource (CLI)
- Anzeigen von Tags für eine Verbindungsressource (CLI)
- Bearbeiten von Tags für eine Verbindungsressource (CLI)
- Entfernen von Tags aus einer Verbindungsressource (CLI)

Hinzufügen von Tags zu einer Verbindungsressource (CLI)

Sie können die verwenden AWS CLI , um Ressourcen in Verbindungen zu kennzeichnen.

Führen Sie am Terminal oder in der Befehlszeile den Befehl tag-resource aus und geben Sie dabei den Amazon-Ressourcennamen (ARN) der Ressource an, für die Sie Tags hinzufügen möchten, sowie den Schlüssel und Wert des hinzuzufügenden Tags. Sie können mehr als ein Tag hinzufügen.

Tags für eine Verbindung hinzufügen

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-connections (zu finden in Auflisten von Verbindungen), um den Verbindungs-ARN abzurufen.
2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl tag-resource aus.

   Verwenden Sie beispielsweise den folgenden Befehl, um eine Verbindung mit zwei Tags zu kennzeichnen, einem Tag-Schlüssel *Project* mit dem Tag-Wert von *ProjectA* und einem Tag-Schlüssel *ReadOnly* mit dem Tag-Wert von*true*.

   ```
   aws codestar-connections tag-resource --resource-arn arn:aws:codestar-
   connections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f --
   tags Key=Project,Value=ProjectA Key=IscontainerBased,Value=true
   ```

   Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Tags für einen Host hinzufügen

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-hosts (zu finden in Auflisten von Hosts), um den Host-ARN abzurufen.

2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl tag-resource aus.

   Verwenden Sie beispielsweise den folgenden Befehl, um einen Host mit zwei Tags zu taggen, einem Tag-Schlüssel *Project* mit dem Tag-Wert von *ProjectA* und einem Tag-Schlüssel *IscontainerBased* mit dem Tag-Wert von*true*.

```
aws codestar-connections tag-resource --resource-arn arn:aws:codestar-
connections:us-west-2:account_id:host/My-Host-28aef605 --tags
 Key=Project,Value=ProjectA Key=IscontainerBased,Value=true
```

   Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück.

Anzeigen von Tags für eine Verbindungsressource (CLI)

Sie können den verwenden AWS CLI , um die AWS Tags für eine Verbindungsressource anzuzeigen. Wenn keine Tags hinzugefügt wurden, ist die zurückgegebene Liste leer. Verwenden Sie den Befehl list-tags-for-resource, um Tags anzuzeigen, die einer Verbindung oder einem Host hinzugefügt wurden.

Tags für eine Verbindung anzeigen

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-connections (zu finden in Auflisten von Verbindungen), um den Verbindungs-ARN abzurufen.
2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl list-tags-for-resource aus. Verwenden Sie beispielsweise den folgenden Befehl, um eine Liste von Tag-Schlüsseln und Tag-Werten für eine Verbindung anzuzeigen.

```
aws codestar-connections list-tags-for-resource --resource-arn arn:aws:codestar-
connections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

   Dieser Befehl gibt die Tags zurück, die der Ressource zugeordnet sind. In diesem Beispiel haben wir zwei Schlüssel-Wert-Paare, die für eine Verbindung zurückgegeben werden.

```
{
    "Tags": [
        {
            "Key": "Project",
            "Value": "ProjectA"
        },
```

```
        {
            "Key": "ReadOnly",
            "Value": "true"
        }
    ]
}
```

Tags für einen Host anzeigen

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-hosts (zu finden in Auflisten von Hosts), um den Host-ARN abzurufen.
2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl list-tags-for-resource aus. Verwenden Sie beispielsweise den folgenden Befehl, um eine Liste von Tag-Schlüsseln und Tag-Werten für einen Host anzuzeigen.

```
aws codestar-connections list-tags-for-resource --resource-arn arn:aws:codestar-
connections:us-west-2:account_id:host/My-Host-28aef605
```

Dieser Befehl gibt die Tags zurück, die der Ressource zugeordnet sind. In diesem Beispiel werden zwei Schlüssel-Wert-Paare angezeigt, die für eine Verbindung zurückgegeben wurden.

```
{
    "Tags": [
        {
            "Key": "IscontainerBased",
            "Value": "true"
        },
        {
            "Key": "Project",
            "Value": "ProjectA"
        }
    ]
}
```

Bearbeiten von Tags für eine Verbindungsressource (CLI)

Sie können den verwenden AWS CLI , um ein Tag für eine Ressource zu bearbeiten. Sie können den Wert für einen vorhandenen Schlüssel ändern oder einen anderen Schlüssel hinzufügen.

Führen Sie am Terminal oder über die Befehlszeile den Befehl tag-resource aus und geben Sie dabei den ARN der Ressource an, für die Sie ein Tag aktualisieren möchten, sowie den Tag-Schlüssel und Tag-Wert.

Wenn Sie Tags bearbeiten, werden alle nicht angegebenen Tag-Schlüssel beibehalten und alles mit demselben Schlüssel aber neuem Wert aktualisiert. Neue Schlüssel, die mit dem Befehl „edit“ hinzugefügt werden, werden als neues Schlüssel-Wert-Paar hinzugefügt.

Tags für eine Verbindung bearbeiten

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-connections (zu finden in Auflisten von Verbindungen), um den Verbindungs-ARN abzurufen.
2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl tag-resource aus.

   In diesem Beispiel wird der Wert für den Schlüssel `Project` in `ProjectB` geändert.

   ```
   aws codestar-connections tag-resource --resource-arn arn:aws:codestar-
   connections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f --
   tags Key=Project,Value=ProjectB
   ```

   Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück. Um die der Verbindung zugeordneten Tags zu überprüfen, führen Sie den Befehl list-tags-for-resource aus.

Tags für einen Host bearbeiten

1. Rufen Sie den ARN für Ihre Ressource ab. Verwenden Sie den Befehl list-hosts (zu finden in Auflisten von Hosts), um den Host-ARN abzurufen.
2. Führen Sie in einem Terminal oder über die Befehlszeile den Befehl tag-resource aus.

   In diesem Beispiel wird der Wert für den Schlüssel `Project` in `ProjectB` geändert.

   ```
   aws codestar-connections tag-resource --resource-arn arn:aws:codestar-
   connections:us-west-2:account_id:host/My-Host-28aef605 --tags
    Key=Project,Value=ProjectB
   ```

   Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück. Um die dem Host zugeordneten Tags zu überprüfen, führen Sie den Befehl list-tags-for-resource aus.

Entfernen von Tags aus einer Verbindungsressource (CLI)

Gehen Sie wie folgt vor, AWS CLI um mit dem ein Tag aus einer Ressource zu entfernen. Wenn Sie Tags aus der zugehörigen Ressource entfernen, werden die Tags gelöscht.

> **Note**
>
> Wenn Sie eine Verbindungsressource löschen, werden alle Tag-Zuordnungen aus der gelöschten Ressource entfernt. Sie müssen keine Tags entfernen, bevor Sie eine Verbindungsressource löschen.

Führen Sie am Terminal oder über die Befehlszeile den Befehl untag-resource aus und geben Sie dabei den ARN der Ressource an, für die Sie Tags entfernen möchten, sowie den Tag-Schlüssel des zu entfernenden Tags. Um beispielsweise mehrere Tags auf einer Verbindung mit den Tag-Schlüsseln *Project* zu entfernen*ReadOnly*, verwenden Sie den folgenden Befehl.

```
aws codestar-connections untag-resource --resource-arn arn:aws:codestar-connections:us-
west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f --tag-
keys Project ReadOnly
```

Bei erfolgreicher Ausführung gibt dieser Befehl nichts zurück. Um die der Ressource zugeordneten Tags zu überprüfen, führen Sie den Befehl list-tags-for-resource aus. Die Ausgabe zeigt, dass alle Tags entfernt wurden.

```
{
    "Tags": []
}
```

Anzeigen von Verbindungsdetails

Sie können mit der Entwicklertools-Konsole oder dem Befehl get-connection (Verbindung löschen) in der AWS Command Line Interface (AWS CLI, Befehlszeilenschnittstelle) die Details für eine Verbindung anzeigen. Um den verwenden zu können AWS CLI, müssen Sie bereits eine aktuelle Version von installiert AWS CLI oder auf die aktuelle Version aktualisiert haben. Weitere Informationen finden Sie unter Installieren der AWS CLI im AWS Command Line Interface - Benutzerhandbuch.

Eine Verbindung anzeigen (Konsole)

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) aus.
3. Wählen Sie die Schaltfläche neben der Verbindung aus, die Sie sich ansehen möchten, und wählen Sie dann View details (Details anzeigen) aus.
4. Sie sehen die folgenden Informationen für Ihre Verbindung:
   - den Verbindungsnamen
   - den Anbietertypen für Ihre Verbindung
   - den Verbindungsstatus
   - den ARN der Verbindung
   - Wenn die Verbindung für einen installierten Anbieter wie GitHub Enterprise Server erstellt wurde, sind die Hostinformationen der Verbindung zugeordnet.
   - Wenn die Verbindung für einen installierten Anbieter wie GitHub Enterprise Server erstellt wurde, die Endpunktinformationen, die dem Host für die Verbindung zugeordnet sind.
5. Wenn sich die Verbindung im Zustand Pending (Ausstehend) befindet, wählen Sie zum Abschließen der Verbindung die Option Update pending connection (Ausstehende Verbindung aktualisieren) aus. Weitere Informationen finden Sie unter Aktualisieren einer ausstehenden Verbindung.

Eine Verbindung anzeigen (CLI)

- Führen Sie am Terminal oder über die Befehlszeile den Befehl get-connection aus. Verwenden Sie beispielsweise den folgenden Befehl, um Details für eine Verbindung mit dem ARN-Wert `arn:aws:codestar-connections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f` anzuzeigen.

  ```
  aws codeconnections get-connection --connection-arn arn:aws:codeconnections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
  ```

  Wenn der Befehl erfolgreich ist, werden die Verbindungsdetails zurückgegeben.

  Beispielausgabe für eine Bitbucket-Verbindung:

```
{
    "Connection": {
        "ConnectionName": "MyConnection",
        "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
cdacd948-EXAMPLE",
        "ProviderType": "Bitbucket",
        "OwnerAccountId": "account_id",
        "ConnectionStatus": "AVAILABLE"
    }
}
```

Beispielausgabe für eine GitHub Verbindung:

```
{
    "Connection": {
        "ConnectionName": "MyGitHubConnection",
        "ConnectionArn": "arn:aws:codeconnections:us-west-2:account_id:connection/
ebcd4a13-EXAMPLE",
        "ProviderType": "GitHub",
        "OwnerAccountId": "account_id",
        "ConnectionStatus": "AVAILABLE"
    }
}
```

Beispielausgabe für eine GitHub Enterprise Server-Verbindung:

```
{
    "Connection": {
        "ConnectionName": "MyConnection",
        "ConnectionArn": "arn:aws:codeconnections:us-
west-2:account_id:connection/2d178fb9-EXAMPLE",
        "ProviderType": "GitHubEnterpriseServer",
        "OwnerAccountId": "account_id",
        "ConnectionStatus": "PENDING",
        "HostArn": "arn:aws:ccodeconnections:us-west-2:account_id:host/sdfsdf-
EXAMPLE"
    }
}
```

Verbindungen teilen mit AWS-Konten

Sie können Resource Sharing mit verwenden AWS RAM , um eine bestehende Verbindung mit einer anderen Person AWS-Konto oder mit Konten in Ihrer Organisation zu teilen. Sie können Ihre gemeinsame Verbindung mit Ressourcen verwenden AWS , die Sie für Quellverbindungen von Drittanbietern verwalten, z. B. in CodePipeline.

> **Important**
>
> Die gemeinsame Nutzung von Verbindungen wird für `codestar-connections` Ressourcen nicht unterstützt. Dies wird nur für `codeconnections` Ressourcen unterstützt.

Bevor Sie beginnen:

- Sie müssen bereits eine Verbindung mit Ihrem hergestellt haben AWS-Konto.
- Sie müssen die gemeinsame Nutzung von Ressourcen aktiviert haben.

> **Note**
>
> Um die Verbindung gemeinsam nutzen zu können, müssen Sie der Eigentümer der Organisation oder des Repositorys sein, sofern Sie keiner Organisation angehören. Das Konto, mit dem Sie die Daten teilen, benötigt ebenfalls Berechtigungen für das Repository.

Themen

- Eine Verbindung teilen (Konsole)
- Eine Verbindung teilen (CLI)
- Geteilte Verbindungen anzeigen (Konsole)
- Geteilte Verbindungen anzeigen (CLI)

Eine Verbindung teilen (Konsole)

Sie können die Konsole verwenden, um gemeinsam genutzte Verbindungsressourcen zu erstellen.

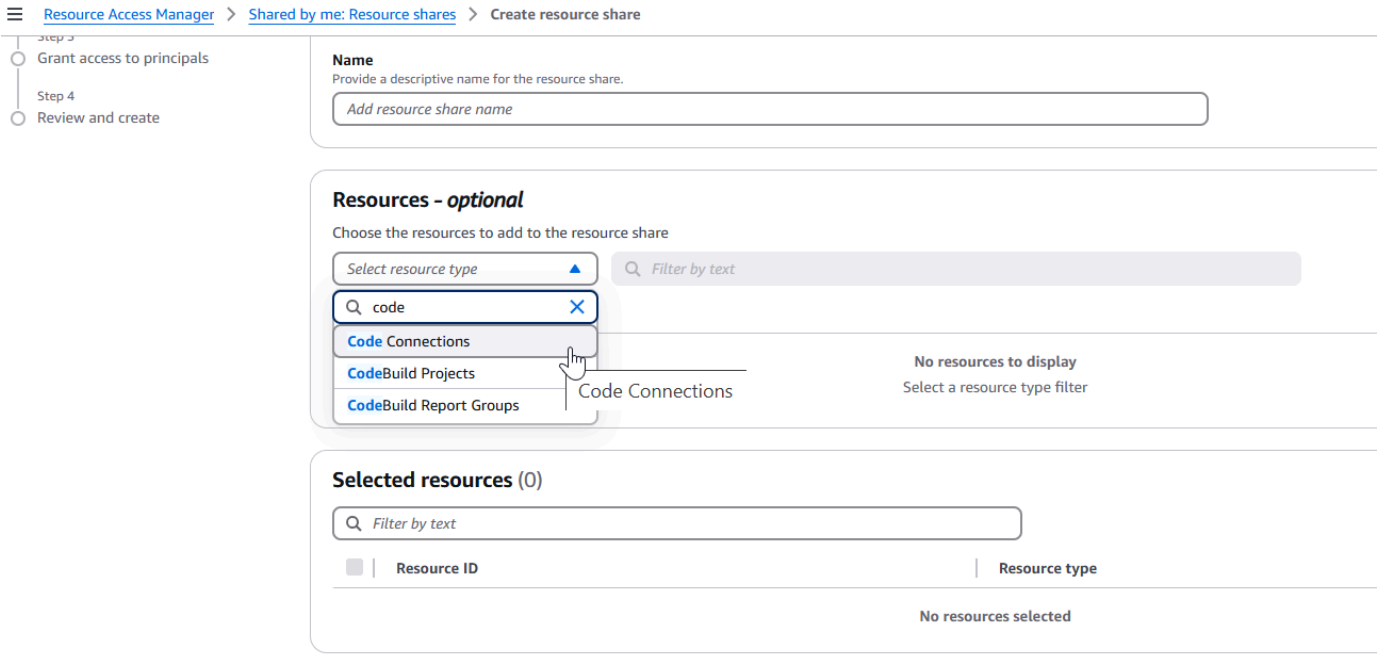
1. Melden Sie sich bei der an AWS Management Console.

Wählen Sie auf der Seite Von mir geteilt: Gemeinsam genutzte Ressourcen in der AWS RAM Konsole die Option Ressourcenfreigabe erstellen aus.

2. Da es in bestimmten AWS-Regionen AWS RAM Ressourcenfreigaben gibt, wählen Sie die entsprechende AWS-Region aus der Dropdownliste in der oberen rechten Ecke der Konsole aus. Um Ressourcenfreigaben zu erstellen, die globale Ressourcen enthalten, müssen Sie die AWS-Region auf USA Ost (Nord-Virginia) festlegen.

   Weitere Informationen zur gemeinsamen Nutzung globaler Ressourcen finden Sie unter Gemeinsame Nutzung regionaler Ressourcen im Vergleich zu globalen Ressourcen.

3. Geben Sie auf der Erstellungsseite im Feld Name einen Namen für Ihre gemeinsam genutzte Ressource ein. Wählen Sie unter Ressourcen die Option Codeverbindungen aus.



4. Wählen Sie Ihre Verbindungsressource und weisen Sie die Principals zu, mit denen Sie sie teilen möchten.

5. Wählen Sie Erstellen aus.

Eine Verbindung teilen (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um eine bestehende Verbindung mit anderen Konten zu teilen und Verbindungen anzuzeigen, die Ihnen gehören oder die Sie mit Ihnen geteilt haben.

Verwenden Sie dazu die `accept-resource-share-invitation` Befehle create-resource-share und für AWS RAM.

Um eine Verbindung gemeinsam zu nutzen

1. Melden Sie sich mit dem Konto an, das die Verbindung teilen wird.
2. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den create-resource-share Befehl auszuführen, und geben Sie dabei `--name``--resource-arns`, und `--principals` für Ihre Verbindungsfreigabe an. In diesem Beispiel ist der Name `my-shared-resource` und der angegebene Verbindungsname `MyConnection` im Ressourcen-ARN ARN. Geben Sie `principals` unter das Zielkonto oder die Zielkonten an, mit denen Sie Inhalte teilen.

   ```
   aws ram create-resource-share --name my-shared-resource --resource-
   arns connection_ARN --principals destination_account
   ```

   Wenn der Befehl erfolgreich ausgeführt wurde, gibt er die ARN-Informationen der Verbindung ähnlich der folgenden zurück.

   ```
   {
       "resourceShare": {
           "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-
   share/4476c27d-8feb-4b21-afe9-7de23EXAMPLE",
           "name": "MyNewResourceShare",
           "owningAccountId": "111111111111",
           "allowExternalPrincipals": true,
           "status": "ACTIVE",
           "creationTime": 1634586271.302,
           "lastUpdatedTime": 1634586271.302
       }
   }
   ```
3. Anfragen zum Teilen können akzeptiert werden, wie im nächsten Verfahren beschrieben.

Um die Verbindung zu authentifizieren und zu akzeptieren, teilen Sie sie mit dem Zielkonto

Das folgende Verfahren ist optional für Zielkonten, die derselben Organisation angehören und für die die gemeinsame Nutzung von Ressourcen in Organizations aktiviert ist.

1. Melden Sie sich mit dem Zielkonto an, das die Einladung erhalten soll.

2. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den get-resource-share-invitations Befehl auszuführen.

```
aws ram get-resource-share-invitations
```

   Erfassen Sie den ARN für die Einladung zur Ressourcenfreigabe für den nächsten Schritt.

3. Führen Sie den accept-resource-share-invitation Befehl aus und geben Sie den an`--resource-share-invitation-arn`.

```
aws ram accept-resource-share-invitation --resource-share-invitation-
arn invitation_ARN
```

   Bei Erfolg gibt dieser Befehl die folgende Ausgabe zurück.

```
{
    "resourceShareInvitation": {
        "resourceShareInvitationArn": "arn:aws:ram:us-west-2:111111111111:resource-
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE",
        "resourceShareName": "MyResourceShare",
        "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
        "senderAccountId": "111111111111",
        "receiverAccountId": "222222222222",
        "invitationTimestamp": "2021-09-22T15:07:35.620000-07:00",
        "status": "ACCEPTED"
    }
}
```

Geteilte Verbindungen anzeigen (Konsole)

Sie können die Konsole verwenden, um gemeinsam genutzte Verbindungsressourcen anzuzeigen.

1. Melden Sie sich bei der an AWS Management Console.

   Öffnen Sie die Seite Von mir gemeinsam genutzt: Gemeinsam genutzte Ressourcen in der AWS-RAM-Konsole.

2. Da AWS-RAM-Ressourcenfreigaben in bestimmten AWS-Regionen existieren, wählen Sie die entsprechende AWS-Region aus der Dropdownliste in der oberen rechten Ecke der Konsole aus.

Um Ressourcenfreigaben zu sehen, die globale Ressourcen enthalten, müssen Sie die AWS-Region auf USA Ost (Nord-Virginia) festlegen.

Weitere Informationen zur gemeinsamen Nutzung globaler Ressourcen finden Sie unter Gemeinsame Nutzung regionaler Ressourcen im Vergleich zu globalen Ressourcen.

3. Für jede freigegebene Ressource sind die folgenden Informationen verfügbar:

   - Ressourcen-ID — Die ID der Ressource. Wählen Sie die ID einer Ressource, um einen neuen Browser-Tab zu öffnen und die Ressource in der nativen Servicekonsole anzuzeigen.
   - Ressourcentyp — Der Ressourcentyp.
   - Datum der letzten gemeinsamen Nutzung — Das Datum, an dem die Ressource zuletzt gemeinsam genutzt wurde.
   - Ressourcenfreigaben — Die Anzahl der Ressourcenfreigaben, zu denen die Ressource gehört. Um die Liste der Ressourcenfreigaben zu sehen, wählen Sie die Anzahl aus.
   - Principals — Die Anzahl der Principals, die auf die Ressource zugreifen können. Wählen Sie den Wert, um die Principals anzuzeigen.

Geteilte Verbindungen anzeigen (CLI)

Sie können die verwenden AWS CLI , um Verbindungen anzuzeigen, die Ihnen gehören oder die Sie mit Ihnen geteilt haben.

Verwenden Sie dazu den Befehl get-resource-shares.

Um geteilte Verbindungen anzuzeigen

- Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den get-resource-shares Befehl auszuführen.

  ```
  aws ram get-resource-shares
  ```

  Die Ausgabe gibt eine Liste der Ressourcenfreigaben für Ihr Konto zurück.

Arbeiten mit Hosts

Um eine Verbindung zu einem installierten Anbietertyp wie GitHub Enterprise Server herzustellen, erstellen Sie zunächst einen Host mit dem AWS Management Console. Ein Host ist eine Ressource,

die die Infrastruktur darstellt, auf der Ihr Anbieter installiert ist. Dann erstellen Sie eine Verbindung mit diesem Host. Weitere Informationen finden Sie unter Arbeiten mit Verbindungen.

Beispielsweise erstellen Sie einen Host für Ihre Verbindung, damit die Drittanbieter-App für Ihren Anbieter registriert werden kann, um Ihre Infrastruktur zu repräsentieren. Sie erstellen einen Host für einen Anbietertyp, den dann alle Verbindungen zu diesem Anbietertyp nutzen.

Wenn Sie die Konsole verwenden, um eine Verbindung zu einem installierten Anbietertyp wie GitHub Enterprise Server herzustellen, erstellt die Konsole Ihre Hostressource für Sie.

Themen

- Erstellen eines Hosts
- Einrichten eines ausstehenden Hosts
- Auflisten von Hosts
- Bearbeiten eines Host
- Löschen eines Hosts
- Anzeigen von Hostdetails

Erstellen eines Hosts

Sie können das AWS Management Console oder das AWS Command Line Interface (AWS CLI) verwenden, um eine Verbindung zu einem Code-Repository eines Drittanbieters herzustellen, das auf Ihrer Infrastruktur installiert ist. Beispielsweise könnten Sie GitHub Enterprise Server als virtuelle Maschine auf einer EC2 Amazon-Instance ausführen. Bevor Sie eine Verbindung zu GitHub Enterprise Server herstellen, erstellen Sie einen Host, der für die Verbindung verwendet werden soll.

Eine Übersicht über den Workflow zur Host-Erstellung für installierte Anbieter finden Sie unter Workflow zum Erstellen oder Aktualisieren eines Hosts.

Bevor Sie beginnen:

- (Optional) Wenn Sie Ihren Host mit einer VPC erstellen möchten, müssen Sie bereits ein Netzwerk oder eine Virtual Private Cloud (VPC) erstellt haben.
- Sie müssen die Instance bereits erstellt und, wenn Sie eine Verbindung mit der VPC erstellen möchten, den Host in der VPC gestartet haben.

> **Note**
>
> Jede VPC kann jeweils nur einem Host zugeordnet sein.

Sie können Ihren Host optional mit einer VPC konfigurieren. Weitere Informationen zur Netzwerk- und VPC-Konfiguration für Ihre Host-Ressource finden Sie in den VPC-Voraussetzungen unter (Optional) Voraussetzungen: Netzwerk- oder Amazon-VPC-Konfiguration für Ihre Verbindung und Fehlerbehebung bei der VPC-Konfiguration für Ihren Host.

Informationen zur Verwendung der Konsole zum Erstellen eines Hosts und einer Verbindung zu GitHub Enterprise Server finden Sie unterErstellen Sie Ihre GitHub Enterprise Server-Verbindung (Konsole). Die Konsole erstellt einen Host für Sie.

Informationen zur Verwendung der Konsole zum Erstellen eines Hosts und einer Verbindung zu GitLab Self-Managed finden Sie unterStellen Sie eine Verbindung zu GitLab Selfmanaged her. Die Konsole erstellt einen Host für Sie.

(Optional) Voraussetzungen: Netzwerk- oder Amazon-VPC-Konfiguration für Ihre Verbindung

Wenn Ihre Infrastruktur mit einer Netzwerkverbindung konfiguriert ist, können Sie diesen Abschnitt überspringen.

Wenn Ihr Host nur in einer VPC zugänglich ist, beachten Sie diese VPC-Anforderungen, bevor Sie fortfahren.

VPC-Anforderungen

Sie haben die Möglichkeit, Ihren Host mit einer VPC zu erstellen. Die folgenden allgemeinen VPC-Anforderungen sind abhängig von der VPC, die Sie für Ihre Installation eingerichtet haben.

- Sie können eine öffentlich (public) VPC mit öffentlichen und privaten Subnetzen konfigurieren. Wenn Sie keine bevorzugten CIDR-Blöcke oder Subnetze haben, können Sie die Standard-VPC für Ihr AWS-Konto verwenden.
- Wenn Sie eine private VPC konfiguriert haben und Ihre GitHub Enterprise Server-Instanz so konfiguriert haben, dass sie die TLS-Validierung mithilfe einer nicht öffentlichen Zertifizierungsstelle durchführt, müssen Sie das TLS-Zertifikat für Ihre Hostressource bereitstellen.

- Wenn Connections Ihren Host erstellt, wird der VPC-Endpunkt (PrivateLink) für Webhooks für Sie erstellt. Weitere Informationen finden Sie unter AWS CodeConnections und Schnittstellen-VPC-Endpunkte ()AWS PrivateLink.
- Konfiguration der Sicherheitsgruppen:
  - Die bei der Hosterstellung verwendeten Sicherheitsgruppen benötigen Regeln für eingehenden und ausgehenden Datenverkehr, die es der Netzwerkschnittstelle ermöglichen, eine Verbindung zu Ihrer Enterprise Server-Instanz herzustellen GitHub
  - Die mit Ihrer GitHub Enterprise Server-Instanz verbundenen Sicherheitsgruppen (die nicht Teil des Host-Setups sind) benötigen eingehenden und ausgehenden Zugriff über die Netzwerkschnittstellen, die durch Verbindungen erzeugt werden.
- Die VPC-Subnetze müssen sich in verschiedenen Availability Zones in Ihrer Region befinden. Availability Zones sind unabhängige Standorte, die von Fehlern in anderen Availability Zones nicht betroffen sind. Jedes Subnetz muss sich vollständig innerhalb einer Availability Zone befinden und darf nicht mehrere Zonen umfassen.

Weitere Informationen zur Arbeit mit VPCs Subnetzen finden Sie unter VPC and Subnet Sizing for IPv4 im Amazon VPC-Benutzerhandbuch.

VPC Informationen, die Sie für die Host-Einrichtung angeben

Wenn Sie die Hostressource für Ihre Verbindungen im nächsten Schritt erstellen, müssen Sie Folgendes angeben:

- VPC-ID: Die ID der VPC für den Server, auf dem Ihre GitHub Enterprise Server-Instance installiert ist, oder für eine VPC, die über VPN oder Direct Connect Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz hat.
- Subnetz-ID oder IDs: Die ID des Subnetzes für den Server, auf dem Ihre GitHub Enterprise Server-Instanz installiert ist, oder ein Subnetz mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.
- Sicherheitsgruppe oder Gruppen: Die Sicherheitsgruppe für den Server, auf dem Ihre GitHub Enterprise Server-Instanz installiert ist, oder eine Sicherheitsgruppe mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.
- Endpoint (Endpunkt): Halten Sie Ihren Server-Endpunkt bereit und fahren Sie mit dem nächsten Schritt fort.

Weitere Informationen sowie Möglichkeiten zur Fehlerbehebung bei VPC- oder Hostverbindungen finden Sie unter Fehlerbehebung bei der VPC-Konfiguration für Ihren Host.

Benötigte Berechtigungen

AWS CodeConnections Erstellt im Rahmen der Hosterstellung in Ihrem Namen Netzwerkressourcen, um die VPC-Konnektivität zu erleichtern. Dazu gehören eine Netzwerkschnittstelle für die AWS CodeConnections Abfrage von Daten von Ihrem Host und ein VPC-Endpunkt oder PrivateLinkfür den Host, um Ereignisdaten über Webhooks an Verbindungen zu senden. Damit Sie diese Netzwerkressourcen erstellen können, müssen Sie sicherstellen, dass die für die Host-Erstellung verwendete Rolle über die folgenden Berechtigungen verfügt:

```
ec2:CreateNetworkInterface
ec2:CreateTags
ec2:DescribeDhcpOptions
ec2:DescribeNetworkInterfaces
ec2:DescribeSubnets
ec2:DeleteNetworkInterface
ec2:DescribeVpcs
ec2:CreateVpcEndpoint
ec2:DeleteVpcEndpoints
ec2:DescribeVpcEndpoints
```

Weitere Informationen sowie Möglichkeiten zur Fehlerbehebung bei Berechtigungen oder Hostverbindungen in einer VPC finden Sie unter Fehlerbehebung bei der VPC-Konfiguration für Ihren Host.

Weitere Informationen über den Webhook-VPC-Endpunkt finden unter AWS CodeConnections und Schnittstellen-VPC-Endpunkte ()AWS PrivateLink.

Themen

- Erstellen eines Hosts für eine Verbindung (Konsole)
- Erstellen eines Hosts für eine Verbindung (CLI)

Erstellen eines Hosts für eine Verbindung (Konsole)

Bei Verbindungen für Installationen, z. B. mit GitHub Enterprise Server oder mit GitLab Self-Managed, verwenden Sie einen Host, der den Endpunkt für die Infrastruktur darstellt, auf der Ihr Drittanbieter installiert ist.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Was sonst noch beim Einrichten eines Hosts in einer VPC zu beachten ist, finden Sie unter Stellen Sie eine Verbindung zu GitLab Selfmanaged her.

Informationen zur Verwendung der Konsole zum Erstellen eines Hosts und einer Verbindung zu GitHub Enterprise Server finden Sie unterErstellen Sie Ihre GitHub Enterprise Server-Verbindung (Konsole). Die Konsole erstellt einen Host für Sie.

Informationen zur Verwendung der Konsole zum Erstellen eines Hosts und einer Verbindung zu GitLab Self-Managed finden Sie unterStellen Sie eine Verbindung zu GitLab Selfmanaged her. Die Konsole erstellt einen Host für Sie.

> **Note**
>
> Sie erstellen einen Host nur einmal pro GitHub Enterprise Server oder GitLab selbstverwaltetem Konto. Alle Ihre Verbindungen zu einem bestimmten GitHub Enterprise Server oder einem GitLab selbstverwalteten Konto verwenden denselben Host.

Erstellen eines Hosts für eine Verbindung (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um einen Host für installierte Verbindungen zu erstellen.

> **Note**
>
> Sie erstellen einen Host nur einmal pro GitHub Enterprise Server-Konto. Alle Ihre Verbindungen zu einem bestimmten GitHub Enterprise Server-Konto verwenden denselben Host.

Sie verwenden Sie einen Host, der den Endpunkt für die Infrastruktur darstellt, in der der Drittanbieter installiert ist. Um einen Host mit der CLI zu erstellen, verwenden Sie den Befehl create-host. Sobald Sie den Host vollständig erstellt haben, befindet sich der Host im Status Pending (Ausstehend).

Danach müssen Sie den Host einrichten (set up), damit er in den Status Available (Verfügbar) kommt. Sobald der Host verfügbar ist, führen Sie die Schritte zum Erstellen einer Verbindung aus.

> **Important**
>
> Ein über den erstellter Host AWS CLI befindet sich standardmäßig im `Pending` Status. Nachdem Sie mit der CLI einen Host erstellt haben, richten Sie mit der Konsole den Host so ein, dass er in den Status `Available` (Verfügbar) kommt.

Informationen zum Erstellen eines Hosts und einer Verbindung zu GitHub Enterprise Server mithilfe der Konsole finden Sie unterErstellen Sie Ihre GitHub Enterprise Server-Verbindung (Konsole). Die Konsole erstellt einen Host für Sie.

Informationen zur Verwendung der Konsole zum Erstellen eines Hosts und einer Verbindung zu GitLab Self-Managed finden Sie unterStellen Sie eine Verbindung zu GitLab Selfmanaged her. Die Konsole erstellt einen Host für Sie.

Einrichten eines ausstehenden Hosts

Ein über das AWS Command Line Interface (AWS CLI) oder SDK erstellter Host befindet sich standardmäßig im `Pending` Status. Nachdem Sie eine Verbindung mit der Konsole oder dem SDK hergestellt haben AWS CLI, verwenden Sie die Konsole, um den Host so einzurichten, dass er seinen Status festlegt`Available`.

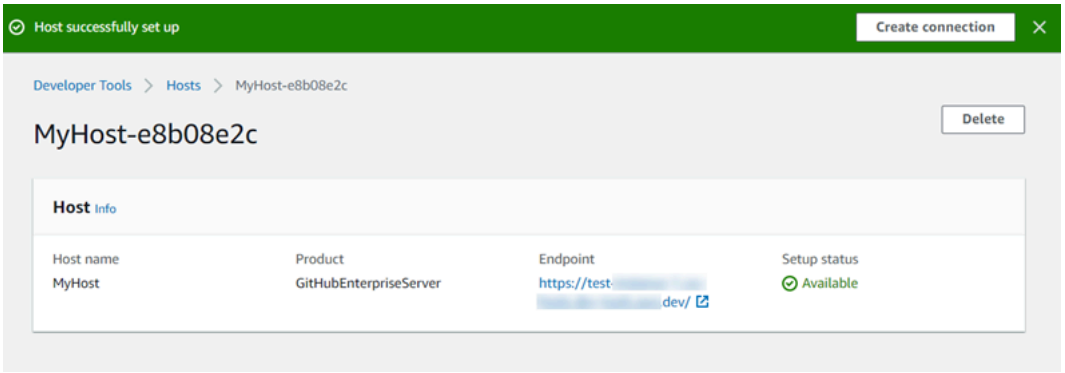
Sie benötigen einen Host. Weitere Informationen finden Sie unter Einen Host erstellen.

Einen ausstehenden Host einrichten

Nachdem Ihr Host erstellt wurde, befindet er sich im Status Pending (Ausstehend). Um den Host vom Status Pending (Ausstehend) in den Status Available (Verfügbar) zu bringen, führen Sie diese Schritte aus. Bei diesem Vorgang wird ein Handshake mit dem Drittanbieter durchgeführt, um die AWS Verbindungs-App auf dem Host zu registrieren.

1. Wenn Ihr Host in der AWS Developer Tools-Konsole den Status Ausstehend erreicht hat, wählen Sie Host einrichten aus.
2. Wenn Sie einen Host für die GitLab Selbstverwaltung erstellen, wird eine Einrichtungsseite angezeigt. Geben Sie unter Persönliches Zugriffstoken bereitstellen Ihrem GitLab PAT nur die folgende eingeschränkte Berechtigung: api.

3. Melden Sie sich auf der Anmeldeseite des Drittanbieters, wie z. B. der GitHub Enterprise Server-Anmeldeseite, mit Ihren Kontoanmeldedaten an, wenn Sie dazu aufgefordert werden.
4. Geben Sie auf der App-Installationsseite im Feld GitHub App-Name einen Namen für die App ein, die Sie für Ihren Host installieren möchten. Wähle GitHubApp erstellen.
5. Nachdem Ihr Host erfolgreich registriert wurde, erscheint die Host-Detailseite und zeit den HoststatusAvailable (Verfügbar) an.



6. Sie können mit dem Erstellen Ihrer Verbindung fortfahren, sobald der Host verfügbar ist. Wählen Sie auf dem Erfolgsmeldungs–Banner Create connection (Verbindung erstellen) aus. Führen Sie die Schritte unter Create a connection (Verbindung erstellen) aus.

Auflisten von Hosts

Sie können die Entwickler-Tools-Konsole oder den Befehl list-connections in AWS Command Line Interface (AWS CLI) verwenden, um eine Liste der Verbindungen in Ihrem Konto anzuzeigen.

Auflisten von Hosts (Konsole)

Hosts auflisten

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie die Registerkarte Hosts. Zeigen Sie den Namen, den Status und den ARN für Ihre Hosts an.

Auflisten von Hosts (CLI)

Sie können die verwenden AWS CLI , um Ihre Hosts für installierte Verbindungen von Drittanbietern aufzulisten.

Verwenden Sie dazu den Befehl list-hosts.

Hosts auflisten

- Öffnen Sie ein Terminal (Linux, macOS oder Unix) oder eine Befehlszeile (Windows) und verwenden Sie die, AWS CLI um den list-hosts Befehl auszuführen.

```
aws codeconnections list-hosts
```

Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "Hosts": [
        {
            "Name": "My-Host",
            "HostArn": "arn:aws:codeconnections:us-west-2:account_id:host/My-
Host-28aef605",
            "ProviderType": "GitHubEnterpriseServer",
            "ProviderEndpoint": "https://my-instance.test.dev",
            "Status": "AVAILABLE"
        }
    ]
}
```

Bearbeiten eines Host

Sie können Host-Einstellungen für einen Host im Status `Pending` (Ausstehend) ändern. Sie können den Hostnamen, die URL oder die VPC-Konfiguration bearbeiten.

Sie können nicht dieselbe URL für mehr als einen Host verwenden.

> **Note**
>
> Was sonst noch beim Einrichten eines Hosts in einer VPC zu beachten ist, finden Sie unter (Optional) Voraussetzungen: Netzwerk- oder Amazon-VPC-Konfiguration für Ihre Verbindung.

Einen Host bearbeiten

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) aus.
3. Wählen Sie die Registerkarte Hosts.

   Die Hosts, die mit deinem AWS Konto verknüpft und in der ausgewählten AWS Region erstellt wurden, werden angezeigt.
4. Geben Sie zum Bearbeiten des Hostnamens einen neuen Wert unter Name ein.
5. Geben Sie zum Bearbeiten des Host-Endpunkts einen neuen Wert unter URL ein.
6. Um die Host-VPC Konfiguration zu bearbeiten, geben Sie neue Werte unter VPC ID ein.
7. Wählen Sie Edit host (Host bearbeiten) aus.
8. Es werden die geänderten Einstellungen angezeigt. Wählen Sie Set up Pending host (Ausstehenden Host einrichten) aus.

Löschen eines Hosts

Sie können den Host mit der Entwicklertools-Konsole oder dem Befehl delete-host in der AWS Command Line Interface (AWS CLI, Befehlszeilenschnittstelle) löschen.

Themen

- Löschen eines Hosts (Konsole)
- Löschen eines Hosts (CLI)

Löschen eines Hosts (Konsole)

Einen Host löschen

1. Öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie die Registerkarte Hosts. Wählen Sie im Feld Name den Namen des Hosts aus, den Sie löschen möchten.
3. Wählen Sie Löschen.

4. Geben Sie zur Bestätigung **`delete`** in das Feld ein. Wählen Sie anschließend Delete (Löschen) aus.

> **Important**
>
> Diese Aktion kann nicht rückgängig gemacht werden.

Löschen eines Hosts (CLI)

Sie können das AWS Command Line Interface (AWS CLI) verwenden, um einen Host zu löschen.

Verwenden Sie dazu den Befehl delete-host.

> **Important**
>
> Bevor Sie einen Host löschen können, müssen Sie alle Verbindungen löschen, die mit dem Host verknüpft sind.
> Nachdem Sie den Befehl ausgeführt haben, wird der Host gelöscht. Es wird kein Bestätigungsdialogfeld angezeigt.

Einen Host löschen

- Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den delete-host Befehl auszuführen, und geben Sie den Amazon-Ressourcennamen (ARN) des Hosts an, den Sie löschen möchten.

  ```
  aws codeconnections delete-host --host-arn "arn:aws:codeconnections:us-
  west-2:account_id:host/My-Host-28aef605"
  ```

  Mit diesem Befehl wird kein Inhalt zurückgegeben.

Anzeigen von Hostdetails

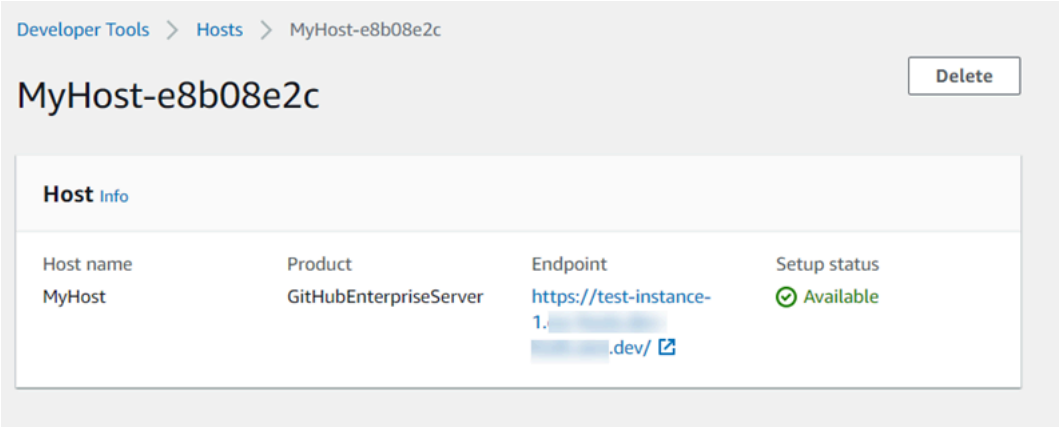
Sie können sich die Host-Details mit der Entwicklertools-Konsole oder dem Befehl get-host in der AWS Command Line Interface (AWS CLI, Befehlszeilenschnittstelle) anzeigen lassen.

Anzeigen von Hostdetails (Konsole)

1. Melden Sie sich in der AWS Management Console (Konsole) an und öffnen Sie die Entwicklertools-Konsole unter https://console.aws.amazon.com/codesuite/settings/connections.
2. Wählen Sie Settings (Einstellungen) > Connections (Verbindungen) und dann die Registerkarte Hosts aus.
3. Wählen Sie die Schaltfläche neben dem Host aus, den Sie sich ansehen möchten, und wählen Sie dann View details (Details anzeigen) aus.
4. Sie sehen die folgenden Informationen für Ihren Host:
   - den Namen des Hosts
   - den Anbietertypen für Ihre Verbindung
   - den Endpunkt für die Infrastruktur, wo Ihr Anbieter installiert ist
   - den Setup-Status für Ihren Host Ein Host ist für eine Verbindung bereit, wenn er sich im Status Available (Verfügbar) befindet. Wenn Ihr Host erstellt, das Setup aber nicht abgeschlossen wurde, hat der Host möglicherweise einen anderen Status.

     Die folgenden Statuswerte gibt es:

     - PENDING (AUSSTEHEND) – Der Host ist erstellt und es kann mit der Einrichtung begonnen werden, indem die Anbieter-App auf dem Host registriert wird.
     - AVAILABLE (VERFÜGBAR) – Die Host-Einrichtung ist abgeschlossen und er kann für Verbindungen verwendet werden.
     - ERROR (FEHLER) – Bei der Erstellung oder Registrierung des Hosts ist ein Fehler aufgetreten.
     - VPC_CONFIG_VPC_INITIALIZING – Die VPC-Konfiguration für den Host wird erstellt.
     - VPC_CONFIG_VPC_FAILED_INITIALIZATION – Die VPC-Konfiguration für den Host ist fehlgeschlagen.
     - VPC_CONFIG_VPC_AVAILABLE – Die VPC-Konfiguration für den Host hat die Einrichtung abgeschlossen und ist verfügbar.
     - VPC_CONFIG_VPC_DELETING – Die VPC-Konfiguration für den Host wird gelöscht.



5. Um den Host zu löschen, wählen Sie Delete (Löschen).
6. Wenn sich der Host im Status Pending (Ausstehend) befindet, wählen Sie zum Abschließen der Einrichtung die Option Set up host (Host einrichten) aus. Weitere Informationen finden Sie unter Einrichten eines ausstehenden Hosts aus.

Hostdetails anzeigen (CLI)

- Öffnen Sie ein Terminal (Linux, macOS oder Unix) oder eine Befehlszeile (Windows) und verwenden Sie die, AWS CLI um den get-host Befehl auszuführen. Geben Sie dabei den Amazon-Ressourcennamen (ARN) des Hosts an, für den Sie Details anzeigen möchten.

```
aws codeconnections get-host --host-arn arn:aws:codeconnections:us-
west-2:account_id:host/My-Host-28aef605
```

Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "Name": "MyHost",
    "Status": "AVAILABLE",
    "ProviderType": "GitHubEnterpriseServer",
    "ProviderEndpoint": "https://test-instance-1.dev/"
}
```

Arbeiten mit Synchronisierungskonfigurationen für verknüpfte Repositorys

In verwendest du eine Verbindung AWS CodeConnections, um AWS Ressourcen mit einem Drittanbieter-Repository wie Bitbucket Cloud GitHub, GitHub Enterprise Server und zu verknüpfen. GitLab Mithilfe des `CFN_STACK_SYNC` Synchronisierungstyps können Sie eine Synchronisierungskonfiguration erstellen, die es ermöglicht, Inhalte aus einem Git-Repository AWS zu synchronisieren, um eine angegebene AWS Ressource zu aktualisieren. AWS CloudFormation integriert sich in Verbindungen, sodass du Git Sync verwenden kannst, um deine Vorlagen- und Parameterdateien in einem verknüpften Repository zu verwalten, mit dem du synchronisierst.

Nachdem Sie eine Verbindung hergestellt haben, können Sie die Verbindungs-CLI oder die AWS CloudFormation Konsole verwenden, um Ihren Repository-Link zu erstellen und die Konfiguration zu synchronisieren.

- Repository-Link: Ein Repository-Link stellt eine Verbindung zwischen Ihrer Verbindung und einem externen Git-Repository her. Der Repository-Link ermöglicht es einer Git-Synchronisierung, Änderungen an Dateien in einem bestimmten Git-Repository zu überwachen und zu synchronisieren.
- Synchronisierungskonfiguration: Verwenden Sie die Synchronisierungskonfiguration, um Inhalte aus einem Git-Repository zu synchronisieren, um eine angegebene AWS Ressource zu aktualisieren.

Weitere Informationen finden Sie in der AWS CodeConnections -API-Referenz.

Ein Tutorial, das dich durch die Erstellung einer Synchronisierungskonfiguration für einen AWS CloudFormation Stack mithilfe der AWS CloudFormation Konsole führt, findest du unter Arbeiten mit AWS CloudFormation Git Sync im CloudFormation Benutzerhandbuch.

Themen

- Arbeiten mit Repository-Links
- Arbeiten mit Synchronisierungskonfigurationen

Arbeiten mit Repository-Links

Ein Repository-Link stellt eine Verbindung zwischen Ihrer Verbindung und einem externen Git-Repository her. Der Repository-Link ermöglicht es Git Sync, Änderungen an Dateien in einem

bestimmten Git-Repository zu überwachen und mit einem AWS CloudFormation Stack zu synchronisieren.

Weitere Informationen zu Repository-Links findest du in der AWS CodeConnections API-Referenz.

Themen

- Einen Repository-Link erstellen
- Einen Repository-Link aktualisieren
- Repository-Links auflisten
- Einen Repository-Link löschen
- Details zu Repository-Links anzeigen

Einen Repository-Link erstellen

Sie können den create-repository-link Befehl in AWS Command Line Interface (AWS CLI) verwenden, um einen Link zwischen Ihrer Verbindung und dem externen Repository zu erstellen, mit dem synchronisiert werden soll.

Bevor Sie einen Repository-Link erstellen können, müssen Sie Ihr externes Repository bereits bei Ihrem Drittanbieter erstellt haben, z. GitHub B.

So erstellen Sie einen Repository-Link

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-repository-link Befehl auszuführen. Geben Sie den ARN der zugehörigen Verbindung, die Besitzer-ID und den Repository-Namen an.

```
aws codeconnections create-repository-link --connection-arn
 arn:aws:codeconnections:us-east-1:account_id:connection/001f5be2-a661-46a4-
b96b-4d277cac8b6e --owner-id account_id --repository-name MyRepo
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "RepositoryLinkInfo": {
        "ConnectionArn": "arn:aws:codeconnections:us-east-1:account_id:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "OwnerId": "account_id",
        "ProviderType": "GitHub",
```

```
        "RepositoryLinkArn": "arn:aws:codeconnections:us-
east-1:account_id:repository-link/be8f2017-b016-4a77-87b4-608054f70e77",
        "RepositoryLinkId": "be8f2017-b016-4a77-87b4-608054f70e77",
        "RepositoryName": "MyRepo",
        "Tags": []
    }
}
```

Einen Repository-Link aktualisieren

Sie können den update-repository-link Befehl in AWS Command Line Interface (AWS CLI) verwenden, um einen angegebenen Repository-Link zu aktualisieren.

Sie können die folgenden Informationen für Ihren Repository-Link aktualisieren:

- `--connection-arn`
- `--owner-id`
- `--repository-name`

Sie können einen Repository-Link aktualisieren, wenn Sie die Verbindung ändern möchten, die mit Ihrem Repository verknüpft ist. Wenn Sie eine andere Verbindung verwenden möchten, müssen Sie den Verbindungs-ARN angeben. Die Schritte zum Anzeigen Ihres Verbindungs-ARN finden Sie unter Verbindungsdetails anzeigen.

So aktualisieren Sie einen Repository-Link

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den update-repository-link Befehl auszuführen und dabei den Wert anzugeben, der für den Repository-Link aktualisiert werden soll. Mit dem folgenden Befehl wird beispielsweise die Verbindung aktualisiert, die mit der Repository-Link-ID verknüpft ist. Er spezifiziert den neuen Verbindungs-ARN mit dem `--connection`-Parameter.

```
aws codestar-connections update-repository-link --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173 --connection-arn arn:aws:codestar-
connections:us-east-1:account_id:connection/aEXAMPLE-f055-4843-adef-4ceaefcb2167
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
```

```
    "RepositoryLinkInfo": {
        "ConnectionArn": "arn:aws:codestar-connections:us-
east-1:account_id:connection/aEXAMPLE-f055-4843-adef-4ceaefcb2167",
        "OwnerId": "owner_id",
        "ProviderType": "GitHub",
        "RepositoryLinkArn": "arn:aws:codestar-connections:us-
east-1:account_id:repository-link/6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryName": "MyRepo",
        "Tags": []
    }
}
```

Repository-Links auflisten

Sie können den list-repository-links Befehl in AWS Command Line Interface (AWS CLI) verwenden, um die Repository-Links für Ihr Konto aufzulisten.

So listen Sie Repository-Links auf

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den list-repository-links Befehl auszuführen.

```
aws codeconnections list-repository-links
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "RepositoryLinks": [
        {
            "ConnectionArn": "arn:aws:codestar-connections:us-
east-1:account_id:connection/001f5be2-a661-46a4-b96b-4d277cac8b6e",
            "OwnerId": "owner_id",
            "ProviderType": "GitHub",
            "RepositoryLinkArn": "arn:aws:codestar-connections:us-
east-1:account_id:repository-link/6053346f-8a33-4edb-9397-10394b695173",
            "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "RepositoryName": "MyRepo",
            "Tags": []
        }
    ]
```

```
}
```

Einen Repository-Link löschen

Sie können den delete-repository-link Befehl in AWS Command Line Interface (AWS CLI) verwenden, um einen Repository-Link zu löschen.

Bevor Sie einen Repository-Link löschen können, müssen Sie alle Synchronisierungskonfigurationen löschen, die mit dem Repository-Link verknüpft sind.

> **Important**
>
> Nachdem Sie den Befehl ausgeführt haben, wird der Repository-Link gelöscht. Es wird kein Bestätigungsdialogfeld angezeigt. Sie können einen neuen Repository-Link erstellen. Der Amazon-Ressourcenname (ARN) wird jedoch nicht wiederverwendet.

So löschen Sie einen Repository-Link

- Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den delete-repository-link Befehl auszuführen, und geben Sie dabei die ID des zu löschenden Repository-Links an.

  ```
  aws codeconnections delete-repository-link --repository-link-id
   6053346f-8a33-4edb-9397-10394b695173
  ```

  Mit diesem Befehl wird kein Inhalt zurückgegeben.

Details zu Repository-Links anzeigen

Sie können den get-repository-link Befehl in AWS Command Line Interface (AWS CLI) verwenden, um Details zu einem Repository-Link anzuzeigen.

So zeigen Sie Details zum Repository-Link an

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den get-repository-link Befehl auszuführen, und geben Sie dabei die Repository-Link-ID an.

```
aws codestar-connections get-repository-link --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "RepositoryLinkInfo": {
        "ConnectionArn": "arn:aws:codestar-connections:us-
east-1:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "OwnerId": "owner_id",
        "ProviderType": "GitHub",
        "RepositoryLinkArn": "arn:aws:codestar-connections:us-
east-1:account_id:repository-link/be8f2017-b016-4a77-87b4-608054f70e77",
        "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryName": "MyRepo",
        "Tags": []
    }
}
```

Arbeiten mit Synchronisierungskonfigurationen

Eine Synchronisierungskonfiguration erstellt eine Zuordnung zwischen einem angegebenen Repository und einer Verbindung. Verwenden Sie die Synchronisierungskonfiguration, um Inhalte aus einem Git-Repository zu synchronisieren und eine bestimmte AWS -Ressource zu aktualisieren.

Weitere Informationen zu Verbindungen finden Sie in der AWS CodeConnections API-Referenz.

Themen

- Eine neue Synchronisierungskonfiguration erstellen
- Eine Synchronisierungskonfiguration aktualisieren
- Synchronisierungskonfigurationen auflisten
- Eine Konfiguration löschen
- Details zu Synchronisierungskonfigurationen anzeigen

Eine neue Synchronisierungskonfiguration erstellen

Sie können den create-repository-link Befehl in AWS Command Line Interface (AWS CLI) verwenden, um einen Link zwischen Ihrer Verbindung und dem externen Repository zu erstellen, mit dem synchronisiert werden soll.

Bevor Sie eine Synchronisierungskonfiguration erstellen können, müssen Sie bereits einen Repository-Link zwischen Ihrer Verbindung und Ihrem Drittanbieter-Repository erstellt haben.

So erstellen Sie eine Synchronisierungskonfiguration

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den create-repository-link Befehl auszuführen. Geben Sie den ARN der zugehörigen Verbindung, die Besitzer-ID und den Repository-Namen an. Der folgende Befehl erstellt eine Synchronisierungskonfiguration mit einem Synchronisierungstyp für eine Ressource in AWS CloudFormation. Er spezifiziert auch die Repository-Verzweigung und die Konfigurationsdatei im Repository. In diesem Beispiel ist die Ressource ein Stack namens **`mystack`**.

```
aws codeconnections create-sync-configuration --branch main --config-file filename
 --repository-link-id be8f2017-b016-4a77-87b4-608054f70e77 --resource-name mystack
 --role-arn arn:aws:iam::account_id:role/myrole --sync-type CFN_STACK_SYNC
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "SyncConfiguration": {
        "Branch": "main",
        "ConfigFile": "filename",
        "OwnerId": "account_id",
        "ProviderType": "GitHub",
        "RepositoryLinkId": "be8f2017-b016-4a77-87b4-608054f70e77",
        "RepositoryName": "MyRepo",
        "ResourceName": "mystack",
        "RoleArn": "arn:aws:iam::account_id:role/myrole",
        "SyncType": "CFN_STACK_SYNC"
    }
```

Eine Synchronisierungskonfiguration aktualisieren

Sie können den update-sync-configuration-Befehl in der AWS Command Line Interface (AWS CLI) verwenden, um eine angegebene Synchronisierungskonfiguration zu aktualisieren.

Sie können die folgenden Informationen für Ihre Synchronisierungskonfiguration aktualisieren:

- `--branch`
- `--config-file`
- `--repository-link-id`
- `--resource-name`
- `--role-arn`

So aktualisieren Sie eine Synchronisierungskonfiguration

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den update-sync-configuration Befehl auszuführen, und geben Sie dabei den Wert an, den Sie aktualisieren möchten, zusammen mit dem Ressourcennamen und dem Synchronisationstyp. Der folgende Befehl aktualisiert beispielsweise den Namen der Verzweigung, der der Synchronisierungskonfiguration zugeordnet ist, mit dem `--branch`-Parameter.

```
aws codeconnections update-sync-configuration --sync-type CFN_STACK_SYNC --
resource-name mystack --branch feature-branch
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "SyncConfiguration": {
        "Branch": "feature-branch",
        "ConfigFile": "filename.yaml",
        "OwnerId": "owner_id",
        "ProviderType": "GitHub",
        "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryName": "MyRepo",
        "ResourceName": "mystack",
        "RoleArn": "arn:aws:iam::account_id:role/myrole",
        "SyncType": "CFN_STACK_SYNC"
    }
```

Synchronisierungskonfigurationen auflisten

Sie können mit dem list-sync-configurations-Befehl in AWS Command Line Interface (AWS CLI) die Repository-Links für Ihr Konto auflisten.

So listen Sie Repository-Links auf

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den list-sync-configurations Befehl auszuführen, und geben Sie dabei den Synchronisationstyp und die Repository-Link-ID an.

```
aws codeconnections list-sync-configurations --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173  --sync-type CFN_STACK_SYNC
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "SyncConfigurations": [
        {
            "Branch": "main",
            "ConfigFile": "filename.yaml",
            "OwnerId": "owner_id",
            "ProviderType": "GitHub",
            "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "RepositoryName": "MyRepo",
            "ResourceName": "mystack",
            "RoleArn": "arn:aws:iam::account_id:role/myrole",
            "SyncType": "CFN_STACK_SYNC"
        }
    ]
}
```

Eine Konfiguration löschen

Mit dem delete-sync-configuration-Befehl in der AWS Command Line Interface (AWS CLI) können Sie eine angegebene Synchronisierungskonfiguration löschen.

> **Important**
>
> Nachdem Sie den Befehl ausgeführt haben, wird die Synchronisierungskonfiguration gelöscht. Es wird kein Bestätigungsdialogfeld angezeigt. Sie können eine neue

Synchronisierungskonfiguration erstellen. Der Amazon-Ressourcenname (ARN) wird jedoch nicht wiederverwendet.

So löschen Sie eine Synchronisierungskonfiguration

- Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den, AWS CLI um den delete-sync-configuration Befehl auszuführen und geben Sie den Synchronisationstyp und den Ressourcennamen für die Synchronisierungskonfiguration an, die Sie löschen möchten.

```
aws codeconnections delete-sync-configuration --sync-type CFN_STACK_SYNC --
resource-name mystack
```

Mit diesem Befehl wird kein Inhalt zurückgegeben.

Details zu Synchronisierungskonfigurationen anzeigen

Sie können den get-sync-configuration Befehl in AWS Command Line Interface (AWS CLI) verwenden, um Details für eine Synchronisierungskonfiguration anzuzeigen.

So zeigen Sie Details für eine Synchronisierungskonfiguration an

1. Öffnen Sie die Eingabeaufforderung (Windows) oder das Terminal (Linux, macOS oder Unix). Verwenden Sie den AWS CLI , um den get-sync-configuration Befehl auszuführen, und geben Sie dabei die Repository-Link-ID an.

```
aws codeconnections get-sync-configuration --sync-type CFN_STACK_SYNC --resource-
name mystack
```

2. Dieser Befehl gibt die folgende Ausgabe zurück.

```
{
    "SyncConfiguration": {
        "Branch": "main",
        "ConfigFile": "filename",
        "OwnerId": "owner_id",
        "ProviderType": "GitHub",
        "RepositoryLinkId": "be8f2017-b016-4a77-87b4-608054f70e77",
        "RepositoryName": "MyRepo",
```

```
        "ResourceName": "mystack",
        "RoleArn": "arn:aws:iam::account_id:role/myrole",
        "SyncType": "CFN_STACK_SYNC"
    }
}
```

AWS CodeConnections API-Aufrufe protokollieren mit AWS CloudTrail

AWS CodeConnections ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der von einem Benutzer, einer Rolle oder einem AWS Dienst ausgeführten Aktionen bereitstellt. CloudTrail erfasst alle API-Aufrufe für Benachrichtigungen als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Developer Tools-Konsole und Code-Aufrufe der AWS CodeConnections -API-Operationen.

Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon Simple Storage Service (Amazon S3) -Bucket aktivieren, einschließlich Ereignissen für Benachrichtigungen. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf einsehen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, an die die Anfrage gestellt wurde AWS CodeConnections, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und andere Details ermitteln.

Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.

AWS CodeConnections Informationen in CloudTrail

CloudTrail ist in Ihrem AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn eine Aktivität in stattfindet AWS CodeConnections, 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 im AWS CloudTrail Benutzerhandbuch unter Ereignisse mit CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich der Ereignisse für AWS CodeConnections, 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 anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen 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 in folgenden Themen im AWS CloudTrail -Benutzerhandbuch:

- 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
- Empfangen von CloudTrail Protokolldateien von mehreren Konten

Alle AWS CodeConnections Aktionen werden von der AWS CodeConnections API-Referenz protokolliert CloudTrail und sind in dieser dokumentiert. Beispielsweise generieren Aufrufe von `DeleteConnection` und `GetConnection` Aktionen Einträge in den CloudTrail Protokolldateien. `CreateConnection`

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

- Gibt an, ob die Anfrage mit Root- oder anderen IAM-Anmeldeinformationen ausgeführt 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 -Element userIdentity.

Grundlagen zu -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.

`CreateConnection`Beispiel für

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


```
{
    "EventId": "b4374fde-c544-4d43-b511-7d899568e55a",
    "EventName": "CreateConnection",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-09T15:13:46-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:user/Mary_Major",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-09T23:03:08Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-09T23:13:46Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "CreateConnection",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
```

```
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.create-connection",
        "requestParameters": {
            "providerType": "GitHub",
            "connectionName": "my-connection"
        },
        "responseElements": {
            "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/df03df74-8e05-45cf-b420-b39e389dd264"
        },
        "requestID": "57640a88-97b7-481d-9665-cfd79a681379",
        "eventID": "b4374fde-c544-4d43-b511-7d899568e55a",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`CreateHost`Beispiel für

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

```
{
    "EventId": "af4ce349-9f21-43fb-8003-267fbf9b1a93",
    "EventName": "CreateHost",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T12:43:06-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
```

```
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T20:43:06Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "CreateHost",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "52.94.133.137",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.create-host",
        "requestParameters": {
            "name": "Demo1",
            "providerType": "GitHubEnterpriseServer",
            "providerEndpoint": "IP"
        },
        "responseElements": {
            "hostArn": "arn:aws:codeconnections:us-east-1:123456789012:host/Demo1-
EXAMPLE"
        },
        "requestID": "974459b3-8a04-4cff-9c8f-0c88647831cc",
        "eventID": "af4ce349-9f21-43fb-8003-267fbf9b1a93",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
```

```
}
```

`CreateSyncConfiguration`Beispiel für

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


```
{
    "EventId": "be1397e1-eefb-49f0-b4ee-2708c45e94e7",
    "EventName": "CreateSyncConfiguration",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T17:38:30+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T17:34:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T17:38:30Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "CreateSyncConfiguration",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/offcommand/
codeconnections.create-sync-configuration",
        "requestParameters": {
            "branch": "master",
            "configFile": "filename",
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "resourceName": "mystack",
            "roleArn": "arn:aws:iam::123456789012:role/my-role",
            "syncType": "CFN_STACK_SYNC"
        },
        "responseElements": {
            "syncConfiguration": {
                "branch": "main",
                "configFile": "filename",
                "ownerId": "owner_ID",
                "providerType": "GitHub",
                "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                "repositoryName": "MyGitHubRepo",
                "resourceName": "mystack",
                "roleArn": "arn:aws:iam::123456789012:role/my-role",
                "syncType": "CFN_STACK_SYNC"
            }
        },
        "requestID": "bad2f662-3f2a-42c0-b638-6115384896f6",
        "eventID": "be1397e1-eefb-49f0-b4ee-2708c45e94e7",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```


`DeleteConnection`Beispiel für

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

```
{
    "EventId": "672837cd-f977-4fe2-95c7-14280b2af76c",
    "EventName": "DeleteConnection",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-10T13:00:50-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::001919387613:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-10T20:41:16Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-10T21:00:50Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "DeleteConnection",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.delete-connection",
        "requestParameters": {
            "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/df03df74-8e05-45cf-b420-b39e389dd264"
        },
```

```
        "responseElements": null,
        "requestID": "4f26ceab-d665-41df-9e15-5ed0fbb4eca6",
        "eventID": "672837cd-f977-4fe2-95c7-14280b2af76c",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`DeleteHost`Beispiel für

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

```
{
    "EventId": "6018ba5c-6f24-4a30-b201-16ec19a1687a",
    "EventName": "DeleteHost",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T12:56:47-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
```

```
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T20:56:47Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "DeleteHost",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.delete-host",
        "requestParameters": {
            "hostArn": "arn:aws:codeconnections:us-east-1:123456789012:host/Demo1-
EXAMPLE"
        },
        "responseElements": null,
        "requestID": "1b244528-143a-4028-b9a4-9479e342bce5",
        "eventID": "6018ba5c-6f24-4a30-b201-16ec19a1687a",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`DeleteSyncConfiguration`Beispiel für

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

```
{
    "EventId": "588660c7-3202-4998-a906-7bb72bcf4438",
    "EventName": "DeleteSyncConfiguration",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
```

```
    "EventTime": "2024-01-24T17:41:59+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T17:34:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T17:41:59Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "DeleteSyncConfiguration",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "52.94.133.142",
        "userAgent": "aws-
cli/2.15.11Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/
offcommand/codeconnections.delete-sync-configuration",
        "requestParameters": {
            "syncType": "CFN_STACK_SYNC",
            "resourceName": "mystack"
        },
        "responseElements": null,
        "requestID": "221e0b1c-a50e-4cf0-ab7d-780154e29c94",
        "eventID": "588660c7-3202-4998-a906-7bb72bcf4438",
        "readOnly": false,
        "eventType": "AwsApiCall",
```

```
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`GetConnection`Beispiel für

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

```
{
    "EventId": "672837cd-f977-4fe2-95c7-14280b2af76c",
    "EventName": "DeleteConnection",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-10T13:00:50-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-10T20:41:16Z",
                    "mfaAuthenticated": "false"
```

```
                }
            }
        },
        "eventTime": "2024-01-10T21:00:50Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "DeleteConnection",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.delete-connection",
        "requestParameters": {
            "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/df03df74-8e05-45cf-b420-b39e389dd264"
        },
        "responseElements": null,
        "requestID": "4f26ceab-d665-41df-9e15-5ed0fbb4eca6",
        "eventID": "672837cd-f977-4fe2-95c7-14280b2af76c",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "001919387613",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```


GetHostBeispiel für

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


```
{
    "EventId": "faa147e7-fe7c-4ab9-a11b-2568a2883c01",
    "EventName": "GetHost",
    "ReadOnly": "true",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T12:44:34-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
```

```
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T20:44:34Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetHost",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "52.94.133.137",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.get-host",
        "requestParameters": {
            "hostArn": "arn:aws:codeconnections:us-east-1:123456789012:host/Demo1-
EXAMPLE"
        },
        "responseElements": null,
        "requestID": "0ad61bb6-f88f-4f96-92fe-997f017ec2bb",
        "eventID": "faa147e7-fe7c-4ab9-a11b-2568a2883c01",
        "readOnly": true,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
```

```
}
```

`GetRepositoryLink`Beispiel für

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

```
{
    "EventId": "b46acb67-3612-41c7-8987-adb6c9ed4ad4",
    "EventName": "GetRepositoryLink",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T02:59:28+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T02:58:52Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T02:59:28Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetRepositoryLink",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11
 Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off
 command/codeconnections.get-repository-link",
        "requestParameters": {
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173"
        },
        "responseElements": {
            "repositoryLinkInfo": {
                "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/7df263cc-f055-4843-adef-4ceaefcb2167",
                "ownerId": "123456789012",
                "providerType": "GitHub",
                "repositoryLinkArn": "arn:aws:codeconnections:us-
east-1:123456789012:repository-link/6053346f-8a33-4edb-9397-10394b695173",
                "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                "repositoryName": "MyGitHubRepo"
            }
        },
        "requestID": "d46704dd-dbe9-462f-96a6-022a8d319fd1",
        "eventID": "b46acb67-3612-41c7-8987-adb6c9ed4ad4",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-ea-1.codeconnections.aws.dev"
        }
    }
}
```

`GetRepositorySyncStatus`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die GetRepositorySyncStatusAktion demonstriert.

```
{
    "EventId": "3e183b74-d8c4-4ad3-9de3-6b5721c522e9",
    "EventName": "GetRepositorySyncStatus",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T03:41:44+00:00",
```

```
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-25T02:56:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T03:41:44Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetRepositorySyncStatus",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "52.94.133.138",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.get-repository-sync-status",
        "errorCode": "ResourceNotFoundException",
        "errorMessage": "Could not find a sync status for repository
 link:6053346f-8a33-4edb-9397-10394b695173",
        "requestParameters": {
            "branch": "feature-branch",
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "syncType": "CFN_STACK_SYNC"
        },
        "responseElements": null,
        "requestID": "e0cee3ee-31e8-4ef5-b749-96cdcabbe36f",
```

```
        "eventID": "3e183b74-d8c4-4ad3-9de3-6b5721c522e9",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`GetResourceSyncStatus`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die GetResourceSyncStatusAktion demonstriert.

```
{
    "EventId": "9c47054e-f6f6-4345-96d0-9a5af3954a8d",
    "EventName": "GetResourceSyncStatus",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T03:44:11+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
```

```
                "attributes": {
                    "creationDate": "2024-01-25T02:56:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T03:44:11Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetResourceSyncStatus",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.get-resource-sync-status",
        "requestParameters": {
            "resourceName": "mystack",
            "syncType": "CFN_STACK_SYNC"
        },
        "responseElements": null,
        "requestID": "e74b5503-d651-4920-9fd2-0f40fb5681e0",
        "eventID": "9c47054e-f6f6-4345-96d0-9a5af3954a8d",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```


`GetSyncBlockerSummary`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die GetSyncBlockerSummaryAktion demonstriert.


```
{
    "EventId": "c16699ba-a788-476d-8c6c-47511d76309e",
    "EventName": "GetSyncBlockerSummary",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T03:03:02+00:00",
```

```
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-25T02:56:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T03:03:02Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetSyncBlockerSummary",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.get-sync-blocker-summary",
        "requestParameters": {
            "syncType": "CFN_STACK_SYNC",
            "resourceName": "mystack"
        },
        "responseElements": {
            "syncBlockerSummary": {
                "resourceName": "mystack",
                "latestBlockers": []
            }
        },
```

```
        "requestID": "04240091-eb25-4138-840d-776f8e5375b4",
        "eventID": "c16699ba-a788-476d-8c6c-47511d76309e",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`GetSyncConfiguration`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die GetSyncConfigurationAktion demonstriert.

```
{
    "EventId": "bab9aa16-4553-4206-a1ea-88219233dd25",
    "EventName": "GetSyncConfiguration",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T17:40:40+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
```

```
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T17:34:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T17:40:40Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "GetSyncConfiguration",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "52.94.133.142",
        "userAgent": "aws-
cli/2.15.11Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/
offcommand/codeconnections.get-sync-configuration",
        "requestParameters": {
            "syncType": "CFN_STACK_SYNC",
            "resourceName": "mystack"
        },
        "responseElements": {
            "syncConfiguration": {
                "branch": "main",
                "configFile": "filename",
                "ownerId": "123456789012",
                "providerType": "GitHub",
                "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                "repositoryName": "MyGitHubRepo",
                "resourceName": "mystack",
                "roleArn": "arn:aws:iam::123456789012:role/my-role",
                "syncType": "CFN_STACK_SYNC"
            }
        },
        "requestID": "0aa8e43a-6e34-4d8f-89fb-5c2d01964b35",
        "eventID": "bab9aa16-4553-4206-a1ea-88219233dd25",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
```

```
}
```

`ListConnections`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListConnectionsAktion demonstriert.


```
{
    "EventId": "3f8d80fe-fbe1-4755-903c-4f58fc8262fa",
    "EventName": "ListConnections",
    "ReadOnly": "true",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-08T14:11:23-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-08T22:11:02Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-08T22:11:23Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListConnections",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/1.18.147 Python/2.7.18
 Linux/5.10.201-168.748.amzn2int.x86_64 botocore/1.18.6",
        "requestParameters": {
            "maxResults": 50
        },
        "responseElements": null,
        "requestID": "5d456d59-3e92-44be-b941-a429df59e90b",
        "eventID": "3f8d80fe-fbe1-4755-903c-4f58fc8262fa",
        "readOnly": true,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`ListHosts`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListHostsAktion demonstriert.

```
{
    "EventId": "f6e9e831-feaf-4ad1-ac47-51681109c401",
    "EventName": "ListHosts",
    "ReadOnly": "true",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T13:00:55-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
```

```
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T21:00:55Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListHosts",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.list-hosts",
        "requestParameters": {
            "maxResults": 50
        },
        "responseElements": null,
        "requestID": "ea87e2cf-6bf1-4cc7-9666-f3fad85d6d83",
        "eventID": "f6e9e831-feaf-4ad1-ac47-51681109c401",
        "readOnly": true,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```


`ListRepositoryLinks`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListRepositoryLinksAktion demonstriert.


```
{
```

```
    "EventId": "4f714bbb-0716-4f6e-9868-9b379b30757f",
    "EventName": "ListRepositoryLinks",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T01:57:29+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T01:43:49Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T01:57:29Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListRepositoryLinks",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-
cli/2.15.11Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/
offcommand/codeconnections.list-repository-links",
        "requestParameters": {
            "maxResults": 50
        },
        "responseElements": {
            "repositoryLinks": [
```

```
                {
                    "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/001f5be2-a661-46a4-b96b-4d277cac8b6e",
                    "ownerId": "123456789012",
                    "providerType": "GitHub",
                    "repositoryLinkArn": "arn:aws:codeconnections:us-
east-1:123456789012:repository-link/be8f2017-b016-4a77-87b4-608054f70e77",
                    "repositoryLinkId": "be8f2017-b016-4a77-87b4-608054f70e77",
                    "repositoryName": "MyGitHubRepo"
                },
                {
                    "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/7df263cc-f055-4843-adef-4ceaefcb2167",
                    "ownerId": "owner",
                    "providerType": "GitHub",
                    "repositoryLinkArn": "arn:aws:codeconnections:us-
east-1:123456789012:repository-link/6053346f-8a33-4edb-9397-10394b695173",
                    "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                    "repositoryName": "MyGitHubRepo"
                }
            ]
        },
        "requestID": "7c8967a9-ec15-42e9-876b-0ef58681ec55",
        "eventID": "4f714bbb-0716-4f6e-9868-9b379b30757f",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`ListRepositorySyncDefinitions`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListRepositorySyncDefinitionsAktion demonstriert.

```
{
    "EventId": "12e52dbb-b00d-49ad-875a-3efec36e5aa1",
    "EventName": "ListRepositorySyncDefinitions",
```

```
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T16:56:19+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-25T16:43:03Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T16:56:19Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListRepositorySyncDefinitions",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.list-repository-sync-definitions",
        "requestParameters": {
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "syncType": "CFN_STACK_SYNC",
            "maxResults": 50
        },
        "responseElements": {
            "repositorySyncDefinitions": []
```

```
        },
        "requestID": "df31d11d-5dc7-459b-9a8f-396b4769cdd9",
        "eventID": "12e52dbb-b00d-49ad-875a-3efec36e5aa1",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
```

`ListSyncConfigurations`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListSyncConfigurationsAktion demonstriert.

```
{
    "EventId": "aa4ae557-ec31-4151-8d21-9e74dd01344c",
    "EventName": "ListSyncConfigurations",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T17:42:06+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
```

```
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T17:34:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T17:42:06Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListSyncConfigurations",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.804.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/offcommand/
codeconnections.list-sync-configurations",
        "requestParameters": {
            "maxResults": 50,
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
            "syncType": "CFN_STACK_SYNC"
        },
        "responseElements": {
            "syncConfigurations": [
                {
                    "branch": "feature-branch",
                    "configFile": "filename.yaml",
                    "ownerId": "owner",
                    "providerType": "GitHub",
                    "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                    "repositoryName": "MyGitHubRepo",
                    "resourceName": "dkstacksync",
                    "roleArn": "arn:aws:iam::123456789012:role/my-role",
                    "syncType": "CFN_STACK_SYNC"
                }
            ]
        },
        "requestID": "7dd220b5-fc0f-4023-aaa0-9555cfe759df",
        "eventID": "aa4ae557-ec31-4151-8d21-9e74dd01344c",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
```

```
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`ListTagsForResource`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die ListTagsForResourceAktion demonstriert.

```
{
    "EventId": "fc501054-d68a-4325-824c-0e34062ef040",
    "EventName": "ListTagsForResource",
    "ReadOnly": "true",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T17:16:56+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "dMary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-25T16:43:03Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T17:16:56Z",
```

```
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "ListTagsForResource",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.list-tags-for-resource",
        "requestParameters": {
            "resourceArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/9703702f-bebe-41b7-8fc4-8e6d2430a330"
        },
        "responseElements": null,
        "requestID": "994584a3-4807-47f2-bb1b-a64f0af6c250",
        "eventID": "fc501054-d68a-4325-824c-0e34062ef040",
        "readOnly": true,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`TagResource`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die TagResourceAktion demonstriert.

```
{
    "EventId": "b7fbc943-2dd1-4c5b-a5ad-fc6d60a011f1",
    "EventName": "TagResource",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T12:22:11-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
```

```
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T20:22:11Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "TagResource",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.tag-resource",
        "requestParameters": {
            "resourceArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/8dcf69d1-3316-4392-ae09-71e038adb6ed",
            "tags": [
                {
                    "key": "Demo1",
                    "value": "hhvh1"
                }
            ]
        },
        "responseElements": null,
        "requestID": "ba382c33-7124-48c8-a23a-25816ce27604",
        "eventID": "b7fbc943-2dd1-4c5b-a5ad-fc6d60a011f1",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
```

```
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`UnTagResource`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die UntagResourceAktion demonstriert.

```
{
    "EventId": "8a85cdee-2586-4679-be18-eec34204bc7e",
    "EventName": "UntagResource",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-11T12:31:14-08:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
```

```
        "eventTime": "2024-01-11T20:31:14Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "UntagResource",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.untag-resource",
        "requestParameters": {
            "resourceArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/8dcf69d1-3316-4392-ae09-71e038adb6ed",
            "tagKeys": [
                "Project",
                "ReadOnly"
            ]
        },
        "responseElements": null,
        "requestID": "05ef26a4-8c39-4f72-89bf-0c056c51b8d7",
        "eventID": "8a85cdee-2586-4679-be18-eec34204bc7e",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`UpdateHost`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die UpdateHostAktion demonstriert.

```
"Events": [{
        "EventId": "4307cf7d-6d1c-40d9-a659-1bb41b31a2b6",
        "EventName": "UpdateHost",
        "ReadOnly": "false",
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "EventTime": "2024-01-11T12:54:32-08:00",
        "EventSource": "codeconnections.amazonaws.com",
        "Username": "Mary_Major",
        "Resources": [],
        "CloudTrailEvent": "eventVersion": "1.08",
```

```
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-11T20:09:35Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-11T20:54:32Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "UpdateHost",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.13.30 Python/3.11.6 Darwin/23.2.0 exe/x86_64 prompt/off
 command/codeconnections.update-host",
        "requestParameters": {
            "hostArn": "arn:aws:codeconnections:us-east-1:123456789012:host/
Demo1-34e70ecb",
            "providerEndpoint": "https://54.218.245.167"
        },
        "responseElements": null,
        "requestID": "b17f46ac-1acb-44ab-a9f5-c35c20233441",
        "eventID": "4307cf7d-6d1c-40d9-a659-1bb41b31a2b6",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
```

```
        }
```

`UpdateRepositoryLink`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die UpdateRepositoryLinkAktion demonstriert.

```
{
    "EventId": "be358c9a-5a8f-467e-8585-2860070be4fe",
    "EventName": "UpdateRepositoryLink",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T02:03:24+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T01:43:49Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T02:03:24Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "UpdateRepositoryLink",
        "awsRegion": "us-east-1",
```

```
        "sourceIPAddress": "IP",
        "userAgent": "aws-
cli/2.15.11Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/
offcommand/codeconnections.update-repository-link",
        "requestParameters": {
            "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/7df263cc-f055-4843-adef-4ceaefcb2167",
            "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173"
        },
        "responseElements": {
            "repositoryLinkInfo": {
                "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/7df263cc-f055-4843-adef-4ceaefcb2167",
                "ownerId": "owner",
                "providerType": "GitHub",
                "repositoryLinkArn": "arn:aws:codeconnections:us-
east-1:123456789012:repository-link/6053346f-8a33-4edb-9397-10394b695173",
                "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                "repositoryName": "MyGitHubRepo"
            }
        },
        "additionalEventData": {
            "providerAction": "UpdateRepositoryLink"
        },
        "requestID": "e01eee49-9393-4983-89e4-d1b3353a70d9",
        "eventID": "be358c9a-5a8f-467e-8585-2860070be4fe",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`UpdateSyncBlocker`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die UpdateSyncBlockerAktion demonstriert.

```
{
```

```
    "EventId": "211d19db-9f71-4d93-bf90-10f9ddefed88",
    "EventName": "UpdateSyncBlocker",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-25T03:01:05+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-25T02:56:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-25T03:01:05Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "UpdateSyncBlocker",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11 Python/3.11.6
 Linux/5.10.205-172.807.amzn2int.x86_64 exe/x86_64.amzn.2 prompt/off command/
codeconnections.update-sync-blocker",
        "requestParameters": {
            "id": "ID",
            "syncType": "CFN_STACK_SYNC",
            "resourceName": "mystack",
            "resolvedReason": "Reason"
```

```
        },
        "responseElements": null,
        "requestID": "eea03b39-b299-4099-ba55-608480f8d96d",
        "eventID": "211d19db-9f71-4d93-bf90-10f9ddefed88",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
        }
    }
}
```

`UpdateSyncConfiguration`Beispiel für

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die UpdateSyncConfigurationAktion demonstriert.

```
{
    "EventId": "d961c94f-1881-4fe8-83bf-d04cb9f22577",
    "EventName": "UpdateSyncConfiguration",
    "ReadOnly": "false",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "EventTime": "2024-01-24T17:40:55+00:00",
    "EventSource": "codeconnections.amazonaws.com",
    "Username": "Mary_Major",
    "Resources": [],
    "CloudTrailEvent": {
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "arn": "arn:aws:iam::123456789012:role/Admin",
                    "accountId": "123456789012",
```

```
                    "userName": "Admin"
                },
                "webIdFederationData": {},
                "attributes": {
                    "creationDate": "2024-01-24T17:34:55Z",
                    "mfaAuthenticated": "false"
                }
            }
        },
        "eventTime": "2024-01-24T17:40:55Z",
        "eventSource": "codeconnections.amazonaws.com",
        "eventName": "UpdateSyncConfiguration",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "IP",
        "userAgent": "aws-cli/2.15.11
 Python/3.11.6Linux/5.10.205-172.804.amzn2int.x86_64exe/x86_64.amzn.2prompt/offcommand/
codeconnections.update-sync-configuration",
        "requestParameters": {
            "branch": "feature-branch",
            "resourceName": "mystack",
            "syncType": "CFN_STACK_SYNC"
        },
        "responseElements": {
            "syncConfiguration": {
                "branch": "feature-branch",
                "configFile": "filename",
                "ownerId": "owner",
                "providerType": "GitHub",
                "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
                "repositoryName": "MyGitHubRepo",
                "resourceName": "mystack",
                "roleArn": "arn:aws:iam::123456789012:role/my-role",
                "syncType": "CFN_STACK_SYNC"
            }
        },
        "requestID": "2ca545ef-4395-4e1f-b14a-2750481161d6",
        "eventID": "d961c94f-1881-4fe8-83bf-d04cb9f22577",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "123456789012",
        "eventCategory": "Management",
        "tlsDetails": {
            "clientProvidedHostHeader": "api.us-east-1.codeconnections.aws.dev"
```

```
        }
    }
}
```

AWS CodeConnections und Schnittstellen-VPC-Endpunkte ()AWS PrivateLink

Sie können eine private Verbindung zwischen Ihrer VPC herstellen und AWS CodeConnections einen VPC-Schnittstellen-Endpunkt erstellen. Schnittstellenendpunkte werden mit einer Technologie betrieben AWS PrivateLink, mit der Sie privat AWS CodeConnections APIs ohne Internet-Gateway, NAT-Gerät, VPN-Verbindung oder AWS Direct Connect-Verbindung zugreifen können. Instances in Ihrer VPC benötigen keine öffentlichen IP-Adressen für die Kommunikation AWS CodeConnections APIs, da der Datenverkehr zwischen Ihrer VPC und dem Amazon-Netzwerk AWS CodeConnections nicht verlässt.

Jeder Schnittstellenendpunkt wird durch eine oder mehrere Elastic-Network-Schnittstellen in Ihren Subnetzen dargestellt.

Weitere Informationen finden Sie unter Schnittstellen-VPC-Endpunkte (AWS PrivateLink) im Amazon-VPC-Benutzerhandbuch.

Überlegungen zu AWS CodeConnections VPC-Endpunkten

Bevor Sie einen Schnittstellen-VPC-Endpunkt für einrichten AWS CodeConnections, sollten Sie die Schnittstellenendpunkte im Amazon VPC-Benutzerhandbuch lesen.

AWS CodeConnections unterstützt Aufrufe aller API-Aktionen von Ihrer VPC aus.

VPC-Endpunkte werden in allen AWS CodeConnections Regionen unterstützt.

VPC-Endpunktkonzepte

Die wichtigsten Konzepte für VPC-Endpunkte sind folgende:

VPC-Endpunkt

Der Eintrittspunkt in Ihrer VPC, über den Sie eine private Verbindung zu einem Service erstellen können. Im Folgenden werden die verschiedenen Arten von VPC-Endpunkten beschrieben. Erstellen Sie die Art von VPC-Endpunkt, die von dem unterstützten Service benötigt wird.

- VPC-Endpunkte für Aktionen AWS CodeConnections
- VPC-Endpunkte für Webhooks AWS CodeConnections

AWS PrivateLink

Eine Technologie, die private Konnektivität zwischen VPCs und Diensten bietet.

VPC-Endpunkte für Aktionen AWS CodeConnections

Sie können VPC-Endpunkte für den AWS CodeConnections Service verwalten.

Schnittstellen-VPC-Endpunkte für Aktionen erstellen AWS CodeConnections

Sie können einen VPC-Endpunkt für den AWS CodeConnections Service entweder mit der Amazon VPC-Konsole oder mit AWS Command Line Interface ()AWS CLI erstellen. Weitere Informationen finden Sie unter Erstellung eines Schnittstellenendpunkts im Benutzerhandbuch für Amazon VPC.

Um mit der Nutzung von Verbindungen mit Ihrer VPC zu beginnen, erstellen Sie einen VPC-Schnittstellen-Endpunkt für. AWS CodeConnections Wenn Sie einen VPC-Endpunkt für erstellen AWS CodeConnections, wählen Sie AWS Services und unter Service Name Folgendes aus:

- com.amazonaws.***region***.codestar-connections.api: Diese Option erstellt einen VPC-Endpunkt für API-Operationen. AWS CodeConnections Wählen Sie diese Option beispielsweise, wenn Ihre Benutzer die AWS CLI, die AWS CodeConnections API oder die verwenden, AWS SDKs AWS CodeConnections um mit Vorgängen wie `CreateConnectionListConnections`, und zu interagieren`CreateHost`.

Wenn Sie für die Option „DNS-Namen aktivieren“ privates DNS für den Endpunkt auswählen, können Sie API-Anfragen an die AWS CodeConnections Verwendung des Standard-DNS-Namens für die Region stellen, `codestar-connections.us-east-1.amazonaws.com` z. B.

> **Important**
>
> Private DNS ist standardmäßig für Endpoints aktiviert, die für AWS Dienste und AWS Marketplace-Partner-Dienste erstellt wurden.

Weitere Informationen finden Sie unter Zugriff auf einen Service über einen Schnittstellenendpunkt im Benutzerhandbuch für Amazon VPC.

Erstellen einer VPC-Endpunktrichtlinie für Aktionen AWS CodeConnections

Sie können eine Endpunktrichtlinie an Ihren VPC-Endpunkt anhängen, der den Zugriff auf AWS CodeConnections steuert. Die Richtlinie gibt die folgenden Informationen an:

- Prinzipal, der die Aktionen ausführen kann.
- Aktionen, die ausgeführt werden können
- Die Ressourcen, für die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie unter Steuerung des Zugriffs auf Services mit VPC-Endpunkten im Amazon-VPC-Benutzerhandbuch.

> **Note**
>
> Die Datei com.amazonaws. *region*Der Endpunkt .codestar-connections.webhooks unterstützt keine Richtlinien.

Beispiel: VPC-Endpunktrichtlinie für Aktionen AWS CodeConnections

Im Folgenden finden Sie ein Beispiel für eine Endpunktrichtlinie für AWS CodeConnections. Wenn diese Richtlinie an einen Endpunkt angehängt ist, gewährt sie allen Prinzipalen auf allen Ressourcen Zugriff auf die aufgelisteten AWS CodeConnections Aktionen.

```
{
  "Statement": [
    {
      "Sid": "GetConnectionOnly",
      "Principal": "*",
      "Action": [
        "codestar-connections:GetConnection"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

VPC-Endpunkte für Webhooks AWS CodeConnections

AWS CodeConnections erstellt Webhook-Endpunkte für Sie, wenn Sie einen Host mit VPC-Konfiguration erstellen oder löschen. Der Endpunktname lautet com.amazonaws. ***region***.codestar-connections.webhooks.

Mit dem VPC-Endpunkt für GitHub Webhooks können Hosts Ereignisdaten über Webhooks über das Amazon-Netzwerk an Ihre integrierten AWS Dienste senden.

> **Important**
>
> Wenn Sie Ihren Host für GitHub Enterprise Server einrichten, AWS CodeConnections erstellt er einen VPC-Endpunkt für Webhooks-Ereignisdaten für Sie. Wenn Sie Ihren Host vor dem 24. November 2020 erstellt haben und PrivateLink VPC-Webhook-Endpunkte verwenden möchten, müssen Sie zuerst Ihren Host löschen und dann einen neuen Host erstellen.

AWS CodeConnections verwaltet den Lebenszyklus dieser Endpunkte. Um den Endpunkt zu löschen, müssen Sie die entsprechende Hostressource löschen.

Wie werden Webhook-Endpunkte für Hosts verwendet AWS CodeConnections

Der Webhook-Endpunkt ist der Ort, an den Webhooks aus Repositorys von Drittanbietern zur Verarbeitung gesendet werden. AWS CodeConnections Ein Webhook beschreibt eine Kundenaktion. Wenn Sie einen `git push` ausführen, erhält der Webhook-Endpunkt einen Webhook vom Anbieten, der den Push detailliert beschreibt. AWS CodeConnections Kann beispielsweise benachrichtigen, dass Ihre Pipeline gestartet CodePipeline werden soll.

Für Cloud-Anbieter wie Bitbucket oder GitHub Enterprise Server-Hosts, die keine VPC verwenden, gilt der Webhook-VPC-Endpunkt nicht, da die Anbieter Webhooks dorthin senden, AWS CodeConnections wo das Amazon-Netzwerk nicht verwendet wird.

Fehlerbehebung bei Verbindungen

Die folgenden Informationen können Ihnen helfen, häufig auftretende Probleme mit Verbindungen zu Ressourcen in AWS CodeBuild AWS CodeDeploy, und zu beheben AWS CodePipeline.

Themen

- Ich kann keine Verbindungen erstellen

- Ein Berechtigungsfehler erscheint, wenn ich versuche, eine Verbindung zu erstellen oder abzuschließen
- Ein Berechtigungsfehler erscheint, wenn ich versuche, eine Verbindung zu verwenden.
- Die Verbindung befindet sich nicht im Status „Available (Verfügbar)“ oder ist nicht mehr ausstehend
- Fügen Sie Berechtigungen für Verbindungen hinzu GitClone
- Der Host befindet sich nicht im Status „Available (Verfügbar)“
- Fehlerbehebung bei einem Host mit Verbindungsfehlern
- Ich kann keine Verbindung für meinen Host erstellen
- Fehlerbehebung bei der VPC-Konfiguration für Ihren Host
- Problembehandlung bei Webhook-VPC-Endpunkten (PrivateLink) für Enterprise Server-Verbindungen GitHub
- Fehlerbehebung für einen Host, der vor dem 24. November 2020 erstellt wurde
- Die Verbindung für ein Repository konnte nicht hergestellt werden GitHub
- Bearbeiten Sie die Berechtigungen Ihrer GitHub Enterprise Server-Verbindungs-App
- Verbindungsfehler bei der Verbindung zu GitHub: „Es ist ein Problem aufgetreten, stellen Sie sicher, dass Cookies in Ihrem Browser aktiviert sind“ oder „Ein Organisationsinhaber muss die GitHub App installieren“
- Das Verbindungsdienstpräfix in Ressourcen muss möglicherweise für IAM-Richtlinien aktualisiert werden
- Berechtigungsfehler aufgrund eines Dienstpräfixes in Ressourcen, die mit der Konsole erstellt wurden
- Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen
- Ich möchte die Limits bei Verbindungen erhöhen

Ich kann keine Verbindungen erstellen

Möglicherweise verfügen Sie nicht über Berechtigungen zum Erstellen einer Verbindung. Weitere Informationen finden Sie unter Berechtigungen und Beispiele für AWS CodeConnections.

Ein Berechtigungsfehler erscheint, wenn ich versuche, eine Verbindung zu erstellen oder abzuschließen

Die folgende Fehlermeldung wird möglicherweise zurückgegeben, wenn Sie versuchen, eine Verbindung in der CodePipeline Konsole herzustellen oder anzuzeigen.

Benutzer: *username* ist nicht berechtigt, Folgendes *permission* auf der Ressource auszuführen: *connection-ARN*

Wenn diese Nachricht erscheint, stellen Sie sicher, dass Sie über ausreichend Berechtigungen verfügen.

Die Berechtigungen zum Erstellen und Anzeigen von Verbindungen in der AWS Command Line Interface (AWS CLI) oder der AWS Management Console sind nur ein Teil der Berechtigungen, die Sie zum Erstellen und Abschließen von Verbindungen auf der Konsole benötigen. Benutzer, die nur bestimmte Aufgaben ausführen müssen, brauchen nur die Berechtigungen zum einfachen Anzeigen, Bearbeiten oder Erstellen einer Verbindung und zum anschließenden Abschluss der ausstehenden Verbindung. Weitere Informationen finden Sie unter Berechtigungen und Beispiele für AWS CodeConnections.

Ein Berechtigungsfehler erscheint, wenn ich versuche, eine Verbindung zu verwenden.

Eine oder beide der folgenden Fehlermeldungen werden möglicherweise zurückgegeben, wenn Sie versuchen, eine Verbindung in der CodePipeline Konsole zu verwenden, obwohl Sie über die erforderlichen Berechtigungen zum Auflisten, Abrufen und Erstellen von Berechtigungen verfügen.

Sie haben Ihr Konto nicht authentifiziert.

Benutzer: *username* ist nicht berechtigt, Folgendes auszuführen: codestar-connections: auf der Ressource: UseConnection *connection-ARN*

Wenn das vorkommt, stellen Sie sicher, dass Sie über ausreichende Berechtigungen verfügen.

Die benötigen die Berechtigungen zum Verwenden einer Verbindung und zum Auflisten der verfügbaren Repositorys am Providerspeicherort. Weitere Informationen finden Sie unter Berechtigungen und Beispiele für AWS CodeConnections.

Die Verbindung befindet sich nicht im Status „Available (Verfügbar)“ oder ist nicht mehr ausstehend

Wenn die Konsole meldet, dass eine Verbindung nicht verfügbar ist, wählen Sie Complete connection (Verbindung abschließen) aus.

Wenn Sie die Verbindung abschließen möchten und eine Meldung erscheint, dass sich die Verbindung nicht im Zustand „pending (asusstehend)“ befindet, können Sie die Anforderung abbrechen: die Verbindung hat denn bereits den Zustand „available (verfügbar)“.

Fügen Sie Berechtigungen für Verbindungen hinzu GitClone

Wenn Sie eine AWS CodeStar Verbindung in einer Quellaktion und einer CodeBuild Aktion verwenden, gibt es zwei Möglichkeiten, wie das Eingabeartefakt an den Build übergeben werden kann:

- Der Standardwert: Die Quellaktion erzeugt eine ZIP-Datei, die den Code enthält, den CodeBuild herunterlädt.
- Git-Klon: Der Quellcode kann direkt in die Build-Umgebung heruntergeladen werden.

  Der Git-Klon-Modus ermöglicht es Ihnen, mit dem Quellcode als funktionierendes Git-Repository zu interagieren. Um diesen Modus verwenden zu können, müssen Sie Ihrer CodeBuild Umgebung Berechtigungen zur Verwendung der Verbindung erteilen.

Um Ihrer CodeBuild Servicerollenrichtlinie Berechtigungen hinzuzufügen, erstellen Sie eine vom Kunden verwaltete Richtlinie, die Sie Ihrer CodeBuild Servicerolle zuordnen. Mit den folgenden Schritten wird eine Richtlinie erstellt, bei der die `UseConnection`-Berechtigung im `action`-Feld und der Amazon-Ressourcenname (ARN) der Verbindung im `Resource`-Feld angegeben wird.

So verwenden Sie die Konsole, um die UseConnection Berechtigungen hinzuzufügen

1. Um den Verbindungs-ARN für Ihre Pipeline zu finden, öffnen Sie die Pipeline und wählen Sie das (i)-Symbol in der Quellaktion aus. Der Konfigurationsbereich wird geöffnet, und der Verbindungs-ARN wird neben angezeigt ConnectionArn. Sie fügen den Verbindungs-ARN zu Ihrer CodeBuild Servicerollenrichtlinie hinzu.
2. Um Ihre CodeBuild Servicerolle zu finden, öffnen Sie das in Ihrer Pipeline verwendete Build-Projekt und navigieren Sie zur Registerkarte Build-Details.
3. Wählen Sie im Bereich „Environment (Umgebung)“ den Link Service role (Servicerolle). Dadurch wird die AWS Identity and Access Management (IAM-) Konsole geöffnet, in der Sie eine neue Richtlinie hinzufügen können, die Zugriff auf Ihre Verbindung gewährt.
4. Wählen Sie in der IAM-Konsole Attach policies (Richtlinien anhängen) und dann Create policy (Richtlinie erstellen).

   Verwenden Sie die folgende Beispielrichtlinienvorlage. Fügen Sie Ihren Verbindungs-ARN in das `Resource`-Feld ein, wie in diesem Beispiel gezeigt:

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

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "codestar-connections:UseConnection",
            "Resource": "insert connection ARN here"
        }
    ]
}
```

Fügen Sie auf der Registerkarte JSON Ihre Richtlinie ein.

5. Wählen Sie Richtlinie prüfen. Geben Sie einen Namen für die Richtlinie ein (beispielsweise **`connection-permissions`**) und wählen Sie dann Create policy (Richtlinie erstellen) aus.
6. Kehren Sie zur Seite Attach Permissions (Berechtigungen anhängen) für die Servicerolle zurück, aktualisieren Sie die Richtlinienliste und wählen Sie die gerade erstellte Richtlinie aus. Wählen Sie Richtlinien anfügen.

Der Host befindet sich nicht im Status „Available (Verfügbar)“

Wenn die Konsole eine Meldung anzeigt, dass sich ein Host nicht im Status `Available` (Verfügbar) befindet, wählen Sie Set up host (Host einrichten) aus.

Der erste Schritt zur Host-Erstellung führt dazu, dass der erstellte Host jetzt im Zustand `Pending` (Ausstehend) ist. Um den Host in den Zustand `Available` (Verfügbar) zu bringen, müssen Sie festlegen, dass der Host in der Konsole eingerichtet werden soll. Weitere Informationen finden Sie unter Einrichten eines ausstehenden Hosts.

> **Note**
>
> Sie können die AWS CLI nicht verwenden, um einen `Pending` Host einzurichten.

Fehlerbehebung bei einem Host mit Verbindungsfehlern

Verbindungen und Hosts können in den Fehlerstatus wechseln, wenn die zugrunde liegende GitHub App gelöscht oder geändert wird. Hosts und Verbindungen im Fehlerstatus können nicht wiederhergestellt werden und der Host muss neu erstellt werden.

- Aktionen wie das Ändern des App-PEM-Schlüssels oder das Ändern des App-Namens (nach der ersten Erstellung) führen dazu, dass der Host und alle zugehörigen Verbindungen in den Fehlerzustand wechseln.

Wenn die Konsole oder die CLI einen Host zurückgibt, der im Zustand `Error` (Fehler) ist, oder eine Verbindung, die mit einem solchen fehlerhaften Host verknüpft ist, müssen Sie den folgenden Schritte ausführen:

- Löschen Sie die Hostressource, erstellen Sie sie neu, und installieren Sie dann die Host-Registrierungs-App neu. Weitere Informationen finden Sie unter Erstellen eines Hosts.

Ich kann keine Verbindung für meinen Host erstellen

Um eine Verbindung oder einen Host zu erstellen, müssen die folgenden Bedingungen erfüllt sein.

- Ihr Host muss sich im Zustand AVAILABLE (VERFÜGBAR) befinden. Weitere Informationen finden Sie unter
- Verbindungen müssen in derselben Region wie der Host erstellt werden.

Fehlerbehebung bei der VPC-Konfiguration für Ihren Host

Wenn Sie eine Hostressource erstellen, müssen Sie Netzwerkverbindungs- oder VPC-Informationen für die Infrastruktur angeben, in der Ihre GitHub Enterprise Server-Instance installiert ist. Verwenden Sie zur Fehlerbehebung bei der VPC- oder Subnetzkonfiguration für Ihren Host die hier gezeigten VPC-Informationen als Beispiel.

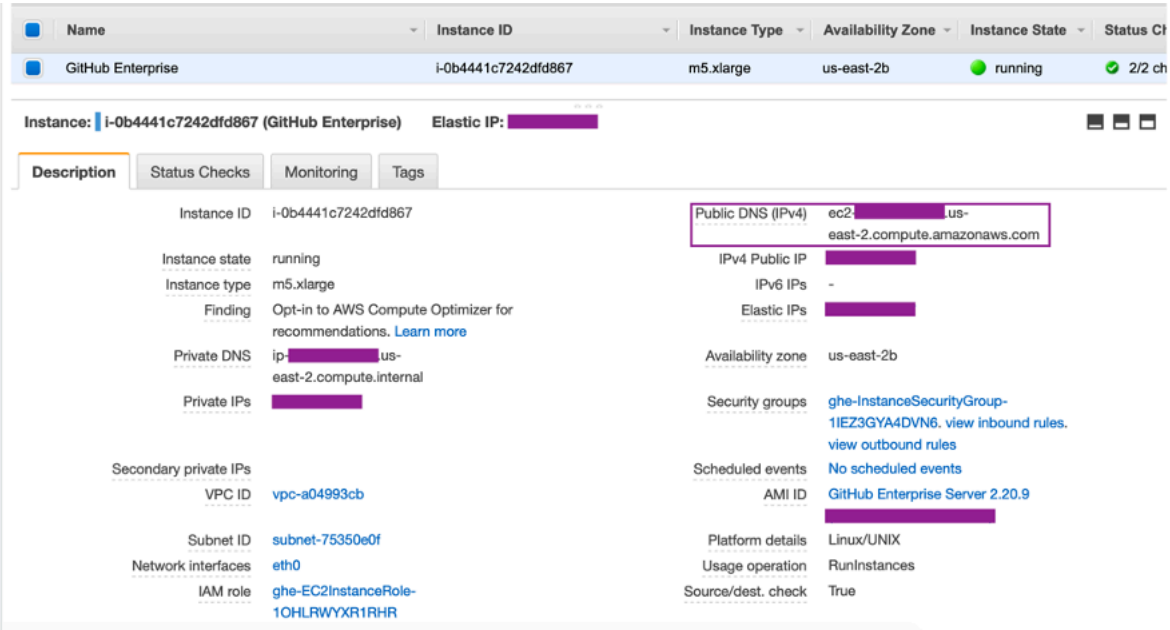
> **Note**
>
> Verwenden Sie diesen Abschnitt zur Fehlerbehebung im Zusammenhang mit Ihrer GitHub Enterprise Server-Host-Konfiguration in einer Amazon VPC. Informationen zur Problembehandlung im Zusammenhang mit Ihrer Verbindung, die für die Verwendung des Webhook-Endpunkts für VPC (PrivateLink) konfiguriert ist, finden Sie unter. Problembehandlung bei Webhook-VPC-Endpunkten (PrivateLink) für Enterprise Server-Verbindungen GitHub

In diesem Beispiel würden Sie den folgenden Prozess verwenden, um die VPC und den Server zu konfigurieren, auf denen Ihre GitHub Enterprise Server-Instanz installiert wird:

1. Erstellen Sie eine VPC. Weitere Informationen finden Sie unter https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#Create-VPC.
2. Erstellen eines Subnetzes in der VPC Weitere Informationen finden Sie unter https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#AddaSubnet.
3. Starten einer Instance in der VPC Weitere Informationen finden Sie unter https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#VPC_Launch_Instance.

> **Note**
>
> Jede VPC kann jeweils nur einem Host (GitHub Enterprise Server-Instanz) zugeordnet werden.

Die folgende Abbildung zeigt eine EC2 Instance, die mit dem GitHub Enterprise AMI gestartet wurde.



Wenn Sie eine VPC für eine GitHub Enterprise Server-Verbindung verwenden, müssen Sie bei der Einrichtung Ihres Hosts Folgendes für Ihre Infrastruktur angeben:

- VPC-ID: Die VPC für den Server, auf dem Ihre GitHub Enterprise Server-Instanz installiert ist, oder eine VPC, die über VPN oder Direct Connect Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz hat.
- Subnetz-ID oder IDs: Das Subnetz für den Server, auf dem Ihre GitHub Enterprise Server-Instanz installiert ist, oder ein Subnetz mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.
- Sicherheitsgruppe oder Gruppen: Die Sicherheitsgruppe für den Server, auf dem Ihre GitHub Enterprise Server-Instanz installiert ist, oder eine Sicherheitsgruppe mit Zugriff auf Ihre installierte GitHub Enterprise Server-Instanz über VPN oder Direct Connect.
- Endpoint (Endpunkt): Halten Sie Ihren Server-Endpunkt bereit und fahren Sie mit dem nächsten Schritt fort.

Weitere Informationen zur Arbeit mit VPCs Subnetzen finden Sie unter VPC and Subnet Sizing for IPv4 im Amazon VPC-Benutzerhandbuch.

Themen

- Ich kann keinen Host im Zustand „pending (ausstehend)“ abrufen
- Ich kann keinen Host im Zustand „available (verfügbar)“ abrufen
- Meine Verbindung/mein Host funktioniert plötzlich nicht mehr
- Ich kann meine Netzwerkschnittstellen nicht löschen

Ich kann keinen Host im Zustand „pending (ausstehend)“ abrufen

Wenn Ihr Host den Zustand VPC_CONFIG_FAILED_INITIALIZATION bekommt, liegt das wahrscheinlich an einem Problem mit der VPC, mit Subnetzen oder mit Sicherheitsgruppen, die Sie für Ihren Host ausgewählt haben.

- Die VPC, Subnetze und Sicherheitsgruppen müssen alle dem Konto gehören, das den Host erstellt.
- Die Subnetze und Sicherheitsgruppen müssen zur ausgewählten VPC gehören.
- Die bereitgestellten Subnetze müssen alle in verschiedenen Availability-Zonen (AZ) liegen.
- Der Benutzer, der den Host erstellt, muss über die folgenden IAM-Berechtigungen verfügen:

```
ec2:CreateNetworkInterface
```

```
ec2:CreateTags
ec2:DescribeDhcpOptionsec2:DescribeNetworkInterfaces
ec2:DescribeSubnets
ec2:DeleteNetworkInterface
ec2:DescribeVpcs
ec2:CreateVpcEndpoint
ec2:DeleteVpcEndpoints
ec2:DescribeVpcEndpoints
```

Ich kann keinen Host im Zustand „available (verfügbar)“ abrufen

Wenn Sie das CodeConnections App-Setup für Ihren Host nicht abschließen können, liegt das möglicherweise an einem Problem mit Ihren VPC-Konfigurationen oder Ihrer GitHub Enterprise Server-Instanz.

- Wenn Sie keine öffentliche Zertifizierungsstelle verwenden, müssen Sie Ihrem Host ein TLS-Zertifikat zur Verfügung stellen, das von Ihrer GitHub Enterprise Instance verwendet wird. Der TLS-Zertifikatwert sollte der öffentliche Schlüssel des Zertifikats sein.
- Sie müssen Administrator der GitHub Enterprise Server-Instanz sein, um GitHub Apps erstellen zu können.

Meine Verbindung/mein Host funktioniert plötzlich nicht mehr

Wenn eine Verbindung/ein Host zuvor funktioniert hat und jetzt nicht funktioniert, kann dies an einer Konfigurationsänderung in Ihrer VPC oder an der Änderung der GitHub App liegen. Überprüfen Sie, ob Folgendes der Fall ist:

- Die Sicherheitsgruppe, die an die Hostressource angehängt ist, die Sie für Ihre Verbindung erstellt haben, hat sich jetzt geändert oder hat keinen Zugriff mehr auf den GitHub Enterprise Server. CodeConnections erfordert eine Sicherheitsgruppe, die mit der GitHub Enterprise Server-Instanz verbunden ist.
- Die IP-Adresse des DNS-Servers wurde kürzlich geändert. Sie können das anhand der DHCP-Optionen für die VPC überprüfen, die in der Hostressource angegeben sind, die Sie für die Verbindung erstellt haben. Beachten Sie, dass der Host/die Verbindung nicht mehr funktionieren würde, wenn Sie kürzlich von AmazonProvided DNS zu einem benutzerdefinierten DNS-Server gewechselt sind oder einen neuen benutzerdefinierten DNS-Server verwenden. Um dies zu beheben, löschen Sie Ihren vorhandenen Host und erstellen Sie ihn neu, wodurch die neuesten DNS-Einstellungen in unserer Datenbank gespeichert werden.

- Die ACLs Netzwerkeinstellungen haben sich geändert und erlauben keine HTTP-Verbindungen mehr zu dem Subnetz, in dem sich Ihre GitHub Enterprise Server-Infrastruktur befindet.
- Alle Konfigurationen der CodeConnections App auf Ihrem GitHub Enterprise Server haben sich geändert. Änderungen an einer der Konfigurationen, z. B. URLs an geheimen Anwendungsschlüsseln, können die Konnektivität zwischen Ihrer installierten GitHub Enterprise Server-Instanz und unterbrechen CodeConnections.

Ich kann meine Netzwerkschnittstellen nicht löschen

Wenn Sie Ihre Netzwerkschnittstellen nicht löschen können, überprüfen Sie Folgendes:

- Die von erstellten Netzwerkschnittstellen CodeConnections können nur gelöscht werden, indem der Host gelöscht wird. Sie können vom Benutzer nicht manuell gelöscht werden.
- Sie benötigen die folgenden Berechtigungen:

```
ec2:DescribeNetworkInterfaces
ec2:DeleteNetworkInterface
```

Problembehandlung bei Webhook-VPC-Endpunkten (PrivateLink) für Enterprise Server-Verbindungen GitHub

Wenn Sie einen Host mit VPC-Konfiguration erstellen, entsteht dabei ein Webhook-VPC-Endpunkt.

> **Note**
>
> Verwenden Sie diesen Abschnitt zur Fehlerbehebung im Zusammenhang mit Ihrer Verbindung, die für die Verwendung des Webhook-Endpunkts für VPC () PrivateLink konfiguriert ist. Informationen zur Fehlerbehebung im Zusammenhang mit Ihrer GitHub Enterprise Server-Hostkonfiguration innerhalb einer Amazon VPC finden Sie unterFehlerbehebung bei der VPC-Konfiguration für Ihren Host.

Wenn Sie eine Verbindung zu einem installierten Anbietertyp herstellen und angegeben haben, dass Ihr Server in einer VPC konfiguriert ist, wird Ihr Host AWS CodeConnections erstellt und der VPC-Endpunkt (PrivateLink) für Webhooks wird für Sie erstellt. Auf diese Weise kann der Host Ereignisdaten über Webhooks über das Amazon-Netzwerk an Ihre integrierten AWS Dienste senden.

Weitere Informationen finden Sie unter AWS CodeConnections und Schnittstellen-VPC-Endpunkte ()AWS PrivateLink.

Themen

- Ich kann meine Webhook-VPC-Endpunkte nicht löschen

Ich kann meine Webhook-VPC-Endpunkte nicht löschen

AWS CodeConnections verwaltet den Lebenszyklus der Webhook-VPC-Endpunkte für Ihren Host. Um den Endpunkt zu löschen, müssen Sie die entsprechende Hostressource löschen.

- Die von erstellten Webhook-VPC-Endpunkte (PrivateLink) CodeConnections können nur gelöscht werden, indem der Host gelöscht wird. Sie können nicht manuell gelöscht werden.
- Sie benötigen die folgenden Berechtigungen:

```
ec2:DescribeNetworkInterfaces
ec2:DeleteNetworkInterface
```

Fehlerbehebung für einen Host, der vor dem 24. November 2020 erstellt wurde

Ab dem 24. November 2020, wenn AWS CodeConnections Sie Ihren Host einrichten, wird eine zusätzliche VPC-Endpunktunterstützung (PrivateLink) für Sie eingerichtet. Dieser Abschnitt behandelt die Problembehandlung bei Hosts, die davor erstellt wurden.

Weitere Informationen finden Sie unter AWS CodeConnections und Schnittstellen-VPC-Endpunkte ()AWS PrivateLink.

Themen

- Ich habe einen Host, der vor dem 24. November 2020 erstellt wurde, und ich möchte VPC-Endpoints (PrivateLink) für Webhooks verwenden
- Ich kann keinen Host im Zustand „available (verfügbar)“ abrufen (VPC-Fehler)

Ich habe einen Host, der vor dem 24. November 2020 erstellt wurde, und ich möchte VPC-Endpoints (PrivateLink) für Webhooks verwenden

Wenn Sie Ihren Host für GitHub Enterprise Server einrichten, wird der Webhook-Endpunkt für Sie erstellt. Verbindungen verwenden jetzt PrivateLink VPC-Webhook-Endpunkte. Wenn Sie Ihren Host

vor dem 24. November 2020 erstellt haben und PrivateLink VPC-Webhook-Endpunkte verwenden möchten, müssen Sie zuerst Ihren Host löschen und dann einen neuen Host erstellen.

Ich kann keinen Host im Zustand „available (verfügbar)“ abrufen (VPC-Fehler)

Wenn Ihr Host vor dem 24. November 2020 erstellt wurde und Sie das CodeConnections App-Setup für Ihren Host nicht abschließen können, liegt das möglicherweise an einem Problem mit Ihren VPC-Konfigurationen oder Ihrer GitHub Enterprise Server-Instanz.

Ihre VPC benötigt ein NAT-Gateway (oder einen ausgehenden Internetzugang), damit Ihre GitHub Enterprise Server-Instance ausgehenden Netzwerkverkehr für Webhooks senden kann. GitHub

Die Verbindung für ein Repository konnte nicht hergestellt werden GitHub

Problem:

Da eine Verbindung zu einem GitHub Repository den AWS Connector für verwendet GitHub, benötigen Sie zum Herstellen der Verbindung die Rechte des Organisationsinhabers oder Administratorberechtigungen für das Repository.

Mögliche Korrekturen: Informationen zu den Berechtigungsstufen für ein GitHub Repository finden Sie unter https://docs.github.com/en/free-pro-team@ latest/github/setting-up-and-managing-organizations-and-teams/permission - levels-for-an-organization.

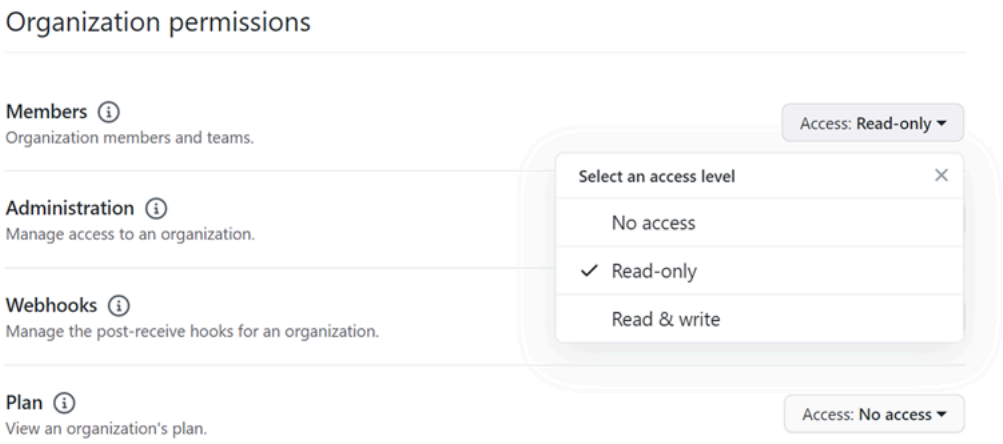
Bearbeiten Sie die Berechtigungen Ihrer GitHub Enterprise Server-Verbindungs-App

Wenn Sie die App für GitHub Enterprise Server am oder vor dem 23. Dezember 2020 installiert haben, müssen Sie der App möglicherweise nur Lesezugriff für Mitglieder der Organisation gewähren. Wenn Sie der Besitzer der GitHub App sind, gehen Sie wie folgt vor, um die Berechtigungen für die App zu bearbeiten, die bei der Erstellung Ihres Hosts installiert wurde.

> **Note**
>
> Sie müssen diese Schritte auf Ihrer GitHub Enterprise Server-Instanz ausführen und müssen der Besitzer der GitHub App sein.

1. Wählen Sie in GitHub Enterprise Server in der Drop-down-Option auf Ihrem Profilfoto die Option Einstellungen aus.
2. Wählen Sie Entwicklereinstellungen und dann GitHubApps aus.

3. Wählen Sie in der Liste der Apps den Namen der App für Ihre Verbindung und dann Permissions and events (Berechtigungen und Ereignisse) in der Einstellungsanzeige aus.
4. Wählen Sie unter Organization permissions (Organisationsberechtigungen) bei Members (Mitglieder) die Option Read-only (schreibgeschützt) im Dropdown-Menü Access (Zugriff) aus.



5. Geben Sie unter Add a note to users (Notiz für Benutzer eingeben) eine Begründung für die Änderung ein. Wählen Sie Änderungen speichern.

Verbindungsfehler bei der Verbindung zu GitHub: „Es ist ein Problem aufgetreten, stellen Sie sicher, dass Cookies in Ihrem Browser aktiviert sind“ oder „Ein Organisationsinhaber muss die GitHub App installieren“

Problem:

Um die Verbindung für ein GitHub Repository herzustellen, müssen Sie der Eigentümer der GitHub Organisation sein. Bei Repositorys, die keiner Organisation angehören, müssen Sie der Repository-Besitzer sein. Erstellt eine andere Person als der Organisationsbesitzer eine Verbindung, wird eine Anfrage für den Organisationsbesitzer erstellt, und einer der folgenden Fehler wird angezeigt:

Es ist ein Problem aufgetreten. Stellen Sie sicher, dass Cookies in Ihrem Browser aktiviert sind

ODER

Ein Organisationsinhaber muss die GitHub App installieren

Mögliche Lösungen: Für Repositorys in einer GitHub Organisation muss der Organisationsinhaber die Verbindung zum GitHub Repository herstellen. Bei Repositorys, die keiner Organisation angehören, müssen Sie der Repository-Besitzer sein.

Das Verbindungsdienstpräfix in Ressourcen muss möglicherweise für IAM-Richtlinien aktualisiert werden

Am 29. März 2024 wurde der Dienst von AWS CodeStar Connections in umbenannt. AWS CodeConnections Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt. Das Dienstpräfix für Ressourcen, die mit der Konsole erstellt wurden, lautet`codeconnections`. Neue SDK/CLI-Ressourcen werden mit `codeconnections` im Ressourcen-ARN erstellt. Erstellte Ressourcen erhalten automatisch das neue Dienstpräfix.

Im Folgenden sind die Ressourcen aufgeführt, die in erstellt werden AWS CodeConnections:

- Verbindungen
- Hosts

Problem:

Ressourcen, die mit `codestar-connections` im ARN erstellt wurden, werden nicht automatisch in das neue Dienstpräfix im Ressourcen-ARN umbenannt. Durch das Erstellen einer neuen Ressource wird eine Ressource mit dem Verbindungsdienstpräfix erstellt. IAM-Richtlinien mit dem `codestar-connections` Dienstpräfix funktionieren jedoch nicht für Ressourcen mit dem neuen Dienstpräfix.

Mögliche Lösungen: Gehen Sie wie folgt vor, um Zugriffs- oder Berechtigungsprobleme für die Ressourcen zu vermeiden:

- Aktualisieren Sie die IAM-Richtlinien für das neue Dienstpräfix. Andernfalls können umbenannte oder erstellte Ressourcen die IAM-Richtlinien nicht verwenden.
- Aktualisieren Sie die Ressourcen für das neue Dienstpräfix, indem Sie sie mit der Konsole oder CLI/CDK/CFN erstellen.

Aktualisieren Sie die Aktionen, Ressourcen und Bedingungen in der Richtlinie nach Bedarf. Im folgenden Beispiel wurde das `Resource` Feld für beide Dienstpräfixe aktualisiert.

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

```
            "codeconnections:UseConnection"
        ],
        "Resource": [
            "arn:aws:codestar-connections:*:*:connection/*",
            "arn:aws:codeconnections:*:*:connection/*"
        ]
    }
}
```

Berechtigungsfehler aufgrund eines Dienstpräfixes in Ressourcen, die mit der Konsole erstellt wurden

Derzeit haben Verbindungsressourcen, die mit der Konsole erstellt werden, nur das `codestar-connections` Dienstpräfix. Bei Ressourcen, die mit der Konsole erstellt wurden, müssen die Richtlinienerklärungen `codestar-connections` als Dienstpräfix angegeben werden.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Problem:

Beim Erstellen einer Verbindungsressource mithilfe der Konsole muss das `codestar-connections` Dienstpräfix in der Richtlinie verwendet werden. Wenn Sie eine Richtlinie mit dem `codeconnections` Dienstpräfix in der Richtlinie verwenden, erhalten Verbindungsressourcen, die mit der Konsole erstellt wurden, die folgende Fehlermeldung:

```
User: user_ARN is not authorized to perform: codestar-connections:action on
 resource: resource_ARN because no identity-based policy allows the codestar-
connections:action action
```

Mögliche Lösungen: Für Ressourcen, die mit der Konsole erstellt wurden, müssen die Aktionen mit Richtlinien-Anweisungen das Dienstpräfix enthalten`codestar-connections`, wie im Richtlinienbeispiel unter gezeigt[Beispiel: Eine Richtlinie für die Erstellung AWS CodeConnections mit der Konsole](#).

Verbindungs- und Host-Setup für installierte Anbieter, die Organisationen unterstützen

Bei installierten Anbietern, die Organizations unterstützen, wie z. B. GitHub Organisationen, übergeben Sie keinen verfügbaren Host. Sie erstellen für jede Verbindung in Ihrer Organisation einen neuen Host und achten Sie darauf, dieselben Informationen in die folgenden Netzwerkfelder einzugeben:

- VPC-ID
- Subnetz-ID
- Sicherheitsgruppe IDs

Informationen zum Erstellen einer GHES-Verbindung oder einer GitLab selbstverwalteten Verbindung finden Sie in den entsprechenden Schritten.

Ich möchte die Limits bei Verbindungen erhöhen

Sie können eine Erhöhung des Limits für bestimmte Limits in CodeConnections beantragen. Weitere Informationen finden Sie unter Kontingente für Verbindungen.

Kontingente für Verbindungen

In der folgenden Tabelle sind die Kontingente (auch als Limits bezeichnet) für Verbindungen in der Entwicklertools-Konsole ausgeführt.

Die Kontingente in dieser Tabelle gelten pro Person AWS-Region und können erhöht werden. Informationen und Kontingente zu AWS-Region , die geändert werden können, finden Sie unter AWS Service Quotas.

> **Note**
>
> Sie müssen Europa (Mailand) aktivieren, AWS-Region bevor Sie es verwenden können. Weitere Informationen finden Sie unter Aktivieren einer Region.

Ressource	Standardlimit
Maximale Anzahl von Verbindungen pro AWS-Konto	250

Diese Kontingente in dieser Tabelle stehen fest und können nicht geändert werden.

Ressource	Standardlimit
Maximale Zeichenanzahl in Verbindungsnamen	32 Zeichen
Maximale Anzahl von Hosts pro AWS-Konto	50
Maximale Anzahl von Repository-Links	100
Maximale Anzahl von AWS CloudFormation - Stack-Sync-Konfigurationen	100
Maximale Anzahl von Synchronisierungsk onfigurationen pro Repository-Link	100
Maximale Anzahl von Stack-Sync-Konfigu rationen pro Verzweigung	50

IP-Adressen, die Sie Ihrer Zulassungsliste hinzufügen möchten

Wenn Sie IP-Filterung implementieren oder bestimmte IP-Adressen auf EC2 Amazon-Instances zulassen, fügen Sie die folgenden IP-Adressen zu Ihrer Zulassungsliste hinzu. Dadurch werden Verbindungen zu Anbietern wie GitHub Bitbucket ermöglicht.

Die folgende Tabelle enthält die IP-Adressen für Verbindungen in der Developer Tools-Konsole nach AWS-Region.

> **Note**
>
> Für die Region Europa (Mailand) müssen Sie diese Region aktivieren, bevor Sie sie verwenden können. Weitere Informationen finden Sie unter Aktivieren einer Region.

Region	IP-Adressen
USA West (Oregon): (us-west-2)	35.160.210.199, 54.71.206.108, 54.71.36.205

Region	IP-Adressen
USA Ost (Nord-Virginia): (us-east-1)	3,216,216,90, 3,216,243,220, 3,217,241,85
Europa (Irland) (eu-west-1)	34,242,64,82, 52,18,37,201, 54,77,75,62
USA Ost (Ohio): (us-east-2)	18,217,188,190, 18,218,158,91, 18,220,4,80
Asien-Pazifik (Singapur): (ap-southeast-1)	18,138.171.151, 18,139,22,70, 3.1.157,1176
Asien-Pazifik (Sydney): (ap-southeast-2)	13,236,59,253, 52,64,164,166,86, 54,206,112
Asien-Pazifik (Tokyo) (ap-northeast-1)	52,196,132,231, 54,95,1333,227, 18,181,13,91
Europa (Frankfurt) (eu-central-1)	18,196,145,164, 3,121,252,59, 52,59,104,195
Asien-Pazifik (Seoul): (ap-northeast-2)	13,125,8,239, 13,209,223,177, 3,37.200,23
Asien-Pazifik (Mumbai): (ap-south-1)	13.234.199.152, 13.235.29.220, 35.154.23 0,124
Südamerika (São Paulo) (sa-east-1)	18,229,77,26, 54,233,226,52, 54,233,207,69
Kanada (Zentral): (ca-central-1)	15.222.219.210, 35.182.166.138, 99.79,111 1.198
Europa (London) (eu-west-2)	3,9.97,205, 35,177,150,185, 35,177,200.225
USA West (Nordkalifornien) (us-west-1)	52,52,16,175, 52,8,63,87
Europa (Paris) (eu-west-3)	35.181.127,138, 35.181.145,22, 35.181.20.200
Europa (Stockholm) (eu-north-1)	13,48,66,148, 13,48,8,79, 13,53,78,182
Europa (Mailand) (eu-south-1)	18,102,28,105, 18,102,35,130, 18,102.8,116
AWS GovCloud (US-Ost)	18,252.168.157, 18,252.207,77, 18,253.18 5,119

Sicherheit für die Funktionen der Entwicklertools-Konsole

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. Weitere Informationen zu den Compliance-Programmen, die für AWS CodeStar Benachrichtigungen gelten AWS CodeConnections, finden Sie unter AWS Services im Umfang der einzelnen Compliance-Programme.
- 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.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von AWS CodeStar Benachrichtigungen und anwenden können AWS CodeConnections. In den folgenden Themen erfahren Sie, wie Sie AWS CodeStar Benachrichtigungen konfigurieren und AWS CodeConnections Ihre Sicherheits- und Compliance-Ziele erreichen. Außerdem erfahren Sie, wie Sie andere AWS Dienste nutzen können, die Ihnen helfen, Ihre AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen zu überwachen und zu schützen.

Weitere Informationen zur Sicherheit für die Services in der Entwicklertools-Konsole finden Sie hier:

- CodeBuild Sicherheit
- CodeCommit Sicherheit
- CodeDeploy Sicherheit
- CodePipeline Sicherheit

Grundlagen zu Benachrichtigungsinhalten und -sicherheit

Benachrichtigungen werden verwendet, um Informationen über Ressourcen für Benutzer bereitzustellen, die die von Ihnen konfigurierten Benachrichtigungsregelziele abonniert haben. Dazu können Informationen über Ihre Entwicklertools-Ressourcen zählen, wozu auch Repository-Inhalte, Build-Status, Bereitstellungsstatus und Pipeline-Ausführungen zählen.

Sie können beispielsweise eine Benachrichtigungsregel für ein Repository so konfigurieren, dass sie Kommentare CodeCommit zu Commits oder Pull-Requests enthält. Wenn dies der Fall ist, enthalten die als Antwort auf diese Regel gesendeten Benachrichtigungen möglicherweise die Codezeile, auf die in diesem Kommentar verwiesen wird. In ähnlicher Weise können Sie eine Benachrichtigungsregel für ein Build-Projekt so konfigurieren, CodeBuild dass sie Erfolge oder Misserfolge für Build-Status und -Phasen berücksichtigt. Benachrichtigungen, die als Reaktion auf diese Regel gesendet werden, enthalten diese Informationen.

Sie können eine Benachrichtigungsregel für eine Pipeline so konfigurieren, dass CodePipeline sie Informationen über manuelle Genehmigungen enthält. Benachrichtigungen, die als Antwort auf diese Regel gesendet werden, können den Namen der Person enthalten, die diese Genehmigung erteilt hat. Sie können eine Benachrichtigungsregel für eine Anwendung konfigurieren, CodeDeploy um anzuzeigen, dass die Bereitstellung erfolgreich war, und Benachrichtigungen, die als Antwort auf diese Regel gesendet werden, können Informationen über das Bereitstellungsziel enthalten.

Benachrichtigungen enthalten projektspezifische Informationen wie Build-Status, Codezeilen mit Kommentaren, Bereitstellungsstatus und Pipeline-Genehmigungen. Um die Sicherheit Ihres Projekts zu gewährleisten, überprüfen Sie regelmäßig sowohl die Ziele der Benachrichtigungsregeln als auch die Abonnenten der Amazon–SNS-Themen, die als Ziel angegeben sind. Darüber hinaus kann sich der Inhalt der Benachrichtigungen, die als Reaktion auf Ereignisse gesendet werden, ändern, da den zugrunde liegenden Services zusätzliche Funktionen hinzugefügt werden. Diese Änderung kann ohne Benachrichtigung an bereits vorhandenen Benachrichtigungsregeln erfolgen. Überprüfen Sie den Inhalt von Benachrichtigungen regelmäßig, um sicherzustellen, dass Sie verstehen, was gesendet wird und an wen sie gesendet werden.

Weitere Informationen zu den Ereignistypen, die für Benachrichtigungsregeln verfügbar sind, finden Sie im Abschnitt Benachrichtigungskonzepte.

Sie können festlegen, dass die in Benachrichtigungen enthaltenen Details auf die in einem Ereignis enthaltenen Informationen beschränkt werden sollen. Dieser Detailtyp wird als Basic (Basis) bezeichnet. Diese Ereignisse enthalten genau dieselben Informationen, die an Amazon EventBridge und Amazon CloudWatch Events gesendet werden.

Konsolendienste der Developer Tools, wie z. B. CodeCommit, können sich dafür entscheiden, Informationen zu einigen oder allen ihrer Ereignistypen in Benachrichtigungsnachrichten hinzuzufügen, die über das hinausgehen, was in einem Ereignis verfügbar ist. Diese zusätzlichen Informationen können jederzeit hinzugefügt werden, um aktuelle Ereignistypen zu verbessern oder zukünftige Ereignistypen zu ergänzen. Sie können alle zusätzlichen Informationen zu dem Ereignis, sofern verfügbar, in die Benachrichtigung aufnehmen, indem Sie den Detailtyp Full (Vollständig) auswählen. Weitere Informationen finden Sie unter Detailtypen.

Datenschutz in AWS CodeStar Benachrichtigungen und AWS CodeConnections

Das AWS Modell der mit gilt für den Datenschutz in AWS CodeStar Benachrichtigungen und AWS CodeConnections. 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 AWS CodeStar Benachrichtigungen und/oder auf andere Weise über die Konsole, die AWS-Services API AWS CodeConnections oder arbeiten. 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.

Identitäts- und Zugriffsmanagement für AWS CodeStar Benachrichtigungen und AWS CodeConnections

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

> **Note**
>
> Aktionen für Ressourcen, die unter dem neuen Dienstpräfix erstellt wurden, `codeconnections` sind verfügbar. Wenn Sie eine Ressource unter dem neuen Dienstpräfix erstellen, wird sie `codeconnections` im Ressourcen-ARN verwendet. Aktionen und Ressourcen für das `codestar-connections` Dienstpräfix bleiben verfügbar. Wenn Sie eine Ressource in der IAM-Richtlinie angeben, muss das Dienstpräfix mit dem der Ressource übereinstimmen.

Themen

- Zielgruppe
- Authentifizierung mit Identitäten

- Verwalten des Zugriffs mit Richtlinien
- Funktionsweise von Funktionen in der Entwicklertools-Konsole mit IAM
- AWS CodeConnections Referenz zu Berechtigungen
- Beispiele für identitätsbasierte Richtlinien
- Verwenden von Tags zur Steuerung des Zugriffs auf AWS CodeConnections -Ressourcen
- Verwenden von Notifications und Connections in der Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer
- Fehlerbehebung bei AWS CodeStar Benachrichtigungen sowie AWS CodeConnections Identität und Zugriff
- Verwenden von serviceverknüpften Rollen für Benachrichtigungen AWS CodeStar
- Verwenden von serviceverknüpften Rollen für AWS CodeConnections
- AWS verwaltete Richtlinien für AWS CodeConnections

Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, unterscheidet sich je nach der Arbeit, die Sie unter AWS CodeStar Benachrichtigungen und ausführen. AWS CodeConnections

Dienstbenutzer — Wenn Sie die AWS CodeStar Benachrichtigungen und den AWS CodeConnections Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr AWS CodeStar Benachrichtigungen und AWS CodeConnections 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. Wenn Sie in AWS CodeStar Benachrichtigungen und nicht auf eine Funktion zugreifen können AWS CodeConnections, finden Sie weitere Informationen unterFehlerbehebung bei AWS CodeStar Benachrichtigungen sowie AWS CodeConnections Identität und Zugriff.

Dienstadministrator — Wenn Sie in Ihrem Unternehmen für AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen zuständig sind, haben Sie wahrscheinlich vollen Zugriff auf AWS CodeStar Benachrichtigungen und AWS CodeConnections. Es ist Ihre Aufgabe, zu bestimmen, auf welche AWS CodeStar Benachrichtigungen und AWS CodeConnections 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 mit AWS CodeStar Benachrichtigungen und verwenden kann AWS CodeConnections, finden Sie unterFunktionsweise von Funktionen in der Entwicklertools-Konsole mit IAM.

IAM-Administrator — Wenn Sie ein IAM-Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff auf Benachrichtigungen und zu verwalten. AWS CodeStar AWS CodeConnections Beispiele für AWS CodeStar Benachrichtigungen und AWS CodeConnections identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter. Beispiele für identitätsbasierte Richtlinien

Authentifizierung 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.

Root-Benutzer des AWS-Kontos

Wenn Sie eine erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle Ressourcen im AWS-Services 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 Ihnen 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 IAMAdminsund 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 Ihnen 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 verknüpft ist. AWS-Service 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 AWS CLI, der 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 AWS-Konto. 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.

Funktionsweise von Funktionen in der Entwicklertools-Konsole mit IAM

Bevor Sie IAM für die Verwaltung des Zugriffs auf Funktionen in der Entwicklertools-Konsole verwenden, sollten Sie wissen, welche IAM-Funktionen Sie damit verwenden werden können.

Einen allgemeinen Überblick darüber, wie Benachrichtigungen und andere AWS Dienste mit IAM funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren.

Themen

- Identitätsbasierte Richtlinien in der Entwicklertools-Konsole
- AWS CodeStar Benachrichtigungen und ressourcenbasierte Richtlinien AWS CodeConnections
- Autorisierung auf der Basis von Markierungen
- IAM-Rollen

Identitätsbasierte Richtlinien in der Entwicklertools-Konsole

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 CodeStar Benachrichtigungen und AWS CodeConnections Support für spezifische 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.

Aktionen

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff 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.

Richtlinienaktionen für Benachrichtigungen in der Entwicklertools-Konsole verwenden die folgenden Präfixe vor der Aktion: `codestar-notifications and codeconnections`. Um beispielsweise jemandem die Berechtigung zum Anzeigen aller Benachrichtigungsregeln in seinem Konto zu

erteilen, fügen Sie die Aktion `codestar-notifications:ListNotificationRules` in seine Richtlinie ein. Richtlinienerklärungen müssen `Action` entweder ein `NotAction` Oder-Element enthalten. AWS CodeStar Benachrichtigungen und AWS CodeConnections definiert eigene Aktionen, die Aufgaben beschreiben, die Sie mit diesem Dienst ausführen können.

Um mehrere AWS CodeStar Benachrichtigungsaktionen in einer einzigen Anweisung anzugeben, trennen Sie sie wie folgt durch Kommas.

```
"Action": [
      "codestar-notifications:action1",
      "codestar-notifications:action2"
```

Um mehrere AWS CodeConnections Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie wie folgt durch Kommas.

```
"Action": [
      "codeconnections:action1",
      "codeconnections: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": "codestar-notifications:List*"
```

AWS CodeStar Zu den API-Aktionen für Benachrichtigungen gehören:

- `CreateNotificationRule`
- `DeleteNotificationRule`
- `DeleteTarget`
- `DescribeNotificationRule`
- `ListEventTypes`
- `ListNotificationRules`
- `ListTagsForResource`
- `ListTargets`
- `Subscribe`

- `TagResource`
- `Unsubscribe`
- `UntagResource`
- `UpdateNotificationRule`

AWS CodeConnections Zu den API-Aktionen gehören die folgenden:

- `CreateConnection`
- `DeleteConnection`
- `GetConnection`
- `ListConnections`
- `ListTagsForResource`
- `TagResource`
- `UntagResource`

Die folgenden Aktionen, die nur für Berechtigungen bestimmt sind, sind erforderlich, AWS CodeConnections um den Auth-Handshake abzuschließen:

- `GetIndividualAccessToken`
- `GetInstallationUrl`
- `ListInstallationTargets`
- `StartOAuthHandshake`
- `UpdateConnectionInstallation`

Für die Verwendung einer Verbindung ist die folgende Aktion nur für Berechtigungen erforderlich: AWS CodeConnections

- `UseConnection`

Die folgende Aktion, bei der nur Berechtigungen erforderlich sind, ist erforderlich, um eine Verbindung AWS CodeConnections zu einem Dienst weiterzuleiten:

- `PassConnection`

Eine Liste der AWS CodeStar Benachrichtigungen und AWS CodeConnections Aktionen finden Sie unter Durch AWS CodeStar Benachrichtigungen definierte Aktionen und Definierte Aktionen von AWS CodeConnections im IAM-Benutzerhandbuch.

Ressourcen

AWS CodeStar Benachrichtigungen und unterstützen AWS CodeConnections nicht die Angabe von Ressourcen ARNs in einer Richtlinie.

Bedingungsschlüssel

AWS CodeStar Benachrichtigungen und AWS CodeConnections definieren ihre eigenen Sätze von Bedingungsschlüsseln und unterstützen auch die Verwendung einiger globaler Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontext-Schlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Alle AWS CodeStar Benachrichtigungsaktionen unterstützen den `codestar-notifications:NotificationsForResource` Bedingungsschlüssel. Weitere Informationen finden Sie unter Beispiele für identitätsbasierte Richtlinien.

AWS CodeConnections definieren Sie die folgenden Bedingungsschlüssel, die im `Condition` Element einer IAM-Richtlinie verwendet werden können. Diese Schlüssel können Sie verwenden, um die Bedingungen zu verfeinern, unter denen die Richtlinienanweisung angewendet wird. Weitere Informationen finden Sie unter AWS CodeConnections Referenz zu Berechtigungen.

Bedingungsschlüssel	Beschreibung
`codeconnections:BranchName`	Filtert den Zugriff nach dem Namen der Verzweigung des Drittanbieter-Repositorys
`codeconnections:FullRepositoryId`	Filtert den Zugriff durch das Repository, das in der Anforderung übergeben wird. Gilt nur für `UseConnection` -Anforderungen für den Zugriff auf ein bestimmtes Repository
`codeconnections:InstallationId`	Filtert den Zugriff durch die Drittanbieter-ID (z. B. die Bitbucket-App-Installations-ID), die zum Ändern einer Verbindung verwendet wird.

Bedingungsschlüssel	Beschreibung
	Schränkt ein, welche App-Installationen von Drittanbietern zum Herstellen einer Verbindung verwendet werden können
`codeconnections:OwnerId`	Filtert den Zugriff nach der ID vom Besitzer bzw. Konto des Drittanbieters
`codeconnections:PassedToService`	Filtert den Zugriff nach dem Service, an den der Prinzipal eine Verbindung übergeben darf
`codeconnections:ProviderAction`	Filtert den Zugriff nach der Anbieteraktion in einer `UseConnection` -Anforderung wie `ListRepositories` .
`codeconnections:ProviderPermissionsRequired`	Filtert den Zugriff nach dem Typ der Drittanbieter-Berechtigungen
`codeconnections:ProviderType`	Filtert den Zugriff nach dem Typ des Drittanbieters, der in der Anforderung übergeben wurde
`codeconnections:ProviderTypeFilter`	Filtert den Zugriff nach dem Typ des Drittanbieters, der zum Filtern der Ergebnisse verwendet wird
`codeconnections:RepositoryName`	Filtert den Zugriff nach dem Namen des Drittanbieter-Repositorys

Beispiele

Beispiele für AWS CodeStar Benachrichtigungen und AWS CodeConnections identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien

AWS CodeStar Benachrichtigungen und ressourcenbasierte Richtlinien AWS CodeConnections

AWS CodeStar Benachrichtigungen und unterstützen AWS CodeConnections keine ressourcenbasierten Richtlinien.

Autorisierung auf der Basis von Markierungen

Sie können Tags an AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen anhängen oder Tags in einer Anfrage weitergeben. Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `codestar-notifications and codeconnections:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden. Weitere Informationen zu Tagging-Strategien finden Sie unter Ressourcen taggen AWS. Weitere Informationen zum Markieren von AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen finden Sie unter. Ressourcen für Tag-Verbindungen

Ein Beispiel für eine identitätsbasierte Richtlinie zur Einschränkung des Zugriffs auf eine Ressource auf der Grundlage der Markierungen dieser Ressource finden Sie unter Verwenden von Tags zur Steuerung des Zugriffs auf AWS CodeConnections -Ressourcen.

IAM-Rollen

Eine IAM-Rolle ist eine Entität innerhalb Ihres AWS Kontos, die über bestimmte Berechtigungen verfügt.

Verwenden temporärer Anmeldeinformationen

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 AssumeRoleoder GetFederationTokenaufrufen.

AWS CodeStar Benachrichtigungen und AWS CodeConnections unterstützt die Verwendung temporärer Anmeldeinformationen.

Service-verknüpfte Rollen

Mit dienstbezogenen Rollen können AWS Dienste auf Ressourcen in anderen Diensten zugreifen, um eine Aktion in Ihrem Namen auszuführen. Serviceverknüpfte Rollen werden in Ihrem IAM-Konto angezeigt und gehören zum Service. Ein IAM-Administrator kann die Berechtigungen für serviceverknüpfte Rollen anzeigen, aber nicht bearbeiten.

AWS CodeStar Notifications unterstützt dienstbezogene Rollen. Einzelheiten zum Erstellen oder Verwalten von AWS CodeStar Benachrichtigungen und AWS CodeConnections dienstbezogenen Rollen finden Sie unter. Verwenden von serviceverknüpften Rollen für Benachrichtigungen AWS CodeStar

CodeConnections unterstützt keine dienstbezogenen Rollen.

AWS CodeConnections Referenz zu Berechtigungen

In den folgenden Tabellen sind die einzelnen AWS CodeConnections API-Operationen, die entsprechenden Aktionen, für die Sie Berechtigungen erteilen können, und das Format des Ressourcen-ARN aufgeführt, der für die Erteilung von Berechtigungen verwendet werden soll. Sie AWS CodeConnections APIs sind auf der Grundlage des Umfangs der Aktionen, die von dieser API zugelassen sind, in Tabellen gruppiert. Verwenden Sie sie als Referenz, wenn Sie Berechtigungsrichtlinien für eine IAM-Identität (identitätsbasierte Richtlinie) verfassen.

Beim Erstellen einer Berechtigungsrichtlinie geben Sie die Aktionen im Feld `Action` der Richtlinie an. Sie geben den Ressourcenwert im Feld `Resource` der Richtlinie als ARN mit oder ohne Platzhalterzeichen (*) an.

Bedingungen in Ihren Verbindungsrichtlinien können Sie mit den Bedingungsschlüsseln ausdrücken, die hier beschrieben und unter Bedingungsschlüssel aufgeführt sind. Sie können auch Bedingungsschlüssel für AWS alle Bereiche verwenden. Eine vollständige Liste der AWS-weiten Schlüssel finden Sie unter Verfügbare Schlüssel im IAM-Benutzerhandbuch.

Um eine Aktion anzugeben, verwenden Sie das Präfix `codeconnections` gefolgt vom Namen der API-Operation (z. B. `codeconnections:ListConnections` oder `codeconnections:CreateConnection`).

Verwenden von Platzhaltern

Sie können ein Platzhalterzeichen (*) in Ihrem ARN verwenden, um mehrere Aktionen oder Ressourcen anzugeben. `codeconnections:*`Gibt beispielsweise alle Aktionen an und gibt alle AWS CodeConnections Aktionen `codeconnections:Get*` an, die mit dem Wort beginnen. AWS CodeConnections `Get` Im folgenden Beispiel wird der Zugriff auf alle Ressourcen erteilt, deren Name mit `MyConnection` beginnt.

```
arn:aws:codeconnections:us-west-2:account-ID:connection/*
```

Sie können Platzhalter nur für die in der folgenden Tabelle aufgeführten *`connection`* Ressourcen verwenden. Sie können Platzhalter nicht zusammen mit *`region`* Ressourcen verwenden. *`account-id`* Weitere Informationen zu Platzhaltern finden Sie unter IAM Identifiers im Benutzerhandbuch von IAM.

Themen

- Berechtigungen zum Verwalten von Verbindungen
- Berechtigungen zum Verwalten von Hosts
- Berechtigungen zum Abschließen von Verbindungen
- Berechtigungen zum Einrichten von Hosts
- Übergeben einer Verbindung an einen Service
- Verwenden einer Verbindung
- Unterstützte Zugriffstypen für ProviderAction
- Unterstütze Berechtigungen für das Markieren von Verbindungsressourcen
- Übergeben einer Verbindung an einen Repository-Link
- Unterstützter Bedingungsschlüssel für Repository-Links

Berechtigungen zum Verwalten von Verbindungen

Eine Rolle oder ein Benutzer, der das SDK AWS CLI oder das SDK zum Anzeigen, Erstellen oder Löschen von Verbindungen verwendet, sollte über folgende Berechtigungen verfügen.

> **Note**
>
> Sie können in der Konsole keine Verbindung nur mit den folgenden Berechtigungen herstellen oder verwenden. Sie müssen die Berechtigungen in Berechtigungen zum Abschließen von Verbindungen hinzufügen.

```
codeconnections:CreateConnection
codeconnections:DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
```

AWS CodeStar Benachrichtigungen und erforderliche Berechtigungen für Aktionen zur Verwaltung von Verbindungen AWS CodeConnections

CreateConnection

Aktion(en): `codeconnections:CreateConnection`

Ist erforderlich, um mit der CLI bzw. der Konsole eine Verbindung zu erstellen.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

DeleteConnection

Aktion(en): `codeconnections:DeleteConnection`

Ist erforderlich, um mit der CLI bzw. der Konsole eine Verbindung zu löschen.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

GetConnection

Aktion(en): `codeconnections:GetConnection`

Ist erforderlich, um mit der CLI bzw. der Konsole Details zu einer Verbindung zu sehen.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListConnections

Aktion(en): `codeconnections:ListConnections`

Ist erforderlich, um mit der CLI bzw. der Konsole alle Verbindungen im Konto zu sehen.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

Diese Vorgänge unterstützen die folgenden Bedingungsschlüssel:

Aktion	Bedingungsschlüssel
`codeconnections:CreateConnection`	`codeconnections:ProviderType`
`codeconnections:DeleteConnection`	N/A
`codeconnections:GetConnection`	N/A
`codeconnections:ListConnections`	`codeconnections:ProviderTypeFilter`

Berechtigungen zum Verwalten von Hosts

Eine Rolle oder ein Benutzer, der AWS CLI bzw. der das SDK zum Anzeigen, Erstellen oder Löschen von Hosts verwenden soll, sollte über folgende Berechtigungen verfügen.

> **Note**
>
> Sie können im Host keine Verbindung herstellen oder verwenden, wenn nur die folgenden Berechtigungen vorhanden sind. Sie müssen die Berechtigungen in Berechtigungen zum Einrichten von Hosts hinzufügen.

```
codeconnections:CreateHost
codeconnections:DeleteHost
codeconnections:GetHost
codeconnections:ListHosts
```

AWS CodeStar Benachrichtigungen und erforderliche Berechtigungen für Aktionen zur Verwaltung von Hosts AWS CodeConnections

CreateHost

Aktion(en): `codeconnections:CreateHost`

Ist erforderlich, um mit der CLI bzw. der Konsole einen Host zu erstellen.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:host/`*`host-id`*

DeleteHost

Aktion(en): `codeconnections:DeleteHost`

Ist erforderlich, um mit der CLI bzw. der Konsole einen Host zu löschen.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:host/`*`host-id`*

GetHost

Aktion(en): `codeconnections:GetHost`

Ist erforderlich, um mit der CLI bzw. der Konsole Details zu einem Host zu sehen.

Ressource:`arn:aws:codeconnections:region:account-id:host/host-id`

ListHosts

Aktion(en): `codeconnections:ListHosts`

Ist erforderlich, um mit der CLI bzw. der Konsole alle Hosts im Konto zu sehen.

Ressource:`arn:aws:codeconnections:region:account-id:host/host-id`

Diese Vorgänge unterstützen die folgenden Bedingungsschlüssel:

Aktion	Bedingungsschlüssel
`codeconnections:CreateHost`	`codeconnections:ProviderType` `codeconnections:VpcId`
`codeconnections:DeleteHost`	N/A
`codeconnections:GetHost`	N/A
`codeconnections:ListHosts`	`codeconnections:ProviderTypeFilter`

Ein Beispiel für eine Richtlinie, die den VpcIdBedingungsschlüssel verwendet, finden Sie unterBeispiel: Beschränken Sie die VPC-Berechtigungen für Hosts mithilfe des VpcIdKontextschlüssels .

Berechtigungen zum Abschließen von Verbindungen

Eine Rolle oder ein Benutzer, die bzw. der zum Verwalten von Verbindungen in der Konsole bestimmt ist, sollte über die erforderlichen Berechtigungen verfügen, um eine Verbindung in der Konsole abzuschließen und eine Installation zu erstellen. Dazu gehören das Autorisieren des Handshakes beim Anbieter und das Erstellen von Installationen für Verbindungen. Verwenden Sie die folgenden Berechtigungen zusätzlich zu den oben genannten Berechtigungen.

Die folgenden IAM-Vorgänge werden von der Konsole verwendet, wenn Sie einen browserbasierten Handshake ausführen. `ListInstallationTargets`,

`GetInstallationUrl`, `StartOAuthHandshake`, `UpdateConnectionInstallation` und `GetIndividualAccessToken` sind IAM-Richtlinienberechtigungen. Es handelt sich dabei nicht um API-Aktionen.

```
codeconnections:GetIndividualAccessToken
codeconnections:GetInstallationUrl
codeconnections:ListInstallationTargets
codeconnections:StartOAuthHandshake
codeconnections:UpdateConnectionInstallation
```

Auf dieser Grundlage sind die folgenden Berechtigungen erforderlich, um eine Verbindung in der Konsole zu verwenden, zu erstellen, zu ändern oder zu löschen:

```
codeconnections:CreateConnection
codeconnections:DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
codeconnections:UseConnection
codeconnections:ListInstallationTargets
codeconnections:GetInstallationUrl
codeconnections:StartOAuthHandshake
codeconnections:UpdateConnectionInstallation
codeconnections:GetIndividualAccessToken
```

AWS CodeConnections erforderliche Berechtigungen für Aktionen zum Herstellen von Verbindungen

GetIndividualAccessToken

Aktion(en): `codeconnections:GetIndividualAccessToken`

Ist erforderlich, um mit der Konsole eine Verbindung abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

GetInstallationUrl

Aktion(en): `codeconnections:GetInstallationUrl`

Ist erforderlich, um mit der Konsole eine Verbindung abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListInstallationTargets

Aktion(en): `codeconnections:ListInstallationTargets`

Ist erforderlich, um mit der Konsole eine Verbindung abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

Starten Sie OAuth Handshake

Aktion(en): `codeconnections:StartOAuthHandshake`

Ist erforderlich, um mit der Konsole eine Verbindung abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

UpdateConnectionInstallation

Aktion(en): `codeconnections:UpdateConnectionInstallation`

Ist erforderlich, um mit der Konsole eine Verbindung abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

Diese Vorgänge unterstützen die folgenden Bedingungsschlüssel:

Aktion	Bedingungsschlüssel
`codeconnections:GetIndividualAccessToken`	`codeconnections:ProviderType`
`codeconnections:GetInstallationUrl`	`codeconnections:ProviderType`

Aktion	Bedingungsschlüssel
`codeconnections:ListInstallationTargets`	N/A
`codeconnections:StartOAuthHandshake`	`codeconnections:ProviderType`
`codeconnections:UpdateConnectionInstallation`	`codeconnections:InstallationId`

Berechtigungen zum Einrichten von Hosts

Eine Rolle oder ein Benutzer, die bzw. der zum Verwalten von Verbindungen in der Konsole bestimmt ist, sollte über die erforderlichen Berechtigungen verfügen, um einen Host in der Konsole zu erstellen. Dazu gehören das Autorisieren des Handshakes beim Anbieter und das Installieren der Host-App. Verwenden Sie die folgenden Berechtigungen zusätzlich zu den oben genannten Berechtigungen für Hosts.

Die folgenden IAM-Vorgänge werden von der Konsole verwendet, wenn Sie eine browserbasierte Hostregistrierung durchführen. `RegisterAppCode` und `StartAppRegistrationHandshake` sind IAM-Richtlinienberechtigungen. Es handelt sich dabei nicht um API-Aktionen.

```
codeconnections:RegisterAppCode
codeconnections:StartAppRegistrationHandshake
```

Dementsprechend sind die folgenden Berechtigungen erforderlich, damit Sie eine Verbindung in der Konsole verwenden, erstellen, ändern oder löschen können, für die ein Host erforderlich ist (z. B. installierte Anbietertypen).

```
codeconnections:CreateConnection
codeconnections:DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
codeconnections:UseConnection
codeconnections:ListInstallationTargets
codeconnections:GetInstallationUrl
codeconnections:StartOAuthHandshake
```

```
codeconnections:UpdateConnectionInstallation
codeconnections:GetIndividualAccessToken
codeconnections:RegisterAppCode
codeconnections:StartAppRegistrationHandshake
```

AWS CodeConnections erforderliche Berechtigungen für Aktionen zum Abschließen des Host-Setups

RegisterAppCode

Aktion(en): `codeconnections:RegisterAppCode`

Erforderlich, um die Hosteinrichtung mit der Konsole abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:host/`*`host-id`*

StartAppRegistrationHandshake

Aktion(en): `codeconnections:StartAppRegistrationHandshake`

Erforderlich, um die Hosteinrichtung mit der Konsole abzuschließen. Dies ist nur eine IAM-Richtlinienberechtigung, keine API-Aktion.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:host/`*`host-id`*

Diese Vorgänge unterstützen die folgenden Bedingungsschlüssel:

Übergeben einer Verbindung an einen Service

Wenn eine Verbindung an einen Service übergeben wird (z. B. wenn ein Verbindungs-ARN in einer Pipeline-Definition bereitgestellt wird, um eine Pipeline zu erstellen oder zu ändern), muss der Benutzer über die `codeconnections:PassConnection`-Berechtigung verfügen.

AWS CodeConnections erforderliche Berechtigungen für das Weiterleiten einer Verbindung

PassConnection

Aktion(en): `codeconnections:PassConnection`

Ist erforderlich, um eine Verbindung an einen Service zu übergeben.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

Dieser Vorgang unterstützt auch den folgenden Bedingungsschlüssel:

- `codeconnections:PassedToService`

Unterstützte Werte für Bedingungsschlüssel

Schlüssel	Gültige Aktionsanbieter
`codeconnections:PassedToService`	• `codeguru-reviewer` • `codepipeline.amazonaws.com` • `proton.amazonaws.com`

Verwenden einer Verbindung

Wenn ein Dienst wie eine Verbindung CodePipeline verwendet, muss die Dienstrolle über die `codeconnections:UseConnection` Berechtigung für eine bestimmte Verbindung verfügen.

Um Verbindungen in der Konsole zu verwalten, muss die Benutzerrichtlinie die `codeconnections:UseConnection`-Berechtigung haben.

AWS CodeConnections erforderliche Aktion für die Verwendung einer Verbindung

UseConnection

Aktion(en): `codeconnections:UseConnection`

Ist erforderlich zum Verwenden einer Verbindung.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

Dieser Vorgang unterstützt auch die folgenden Bedingungsschlüssel:

- `codeconnections:BranchName`
- `codeconnections:FullRepositoryId`
- `codeconnections:OwnerId`
- `codeconnections:ProviderAction`
- `codeconnections:ProviderPermissionsRequired`

- `codeconnections:RepositoryName`

Unterstützte Werte für Bedingungsschlüssel

Schlüssel	Gültige Aktionsanbieter
`codeconnections:FullRepositoryId`	Der Benutzername und der Repository-Name eines Repositorys, wie etwa `my-owner/my-repository` . Wird nur unterstützt, wenn die Verbindung für den Zugriff auf ein bestimmtes Repository verwendet wird.
`codeconnections:ProviderPermissionsRequired`	read_only oder read_write
`codeconnections:ProviderAction`	`GetBranch` , `ListRepositories` , `ListOwners` , `ListBranches` , `StartUploadArchiveToS3` , `GitPush`, `GitPull`, `GetUploadArchiveToS3Status` , `CreatePullRequestDiffComment` , `GetPullRequest` , `ListBranchCommits` , `ListCommitFiles` , `ListPullRequestComments` , `ListPullRequestCommits` . Weitere Informationen finden Sie im folgenden Abschnitt.

Die erforderlichen Bedingungsschlüssel für einige Funktionen können sich im Laufe der Zeit ändern. Es wird empfohlen, den Zugriff auf eine Verbindung mit `codeconnections:UseConnection` zu kontrollieren, es sei denn, Ihre Zugriffskontrollanforderungen erfordern andere Berechtigungen.

Unterstützte Zugriffstypen für **`ProviderAction`**

Wenn eine Verbindung von einem AWS Dienst verwendet wird, führt dies dazu, dass API-Aufrufe an Ihren Quellcode-Anbieter getätigt werden. Beispielsweise kann ein Service Repositorys für eine Bitbucket-Verbindung auflisten, indem er die `https://api.bitbucket.org/2.0/repositories/`*`username`*-API aufruft.

Mit dem `ProviderAction` Bedingungsschlüssel können Sie einschränken, welcher APIs Anbieter aufgerufen werden kann. Da der API-Pfad möglicherweise dynamisch generiert wird und der Pfad von Anbieter zu Anbieter variiert, wird der `ProviderAction`-Wert einem abstrakten Aktionsnamen und nicht der URL der API zugeordnet. Auf diese Weise können Sie Richtlinien schreiben, die unabhängig vom Anbietertyp für die Verbindung dieselbe Wirkung haben.

Im Folgenden sind die Zugriffstypen aufgeführt, die für jeden der unterstützten `ProviderAction`-Werte gewährt werden. Das folgende Beispiel zeigt IAM-Richtlinienberechtigungen. Es handelt sich dabei nicht um API-Aktionen.

AWS CodeConnections unterstützte Zugriffstypen für **`ProviderAction`**

GetBranch

Aktion(en): `codeconnections:GetBranch`

Ist erforderlich zum Zugreifen auf Informationen über eine Verzweigung, z. B. das letzte Commit für diese Verzweigung.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

ListRepositories

Aktion(en): `codeconnections:ListRepositories`

Ist erforderlich zum Abrufen einer Liste von öffentlichen und privaten Repositorys, einschließlich Details zu den Repositorys, die einem Besitzer gehören.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

ListOwners

Aktion(en): `codeconnections:ListOwners`

Ist erforderlich zum Aufrufen einer Liste von Besitzern, auf die die Verbindung Zugriff hat.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:connection/`*`connection-id`*

ListBranches

Aktion(en): `codeconnections:ListBranches`

Ist erforderlich zum Abrufen einer Liste der Verzweigungen, die in einem bestimmten Repository vorhanden sind.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

StartUploadArchiveToS3

Aktion(en): `codeconnections:StartUploadArchiveToS3`

Ist erforderlich, um den Quellcode zu lesen und auf Amazon S3 hochzuladen.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

GitPush

Aktion(en): `codeconnections:GitPush`

Ist erforderlich zum Schreiben eines Repositorys mit Git.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

GitPull

Aktion(en): `codeconnections:GitPull`

Ist erforderlich zum Lesen eines Git aus einem Repository.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

GetUploadArchiveToS3-Status

Aktion(en): `codeconnections:GetUploadArchiveToS3Status`

Ist erforderlich zum Aufrufen des Status eines Uploads, einschließlich aller Fehlermeldungen, die von `StartUploadArchiveToS3` gestartet werden.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

CreatePullRequestDiffComment

Aktion(en): `codeconnections:CreatePullRequestDiffComment`

Ist erforderlich für den Zugriff auf Kommentare zu einer Pull-Anforderung.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

GetPullRequest

Aktion(en): `codeconnections:GetPullRequest`

Ist erforderlich zum Anzeigen von Pull-Anforderungen für ein Repository.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListBranchCommits

Aktion(en): `codeconnections:ListBranchCommits`

Ist erforderlich zum Anzeigen einer Liste von Commits für eine Repository-Verzweigung.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListCommitFiles

Aktion(en): `codeconnections:ListCommitFiles`

Ist erforderlich zum Anzeigen einer Liste von Dateien für ein Commit.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListPullRequestComments

Aktion(en): `codeconnections:ListPullRequestComments`

Ist erforderlich zum Anzeigen einer Liste mit Kommentaren für eine Pull-Anforderung.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

ListPullRequestCommits

Aktion(en): `codeconnections:ListPullRequestCommits`

Ist erforderlich zum Anzeigen einer Liste von Commits für eine Pull-Anforderung.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`

Unterstütze Berechtigungen für das Markieren von Verbindungsressourcen

Die folgenden IAM-Vorgänge werden beim Markieren von Verbindungsressourcen verwendet.

```
codeconnections:ListTagsForResource
codeconnections:TagResource
codeconnections:UntagResource
```

AWS CodeConnections erforderliche Aktionen zum Markieren von Verbindungsressourcen

ListTagsForResource

Aktion(en): `codeconnections:ListTagsForResource`

Ist erforderlich zum Anzeigen einer Liste von Markierungen, die mit der Verbindungsressource verknüpft sind.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`, `arn:aws:codeconnections:region:account-id:host/host-id`

TagResource

Aktion(en): `codeconnections:TagResource`

Ist erforderlich zum Markieren einer Verbindungsressource.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`, `arn:aws:codeconnections:region:account-id:host/host-id`

UntagResource

Aktion(en): `codeconnections:UntagResource`

Ist erforderlich zum Entfernen von Markierungen bei einer Ressource.

Ressource:`arn:aws:codeconnections:region:account-id:connection/connection-id`, `arn:aws:codeconnections:region:account-id:host/host-id`

Übergeben einer Verbindung an einen Repository-Link

Wenn ein Repository-Link in einer Synchronisierungskonfiguration bereitgestellt wird, muss der Benutzer über die `codeconnections:PassRepository`-Berechtigung für den Repository-Link ARN/die Resource verfügen.

AWS CodeConnections erforderliche Berechtigungen für das Weiterleiten einer Verbindung

PassRepository

Aktion(en): `codeconnections:PassRepository`

Dies ist erforderlich, um einen Repository-Link an eine Synchronisierungskonfiguration zu übergeben.

Ressource:`arn:aws:codeconnections:`*`region`*`:`*`account-id`*`:repository-link/`*`repository-link-id`*

Dieser Vorgang unterstützt auch den folgenden Bedingungsschlüssel:

- `codeconnections:PassedToService`

Unterstützte Werte für Bedingungsschlüssel

Schlüssel	Gültige Aktionsanbieter
`codeconnections:PassedToService`	• `cloudformation.sync.codeconnections.amazonaws.com`

Unterstützter Bedingungsschlüssel für Repository-Links

Operationen für Repository-Links und Sync-Konfigurationsressourcen werden durch den folgenden Bedingungsschlüssel unterstützt:

- `codeconnections:Branch`

  Filtert den Zugriff nach dem Zweignamen, der in der Anforderung übergeben wird.

Unterstützte Aktionen für den Bedingungsschlüssel

Schlüssel	Zulässige Werte
`codeconnections:Branch`	Die folgenden Aktionen werden für diesen Bedingungsschlüssel unterstützt: • CreateSyncConfiguration • UpdateSyncConfiguration • GetRepositorySyncStatus

Beispiele für identitätsbasierte Richtlinien

Standardmäßig verfügen IAM-Benutzer und -Rollen, auf die eine der verwalteten Richtlinien für AWS CodeCommit, AWS CodeBuild, oder AWS CodePipeline angewendet wurde AWS CodeDeploy, über Berechtigungen für Verbindungen, Benachrichtigungen und Benachrichtigungsregeln, die der Absicht dieser Richtlinien entsprechen. Beispielsweise haben IAM-Benutzer oder -Rollen, auf die eine der Vollzugriffsrichtlinien (AWSCodeCommitFullAccess, AWSCodeBuildAdminAccessAWSCodeDeployFullAccess, oder AWSCodePipeline_FullAccess) angewendet wurde, auch vollen Zugriff auf Benachrichtigungen und Benachrichtigungsregeln, die für die Ressourcen dieser Dienste erstellt wurden.

Andere IAM-Benutzer und -Rollen sind nicht berechtigt, AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen zu erstellen oder zu ändern. 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.

Berechtigungen und Beispiele für AWS CodeStar Benachrichtigungen

Die folgenden Richtlinienerklärungen und Beispiele können Ihnen bei der Verwaltung von AWS CodeStar Benachrichtigungen helfen.

Berechtigungen in Zusammenhang mit Benachrichtigungen in verwalteten Vollzugriffsrichtlinien

Die AWSCodePipeline_FullAccessverwalteten Richtlinien AWSCodeCommitFullAccessAWSCodeBuildAdminAccessAWSCodeDeployFullAccess,, und

enthalten die folgenden Anweisungen, um vollen Zugriff auf Benachrichtigungen in der Developer Tools-Konsole zu ermöglichen. Benutzer, für die eine dieser verwalteten Richtlinien angewendet ist, können auch Amazon-SNS-Themen für Benachrichtigungen erstellen und verwalten, Benutzer für Themen abonnieren und Abonnements beenden sowie Themen auflisten, die als Ziele für Benachrichtigungsregeln ausgewählt werden sollen.

> **Note**
>
> In der verwalteten Richtlinie hat der Bedingungsschlüssel `codestar-notifications:NotificationsForResource` einen für den Ressourcentyp für den Service spezifischen Wert. In der Vollzugriffsrichtlinie für CodeCommit lautet der Wert beispielsweise`arn:aws:codecommit:*`.

```
    {
        "Sid": "CodeStarNotificationsReadWriteAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:CreateNotificationRule",
            "codestar-notifications:DescribeNotificationRule",
            "codestar-notifications:UpdateNotificationRule",
            "codestar-notifications:DeleteNotificationRule",
            "codestar-notifications:Subscribe",
            "codestar-notifications:Unsubscribe"
        ],
        "Resource": "*",
        "Condition" : {
            "StringLike" : {"codestar-notifications:NotificationsForResource" :
 "arn:aws:<vendor-code>:*"}
        }
    },
    {
        "Sid": "CodeStarNotificationsListAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource",
            "codestar-notifications:ListEventTypes"
        ],
        "Resource": "*"
```

```
    },
    {
        "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
        "Effect": "Allow",
        "Action": [
            "sns:CreateTopic",
            "sns:SetTopicAttributes"
        ],
        "Resource": "arn:aws:sns:*:*:codestar-notifications*"
    },
    {
        "Sid": "SNSTopicListAccess",
        "Effect": "Allow",
        "Action": [
            "sns:ListTopics"
        ],
        "Resource": "*"
    },
    {
        "Sid": "CodeStarNotificationsChatbotAccess",
        "Effect": "Allow",
        "Action": [
            "chatbot:DescribeSlackChannelConfigurations",
            "chatbot:ListMicrosoftTeamsChannelConfigurations"
          ],
       "Resource": "*"
    }
```

Berechtigungen in Zusammenhang mit Benachrichtigungen in schreibgeschützten verwalteten Richtlinien

Die AWSCodePipeline_ReadOnlyAccessverwalteten Richtlinien AWSCodeCommitReadOnlyAccessAWSCodeBuildReadOnlyAccessAWSCodeDeployReadOnlyAccess,, und enthalten die folgenden Anweisungen, um den schreibgeschützten Zugriff auf Benachrichtigungen zu ermöglichen. So können beispielsweise Benachrichtigungen für Ressourcen in der Entwicklertools-Konsole angezeigt, nicht jedoch erstellt, verwaltet oder abonniert werden.

> **Note**
>
> In der verwalteten Richtlinie hat der Bedingungsschlüssel `codestar-notifications:NotificationsForResource` einen für den Ressourcentyp für den

Service spezifischen Wert. In der Vollzugriffsrichtlinie für CodeCommit lautet der Wert beispielsweise. `arn:aws:codecommit:*`

```
  {
       "Sid": "CodeStarNotificationsPowerUserAccess",
       "Effect": "Allow",
       "Action": [
           "codestar-notifications:DescribeNotificationRule"
       ],
       "Resource": "*",
       "Condition" : {
           "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:<vendor-code>:*"}
       }
   },
   {
       "Sid": "CodeStarNotificationsListAccess",
       "Effect": "Allow",
       "Action": [
           "codestar-notifications:ListNotificationRules",
           "codestar-notifications:ListEventTypes",
           "codestar-notifications:ListTargets"
       ],
       "Resource": "*"
   }
```

Berechtigungen in Zusammenhang mit Benachrichtigungen in anderen verwalteten Richtlinien

Die AWSCodeBuildDeveloperAccessverwalteten Richtlinien AWSCodeCommitPowerUserAWSCodeBuildDeveloperAccess, und enthalten die folgenden Anweisungen, damit Entwickler, auf die eine dieser verwalteten Richtlinien angewendet wurde, Benachrichtigungen erstellen, bearbeiten und abonnieren können. Sie können Benachrichtigungsregeln nicht löschen und auch keine Tags für Ressourcen verwalten.

> **Note**
>
> In der verwalteten Richtlinie hat der Bedingungsschlüssel `codestar-notifications:NotificationsForResource` einen für den Ressourcentyp für den

Service spezifischen Wert. In der Vollzugriffsrichtlinie für CodeCommit lautet der Wert beispielsweise`arn:aws:codecommit:*`.

```
    {
        "Sid": "CodeStarNotificationsReadWriteAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:CreateNotificationRule",
            "codestar-notifications:DescribeNotificationRule",
            "codestar-notifications:UpdateNotificationRule",
            "codestar-notifications:Subscribe",
            "codestar-notifications:Unsubscribe"
        ],
        "Resource": "*",
        "Condition" : {
            "StringLike" : {"codestar-notifications:NotificationsForResource" :
 "arn:aws:<vendor-code>:*"}
        }
    },
    {
        "Sid": "CodeStarNotificationsListAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource",
            "codestar-notifications:ListEventTypes"
        ],
        "Resource": "*"
    },
    {
        "Sid": "SNSTopicListAccess",
        "Effect": "Allow",
        "Action": [
            "sns:ListTopics"
        ],
        "Resource": "*"
    },
    {
        "Sid": "CodeStarNotificationsChatbotAccess",
        "Effect": "Allow",
        "Action": [
```

```
            "chatbot:DescribeSlackChannelConfigurations",
            "chatbot:ListMicrosoftTeamsChannelConfigurations"
          ],
        "Resource": "*"
    }
```

Beispiel: Eine Richtlinie auf Administratorebene für die Verwaltung von Benachrichtigungen AWS CodeStar

In diesem Beispiel möchten Sie einem IAM-Benutzer in Ihrem AWS Konto vollen Zugriff auf AWS CodeStar Benachrichtigungen gewähren, sodass der Benutzer die Details der Benachrichtigungsregeln überprüfen und Benachrichtigungsregeln, Ziele und Ereignistypen auflisten kann. Sie möchten dem Benutzer außerdem Berechtigungen zum Hinzufügen, Ändern und Löschen von Benachrichtigungsregeln gewähren. Dabei handelt es sich um eine Vollzugriffsrichtlinie, die den Benachrichtigungsberechtigungen entspricht, die in den Richtlinien AWSCodeBuildAdminAccess, AWSCodeCommitFullAccessAWSCodeDeployFullAccess, und den AWSCodePipeline_FullAccessverwalteten Richtlinien enthalten sind. Wie bei diesen verwalteten Richtlinien sollten Sie diese Art von Richtlinienerklärung nur an IAM-Benutzer, -Gruppen oder -Rollen anhängen, die vollen Administratorzugriff auf Benachrichtigungen und Benachrichtigungsregeln in Ihrem gesamten AWS Konto benötigen.

> **Note**
>
> Diese Richtlinie beinhaltet `CreateNotificationRule`. Jeder Benutzer, für den diese Richtlinie auf seinen IAM-Benutzer oder seine IAM-Rolle angewendet wird, kann Benachrichtigungsregeln für alle Ressourcentypen erstellen, die von AWS CodeStar Benachrichtigungen im AWS Konto unterstützt werden, auch wenn dieser Benutzer selbst keinen Zugriff auf diese Ressourcen hat. Ein Benutzer mit dieser Richtlinie könnte beispielsweise eine Benachrichtigungsregel für ein CodeCommit Repository erstellen, ohne über die erforderlichen Zugriffsrechte zu verfügen CodeCommit.

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

```
            "codestar-notifications:CreateNotificationRule",
            "codestar-notifications:DeleteNotificationRule",
            "codestar-notifications:DescribeNotificationRule",
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:UpdateNotificationRule",
            "codestar-notifications:Subscribe",
            "codestar-notifications:Unsubscribe",
            "codestar-notifications:DeleteTarget",
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource",
            "codestar-notifications:TagResource",
            "codestar-notifications:UntagResource"
        ],
        "Resource": "*"
    }
  ]
}
```

Beispiel: Eine Richtlinie auf Mitwirkendenebene für die Verwendung von Benachrichtigungen AWS CodeStar

In diesem Beispiel möchten Sie Zugriff auf die day-to-day Verwendung von AWS CodeStar Benachrichtigungen gewähren, z. B. auf das Erstellen und Abonnieren von Benachrichtigungen, aber nicht auf destruktivere Aktionen wie das Löschen von Benachrichtigungsregeln oder -zielen. Dies entspricht dem Zugriff, der in den AWSCodeCommitPowerUserverwalteten Richtlinien AWSCodeBuildDeveloperAccessAWSCodeDeployDeveloperAccess, und bereitgestellt wird.

> **Note**
>
> Diese Richtlinie beinhaltet `CreateNotificationRule`. Jeder Benutzer, für den diese Richtlinie auf seinen IAM-Benutzer oder seine IAM-Rolle angewendet wurde, kann Benachrichtigungsregeln für alle Ressourcentypen erstellen, die von AWS CodeStar Benachrichtigungen im AWS Konto unterstützt werden, auch wenn dieser Benutzer selbst keinen Zugriff auf diese Ressourcen hat. Ein Benutzer mit dieser Richtlinie könnte beispielsweise eine Benachrichtigungsregel für ein CodeCommit Repository erstellen, ohne über die erforderlichen Zugriffsrechte zu verfügen CodeCommit.

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

```
    "Sid": "AWSCodeStarNotificationsPowerUserAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:CreateNotificationRule",
            "codestar-notifications:DescribeNotificationRule",
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:UpdateNotificationRule",
            "codestar-notifications:Subscribe",
            "codestar-notifications:Unsubscribe",
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource"
        ],
        "Resource": "*"
        }
    ]
}
```

Beispiel: Eine read-only-level Richtlinie für die Verwendung von AWS CodeStar Benachrichtigungen

In diesem Beispiel möchten Sie einem IAM-Benutzer in Ihrem Konto schreibgeschützten Zugriff auf die Benachrichtigungsregeln, Ziele und Ereignistypen in Ihrem AWS -Konto gewähren. Dieses Beispiel zeigt, wie Sie eine Richtlinie erstellen könnten, die die Anzeige dieser Elemente gestattet. Dies entspricht den Berechtigungen, die in den AWSCodePipeline_ReadOnlyAccessverwalteten Richtlinien AWSCodeBuildReadOnlyAccessAWSCodeCommitReadOnly, und enthalten sind.

```
{
    "Version": "2012-10-17",
    "Id": "CodeNotification__ReadOnly",
    "Statement": [
        {
            "Sid": "Reads_API_Access",
            "Effect": "Allow",
            "Action": [
                "CodeNotification:DescribeNotificationRule",
                "CodeNotification:ListNotificationRules",
                "CodeNotification:ListTargets",
                "CodeNotification:ListEventTypes"
            ],
            "Resource": "*"
        }
    ]
}
```

Berechtigungen und Beispiele für AWS CodeConnections

Die folgenden Richtlinienanweisungen und -beispiele helfen Ihnen bei der Verwaltung von AWS CodeConnections.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von Richtlinien auf der JSON-Registerkarte im Benutzerhandbuch von IAM.

Beispiel: Eine Richtlinie zum Erstellen AWS CodeConnections mit der CLI und zum Anzeigen mit der Konsole

Eine Rolle oder ein Benutzer, der das AWS CLI oder das SDK zum Anzeigen, Erstellen, Markieren oder Löschen von Verbindungen verwenden soll, sollte über Berechtigungen verfügen, die auf Folgendes beschränkt sind.

> **Note**
>
> Sie können in der Konsole nur mit den folgenden Berechtigungen keine Verbindung herstellen. Sie müssen die Berechtigungen im nächsten Abschnitt hinzufügen.

Verwenden Sie die folgende Richtlinie, um mithilfe der Konsole eine Liste verfügbarer Verbindungen anzuzeigen, Tags anzuzeigen und eine Verbindung zu verwenden.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "ConnectionsFullAccess",
        "Effect": "Allow",
        "Action": [
            "codeconnections:CreateConnection",
            "codeconnections:DeleteConnection",
            "codeconnections:UseConnection",
            "codeconnections:GetConnection",
            "codeconnections:ListConnections",
            "codeconnections:TagResource",
            "codeconnections:ListTagsForResource",
            "codeconnections:UntagResource"
        ],
```

```
        "Resource": "*"
      }
    ]
}
```

Beispiel: Eine Richtlinie für die Erstellung AWS CodeConnections mit der Konsole

Eine Rolle oder ein Benutzer, die bzw. der zum Verwalten von Verbindungen in der Konsole bestimmt ist, sollte über die erforderlichen Berechtigungen verfügen, um eine Verbindung in der Konsole abzuschließen und eine Installation zu erstellen. Dazu gehören das Autorisieren des Handshakes beim Anbieter und das Erstellen von Installationen für Verbindungen. `UseConnection` sollte auch hinzugefügt werden, um die Verbindung in der Konsole zu verwenden. Verwenden Sie die folgende Richtlinie, um eine Verbindung in der Konsole anzuzeigen, zu verwenden, zu markieren, zu erstellen oder zu löschen.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

> **Note**
>
> Bei Ressourcen, die mit der Konsole erstellt wurden, müssen die Aktionen in `codestar-connections` Richtlinienanweisungen das Dienstpräfix enthalten, wie im folgenden Beispiel gezeigt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "codestar-connections:CreateConnection",
                "codestar-connections:DeleteConnection",
                "codestar-connections:GetConnection",
                "codestar-connections:ListConnections",
                "codestar-connections:GetInstallationUrl",
```

```
                "codestar-connections:GetIndividualAccessToken",
                "codestar-connections:ListInstallationTargets",
                "codestar-connections:StartOAuthHandshake",
                "codestar-connections:UpdateConnectionInstallation",
                "codestar-connections:UseConnection",
                "codestar-connections:TagResource",
                "codestar-connections:ListTagsForResource",
                "codestar-connections:UntagResource"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Beispiel: Eine Richtlinie auf Administratorebene für die Verwaltung AWS CodeConnections

In diesem Beispiel möchten Sie einem IAM-Benutzer in Ihrem AWS Konto vollen Zugriff gewähren, CodeConnections sodass der Benutzer Verbindungen hinzufügen, aktualisieren und löschen kann. Dies ist eine Vollzugriffsrichtlinie, die der AWSCodePipeline_FullAccessverwalteten Richtlinie entspricht. Wie bei dieser verwalteten Richtlinie sollten Sie diese Art von Richtlinienerklärung nur an IAM-Benutzer, -Gruppen oder -Rollen anhängen, die vollen Administratorzugriff auf Verbindungen in Ihrem AWS Konto benötigen.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "ConnectionsFullAccess",
        "Effect": "Allow",
        "Action": [
            "codeconnections:CreateConnection",
            "codeconnections:DeleteConnection",
            "codeconnections:UseConnection",
            "codeconnections:GetConnection",
            "codeconnections:ListConnections",
            "codeconnections:ListInstallationTargets",
            "codeconnections:GetInstallationUrl",
            "codeconnections:StartOAuthHandshake",
            "codeconnections:UpdateConnectionInstallation",
            "codeconnections:GetIndividualAccessToken",
            "codeconnections:TagResource",
```

```
            "codeconnections:ListTagsForResource",
            "codeconnections:UntagResource"
        ],
        "Resource": "*"
     }
   ]
}
```

Beispiel: Eine Richtlinie auf Mitwirkendenebene für die Verwendung AWS CodeConnections

In diesem Beispiel möchten Sie Zugriff auf die day-to-day Nutzung von gewähren CodeConnections, z. B. auf das Erstellen und Anzeigen von Verbindungsdetails, jedoch nicht auf destruktivere Aktionen wie das Löschen von Verbindungen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AWSCodeConnectionsPowerUserAccess",
            "Effect": "Allow",
            "Action": [
                "codeconnections:CreateConnection",
                "codeconnections:UseConnection",
                "codeconnections:GetConnection",
                "codeconnections:ListConnections",
                "codeconnections:ListInstallationTargets",
                "codeconnections:GetInstallationUrl",
                "codeconnections:GetIndividualAccessToken",
                "codeconnections:StartOAuthHandshake",
                "codeconnections:UpdateConnectionInstallation",
                "codeconnections:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

Beispiel: Eine read-only-level Richtlinie für die Verwendung AWS CodeConnections

In diesem Beispiel möchten Sie einem IAM-Benutzer in Ihrem Konto schreibgeschützten Zugriff auf die Verbindungen in Ihrem Konto gewähren. AWS Dieses Beispiel zeigt, wie Sie eine Richtlinie erstellen könnten, die die Anzeige dieser Elemente gestattet.

```
{
    "Version": "2012-10-17",
    "Id": "Connections__ReadOnly",
    "Statement": [
        {
            "Sid": "Reads_API_Access",
            "Effect": "Allow",
            "Action": [
            "codeconnections:GetConnection",
            "codeconnections:ListConnections",
            "codeconnections:ListInstallationTargets",
            "codeconnections:GetInstallationUrl",
            "codeconnections:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

Beispiel: Beschränken Sie die VPC-Berechtigungen für Hosts mithilfe des VpcIdKontextschlüssels

Im folgenden Beispiel kann der Kunde den VpcIdKontextschlüssel verwenden, um die Erstellung oder Verwaltung von Hosts auf Hosts mit spezifizierter VPC zu beschränken.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "codeconnections:CreateHost",
                "codeconnections:UpdateHost"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "codeconnections:VpcId": "vpc-EXAMPLE"
                }
            }
        }
    ]
```

```
}
```

Verwenden von Tags zur Steuerung des Zugriffs auf AWS CodeConnections -Ressourcen

Markierungen können an die Ressource angehängt oder in der Anfrage an Services übergeben werden, die das Markieren unterstützen. In AWS CodeConnections können Ressourcen Tags haben, und einige Aktionen können Tags enthalten. Wenn Sie eine IAM-Richtlinie erstellen, können Sie Markierungs-Bedingungsschlüssel verwenden, um Folgendes zu kontrollieren:

- Welche Benutzer Aktionen für eine Pipeline-Ressource ausführen können, basierend auf den Tags, über die diese bereits verfügt.
- Welche Tags in der Anforderung einer Aktion übergeben werden können.
- Ob bestimmte Tag-Schlüssel in einer Anforderung verwendet werden können.

Die folgenden Beispiele zeigen, wie Tag-Bedingungen in Richtlinien für AWS CodeConnections Benutzer angegeben werden.

Example 1: Zulassen von Aktionen basierend auf Markierungen in der Anforderung

Die folgende Richtlinie gewährt Benutzern die Berechtigung, Verbindungen in herzustellen AWS CodeConnections.

Hierfür werden die Aktionen `CreateConnection` und `TagResource` zugelassen, wenn die Anforderung ein Tag mit dem Namen `Project` und dem Wert `ProjectA` angibt. (Der Bedingungsschlüssel `aws:RequestTag` wird verwendet, um zu steuern, welche Tags in einer IAM-Anforderung übergeben werden können.) Die Bedingung `aws:TagKeys` stellt sicher, dass bei Tag-Schlüsseln die Groß- und Kleinschreibung beachtet wird.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codeconnections:CreateConnection",
        "codeconnections:TagResource"
      ],
```

```
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/Project": "ProjectA"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["Project"]
        }
      }
    }
  ]
}
```

Example 2: Zulassen von Aktionen basierend auf Ressourcen-Tags

Mit der folgenden Richtlinie wird Benutzern die Berechtigung zum Ausführen von Aktionen auf und Abrufen von Informationen zu Ressourcen in AWS CodeConnections gewährt.

Hierfür werden bestimmte Aktionen zugelassen, wenn die Pipeline ein Tag mit dem Namen `Project` und dem Wert `ProjectA` enthält. (Der Bedingungsschlüssel `aws:RequestTag` wird verwendet, um zu steuern, welche Tags in einer IAM-Anforderung übergeben werden können.) Die Bedingung `aws:TagKeys` stellt sicher, dass bei Tag-Schlüsseln die Groß- und Kleinschreibung beachtet wird.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codeconnections:CreateConnection",
        "codeconnections:DeleteConnection",
        "codeconnections:ListConnections"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "ProjectA"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["Project"]
        }
      }
    }
```

```
  ]
}
```

Verwenden von Notifications und Connections in der Konsole

Die Benachrichtigungserfahrung ist in die CodePipeline Konsolen CodeBuild CodeCommit CodeDeploy,, und sowie in die Developer Tools-Konsole in der Navigationsleiste „Einstellungen“ selbst integriert. Um auf Benachrichtigungen in den Konsolen zuzugreifen, muss entweder eine der verwalteten Richtlinien für diese Services für Sie angewendet sein oder Sie müssen über einen Mindestsatz an Berechtigungen verfügen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen in Ihrem AWS Konto aufzulisten und einzusehen. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (IAM-Benutzer oder -Rollen) mit dieser Richtlinie. Weitere Informationen zur Gewährung von Zugriff auf AWS CodeBuild, AWS CodeCommit, und AWS CodeDeploy AWS CodePipeline, einschließlich Zugriff auf diese Konsolen, finden Sie in den folgenden Themen:

- CodeBuild: Verwendung identitätsbasierter Richtlinien für CodeBuild
- CodeCommit: Verwendung identitätsbasierter Richtlinien für CodeCommit
- AWS CodeDeploy: Identitäts- und Zugriffsmanagement für AWS CodeDeploy
- CodePipeline: Zugriffskontrolle mit IAM-Richtlinien

AWS CodeStar Für Benachrichtigungen gibt AWS es keine verwalteten Richtlinien. Um Zugriff auf Benachrichtigungsfunktionen zu ermöglichen, müssen Sie entweder eine der verwalteten Richtlinien für einen der oben aufgeführten Services anwenden oder Richtlinien mit der Berechtigungsstufe erstellen, die Sie Benutzern oder Entitäten erteilen möchten, und diese Richtlinien dann den Benutzern, Gruppen oder Rollen anfügen, die die Berechtigungen benötigen. Weitere Informationen finden Sie in den folgenden Beispielen:

- Beispiel: Eine Richtlinie auf Administratorebene für die Verwaltung von Benachrichtigungen AWS CodeStar
- Beispiel: Eine Richtlinie auf Mitwirkendenebene für die Verwendung von Benachrichtigungen AWS CodeStar
- Beispiel: Eine read-only-level Richtlinie für die Verwendung von AWS CodeStar Benachrichtigungen.

AWS CodeConnections hat keine AWS verwalteten Richtlinien. Sie verwenden die Berechtigungen und Kombinationen von Berechtigungen für den Zugriff, z. B. die Berechtigungen, die unter Berechtigungen zum Abschließen von Verbindungen beschrieben ist.

Weitere Informationen finden Sie hier:

- Beispiel: Eine Richtlinie auf Administratorebene für die Verwaltung AWS CodeConnections
- Beispiel: Eine Richtlinie auf Mitwirkendenebene für die Verwendung AWS CodeConnections
- Beispiel: Eine read-only-level Richtlinie für die Verwendung AWS CodeConnections

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Konsolenberechtigungen gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die den API-Operation entsprechen, die Sie ausführen möchten.

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:ListGroupsForUser",
                "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": "*"
        }
    ]
}
```

Fehlerbehebung bei AWS CodeStar Benachrichtigungen sowie AWS CodeConnections Identität und Zugriff

Mit den folgenden Informationen können Sie häufige Probleme diagnostizieren und beheben, die beim Arbeiten mit Benachrichtigungen und IAM auftreten könnten.

Themen

- Ich bin Administrator und möchte anderen den Zugriff auf Benachrichtigungen ermöglichen
- Ich habe ein Amazon-SNS-Thema erstellt und es als Benachrichtigungsregelziel hinzugefügt, aber ich empfange keine E-Mail-Nachrichten zu Ereignissen
- Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen ermöglichen

Ich bin Administrator und möchte anderen den Zugriff auf Benachrichtigungen ermöglichen

Um anderen den Zugriff auf AWS CodeStar Benachrichtigungen und zu ermöglichen AWS CodeConnections, müssen Sie den Personen oder Anwendungen, die Zugriff benötigen, die entsprechenden Berechtigungen erteilen. Wenn Sie Benutzer 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 unter AWS CodeStar Benachrichtigungen und eine Richtlinie hinzufügen, die ihr die richtigen Berechtigungen gewährt. AWS CodeConnections Nachdem die Berechtigungen erteilt wurden, geben Sie die Anmeldeinformationen an den Benutzer oder Anwendungsentwickler weiter. 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.

Spezifische Informationen zu AWS CodeStar Benachrichtigungen finden Sie unter. Berechtigungen und Beispiele für AWS CodeStar Benachrichtigungen

Ich habe ein Amazon-SNS-Thema erstellt und es als Benachrichtigungsregelziel hinzugefügt, aber ich empfange keine E-Mail-Nachrichten zu Ereignissen

Um Benachrichtigungen zu Ereignissen zu erhalten, müssen Sie ein gültiges Amazon-SNS-Thema als Ziel für die Benachrichtigungsregel abonniert haben, und Ihre E-Mail-Adresse muss das Amazon-SNS-Thema abonniert haben. Um Probleme mit dem Amazon-SNS-Thema zu beheben, überprüfen Sie Folgendes:

- Stellen Sie sicher, dass sich das Amazon SNS SNS-Thema in derselben AWS Region wie die Benachrichtigungsregel befindet.
- Stellen Sie sicher, dass Ihr E-Mail-Alias das richtige Thema abonniert hat, und dass Sie das Abonnement bestätigt haben. Weitere Informationen finden Sie unter Abonnieren eines Endpunkts für ein Amazon-SNS-Thema.
- Vergewissern Sie sich, dass die Themenrichtlinie so geändert wurde, dass AWS CodeStar Benachrichtigungen Push-Benachrichtigungen zu diesem Thema senden können. Die Themarichtlinie sollte eine Anweisung ähnlich der folgenden enthalten:

```
{
    "Sid": "AWSCodeStarNotifications_publish",
    "Effect": "Allow",
    "Principal": {
        "Service": [
            "codestar-notifications.amazonaws.com"
        ]
    },
    "Action": "SNS:Publish",
    "Resource": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopicName",
```

```
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "123456789012"
        }
    }
}
```

Weitere Informationen finden Sie unter Einrichtung.

Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine AWS CodeStar Benachrichtigungen und AWS CodeConnections Ressourcen ermöglichen

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, kannst du diese Richtlinien verwenden, um Personen Zugriff auf deine Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob AWS CodeStar Benachrichtigungen und diese Funktionen AWS CodeConnections unterstützen, finden Sie unter. Funktionsweise von Funktionen in der Entwicklertools-Konsole 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 für 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.

Verwenden von serviceverknüpften Rollen für Benachrichtigungen AWS CodeStar

AWS CodeStar Notifications verwendet AWS Identity and Access Management dienstverknüpfte Rollen (IAM). Eine serviceverknüpfte Rolle ist ein einzigartiger Typ von IAM-Rolle, die direkt mit Benachrichtigungen verknüpft ist. AWS CodeStar Dienstbezogene Rollen sind in AWS CodeStar Notifications vordefiniert und beinhalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen. Die Rolle wird für Sie erstellt, wenn Sie zum ersten Mal eine Benachrichtigungsregel erstellen. Sie müssen die Rolle nicht erstellen.

Eine dienstbezogene Rolle erleichtert die Einrichtung von AWS CodeStar Benachrichtigungen, da Sie Berechtigungen nicht manuell hinzufügen müssen. AWS CodeStar Notifications definiert die Berechtigungen ihrer dienstbezogenen Rollen, und sofern nicht anders definiert, können nur AWS CodeStar Benachrichtigungen diese Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Um eine serviceverknüpfte Rolle zu löschen, müssen Sie zunächst die verwandten Ressourcen löschen. Dadurch werden Ihre AWS CodeStar Benachrichtigungsressourcen geschützt, da Sie die Zugriffsberechtigung für die Ressourcen nicht versehentlich entziehen können.

Weitere Informationen zu anderen Services, die serviceverknüpfte Rollen unterstützen, finden Sie unter AWS -Services, die mit IAM funktionieren.

Mit dem Dienst verknüpfte Rollenberechtigungen für Benachrichtigungen AWS CodeStar

AWS CodeStar Notifications verwendet die AWSService RoleForCodeStarNotifications dienstbezogene Rolle, um Informationen über Ereignisse abzurufen, die in Ihrer Toolchain auftreten, und um Benachrichtigungen an die von Ihnen angegebenen Ziele zu senden.

Die mit dem AWSService RoleForCodeStarNotifications Dienst verknüpfte Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `codestar-notifications.amazonaws.com`

Die Richtlinie für Rollenberechtigungen ermöglicht AWS CodeStar Notifications, die folgenden Aktionen für die angegebenen Ressourcen auszuführen:

- Aktion: `PutRule` für `CloudWatch Event rules that are named awscodestar-notifications-*`
- Aktion: `DescribeRule` für `CloudWatch Event rules that are named awscodestar-notifications-*`
- Aktion: `PutTargets` für `CloudWatch Event rules that are named awscodestar-notifications-*`
- Aktion: `CreateTopic`, um `create Amazon SNS topics for use with AWS CodeStar Notifications with the prefix CodeStarNotifications-`
- Aktion: `GetCommentsForPullRequests` für `all comments on all pull requests in all CodeCommit repositories in the AWS account`
- Aktion: `GetCommentsForComparedCommit` für `all comments on all commits in all CodeCommit repositories in the AWS account`
- Aktion: `GetDifferences` für `all commits in all CodeCommit repositories in the AWS account`
- Aktion: `GetCommentsForComparedCommit` für `all comments on all commits in all CodeCommit repositories in the AWS account`
- Aktion: `GetDifferences` für `all commits in all CodeCommit repositories in the AWS account`
- Aktion: `DescribeSlackChannelConfigurations` für `all AWS Chatbot clients in the AWS account`
- Aktion: `UpdateSlackChannelConfiguration` für `all AWS Chatbot clients in the AWS account`
- Aktion: `ListActionExecutions` für `all actions in all pipelines in the AWS account`
- Aktion: `GetFile` für `all files in all CodeCommit repositories in the AWS account unless otherwise tagged`

Sie können diese Aktionen in der Richtlinienerklärung für die Rolle, die mit dem AWSService RoleForCodeStarNotifications Dienst verknüpft ist, nachlesen.

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

```
            "Action": [
                "events:PutTargets",
                "events:PutRule",
                "events:DescribeRule"
            ],
            "Resource": "arn:aws:events:*:*:rule/awscodestarnotifications-*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "sns:CreateTopic"
            ],
            "Resource": "arn:aws:sns:*:*:CodeStarNotifications-*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "codecommit:GetCommentsForPullRequest",
                "codecommit:GetCommentsForComparedCommit",
                "codecommit:GetDifferences",
                "chatbot:DescribeSlackChannelConfigurations",
                "chatbot:UpdateSlackChannelConfiguration",
                "codepipeline:ListActionExecutions"
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "codecommit:GetFile"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:ResourceTag/ExcludeFileContentFromNotifications": "true"
                }
            },
            "Effect": "Allow"
        }
    ]
}
```

Sie müssen Berechtigungen konfigurieren, damit eine juristische Stelle von IAM (z. B. Benutzer, Gruppe oder Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigungen im IAM-Benutzerhandbuch.

Eine dienstbezogene Rolle für Benachrichtigungen erstellen AWS CodeStar

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Sie können die Developer Tools-Konsole oder die CreateNotificationRule API von oder aus verwenden, SDKs um eine Benachrichtigungsregel zu erstellen. AWS CLI Sie können die API auch direkt aufrufen. Die serviceverknüpfte Rolle wird für Sie erstellt, unabhängig von der verwendeten Methode.

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Sie können die Developer Tools-Konsole oder die CreateNotificationRule API von AWS CLI oder aus verwenden SDKs , um eine Benachrichtigungsregel zu erstellen. Sie können die API auch direkt aufrufen. Die serviceverknüpfte Rolle wird für Sie erstellt, unabhängig von der verwendeten Methode.

Eine serviceverknüpfte Rolle für AWS CodeStar Benachrichtigungen bearbeiten

Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, ist es nach dem Erstellen einer serviceverknüpften Rolle nicht mehr möglich, den Namen der Rolle zu ändern. Sie können mithilfe von IAM jedoch die Beschreibung der Rolle bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer dienstbezogenen Rolle für Benachrichtigungen AWS CodeStar

Wenn Sie eine Funktion oder einen Service nicht mehr benötigen, die bzw. der eine serviceverknüpfte Rolle erfordert, sollten Sie die Rolle löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird. Sie müssen die Ressourcen für Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie löschen können. Für AWS CodeStar Benachrichtigungen bedeutet dies, dass alle Benachrichtigungsregeln gelöscht werden, die die Servicerolle in Ihrem AWS Konto verwenden.

> **Note**
>
> Wenn der AWS CodeStar Benachrichtigungsdienst die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um Ressourcen für AWS CodeStar Benachrichtigungen zu löschen, die verwendet werden von AWSService RoleForCodeStarNotifications

1. Öffnen Sie die AWS Developer Tools-Konsole unter https://console.aws.amazon.com/codesuite/Einstellungen/Benachrichtigungen.

   > **Note**
   >
   > Benachrichtigungsregeln gelten für die AWS Region, in der sie erstellt wurden. Wenn Sie Benachrichtigungsregeln in mehr als einer AWS Region haben, verwenden Sie die Regionsauswahl, um die AWS-Region zu ändern.

2. Wählen Sie alle Benachrichtigungsregeln aus, die in der Liste angezeigt werden und dann Delete (Löschen).
3. Wiederholen Sie diese Schritte in allen AWS Regionen, in denen Sie Benachrichtigungsregeln erstellt haben.

IAM verwenden, um die serviceverknüpfte Rolle zu löschen

Verwenden Sie die IAM-Konsole oder AWS Identity and Access Management API AWS CLI, um die AWSService RoleForCodeStarNotifications serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Leitfaden.

Unterstützte Regionen für Rollen, die mit dem Dienst für AWS CodeStar Benachrichtigungen verknüpft sind

AWS CodeStar Notifications unterstützt die Verwendung von dienstbezogenen Rollen in allen AWS Regionen, in denen der Dienst verfügbar ist. Weitere Informationen finden Sie unter AWS Regionen und Endpunkte und AWS CodeStar Benachrichtigungen.

Verwenden von serviceverknüpften Rollen für AWS CodeConnections

AWS CodeConnections verwendet AWS Identity and Access Management (IAM) serviceverknüpfte Rollen. Eine serviceverknüpfte Rolle ist ein einzigartiger Typ von IAM-Rolle, mit der direkt verknüpft ist. AWS CodeConnections Mit Diensten verknüpfte Rollen sind vordefiniert AWS CodeConnections und enthalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen. Die Rolle wird für Sie erstellt, wenn Sie zum ersten Mal eine Verbindung erstellen. Sie müssen die Rolle nicht erstellen.

Eine dienstbezogene Rolle AWS CodeConnections erleichtert die Einrichtung, da Sie Berechtigungen nicht manuell hinzufügen müssen. AWS CodeConnections definiert die Berechtigungen ihrer dienstbezogenen Rollen und AWS CodeConnections kann, sofern nicht anders definiert, nur ihre Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Um eine serviceverknüpfte Rolle zu löschen, müssen Sie zunächst die verwandten Ressourcen löschen. Dadurch werden Ihre AWS CodeConnections Ressourcen geschützt, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Weitere Informationen zu anderen Services, die serviceverknüpfte Rollen unterstützen, finden Sie unter AWS -Services, die mit IAM funktionieren.

> **Note**
>
> Aktionen für Ressourcen, die unter dem neuen Dienstpräfix `codeconnections` erstellt wurden, sind verfügbar. Wenn Sie eine Ressource unter dem neuen Dienstpräfix erstellen, wird sie `codeconnections` im Ressourcen-ARN verwendet. Aktionen und Ressourcen für das `codestar-connections` Dienstpräfix bleiben verfügbar. Bei der Angabe einer Ressource in der IAM-Richtlinie muss das Dienstpräfix mit dem der Ressource übereinstimmen.

Mit dem Dienst verknüpfte Rollenberechtigungen für AWS CodeConnections

AWS CodeConnections verwendet die AWSService RoleForGitSync serviceverknüpfte Rolle, um Git Sync mit verbundenen Git-basierten Repositorys zu verwenden.

Die AWSService RoleForGitSync dienstverknüpfte Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `repository.sync.codeconnections.amazonaws.com`

Die genannte Rollenberechtigungsrichtlinie AWSGit SyncServiceRolePolicy ermöglicht es AWS CodeConnections , die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Aktion: Gewährt Benutzern Berechtigungen, die es Benutzern ermöglichen, Verbindungen mit externen Git-basierten Repositorys herzustellen und eine Git-Synchronisierung mit diesen Repositorys zu verwenden.

Sie müssen Berechtigungen konfigurieren, damit eine juristische Stelle von IAM (z. B. Benutzer, Gruppe oder Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigungen im IAM-Benutzerhandbuch.

Erstellen einer serviceverknüpften Rolle für AWS CodeConnections

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Sie erstellen die Rolle, wenn Sie eine Ressource für Ihr Git-synchronisiertes Projekt mit der API erstellen. CreateRepositoryLink

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen.

Bearbeiten einer serviceverknüpften Rolle für AWS CodeConnections

Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, ist es nach dem Erstellen einer serviceverknüpften Rolle nicht mehr möglich, den Namen der Rolle zu ändern. Sie können mithilfe von IAM jedoch die Beschreibung der Rolle bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für AWS CodeConnections

Wenn Sie eine Funktion oder einen Service nicht mehr benötigen, die bzw. der eine serviceverknüpfte Rolle erfordert, sollten Sie die Rolle löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird. Sie müssen die Ressourcen für Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie löschen können. Dies bedeutet, dass alle Verbindungen gelöscht werden, die die Servicerolle in Ihrem Konto verwenden. AWS

> **Note**
>
> Wenn der AWS CodeConnections Dienst die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um AWS CodeConnections Ressourcen zu löschen, die verwendet werden von AWSService RoleForGitSync

1. Öffnen Sie die Developer-Tools-Konsole und wählen Sie dann Einstellungen aus.
2. Wählen Sie alle in der Liste angezeigten Benachrichtigungsregeln und dann Löschen aus.
3. Wiederholen Sie diese Schritte in allen AWS Regionen, in denen Sie Verbindungen erstellt haben.

IAM verwenden, um die serviceverknüpfte Rolle zu löschen

Verwenden Sie die IAM-Konsole oder AWS Identity and Access Management API AWS CLI, um die AWSService RoleForGitSync serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Leitfaden.

Unterstützte Regionen für serviceverknüpfte Rollen AWS CodeConnections

AWS CodeConnections unterstützt die Verwendung von dienstbezogenen Rollen in allen AWS Regionen, in denen der Dienst verfügbar ist. Weitere Informationen finden Sie unter AWS Regionen und Endpunkte.

AWS verwaltete Richtlinien für AWS CodeConnections

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien dienen dazu, Berechtigungen für viele gängige Anwendungsfälle bereitzustellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie für alle AWS Kunden verfügbar sind. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

> **Note**
>
> Aktionen für Ressourcen, die unter dem neuen Dienstpräfix erstellt wurden, `codeconnections` sind verfügbar. Wenn Sie eine Ressource unter dem neuen Dienstpräfix erstellen, wird sie `codeconnections` im Ressourcen-ARN verwendet. Aktionen und Ressourcen für das `codestar-connections` Dienstpräfix bleiben verfügbar. Bei der Angabe einer Ressource in der IAM-Richtlinie muss das Dienstpräfix mit dem der Ressource übereinstimmen.

AWS verwaltete Richtlinie: AWSGit SyncServiceRolePolicy

Sie können keine Verbindungen AWSGit SyncServiceRolePolicy zu Ihren IAM-Entitäten herstellen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, mit der Sie Aktionen AWS CodeConnections in Ihrem Namen ausführen können. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für AWS CodeConnections.

Diese Richtlinie ermöglicht Kunden den Zugriff auf Git-basierte Repositorys zur Verwendung mit Verbindungen. Kunden werden nach der Nutzung der CreateRepositoryLink API auf diese Ressourcen zugreifen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `codeconnections`: Gewährt Benutzern Berechtigungen, die es Benutzern ermöglichen, Verbindungen mit externen Git-basierten Repositorys herzustellen.

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

```
 "Statement": [
  {
   "Sid": "AccessGitRepos",
   "Effect": "Allow",
   "Action": [
    "codestar-connections:UseConnection",
    "codeconnections:UseConnection"
   ],
   "Resource": [
    "arn:aws:codestar-connections:*:*:connection/*",
    "arn:aws:codeconnections:*:*:connection/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  }
 ]
}
```

AWS CodeConnections Aktualisierungen der AWS verwalteten Richtlinien

Hier finden Sie Informationen zu Aktualisierungen AWS verwalteter Richtlinien AWS CodeConnections seit Beginn der Nachverfolgung dieser Änderungen durch diesen Dienst. Abonnieren Sie den RSS-Feed auf der Seite AWS CodeConnections Dokumentenverlauf, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten.

Änderung	Beschreibung	Datum
AWSGitSyncServiceRolePolicy— Aktualisierte Richtlinie	Der Name des AWS CodeStar Connections-Service wurde geändert in AWS CodeConnections. Die Richtlinie für Ressourcen, die beide Servicepräfixe enthalten ARNs , wurde aktualisiert.	26. April 2024
AWSGitSyncServiceRolePolicy – Neue Richtlinie	AWS CodeStar Connections hat die Richtlinie hinzugefügt.	26. November 2023

Änderung	Beschreibung	Datum
	Erteilt Berechtigungen, die es Verbindungsbenutzern ermöglichen, Git Sync mit verbundenen Git-basierten Repositorys zu verwenden.	
AWS CodeConnections hat begonnen, Änderungen zu verfolgen	AWS CodeConnections hat begonnen, Änderungen für die AWS verwalteten Richtlinien zu verfolgen.	26. November 2023

Konformitätsvalidierung für AWS CodeStar Benachrichtigungen und AWS CodeConnections

Eine Liste der AWS Services im Rahmen bestimmter Compliance-Programme finden Sie unter AWS Services im Umfang der einzelnen Compliance-Programme. Allgemeine Informationen finden Sie unter AWS -Compliance-Programme.

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte in AWS Artifact herunterladen.

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Verwendung von AWS CodeStar Benachrichtigungen hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS CodeConnections 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 für die Implementierung von Umgebungen beschrieben, auf denen auf Sicherheit und Compliance ausgerichtete Basisumgebungen eingerichtet werden. AWS
- AWS Ressourcen zur Einhaltung von Vorschriften — Diese Sammlung von Arbeitsmappen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.
- AWS Config— Mit diesem AWS Service wird 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 bei AWS CodeStar Benachrichtigungen und AWS CodeConnections

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.

- Benachrichtigungsregeln sind spezifisch für den AWS-Region Ort, an dem sie erstellt wurden. Wenn Sie in mehr als einer Benachrichtigungsregeln haben AWS-Region, verwenden Sie die Regionsauswahl, um die jeweiligen AWS-Region Benachrichtigungsregeln zu überprüfen.
- AWS CodeStar Benachrichtigungen basieren auf Amazon Simple Notification Service (Amazon SNS) -Themen als Ziele für Benachrichtigungsregeln. Informationen zu Ihren Amazon-SNS-Themen und Benachrichtigungsregelzielen könnten in einer AWS -Region gespeichert werden, bei der es sich nicht um die Region handelt, in der Sie die Benachrichtigungsregel konfiguriert haben.

Infrastruktursicherheit in AWS CodeStar Benachrichtigungen und AWS CodeConnections

Als Funktionen in einem verwalteten Service AWS CodeConnections sind AWS CodeStar Benachrichtigungen und durch die AWS globalen Netzwerksicherheitsverfahren geschützt, die im Whitepaper Amazon Web Services: Sicherheitsprozesse im Überblick beschrieben sind.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff auf AWS CodeStar Benachrichtigungen und AWS CodeConnections über das Netzwerk. Kunden müssen Transport Layer Security (TLS) 1.0 oder neuer unterstützen. Clients müssen außerdem

Verschlüsselungssammlungen mit PFS (Perfect Forward Secrecy) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) unterstützen. Die meisten modernen Systeme unterstützen diese Modi.

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

Datenverkehr zwischen Ressourcen von AWS CodeConnections über Regionen hinweg

Wenn Sie die Verbindungsfunktion verwenden, um die Verbindung Ihrer Ressourcen zu ermöglichen, erklären Sie sich damit einverstanden und weisen uns an, Informationen im Zusammenhang mit diesen Verbindungsressourcen AWS-Regionen außerhalb des Bereichs zu speichern und zu verarbeiten, in AWS-Regionen dem Sie den zugrunde liegenden Dienst nutzen, und zwar ausschließlich in Verbindung mit und zu dem alleinigen Zweck, Verbindungen zu solchen Ressourcen in anderen Regionen als der, in der die Ressource erstellt wurde, herzustellen.

Weitere Informationen finden Sie unter Globale Ressourcen in AWS CodeConnections.

> **Note**
>
> Wenn Sie die Verbindungsfunktion verwenden, um die Verbindung für Ihre Ressourcen in Regionen zu aktivieren, für die keine vorherige Aktivierung erforderlich ist, speichern und verarbeiten wir Informationen, wie in den vorherigen Themen beschrieben.
> Bei Verbindungen, die in Regionen hergestellt werden, die zuerst aktiviert werden müssen, wie z. B. in der Region Europa (Mailand), speichern und verarbeiten wir nur Informationen für diese Verbindung in dieser Region.

Verbindungen umbenennen — Zusammenfassung der Änderungen

Mit der Verbindungsfunktion in der Developer Tools-Konsole können Sie Ihre AWS Ressourcen mit Quell-Repositorys von Drittanbietern verbinden. Am 29. März 2024 wurde AWS CodeStar Connections in umbenannt. AWS CodeConnections In den folgenden Abschnitten werden die verschiedenen Teile der Funktion beschrieben, die sich durch die Umbenennung geändert haben, und welche Maßnahmen Sie ergreifen müssen, um sicherzustellen, dass Ihre Ressourcen weiterhin ordnungsgemäß funktionieren.

Beachten Sie, dass diese Liste nicht vollumfänglich ist. Auch andere Teile des Produkts haben sich geändert, aber diese Updates sind am relevantesten.

> **Note**
>
> Aktionen für Ressourcen, die unter dem neuen Dienstpräfix erstellt wurden, `codeconnections` sind verfügbar. Wenn Sie eine Ressource unter dem neuen Dienstpräfix erstellen, wird sie `codeconnections` im Ressourcen-ARN verwendet. Aktionen und Ressourcen für das `codestar-connections` Dienstpräfix bleiben verfügbar. Wenn Sie eine Ressource in der IAM-Richtlinie angeben, muss das Dienstpräfix mit dem der Ressource übereinstimmen.

> **Note**
>
> Ab dem 1. Juli 2024 stellt die Konsole Verbindungen mit `codeconnections` der Ressource ARN her. Ressourcen mit beiden Dienstpräfixen werden weiterhin in der Konsole angezeigt.

Dienstpräfix umbenannt

Verbindungen APIs verwenden ein umbenanntes Dienstpräfix:codeconnections.

Um das neue Präfix in CLI-Befehlen zu verwenden, laden Sie die Version 2 von herunter AWS CLI. Im Folgenden finden Sie einen Beispielbefehl mit dem aktualisierten Präfix.

```
aws codeconnections delete-connection --connection-arn arn:aws:codeconnections:us-
west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Aktionen in IAM wurden umbenannt

Die Aktionen in IAM verwenden das neue Präfix, wie in den folgenden Beispielen gezeigt:

```
codeconnections:CreateConnection
codeconnections:DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
```

Neue Ressource ARN

Verbindungsressourcen, die erstellt werden, erhalten einen neuen ARN:

```
arn:aws:codeconnections:us-west-2:account-ID:connection/*
```

Betroffene Richtlinien für Servicerollen

Für die folgenden Dienste verwenden die Richtlinien für Diensterollen das neue Präfix in den Richtlinienerklärungen. Sie können auch Ihre bestehenden Richtlinien für Servicerollen aktualisieren, um die neuen Berechtigungen zu verwenden. Richtlinien, die mit dem alten Präfix erstellt wurden, werden jedoch weiterhin unterstützt.

- Die vom CodePipeline Kunden verwaltete Richtlinie für Servicerollen
- Die Richtlinie AWS CodeStar für Servicerollen `AWSCodeStarServiceRole`

Neue CloudFormation Ressource

Um die AWS CloudFormation Ressourcen für Verbindungen zu verwenden, wird eine neue Ressource verfügbar sein. Die bestehende Ressource wird weiterhin unterstützt.

- Die neue AWS CloudFormationRessource heißt AWS::CodeConnections: :Connection. Weitere Informationen finden Sie AWS::CodeConnections::Connectionim CloudFormation Benutzerhandbuch.

- Die bestehende Ressource AWS:CodeStarConnections:: :Connection wird weiterhin unterstützt. Weitere Informationen finden Sie AWS::CodeStarConnections::Connectionim CloudFormation Benutzerhandbuch.

Dokumentverlauf

Die folgende Tabelle beschreibt die Dokumentation zu dieser Version der Entwicklertools-Konsole.

- AWS CodeStar API-Version für Benachrichtigungen: 2019-10-15
- AWS CodeConnections API-Version: 2023-12-01

Änderung	Beschreibung	Datum
Neuer Bedingungsschlüssel für Host-VPC IDs	Sie können den Hostzugriff für GitHub Enterprise Server und GitLab selbstverwaltete Hosts mithilfe des VpcId Bedingungsschlüssels verwalten. Mit dem Bedingungsschlüssel können Sie Richtlinien im Zusammenhang mit der Erstellung oder Aktualisierung von Hosts zur Verwendung einer bestimmten VPC-ID durchsetzen. Weitere Informationen finden Sie in der Berechtigungsreferenz für Verbindungen.	13. März 2025
Unterstützung für die gemeinsame Nutzung von Verbindungen zwischen Konten hinzufügen	Sie können Verbindungen als Ressourcen in anzeigen und verwalten AWS Resource Access Manager, und Sie können Verbindungen zwischen ihnen teilen AWS-Konten. Weitere Informationen finden Sie unter Verbindungen teilen mit AWS-Konten.	6. März 2025

Aktualisierung, um Informationen hinzuzufügen und zu korrigieren, die beschreiben, wie Verbindungen mit Benutzerkonten oder Organisationen funktionieren	Die Übersichten und Informationen zur Problembehandlung wurden aktualisiert und beschreiben nun korrekt, wie Verbindungen mit Benutzerkonten oder Organisationen funktionieren. Weitere Informationen finden Sie unter So funktionieren Verbindungen, So AWS CodeConnections funktionieren Verbindungen mit Organisationen und Einrichtung von Verbindungen und Hosts für installierte Anbieter, die Organisationen unterstützen.	9. Dezember 2024
Aktualisierung der verwalteten Richtlinie für Verbindungen service-linked-role	Die verwaltete Richtlinie für die dienstverknüpfte Rolle zur Verwendung von Git-Synchronisierung mit Git-Repositorys wurde für Ressourcen mit beiden Dienstpräfixen aktualisiert. Weitere Informationen findest du unter Verwenden von dienstbezogenen Rollen für AWS CodeConnections und Verwaltete Richtlinien.	26. April 2024
AWS CodeStar Verbindungen wurden umbenannt in AWS CodeConnections	Einführung AWS CodeConnections, mit der Sie Verbindungen zwischen AWS Ressourcen, z. B. Pipelines in CodePipeline, zu Git-Drittanbietern herstellen und verwalten können.	29. März 2024

Verbindungen zu werden GitLab jetzt unterstützt in CodeBuild	Support CodeBuild für die Konfiguration von Verbindungen zu hinzugefügt GitLab. Weitere Informationen finden Sie unter Produkt- und Serviceintegrationen mit AWS CodeConnections.	27. März 2024
Support für GitLab selbstverwaltete	Support für die Konfiguration von Verbindungen und Hosts für AWS Ressourcen zur Interaktion mit GitLab selbstverwalteten Ressourcen hinzugefügt. Weitere Informationen finden Sie unter Arbeitsablauf zum Erstellen oder Aktualisieren eines Hosts und Herstellen einer Verbindung zu GitLab selbstverwalteten Hosts.	28. Dezember 2023
Neue Repository-Links und Synchronisierungskonfigurationen für Verbindungen	Es wurden Informationen zur Konfiguration von Repository-Links und Synchronisierungskonfigurationen hinzugefügt. Verwenden Sie die Synchronisierungskonfiguration, um Inhalte aus einem Git-Repository zu synchronisieren und Ihre AWS CloudFormation Stack-Ressourcen zu aktualisieren. Weitere Informationen finden Sie unter Arbeiten mit Repository-Links und Arbeiten mit Synchronisierungskonfigurationen.	8. November 2023

Support für Verbindungen service-linked-role	Unterstützung für die Konfiguration von Verbindungen zur Verwendung der Git-Synchronisierung mit Git-Repositorys wurde hinzugefügt. Weitere Informationen finden Sie unter Verwenden von dienstbezogenen Rollen für AWS CodeConnections und unter Verwaltete Richtlinien.	26. November 2023
Support für GitLab Gruppen	Support für die Konfiguration von Verbindungen für AWS Ressourcen zur Interaktion mit GitLab Gruppen hinzugefügt. Weitere Informationen finden Sie unter Verbindung erstellen und Verbindung herstellen zu GitLab.	15. September 2023
Neuer GitLab Anbietertyp	Sie können jetzt Verbindungen zu herstellen GitLab. Weitere Informationen finden Sie unter Verbindung erstellen und Verbindung herstellen zu GitLab.	10. August 2023

Neuer Zieltyp für Benachrichtigungsregeln	Sie können jetzt AWS Chatbot-Clients, die für Microsoft Teams-Kanäle konfiguriert sind, als Ziel für Benachrichtigungsregeln auswählen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel und Arbeiten mit Benachrichtigungsregelzielen.	17. Mai 2023
Verbindungen sind in der Region Europa (Mailand) verfügbar	Es wurden Informationen über Verbindungen in der Region Europa (Mailand) hinzugefügt. Weitere Informationen finden Sie unter Verkehr zwischen AWS CodeConnections Ressourcen in verschiedenen Regionen.	17. Mai 2023
Fehlerbehebung für Verbindungsfehler mit Repository-Berechtigungen hinzugefügt	Wenn Sie eine Verbindung zu einem Repository in einer GitHub Organisation herstellen, müssen Sie der Eigentümer der GitHub Organisation sein. Weitere Informationen finden Sie unter Verbindungsfehler beim Herstellen einer Verbindung zu GitHub.	2p. August 2022
Zusätzliche Informationen zum Markieren von Host-Ressourcen	Sie können Hosts jetzt mithilfe der Konsole und des CLI markieren. Weitere Informationen finden Sie unter Ressourcen taggen in AWS CodeConnections.	19. April 2021

Support für VPC-Endpunkte für Verbindungen	Sie können jetzt VPC-Endpunkte mit Verbindungen verwenden. Weitere Informationen finden Sie unter VPC-Endpoints ()AWS CodeConnections AWS PrivateLink und stellen Sie eine Schnittstelle her.	24. November 2020
Neue GitHub und GitHub Enterprise Cloud-Anbietertypen	Sie können jetzt Verbindungen zu einer GitHub GitHub Enterprise Cloud herstellen. Weitere Informationen finden Sie unter Verbindung herstellen und Verbindung herstellen zu GitHub.	30. September 2020
Der GitHub Enterprise Server-Providertyp und die Hostressourcen wurden hinzugefügt	Dieses Handbuch enthält jetzt Informationen zur Hostressource für Verbindungen. Sie können jetzt Verbindungen zu GitHub Enterprise Server herstellen. Weitere Informationen finden Sie unter Erstellen von Verbindungen und Arbeiten mit Hosts. Das ist die allgemein verfügbare Version der Verbindungsfunktion im Benutzerhandbuch zur Entwicklertools-Konsole.	29. Juni 2020

Neue Informationen zum Verwenden und Markieren von Verbindungen	Dieses Handbuch enthält nun Informationen über die Verbindungsfunktion in der Konsole. Sie erfahren darin mehr über Konzepte und erste Schritte, finden eine Berechtigungsreferenz einschließlich Beispielrichtlinien sowie Schritte zum Erstellen, Anzeigen und Markieren von Verbindungen. Weitere Informationen finden Sie unter Was sind Verbindungen, Verbindungskonzepte, Erste Schritte mit Verbindungen, Eine Verbindung erstellen, Ressourcen taggen in AWS CodeConnections, Sicherheit, Kontingente für Verbindungen, Problembehandlung und AWS CodeConnections API-Aufrufe mit AWS CloudTrail. Eine Liste zusätzlicher Anbieteraktionen (Aktionen nur für Berechtigungen) finden Sie unter Aktionen für ProviderType.	28. Juni 2020

Neuer Zieltyp für Benachrichtigungsregeln	Du kannst jetzt AWS Chatbot-Clients, die für Slack-Channels konfiguriert sind, als Ziel für Benachrichtigungsregeln auswählen. Weitere Informationen finden Sie unter Erstellen einer Benachrichtigungsregel und Arbeiten mit Benachrichtigungsregelzielen.	2. April 2020
Benachrichtigungen über zusätzliche Ereignisse wurden hinzugefügt AWS CodeCommit	Sie können jetzt Benachrichtigungen für Ereignisse konfigurieren, die mit Genehmigungen für Pull-Anforderungen zusammenhängen. Weitere Informationen findest du unter Ereignisse für Benachrichtigungsregeln in Repositorys und Mit Pull-Requests arbeiten in CodeCommit.	10. Februar 2020
Benachrichtigungen sind in zwei weiteren AWS Regionen verfügbar	Die Entwicklertools-Konsole unterstützt jetzt Benachrichtigungen in den Regionen Naher Osten (Bahrain) und Asien-Pazifik (Hongkong). Weitere Informationen finden Sie unter AWS CodeStar Benachrichtigungen in der Allgemeine AWS-Referenz.	5. Februar 2020

Verschlüsselte Amazon-SNS-Themen werden jetzt unterstützt	Es gibt jetzt eine Anleitung für die Verwendung verschlüsselter Amazon-SNS-Themen als Benachrichtigungsziele. Weitere Informationen finden Sie unter Konfigurieren von Amazon SNS-Themen für Benachrichtigungen.	4. Februar 2020
Benachrichtigungen können Sitzungs-Tag-Informationen für enthalten CodeCommit	Benachrichtigungen für CodeCommit können nun mithilfe von Sitzungs-Tags Informationen zur Benutzeridentität enthalten, wie z. B. einen Anzeigenamen oder eine E-Mail-Adresse. Weitere Informationen finden Sie unter Konzepte und Verwendung von Tags zur Bereitstellung von Identitätsinformationen in CodeCommit.	19. Dezember 2019
Erstversion	Dies ist die erste Version des Handbuchs zur Entwicklertools-Konsole.	5. November 2019

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.