

ユーザーガイド

# デベロッパーツールコンソール



# デベロッパーツールコンソール: ユーザーガイド

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

Amazon の商標とトレードドレスは、Amazon 以外の製品またはサービスとの関連において、顧客に混乱を招いたり、Amazon の名誉または信用を毀損するような方法で使用することはできません。Amazon が所有しない他の商標はすべてそれぞれの所有者に帰属します。所有者は必ずしも Amazon との提携や関連があるわけではありません。また、Amazon の支援を受けているとはかぎりません。

# Table of Contents

- デベロッパーツールコンソールとは ..... 1
  - を初めてお使いになる方向けの情報 ..... 3
  - デベロッパーツールコンソールの機能 ..... 3
  - 通知とは何ですか? ..... 3
    - 通知でどのようなことができますか? ..... 4
    - 通知はどのような仕組みで機能しますか? ..... 4
    - 通知の使用を開始する方法 ..... 4
    - 通知の概念 ..... 5
    - セットアップ ..... 13
    - 通知の使用開始 ..... 19
    - 通知ルールの使用 ..... 27
    - 通知ルールのターゲットの使用 ..... 40
    - 通知と AWS Chatbot との統合の設定 ..... 49
    - AWS CloudTrail を使用した AWS CodeStar Notifications API コールのログ記録 ..... 54
    - トラブルシューティング ..... 58
    - クォータ ..... 61
- 接続とは? ..... 61
  - 接続では何ができますか? ..... 62
  - どのサードパーティープロバイダーの接続を作成できますか? ..... 62
  - 接続と AWS のサービス 統合するもの ..... 63
  - 接続はどのように機能しますか? ..... 63
  - 接続を開始するにはどうしたらいいですか? ..... 68
  - 接続概念 ..... 68
  - AWS CodeConnections サポートされているプロバイダーとバージョン ..... 69
  - と製品およびサービスの統合 AWS CodeConnections ..... 70
  - 接続のセットアップ ..... 73
  - 接続の使用開始 ..... 77
  - 接続の使用 ..... 83
  - ホストの使用 ..... 136
  - リンクされたりリポジトリの同期設定を操作する ..... 148
  - を使用した接続 API コールのログ記録 CloudTrail ..... 158
  - VPC エンドポイントAWS PrivateLink ..... 198
  - 接続のトラブルシューティング ..... 202
  - クォータ ..... 214

許可リストに追加する IP アドレス .....	215
セキュリティ .....	218
通知の内容とセキュリティについて .....	219
データ保護 .....	220
ID およびアクセス管理 .....	221
対象者 .....	222
アイデンティティを使用した認証 .....	222
ポリシーを使用したアクセスの管理 .....	226
デベロッパーツールコンソールの機能と IAM との連携方法 .....	227
AWS CodeConnections アクセス許可リファレンス .....	233
アイデンティティベースポリシーの例 .....	248
タグを使用して AWS CodeConnections リソースへのアクセスを制御する .....	261
コンソールを使用する場合 .....	263
ユーザーが自分の許可を表示できるようにする .....	264
トラブルシューティング .....	265
AWS CodeStar 通知のサービスにリンクされたロールの使用 .....	268
のサービスにリンクされたロールの使用 AWS CodeConnections .....	272
AWS マネージドポリシー .....	275
コンプライアンス検証 .....	278
耐障害性 .....	278
インフラストラクチャセキュリティ .....	279
リージョンをまたぐ AWS CodeConnections リソース間のトラフィック .....	279
接続の名前変更 - 変更の概要 .....	281
名前が変更されたサービスプレフィックス .....	281
IAM での名前変更アクション .....	282
新しいリソース ARN .....	282
影響を受けるサービスロールポリシー .....	4
新しい CloudFormation リソース .....	4
ドキュメント履歴 .....	283
AWS 用語集 .....	291
.....	ccxcii

# デベロッパーツールコンソールとは

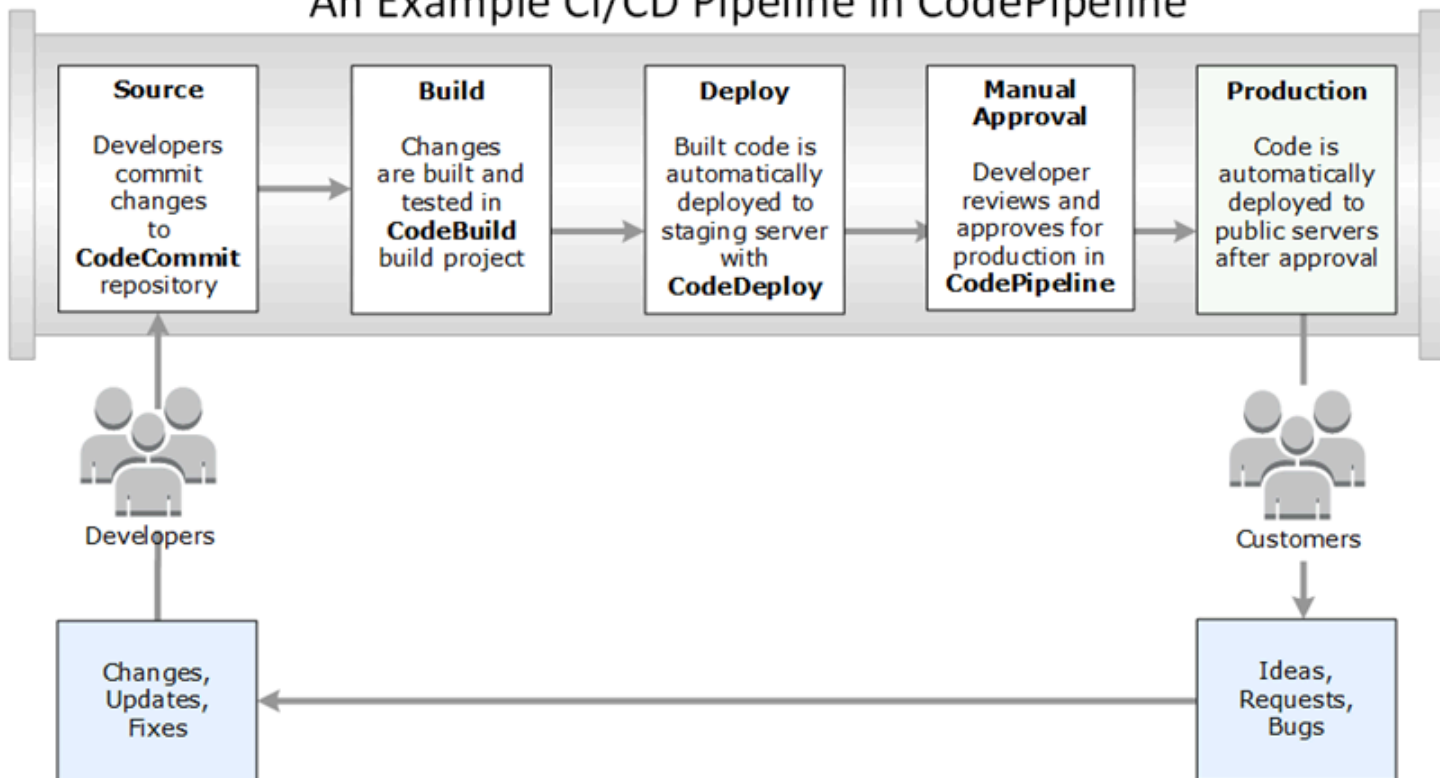
デベロッパーツールコンソールには、ソフトウェアを開発するために個別にまたはまとめて使用できる、一連のサービスと機能があります。デベロッパーツールは、ソフトウェアを安全に保存、ビルド、テスト、およびデプロイするのに役立ちます。これらのツールは個別またはまとめて使用され、継続的インテグレーション DevOps、継続的デリバリー (CI/CD) をサポートします。

デベロッパーツールコンソールには、以下のサービスが含まれます。

- [AWS CodeCommit](#) は、プライベートの Git リポジトリをホストする、完全に管理されたソースコントロールサービスです。リポジトリを使用することで、アセット (ドキュメント、ソースコード、バイナリファイルなど) を AWS クラウドに非公開で保存および管理することができます。リポジトリには、最初のコミットから最新の変更までのプロジェクト履歴が保存されます。コードにコメントし、プルリクエストを作成して、コードの品質を確保することで、リポジトリ内のコードで共同で作業できます。
- [AWS CodeBuild](#) は完全マネージド型の構築サービスです。ソースコードのコンパイル、ユニットテストの実行、すぐにデプロイできるアーティファクトの生成を行います。Apache Maven、Gradle などの一般的なプログラミング言語とビルドツール用のパッケージ済みのビルド環境を提供します。ビルド環境をカスタマイズ CodeBuild して、独自のビルドツールを使用することもできます。
- [AWS CodeDeploy](#) は、Amazon EC2 AWS Lambdaやオンプレミスサーバーなどのコンピューティングサービスへのソフトウェアデプロイを自動化するフルマネージドデプロイサービスです。これにより、新しい機能を迅速にリリースし、アプリケーションのデプロイ中のダウンタイムを回避し、アプリケーションの更新に伴う複雑さを処理できます。
- [AWS CodePipeline](#) は、ソフトウェアをリリースするために必要な手順のモデル化、可視化、および自動化に使用できる継続的な統合および継続的な配信サービスです。ソフトウェアリリースプロセスのさまざまなステージを素早くモデル化して設定できます。お客様は、お客様が定義するリリースプロセスモデルに基づいて、コードの変更があるたびに、コードのビルド、テスト、デプロイを実施できます。

ここでは、デベロッパーツールコンソールのサービスを一緒に使用して、ソフトウェアの開発を支援する方法の例を示します。

## An Example CI/CD Pipeline in CodePipeline



この例では、デベロッパーは リポジトリを作成し CodeCommit、それを使用してコードの開発と共同作業を行います。ビルドプロジェクトを作成してコード CodeBuild を構築およびテストし、CodeDeploy を使用してコードをテスト環境と本番環境にデプロイします。迅速に反復処理を行いたいため、パイプラインを作成して CodeCommit リポジトリ内の変更 CodePipeline を検出します。これらの変更がビルドされ、テストが実行され、正常にビルドされ、テストされたコードがテストサーバーにデプロイされます。チームは、テストステージをパイプラインに追加して、統合テストや負荷テストなど、ステージングサーバーでさらに多くのテストを実行します。これらのテストが正常に完了すると、チームメンバーは結果を確認し、問題がなければ本番稼働用の変更を手動で承認します。はテスト済みおよび承認済みのコードを本番稼働用インスタンスに CodePipeline デプロイします。

これは、デベロッパーツールコンソールで提供されている 1 つまたは複数のサービスを使用してソフトウェアを開発する方法を示す簡単な例の 1 つです。各サービスは、ニーズに合わせてカスタマイズできます。では、他の製品やサービスと多くの統合が提供されています。また、AWS や他のサードパーティ製ツールとも統合されています。詳細については、次のトピックを参照してください。

- CodeCommit: [製品とサービスの統合](#)
- CodeBuild: [Jenkins CodeBuild で使用する](#)

- CodeDeploy: [製品とサービスの統合](#)
- CodePipeline: [製品とサービスの統合](#)

## を初めてお使いになる方向けの情報

デベロッパーツールコンソールで利用可能なサービスを初めて使用する場合は、まず以下のトピックを読むことをお勧めします。

- [CodeCommit の開始方法](#)
- [の開始方法 CodeBuild](#)、[概念](#)
- [の開始方法 CodeDeploy](#)、[プライマリコンポーネント](#)
- [の開始方法 CodePipeline](#)、[概念](#)

## デベロッパーツールコンソールの機能

デベロッパーツールコンソールには、以下の機能も含まれます。

- デベロッパーツールコンソールには、AWS CodeBuild AWS CodeCommit AWS CodeDeploy、およびのイベントをサブスクライブするために使用できる通知マネージャー機能が含まれています。AWS CodePipeline。この機能には独自の API である AWS CodeStar Notifications があります。通知機能を使用して、のユーザーに対して、作業に最も重要なレポジトリ、ビルドプロジェクト、デプロイアプリケーション、パイプラインのイベントについてすばやく通知できます。通知マネージャーは、リポジトリ、ビルド、デプロイ、またはパイプラインで発生するイベントをユーザーに認識させ、変更の承認やエラーの修正などのアクションをすばやく実行できるようにします。詳細については、「[通知とは何ですか?](#)」を参照してください。
- デベロッパーツールコンソールには、AWS リソースをサードパーティーのソースコードプロバイダーに関連付けるための接続機能も含まれています。この機能には独自の API があります AWS CodeConnections。接続機能を使用して、サードパーティープロバイダーとの認可された接続を設定し、その接続リソースを他の AWS のサービスで使用できます。詳細については、「[接続とは?](#)」を参照してください。

## 通知とは何ですか?

開発ツールコンソールの通知機能は、AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy、および AWS CodePipeline のイベントをサブスクライブするための通知マネージャーです。独自の

API、AWS CodeStar Notifications があります。通知機能を使用して、のユーザーに対して、作業に最も重要なレポジトリ、ビルドプロジェクト、デプロイアプリケーション、パイプラインのイベントについてすばやく通知できます。通知マネージャーは、リポジトリ、ビルド、デプロイ、またはパイプラインで発生するイベントをユーザーに認識させ、変更の承認やエラーの修正などのアクションをすばやく実行できるようにします。

## 通知でどのようなことができますか？

通知機能を使用して通知ルールを作成および管理することで、リソースに対する以下のような重要な変更をユーザーに通知できます。

- CodeBuild のビルドプロジェクトにおけるビルドの成功と失敗。
- CodeDeploy アプリケーションのデプロイの成功と失敗。
- CodeCommit リポジトリ内のプルリクエスト (コードに対するコメントを含む) の作成と更新。
- CodePipeline での手動による承認のステータスとパイプラインの実行。

通知は、Amazon SNS トピックにサブスクライブしているユーザーの E メールアドレスに配信されるように設定できます。また、この機能を [AWS Chatbot](#) と統合し、Slack チャンネル、Microsoft Teams チャンネル、または Amazon Chime チャットルームに通知を配信することもできます。

## 通知はどのような仕組みで機能しますか？

リポジトリ、ビルドプロジェクト、アプリケーション、またはパイプラインなど、サポートされているリソースに対する通知ルールを設定すると、通知機能は指定されたイベントをモニタリングする Amazon EventBridge ルールを作成します。このタイプのイベントが発生すると、通知ルールはそのルールのターゲットとして指定された Amazon SNS トピックに通知を送信します。これらのターゲットの受信者は、該当するイベントに関する通知を受け取ります。

## 通知の使用を開始する方法

使用を開始するには、次のいくつかのトピックが役立ちます。

- 通知の [概念](#) について説明します。
- 通知の操作を開始するために [必要なリソース](#) を設定します。
- [最初の通知ルール](#) を開始し、最初の通知を受け取ります。



## 通知の概念

概念と用語を理解すれば、通知の設定と使用が容易になります。ここでは、通知を使用する際に知っておかなければならないいくつかの概念を次に示します。

### トピック

- [通知](#)
- [通知ルール](#)
- [イベント](#)
- [詳細タイプ](#)
- [ターゲット](#)
- [通知および AWS CodeStar Notifications](#)
- [リポジトリでの通知ルールのイベント](#)
- [ビルドプロジェクトでの通知ルールのイベント](#)
- [デプロイアプリケーションでの通知ルールのイベント](#)
- [パイプラインでの通知ルールのイベント](#)

### 通知

通知とは、お客様と開発者が使用するリソースで発生するイベントに関する情報を示すメッセージです。ビルドプロジェクト、リポジトリ、デプロイアプリケーション、パイプラインなどのリソースのユーザーに対して、作成した通知ルールに従って、指定したイベントタイプに関する E メールを送信するように通知を設定できます。

セッションタグを使用して、AWS CodeCommit の通知に表示名や電子メールアドレスなどのユーザー ID 情報を含めることができます。CodeCommit では、セッションタグの使用がサポートされています。セッションタグは、IAM ロールを引き受けるとき、一時的な認証情報を使用するとき、または AWS Security Token Service (AWS STS) でユーザーをフェデレートするときに渡すキーと値のペアの属性です。タグを IAM ユーザーに関連付けることもできます。CodeCommit は、displayName と emailAddress のタグが存在する場合、それらの値を通知コンテンツに含めます。詳細については、「[CodeCommit で ID 情報を提供するためのタグの使用](#)」を参照してください。

### ⚠ Important

通知には、ビルドのステータス、デプロイのステータス、コメントのあるコード行、パイプラインの承認など、プロジェクト固有の情報が含まれます。通知の内容は、新機能が追加されると変更されることがあります。セキュリティのベストプラクティスとして、通知ルールのターゲットと Amazon SNS トピックのサブスクライバーを定期的に確認する必要があります。詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

## 通知ルール

通知ルールは、通知を送信するタイミングと送信先を指定するために作成する AWS リソースです。通知ルールでは、以下を定義します。

- 通知の作成条件。これらの条件は、選択したイベントに基づきます。イベントはリソースタイプに固有です。サポートされているリソースタイプには、AWS CodeBuild のビルドプロジェクト、AWS CodeDeploy のデプロイアプリケーション、AWS CodePipeline のパイプライン、AWS CodeCommit のリポジトリなどがあります。
- 通知の送信先のターゲット。通知ルールには最大 10 個のターゲットを指定できます。

通知ルールの送信先は、個別のビルドプロジェクト、デプロイアプリケーション、パイプライン、およびリポジトリです。通知ルールには、ユーザー定義のフレンドリ名と Amazon リソースネーム (ARN) の両方があります。通知ルールは、リソースが存在する AWS リージョンと同じリージョンで作成する必要があります。例えば、ビルドプロジェクトが 米国東部 (オハイオ) リージョンにある場合、通知ルールも 米国東部 (オハイオ) リージョンで作成する必要があります。

1 つのリソースに対して最大 10 個の通知ルールを定義できます。

## イベント

イベントとは、モニタリングするリソースの状態の変化です。各リソースには、選択できるイベントタイプのリストがあります。リソースに通知ルールを設定する際に、発生したときに通知が送信されるイベントを指定します。例えば、CodeCommit でリポジトリの通知を設定し、[Pull request] (プルリクエスト) と [Branches and tags] (ブランチとタグ) の両方で [Created] (作成済み) を選択した場合、そのリポジトリ内のユーザーがプルリクエスト、ブランチ、または Git タグを作成するたびに通知が送信されます。

## 詳細タイプ

通知ルールを作成するとき、通知に含まれる詳細レベルまたは詳細タイプ ([フル] または [ベーシック]) を選択できます。[フル] 設定 (デフォルト) では、通知にあるイベントについて入手可能な情報 (特定のイベントについてサービスから提供される拡張情報も含む) のすべてが含まれます。[ベーシック] 設定では、入手可能な情報のサブセットのみが含まれます。

以下の表では、特定のイベントタイプについて入手可能な拡張情報を一覧表示し、詳細タイプ間の違いについて説明します。

サービス	イベント	フルに含まれる	ベーシックには含まれない
CodeCommit	コミットに関するコメント  プルリクエストに関するコメント	返信やコメントスレッドなど、すべてのイベントの詳細とコメントの内容。コメントが作成された行番号とコード行も含まれます。	コメントの内容、行番号、コード行、コメントスレッド。
CodeCommit	プルリクエストが作成された	すべてのイベントの詳細、および送信先ブランチに関連するプルリクエストで追加、変更、または削除されたファイルの数。	プルリクエストの送信元ブランチによって追加、変更、または削除されたファイルのリストや詳細。
CodePipeline	手動承認を求められた	すべてのイベントの詳細とカスタムデータ (設定されている場合)。通知には、パイプラインで求められる承認へのリンクも含まれます。	カスタムデータまたはリンク。

サービス	イベント	フルに含まれる	ベーシックには含まれない
CodePipeline	アクションの実行に失敗した  パイプラインの実行に失敗した  ステージの実行に失敗した	すべてのイベントの詳細と失敗のエラーメッセージの内容。	エラーメッセージの内容。

## ターゲット

ターゲットとは、通知ルールからの通知が届く場所です。許可されるターゲットタイプは、Slack チャンネルまたは Microsoft Teams チャンネル用に設定された AWS Chatbot クライアント、および Amazon SNS トピックです。ターゲットにサブスクライブしているすべてのユーザーに、通知ルールで指定したイベントに関する通知が送信されます。

通知の配信先を広げたい場合は、通知と AWS Chatbot との統合を手動で設定することで、通知を Amazon Chime チャットルームに送信できます。次に、通知ルールのターゲットとして、その AWS Chatbot クライアント用に設定された Amazon SNS トピックを選択できます。詳細については、「[通知を AWS Chatbot および Amazon Chime と統合するには](#)」を参照してください。

AWS Chatbot クライアントをターゲットとして使用する場合は、最初にこのクライアントを AWS Chatbot で作成する必要があります。通知ルールのターゲットとして AWS Chatbot クライアントを選択すると、その AWS Chatbot クライアント用の Amazon SNS トピックが、Slack チャンネルまたは Microsoft Teams チャンネルへの通知の送信に必要なすべてのポリシーと共に設定されます。既存の Amazon SNS トピックを AWS Chatbot クライアント用に設定する必要はありません。

通知ルールの作成の一環として、Amazon SNS トピックをターゲットとして作成を選択できます (推奨)。通知ルールと同じ AWS リージョンにある既存の Amazon SNS トピックを選択することもできます。ただし、このトピックには、必要なポリシーを設定する必要があります。ターゲットとして使用する Amazon SNS トピックは、AWS アカウント内に存在する必要があります。また、通知ルールやこのルールを作成した対象の AWS リソースと同じ AWS リージョンに存在する必要があります。

例えば、米国東部 ( オハイオ ) リージョンでリポジトリの通知ルールを作成した場合、Amazon SNS トピックもそのリージョンに存在する必要があります。通知ルールの作成の一部として Amazon SNS トピックを作成すると、トピックへのイベントの公開を許可するために必要なポリシーによりトピックが設定されます。これは、ターゲットと通知ルールを操作するのに最適な方法です。既存のトピックを使用するか、手動でトピックを作成する場合は、ユーザーが通知を受け取る前に、必要なアクセス許可でトピックを設定する必要があります。詳細については、「[通知用に Amazon SNS トピックを設定する](#)」を参照してください。

### Note

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーリストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用された場合、このトピックは AWS CodeStar Notifications に必要なアクセス許可とは異なるアクセス許可を含む、CodeCommit の発行を許可するポリシーを含みます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

## 通知および AWS CodeStar Notifications

デベロッパーツールコンソールの機能ですが、通知には独自の API である AWS CodeStar Notifications があります。また、独自の AWS リソースタイプ ( 通知ルール )、アクセス許可、イベントもあります。通知ルールのイベントはログインした AWS CloudTrail です。API アクションは、IAM ポリシーを通じて許可または拒否できます。

### リポジトリでの通知ルールのイベント

カテゴリ	イベント	イベント ID
コメント	コミット時 プルリクエスト時	codecommit-repository- comments-on-commits

カテゴリ	イベント	イベント ID
		codecommit-repository-comments-on-pull-requests
承認	ステータス変更 ルールの上書き	codecommit-repository-approvals-status-changed  codecommit-repository-approvals-rule-override
プルリクエスト	作成  ソース更新  ステータス変更  マージ	codecommit-repository-pull-request-created  codecommit-repository-pull-request-source-updated  codecommit-repository-pull-request-status-changed  codecommit-repository-pull-request-merged
ブランチとタグ	作成  [Deleted] (削除済み)  更新	codecommit-repository-branches-and-tags-created  codecommit-repository-branches-and-tags-deleted  codecommit-repository-branches-and-tags-updated

## ビルドプロジェクトでの通知ルールのイベント

カテゴリ	イベント	イベント ID
ビルド状態	[Failed] (失敗)	codebuild-project-build-state-failed
	成功	codebuild-project-build-state-succeeded
	進行中	codebuild-project-build-state-in-progress
	停止	codebuild-project-build-state-stopped
		codebuild-project-build-phase-failure
ビルドフェーズ	失敗	codebuild-project-build-phase-success
	成功	codebuild-project-build-phase-success

## デプロイアプリケーションでの通知ルールのイベント

カテゴリ	イベント	イベント ID
デプロイ	[Failed] (失敗)	codedeploy-application-deployment-failed
	成功	codedeploy-application-deployment-succeeded
	Started	codedeploy-application-deployment-started

## パイプラインでの通知ルールのイベント

カテゴリ	イベント	イベント ID
アクションの実行	成功	codepipeline-pipeline-action-execution-succeeded
	[Failed] (失敗)	codepipeline-pipeline-action-execution-failed
	キャンセル	codepipeline-pipeline-action-execution-canceled
	Started	codepipeline-pipeline-action-execution-started
		codepipeline-pipeline-action-execution-started
ステージの実行	Started	codepipeline-pipeline-stage-execution-started
	成功	codepipeline-pipeline-stage-execution-succeeded
	再開	codepipeline-pipeline-stage-execution-resumed
	Canceled	codepipeline-pipeline-stage-execution-canceled
	[Failed] (失敗)	codepipeline-pipeline-stage-execution-failed
		codepipeline-pipeline-stage-execution-failed
パイプラインの実行	[Failed] (失敗)	codepipeline-pipeline-pipeline-execution-failed
	キャンセル	codepipeline-pipeline-pipeline-execution-canceled
	Started	codepipeline-pipeline-pipeline-execution-started
	再開	codepipeline-pipeline-pipeline-execution-started
	成功	codepipeline-pipeline-pipeline-execution-started



カテゴリ	イベント	イベント ID
	置換	codepipeline-pipeline-pipeline-execution-resumed codepipeline-pipeline-pipeline-execution-succeeded codepipeline-pipeline-pipeline-execution-superseded
手動の承認	[Failed] (失敗) 必要 成功	codepipeline-pipeline-manual-approval-failed codepipeline-pipeline-manual-approval-needed codepipeline-pipeline-manual-approval-succeeded

## セットアップ

AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy、または IAM ユーザーまたはロールが AWS CodePipeline に適用される マネージドポリシーがある場合は、ポリシーによって提供されるロールとアクセス許可の制限内で通知を操作するために必要なアクセス許可があります。例えば、AWSCodeBuildAdminAccess、AWSCodeCommitFullAccess、AWSCodeDeployFullAccess、または AWSCodePipeline\_FullAccess 管理ポリシーが適用されたユーザーには、通知に対する完全な管理アクセスがあります。

詳細とポリシーの例については、「[アイデンティティベースのポリシー](#)」を参照してください。

これらのポリシーのいずれかが IAM ユーザーまたはロールに適用され、のビルドプロジェクト CodeBuild、のリポジトリ CodeCommit、のデプロイアプリケーション CodeDeploy、またはのパイプラインに適用されている場合は CodePipeline、最初の通知ルールを作成する準備が整います。「[通知の使用開始](#)」に進みます。そうでない場合は、以下のトピックを参照してください。

- CodeBuild: [の開始方法 CodeBuild](#)
- CodeCommit: [の開始方法 CodeCommit](#)
- CodeDeploy: [チュートリアル](#)

- CodePipeline: [の開始方法 CodePipeline](#)

IAM ユーザー、グループ、またはロールの通知の管理アクセス許可を自分で管理する場合は、このトピックの手順に従って、サービスを使用するために必要なアクセス許可とリソースを設定します。

通知専用のトピックを作成する代わりに、以前に作成した Amazon SNS トピックを通知に使用する場合は、そのトピックへのイベントの発行を許可するポリシーを適用して、通知ルールのターゲットとして使用する Amazon SNS トピックを設定する必要があります。

#### Note

以下の手順を実行するには、管理者権限を持つアカウントでサインインする必要があります。詳細については、「[最初の IAM 管理者ユーザーおよびユーザーグループの作成](#)」を参照してください。

## トピック

- [通知への管理アクセスのためのポリシーの作成と適用](#)
- [通知用に Amazon SNS トピックを設定する](#)
- [ターゲットである Amazon SNS トピックへのユーザーのサブスクライブ](#)

## 通知への管理アクセスのためのポリシーの作成と適用

通知を管理するには、IAM ユーザーでサインインするか、通知を作成するサービスおよびサービス (AWS CodeBuild AWS CodeCommit、AWS CodeDeploy、または AWS CodePipeline) へのアクセス許可を持つロールを使用します。独自のポリシーを作成し、ユーザーまたはグループに適用することもできます。

次の手順では、通知を管理し、IAM ユーザーを追加するアクセス許可を持つ IAM グループを設定する方法を示します。グループをセットアップしない場合は、このポリシーを IAM ユーザーに直接適用するか、ユーザーが引き受けることができる IAM ロールに直接適用できます。CodeBuild、CodeCommit CodeDeploy、またはのマネージドポリシーを使用することもできます。これには CodePipeline、ポリシーの範囲に応じて通知機能へのポリシーに適したアクセスが含まれます。

以下のポリシーに、このポリシーの名前 (例: `AWSCodeStarNotificationsFullAccess`) と説明 (省略可能) を入力します。この説明は、ポリシーの目的を思い出すのに役立ちます (例: **This policy provides full access to AWS CodeStar Notifications.**)。

## JSON ポリシーエディタでポリシーを作成するには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. 左側のナビゲーションペインで、[ポリシー] を選択します。

初めて [ポリシー] を選択する場合には、[管理ポリシーによるこそ] ページが表示されます。[今すぐ始める] を選択します。

3. ページの上部で、[ポリシーを作成] を選択します。
4. [ポリシーエディタ] セクションで、[JSON] オプションを選択します。
5. 次の JSON ポリシードキュメントを入力します。

```
{
  "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. [次へ] をクリックします。

**Note**

いつでも [Visual] と [JSON] エディタオプションを切り替えることができます。ただし、[Visual] エディタで [次] に変更または選択した場合、IAM はポリシーを再構成して visual エディタに合わせて最適化することがあります。詳細については、「IAM ユーザーガイド」の「[ポリシーの再構成](#)」を参照してください。

7. [確認と作成] ページで、作成するポリシーの [ポリシー名] と [説明] (オプション) を入力します。[このポリシーで定義されているアクセス許可] を確認して、ポリシーによって付与されたアクセス許可を確認します。
8. [ポリシーの作成] をクリックして、新しいポリシーを保存します。

## 通知用に Amazon SNS トピックを設定する

通知を設定する最も簡単な方法は、通知ルールを作成するときに Amazon SNS トピックを作成することです。以下の要件を満たしている場合は、既存の Amazon SNS トピックを使用できます。

- これは、通知ルールを作成するリソース (ビルドプロジェクト、デプロイアプリケーション、リポジトリ、またはパイプライン) AWS リージョンと同じに作成されました。
- 2019 年 11 月 5 日より CodeCommit 前に の通知を送信するために使用されていません。使用している場合は、その機能を有効にしたポリシーステートメントが含まれます。このトピックを使用することもできますが、手順で指定されているように、追加のポリシーを追加する必要があります。2019 年 11 月 5 日より前に通知用に 1 つ以上のリポジトリが設定されている場合は、既存のポリシーステートメントを削除しないでください。
- これには、AWS CodeStar 通知がトピックに通知を発行することを許可するポリシーがあります。

AWS CodeStar 通知通知ルールのターゲットとして使用するよう Amazon SNS トピックを設定するには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/sns/v3/home> で Amazon SNS コンソールを開きます。
2. ナビゲーションバーで、[トピック] を選択し、設定するトピックを選択して、[編集] を選択します。
3. [アクセスポリシー]を展開し、アドバンストを選択します。

- JSON エディタで、ポリシーに次のポリシーステートメントを追加します。トピック ARN、AWS リージョン、AWS アカウント ID、トピック名を含めます。

```
{
  "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"
}
```

このポリシーステートメントは、次のようになります。

```
{
  "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. [変更を保存] を選択します。
6. で AWS KMS暗号化された Amazon SNS トピックを使用して通知を送信する場合は、次のステートメントを のポリシーに追加して、イベントソース (AWS CodeStar 通知) と暗号化されたトピックとの互換性も有効にする必要があります AWS KMS key。 ( AWS リージョン この例では us-east-2) を、キーが作成された に置き換え AWS リージョン ます。

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

```
    }  
  ]  
}
```

詳細については、「[保管時の暗号化](#)」および「[AWS KMSでのポリシー条件の使用](#)」のAWS Key Management Service デベロッパーガイドを参照してください。

## ターゲットである Amazon SNS トピックへのユーザーのサブスクライブ

ユーザーが通知を受信できるようにするには、通知ルールのターゲットである Amazon SNS トピックにサブスクライブする必要があります。ユーザーが E メールアドレスでサブスクライブしている場合、通知を受け取る前にサブスクリプションを確認する必要があります。Slack チャンネル、Microsoft Teams チャンネル、または Amazon Chime チャットルームのユーザーに通知を送信するには、「[通知と AWS Chatbot との統合の設定](#)」を参照してください。

通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/sns/v3/home> で Amazon SNS コンソールを開きます。
2. ナビゲーションバーで、[トピック] を選択し、ユーザーをサブスクライブするトピックを選択します。
3. [サブスクリプション] で、[サブスクリプションの作成] を選択します。
4. [プロトコル] で、[E メール] を選択します。[エンドポイント] にメールアドレスを入力し、[サブスクリプションの作成] を選択します。

## 通知の使用開始

通知の使用を開始する最も簡単な方法は、ビルドプロジェクト、デプロイアプリケーション、パイプライン、またはリポジトリのいずれかに通知ルールを設定することです。

### Note

通知ルールを初めて作成すると、サービスにリンクされたロールがアカウントに作成されます。詳細については、「[AWS CodeStar 通知のサービスにリンクされたロールの使用](#)」を参照してください。

## トピック

- [前提条件](#)
- [リポジトリの通知ルールを作成する](#)
- [ビルドプロジェクトの通知ルールを作成する](#)
- [デプロイアプリケーションの通知ルールを作成する](#)
- [パイプラインの通知ルールを作成する](#)

## 前提条件

[セットアップ](#) のステップを完了します。通知ルールを作成するリソースも必要です。

- [CodeBuild でビルドプロジェクトを作成](#)するか、既存のプロジェクトを使用します。
- [アプリケーションを作成](#)するか、既存のデプロイアプリケーションを使用します。
- [CodePipeline でパイプラインを作成](#)するか、既存のパイプラインを使用します。
- [AWS CodeCommit リポジトリを作成](#)するか、既存のリポジトリを使用します。

## リポジトリの通知ルールを作成する

通知ルールを作成して、重要なリポジトリイベントに関する通知を送信できます。以下のステップは、単一のリポジトリイベントに関する通知ルールを設定する方法を示しています。これらの手順は、AWS アカウントにリポジトリが設定されていることを前提としています。

### Important

2019 年 11 月 5 日より前に CodeCommit で通知を設定すると、それらの通知に使用される Amazon SNS トピックには、トピックへの発行を CodeCommit に許可し、AWS CodeStar Notifications に必要なアクセス許可とは異なるアクセス許可を含むポリシーが含まれます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

1. <https://console.aws.amazon.com/codecommit/> で CodeCommit コンソールを開きます。
2. リストからリポジトリを選択して開きます。



3. [Notify (通知)], [Create notification rule (通知ルールの作成)] の順に選択します。[設定]、[通知]、[通知ルールの作成] の順に選択することもできます。
4. [通知名] に、ルールの名前を入力します。
5. Amazon EventBridge に提供された情報のみを通知に含める場合は、[Detail type (詳細タイプ)] で [Basic (基本)] を選択します。Amazon EventBridge に提供される情報に加えて、リソースサービスまたは通知マネージャから提供される場合がある情報も含める場合は、[Full] (完全) を選択します。

詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

6. [Events that trigger notifications (通知をトリガーするイベント)] の [ブランチとタグ] で、[作成済み] を選択します。
7. [ターゲット] で、[SNS トピックの作成] を選択します。

#### Note

通知ルールの作成の一環としてトピックを作成すると、CodeCommit にトピックへのイベントの発行を許可するポリシーが適用されます。通知ルール用に作成されたトピックを使用すると、このリポジトリに関する通知の受信を希望するユーザーのみをサブスクライブできます。

[codestar-notifications-] プレフィックスの後にトピックの名前を入力し、[送信] を選択します。

#### Note

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用された場合、このトピックは AWS CodeStar Notifications に必要な許可とは異なるアクセス許可を含む、CodeCommit の発行を許可するポリシーを含みます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

8. [送信] を選択し、通知ルールを確認します。
9. 自分のメールアドレスを作成した Amazon SNS トピックにサブスクライブします。詳細については、「[通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには](#)」を参照してください。
10. リポジトリに移動し、デフォルトブランチからテストブランチを作成します。
11. ブランチを作成すると、通知ルールによって、そのイベントに関する情報を含む通知がすべてのトピックサブスクライバーに送信されます。

## ビルドプロジェクトの通知ルールを作成する

通知ルールを作成して、ビルドプロジェクトでの重要なイベントに関する通知を送信できます。以下のステップは、単一のビルドプロジェクトイベントに関する通知ルールを設定する方法を示しています。これらの手順は、AWS アカウントにビルドプロジェクトが設定されていることを前提としています。

1. CodeBuild コンソール (<https://console.aws.amazon.com/codebuild/>) を開きます。
2. リストからビルドプロジェクトを選択して開きます。
3. [Notify (通知)]、[Create notification rule (通知ルールの作成)] の順に選択します。[設定]、[通知ルールの作成] の順に選択することもできます。
4. [通知名] に、ルールの名前を入力します。
5. Amazon EventBridge に提供された情報のみを通知に含める場合は、[Detail type (詳細タイプ)] で [Basic (基本)] を選択します。Amazon EventBridge に提供される情報に加えて、リソースサービスまたは通知マネージャから提供される場合がある情報も含める場合は、[Full] (完全) を選択します。

詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

6. [Events that trigger notifications (通知をトリガーするイベント)] の [ビルドフェーズ] で、[成功] を選択します。
7. [ターゲット] で、[SNS トピックの作成] を選択します。

### Note

通知ルールの作成の一環としてトピックを作成すると、CodeBuild にトピックへのイベントの発行を許可するポリシーが適用されます。通知ルール用に作成されたトピックを

使用すると、このビルドプロジェクトに関する通知の受信を希望するユーザーのみをサブスクライブできます。

[codestar-notifications-] プレフィックスの後にトピックの名前を入力し、[送信] を選択します。

#### Note

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーリストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用された場合、このトピックは AWS CodeStar Notifications に必要な許可とは異なるアクセス許可を含む、CodeCommit の発行を許可するポリシーを含みます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

- [送信] を選択し、通知ルールを確認します。
- 自分のメールアドレスを作成した Amazon SNS トピックにサブスクライブします。詳細については、「[通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには](#)」を参照してください。
- ビルドプロジェクトに移動し、ビルドを開始します。
- ビルドフェーズが正常に完了すると、通知ルールは、そのイベントに関する情報を含む通知をすべてのトピックサブスクライバーに送信します。

## デプロイアプリケーションの通知ルールを作成する

通知ルールを作成して、デプロイアプリケーションでの重要なイベントに関する通知を送信できます。以下のステップは、単一のビルドプロジェクトイベントに関する通知ルールを設定する方法を示しています。これらの手順は、AWS アカウントにデプロイアプリケーションが設定されていることを前提としています。

- CodeDeploy コンソールは次の URL で開きます。 <https://console.aws.amazon.com/codedeploy/>

2. リストからアプリケーションを選択して開きます。
3. [Notify (通知)]、[Create notification rule (通知ルールの作成)] の順に選択します。[設定]、[通知ルールの作成] の順に選択することもできます。
4. [通知名] に、ルールの名前を入力します。
5. Amazon EventBridge に提供された情報のみを通知に含める場合は、[Detail type (詳細タイプ)] で [Basic (基本)] を選択します。Amazon EventBridge に提供される情報に加えて、リソースサービスまたは通知マネージャから提供される場合がある情報も含める場合は、[Full] (完全) を選択します。

詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

6. [Events that trigger notifications (通知をトリガーするイベント)] の [デプロイ] で、[成功] を選択します。
7. [ターゲット] で、[SNS トピックの作成] を選択します。

#### Note

通知ルールの作成の一環としてトピックを作成すると、CodeDeploy にトピックへのイベントの発行を許可するポリシーが適用されます。通知ルール用に作成されたトピックを使用すると、このデプロイアプリケーションに関する通知の受信を希望するユーザーのみをサブスクライブできます。

[codestar-notifications-] プレフィックスの後にトピックの名前を入力し、[送信] を選択します。

#### Note

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーリストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用された場合、このトピックは AWS CodeStar Notifications に必要な許可とは異なるアクセス許可を含む、CodeCommit の発行を許可するポリシーを含みます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon](#)

[SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

8. [送信] を選択し、通知ルールを確認します。
9. 自分のメールアドレスを作成した Amazon SNS トピックにサブスクライブします。詳細については、「[通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには](#)」を参照してください。
10. デプロイアプリケーションに移動し、デプロイを開始します。
11. デプロイが成功すると、通知ルールによって、そのイベントに関する情報を含む通知がすべてのトピックサブスクライバーに送信されます。

## パイプラインの通知ルールを作成する

通知ルールを作成して、パイプラインの重要なイベントに関する通知を送信できます。以下のステップは、単一のパイプラインイベントに関する通知ルールを設定する方法を示しています。これらの手順は、AWS アカウントにパイプラインが設定されていることを前提としています。

1. CodePipeline コンソールは次の URL で開きます。 <https://console.aws.amazon.com/codesuite/codepipeline/home>
2. リストからパイプラインを選択して開きます。
3. [Notify (通知)]、[Create notification rule (通知ルールの作成)] の順に選択します。[設定]、[通知ルールの作成] の順に選択することもできます。
4. [通知名] に、ルールの名前を入力します。
5. Amazon EventBridge に提供された情報のみを通知に含める場合は、[Detail type (詳細タイプ)] で [Basic (基本)] を選択します。Amazon EventBridge に提供される情報に加えて、リソースサービスまたは通知マネージャから提供される場合がある情報も含める場合は、[Full] (完全) を選択します。

詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

6. [Events that trigger notifications (通知をトリガーするイベント)] の [アクションの実行] で、[開始済] を選択します。
7. [ターゲット] で、[SNS トピックの作成] を選択します。

**Note**

通知ルールの作成の一環としてトピックを作成すると、CodePipeline にトピックへのイベントの発行を許可するポリシーが適用されます。通知ルール用に作成されたトピックを使用すると、このパイプラインに関する通知の受信を希望するユーザーのみをサブスクライブできます。

[codestar-notifications-] プレフィックスの後にトピックの名前を入力し、[送信] を選択します。

**Note**

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーリストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用された場合、このトピックは AWS CodeStar Notifications に必要なアクセス許可とは異なるアクセス許可を含む、CodeCommit の発行を許可するポリシーを含みます。これらのトピックの使用は非推奨です。そのような経緯で作成されたトピックの使用を求める場合、必要なポリシーをその他の既存のポリシーに加えて AWS CodeStar Notifications に追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

8. [送信] を選択し、通知ルールを確認します。
9. 自分のメールアドレスを作成した Amazon SNS トピックにサブスクライブします。詳細については、「[通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには](#)」を参照してください。
10. パイプラインに移動し、[Release change (変更のリリース)] を選択します。
11. アクションが開始されると、通知ルールによって、そのイベントに関する情報を含む通知がすべてのトピックサブスクライバーに送信されます。

## 通知ルールの使用

通知ルールでは、ユーザーに通知するイベントを設定し、これらの通知を受け取るターゲットを指定します。通知は、Amazon SNS を介するか、Slack チャンネルまたは Microsoft Teams チャンネル用に設定された AWS Chatbot クライアントを介してユーザーに直接送信できます。通知の配信先を広げたい場合は、通知と AWS Chatbot との統合を手動で設定することで、通知を Amazon Chime チャットルームに送信できます。詳細については、[ターゲット](#) および [通知を AWS Chatbot および Amazon Chime と統合するには](#) を参照してください。




# Create notification rule

Notification rules set up a subscription to events that happen with your resources. When these events occur, you will receive notifications sent to the targets you designate. You can manage your notification preferences in Settings. [Info](#)

## Notification rule settings

Notification name

Detail type

Choose the level of detail you want in notifications. [Learn more about notifications and security](#) 

**Full**  
Includes any supplemental information about events provided by the resource or the notifications feature.

**Basic**  
Includes only information provided in resource events.

## Events that trigger notifications

Comments

On commits  
 On pull requests

Approvals

Status changed  
 Rule override


Pull request

Source updated  
 Created  
 Status changed  
 Merged

Branches and tags

Created  
 Deleted  
 Updated

## Targets

Choose a target type for the notification rule. SNS topics can be created specifically for use with the notification rule, or existing topics can be modified for use with notifications. AWS Chatbot clients for Slack integration must be created before you can choose them as a target type. [Learn more](#) 

デベロッパーツールコンソールまたは AWS CLI を使用して、通知ルールを作成および管理できます。

トピック

- [通知ルールの作成](#)



- [通知ルールの表示](#)
- [通知ルールの編集](#)
- [通知ルールの通知の有効化または無効化](#)
- [通知ルールの削除](#)

## 通知ルールの作成

デベロッパーツールコンソールまたは AWS CLI を使用して、通知ルールを作成できます。通知ルールの作成の一環として、通知ルールのターゲットとして使用する Amazon SNS トピックを作成できます。AWS Chatbot クライアントをターゲットとして使用する場合は、通知ルールを作成する前に、そのクライアントを作成する必要があります。詳細については、「[Slack チャンネルの AWS Chatbot クライアントの設定](#)」を参照してください。

通知ルールを作成するには (コンソール)

1. AWS デベロッパーツールコンソールは、次の URL で開きます。<https://console.aws.amazon.com/codesuite/settings/notifications>
2. ナビゲーションバーを使用して、リソースに移動します。
  - CodeBuild では、[Build] (ビルド)、[Build projects] (ビルドプロジェクト) の順に選択し、ビルドプロジェクトを選択します。
  - CodeCommit では、[Source] (ソース)、[Repositories] (リポジトリ) の順に選択し、リポジトリを選択します。
  - CodeDeploy では、[アプリケーション] を選択し、アプリケーションを選択します。
  - CodePipeline では、[Pipeline] (パイプライン)、[Pipelines] (パイプライン) の順に選択し、パイプラインを選択します。
3. リソースページで、[Notify (通知)]、[Create notification rule (通知ルールの作成)] の順に選択します。リソースの [設定] ページの [通知] または [通知ルール] に移動し、[通知ルールの作成] を選択することもできます。
4. [通知名] に、ルールの名前を入力します。
5. Amazon EventBridge に提供された情報のみを通知に含める場合は、[Detail type (詳細タイプ)] で [Basic (基本)] を選択します。Amazon EventBridge に提供される情報に加えて、リソースサービスまたは通知マネージャから提供される場合がある情報も含める場合は、[Full] (完全) を選択します。

詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

6. [Events that trigger notifications (通知をトリガーするイベント)] で、通知を送信するイベントを選択します。リソースのイベントタイプについては、以下を参照してください。
  - CodeBuild: [ビルドプロジェクトでの通知ルールのイベント](#)
  - CodeCommit: [リポジトリでの通知ルールのイベント](#)
  - CodeDeploy: [デプロイアプリケーションでの通知ルールのイベント](#)
  - CodePipeline: [パイプラインでの通知ルールのイベント](#)
7. [Targets (ターゲット)] で、次のいずれかの操作を行います。
  - 通知で使用するリソースを設定済みである場合は、[ターゲットタイプを選択] で、[AWS Chatbot (Slack)]、[AWS Chatbot (Microsoft Teams)]、または [SNS トピック] を選択します。[ターゲットを選択] で、クライアントの名前 (AWS Chatbot で設定された Slack や Microsoft Teams クライアントの場合) を選択するか、Amazon SNS トピックの Amazon リソースネーム (ARN) (通知に必要なポリシーと共に設定済みである Amazon SNS トピックの場合) を選択します。
  - 通知で使用するリソースを設定していない場合は、[Create target]、[SNS topic] の順に選択します。codestar-notifications- の後にトピックの名前を指定し、[Create] を選択します。

#### Note

- 通知ルールの作成の一環として Amazon SNS トピックを作成すると、トピックへのイベント発行を通知機能に許可するポリシーが適用されます。通知ルール用に作成したトピックを使用すると、このリソースに関する通知を受信するユーザーのみをサブスクライブできます。
- 通知ルールの作成の一環として AWS Chatbot クライアントを作成することはできません。AWS Chatbot (Slack) または AWS Chatbot (Microsoft Teams) を選択すると、AWS Chatbot でクライアントを設定するように促すボタンが表示されます。このオプションを選択すると、AWS Chatbot コンソールが開きます。詳細については、「[Slack チャンネルの AWS Chatbot クライアントの設定](#)」を参照してください。
- 既存の Amazon SNS トピックをターゲットとして使用する場合は、このトピック用の他のすべてのポリシーに加えて、AWS CodeStar Notifications に必要なポリシーを追加する必要があります。詳細については、[通知用に Amazon SNS トピックを設定する](#) および [通知の内容とセキュリティについて](#) を参照してください。

## 8. [送信] を選択し、通知ルールを確認します。

### Note

ユーザーは、通知を受け取る前に、ルールのターゲットとして指定した Amazon SNS トピックにサブスクライブしてサブスクライブを確認する必要があります。詳細については、「[通知に使用する Amazon SNS トピックにユーザーをサブスクライブするには](#)」を参照してください。

### 通知ルールを作成するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、`create-notification rule` コマンドを実行して JSON スケルトンを生成します。

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

ファイルには任意の名前を付けることができます。この例では、ファイルの名前を *rule.json* とします。

2. プレーンテキストエディタで JSON ファイルを開き、これを編集してルールに必要なリソース、イベントタイプ、および Amazon SNS ターゲットを含めます。

次の例は、ID *123456789012* の AWS アカウントにある *MyDemoRepo* というリポジトリのための **MyNotificationRule** という通知ルールを示します。ブランチとタグが作成されると、完全な詳細タイプの通知は、*MyNotificationTopic* という Amazon SNS トピックに送信されます。

```
{  
  "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"
}
```

ファイルを保存します。

3. 先ほど編集したファイルを使用して、ターミナルまたはコマンドラインで `create-notification-rule` コマンドを再度実行し、通知ルールを作成します。

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

4. 成功すると、次に示すような通知ルールの ARN がコマンドから返されます。

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

通知ルールのイベントタイプを一覧表示するには (AWS CLI)

1. ターミナルまたはコマンドラインプロンプトで、`list-event-types` コマンドを実行します。 `--filters` オプションを使用して、応答を特定のリソースタイプまたは他の属性に制限できます。例えば、次のコマンドは CodeDeploy アプリケーションのイベントタイプのリストを返します。

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

2. このコマンドでは、次のような出力が生成されます。

```
{
  "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"
}
]
```

## 通知ルールにタグを追加するには (AWS CLI)

1. ターミナルまたはコマンドラインプロンプトで、`tag-resource` コマンドを実行します。例えば、次のコマンドを使用して、*Team* という名前と *Li\_Juan* という値を持つタグキーと値のペアを追加します。

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

2. このコマンドでは、次のような出力が生成されます。

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

## 通知ルールの表示

デベロッパーツールコンソールまたは AWS CLI を使用して、AWS リージョン内のすべてのリソースの通知ルールをすべて表示できます。各通知ルールの詳細を表示することもできます。通知ルールを作成するプロセスとは異なり、リソースのリソースページに移動する必要はありません。

## 通知ルールを表示するには (コンソール)

1. AWS デベロッパーツールコンソールは、次の URL で開きます。 <https://console.aws.amazon.com/codesuite/settings/notifications>
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。
3. [Notification rules] (通知ルール) で、現在サインインしている AWS リージョンで AWS アカウントのリソースに設定されているルールのリストを確認します。セレクトタを使用して AWS リージョンを変更します。
4. 通知ルールの詳細を表示するには、リストからルールを選択し、[詳細を表示] を選択します。リストで名前を選択することもできます。

## 通知ルールのリストを表示するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、list-notification-rules コマンドを実行し、指定した AWS リージョンのすべての通知ルールを表示します。

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

2. 成功すると、次に示すように AWS リージョンの通知ルールごとの ID と ARN がコマンドから返されます。

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

## 通知ルールの詳細を表示するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、describe-notification-rule コマンドを実行します。実行する際に通知ルールの ARN を指定します。

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

2. 成功すると、コマンドは以下のような出力を返します。

```
{
  "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"
}
```

## 通知ルールのタグのリストを表示するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、`list-tags-for-resource` コマンドを実行し、指定した通知ルール ARN のすべてのタグを表示します。

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

2. 正常に完了した場合、このコマンドは以下のような出力を返します。

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

## 通知ルールの編集

通知ルールを編集して、その名前、通知を送信する対象のイベント、詳細タイプまたは通知の送信先のターゲットを変更できます。デベロッパーツールコンソールまたは AWS CLI を使用して、通知ルールを編集できます。

### 通知ルールを編集するには (コンソール)

1. AWS デベロッパーツールコンソールは、次の URL で開きます。 <https://console.aws.amazon.com/codesuite/settings/notifications>
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。
3. [Notification rules] (通知ルール) で、現在サインインしている AWS リージョンで AWS アカウントのリソースに設定されているルールを確認します。セレクトタを使用して AWS リージョンを変更します。
4. リストからルールを選択し、[編集] を選択します。変更を行ってから、[送信] を選択します。

### 通知ルールを編集するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、 [describe-notification-rule コマンド](#) を実行し、通知ルールの構造を表示します。



2. `update-notification rule` コマンドを実行して JSON スケルトンを生成し、それをファイルに保存します。

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

ファイルには任意の名前を付けることができます。この例では、ファイルは *update.json* です。

3. プレーンテキストエディタで JSON ファイルを開き、そのルールを変更します。

次の例は、ID *123456789012* の AWS アカウントにある *MyDemoRepo* というリポジトリのための *MyNotificationRule* という通知ルールを示します。ブランチとタグが作成されると、通知は、*MyNotificationTopic* という Amazon SNS トピックに送信されます。ルール名は、*MyNewNotificationRule* に変更されます。

```
{  
  "Name": "MyNewNotificationRule",  
  "EventIds": [  
    "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"  
}
```

ファイルを保存します。

4. 先ほど編集したファイルを使用して、ターミナルまたはコマンドラインで `update-notification rule` コマンドを再度実行し、通知ルールを更新します。

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

5. 成功すると、次に示すような通知ルールの Amazon リソースネーム (ARN) がコマンドから返されます。

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

### 通知ルールからタグを削除するには (AWS CLI)

1. ターミナルまたはコマンドラインプロンプトで、`untag-resource` コマンドを実行します。例えば、次のコマンドは *Team* という名前のタグを削除します。

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

2. 成功すると、このコマンドは何も返しません。

以下も参照してください。

- [通知ルールのターゲットの追加または削除](#)
- [通知ルールの通知の有効化または無効化](#)
- [イベント](#)

### 通知ルールの通知の有効化または無効化

通知ルールを作成すると、通知はデフォルトで有効になります。ルールを削除して通知を送信しないようにする必要はありません。通知ステータスを変更するだけです。

#### 通知ルールの通知ステータスを変更するには (コンソール)

1. AWS デベロッパーツールコンソールは、次の URL で開きます。 <https://console.aws.amazon.com/codesuite/settings/notifications>
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。

3. [Notification rules] (通知ルール) で、現在サインインしている AWS リージョン で AWS アカウントのリソースに設定されているルールを確認します。セレクタを使用して AWS リージョンを変更します。
4. 有効または無効にする通知ルールを見つけ、そのルールを選択して詳細を表示します。
5. Notification (通知) ステータスで、スライダーを選択してルールのステータスを変更します。
  - [通知を送信する]: これがデフォルト値です。
  - [Notifications paused (通知が一時停止されました)]: 指定されたターゲットに通知は送信されません。

通知ルールの通知ステータスを変更するには (AWS CLI)

1. [通知ルールを編集するには \(AWS CLI\)](#) の手順に従って、通知ルールの JSON を取得します。
2. [Status] フィールドを [ENABLED] ( デフォルト ) または [DISABLED] ( 通知なし ) に編集し、update-notification-rule コマンドを実行してステータスを変更します。

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

## 通知ルールの削除

リソースに対して設定できる通知ルールは 10 個のみであるため、不要になったルールは削除することを検討してください。デベロッパーツールコンソールまたは AWS CLI を使用して、通知ルールを削除できます。

### Note

通知ルールの削除を元に戻すことはできませんが、再作成することはできます。通知ルールを削除しても、ターゲットは削除されません。

通知ルールを削除するには (コンソール)

1. AWS デベロッパーツールコンソールは、次の URL で開きます。 <https://console.aws.amazon.com/codesuite/settings/notifications>
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。

3. [Notification rules] (通知ルール) で、現在サインインしている AWS リージョンで AWS アカウントのリソースに設定されているルールを確認します。セレクタを使用して AWS リージョンを変更します。
4. 通知ルールを選択し、[削除] を選択します。
5. 「delete」と入力後、[削除] を選択します。

### 通知ルールを削除するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、delete-notification-rule コマンドを実行します。実行する際に通知ルールの ARN を指定します。

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

2. 成功すると、次に示すように、削除された通知ルールの ARN がコマンドから返されます。

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

## 通知ルールのターゲットの使用

通知ルールのターゲットとは送信先であり、通知ルールのイベント条件が満たされたときに通知を送信する先を定義します。Slack または Microsoft Teams チャンネル用に設定された Amazon SNS トピックと AWS Chatbot クライアントを選択できます。通知ルールの作成の一環として、Amazon SNS トピックをターゲットとして作成できます (推奨)。通知ルールと同じ AWS リージョンにある既存の Amazon SNS トピックを選択することもできますが、必要なポリシーで設定する必要があります。AWS Chatbot クライアントをターゲットとして使用する場合は、まず Chatbot AWS でそのクライアントを作成する必要があります。

通知の到達範囲を拡張する場合は、通知と AWS Chatbot の統合を手動で設定して、通知が Amazon Chime チャットルームに送信されるようにできます。その後、その AWS Chatbot クライアント用に設定された Amazon SNS トピックを通知ルールのターゲットとして選択できます。詳細については、「[通知を AWS Chatbot および Amazon Chime と統合するには](#)」を参照してください。

デベロッパーツールコンソールまたは [awscli](#) を使用して AWS CLI、通知ターゲットを管理できます。コンソールまたは [awscli](#) を使用して AWS CLI、Amazon SNS トピックと AWS Chatbot クライアントを [ター](#)

[ゲット](#) として作成および設定できます。ターゲットとして設定した Amazon SNS トピックと AWS Chatbot の統合を設定することもできます。これにより、Amazon Chime チャットルームに通知を送信できます。詳細については、「[通知と AWS Chatbot との統合の設定](#)」を参照してください。

## トピック

- [通知ルールのターゲットの作成または設定](#)
- [通知ルールのターゲットの表示](#)
- [通知ルールのターゲットの追加または削除](#)
- [通知ルールのターゲットの削除](#)

## 通知ルールのターゲットの作成または設定

通知ルールのターゲットは、Slack または Microsoft Teams チャンnel用に設定された Amazon SNS トピックまたは AWS Chatbot クライアントです。

ターゲットとしてクライアントを選択する前に、AWS Chatbot クライアントを作成する必要があります。AWS Chatbot クライアントを通知ルールのターゲットとして選択すると、その AWS Chatbot クライアントに対して Amazon SNS トピックが設定され、通知が Slack または Microsoft Teams チャンnelに送信されるために必要なすべてのポリシーが設定されます。既存の Amazon SNS トピックを AWS Chatbot クライアント用に設定する必要はありません。

通知ルールを作成するときに、デベロッパーツールコンソールで Amazon SNS 通知ルールターゲットを作成できます。そのトピックへの通知の送信を許可するポリシーが適用されます。これは、通知ルールのターゲットを作成する最も簡単な方法です。詳細については、「[通知ルールの作成](#)」を参照してください。

既存の Amazon SNS トピックを使用する場合は、リソースがそのトピックに通知を送信できるようにするアクセスポリシーを使用して設定する必要があります。例については、[通知用に Amazon SNS トピックを設定する](#)を参照してください。

### Note

新しいトピックを作成する代わりに既存の Amazon SNS トピックを使用する場合は、[Targets (ターゲット)] でその ARN を選択します。トピックに適切なアクセスポリシーがあり、リソースに関する情報を表示できるユーザーのみがサブスクライバーリストに含まれていることを確認します。Amazon SNS トピックが 2019 年 11 月 5 日より前に CodeCommit 通知に使用されたトピックである場合、AWS CodeStar が通知に必要なアクセス許可とは異

なるアクセス許可を含む公開 CodeCommit を許可するポリシーが含まれます。これらのトピックの使用は非推奨です。そのエクスペリエンス用に作成されたポリシーを使用する場合は、既に存在するポリシーに加えて、AWS CodeStar 通知に必要なポリシーを追加する必要があります。詳細については、「[通知用に Amazon SNS トピックを設定する](#)」および「[通知の内容とセキュリティについて](#)」を参照してください。

通知の到達範囲を拡張する場合は、通知と AWS Chatbot の統合を手動で設定して、通知が Amazon Chime チャットルームに送信されるようにできます。詳細については、「[ターゲット](#)」および「[通知を AWS Chatbot および Amazon Chime と統合するには](#)」を参照してください。

通知ルールのターゲットとして使用する既存の Amazon SNS トピックを設定するには (コンソール)

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/sns/v3/home> で Amazon SNS コンソールを開きます。
2. ナビゲーションバーで、[トピック] を選択します。トピックを選択し、[編集] を選択します。
3. [アクセスポリシー] を展開し、アドバンストを選択します。
4. JSON エディタで、ポリシーに次のポリシーステートメントを追加します。トピック ARN、AWS リージョン、AWS アカウント ID、トピック名を含めます。

```
{
  "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"
}
```

このポリシーステートメントは、次のようになります。

```
{
  "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. [変更を保存] を選択します。
6. [サブスクリプション] で、トピックサブスクライバーのリストを確認します。この通知ルールのターゲットに合わせて、受信者を追加、編集、または削除します。サブスクライバーのリストに

は、リソースに関する情報を表示できるユーザーだけが記載されていることを確認します。詳細については、「[通知の内容とセキュリティについて](#)」を参照してください。

ターゲットとして使用する Slack で AWS Chatbot クライアントを作成するには

1. 「AWS Chatbot 管理者ガイド」の「[AWS Chatbot を Slack で設定する](#)」の手順に従ってください。この場合、通知との統合を最適化するために以下の選択肢を検討してください。
  - IAM ロールを作成するときに、このロールの目的を端的に示すロール名 (**AWSCodeStarNotifications-Chatbot-Slack-Role** など) を選択します。これにより、以後、ロールの使用目的がわかりやすくなります。
  - SNS トピック では、トピックまたは AWS リージョンを選択する必要はありません。AWS Chatbot クライアントを **ターゲット** として選択すると、通知ルールの作成プロセスの一環として、必要なすべてのアクセス許可を持つ Amazon SNS トピックが AWS Chatbot クライアント用に作成および設定されます。
2. クライアントの作成プロセスを完了します。通知ルールの作成時に、このクライアントをターゲットとして選択できます。詳細については、「[通知ルールの作成](#)」を参照してください。

 Note

設定後に Chatbot クライアントから Amazon SNS AWS トピックを削除しないでください。削除すると、Slack に通知が送信されなくなります。

ターゲットとして使用する Microsoft Teams で AWS Chatbot クライアントを作成するには

1. 「AWS Chatbot 管理者ガイド」の「[AWS Chatbot を Microsoft Teams で設定する](#)」の手順に従ってください。この場合、通知との統合を最適化するために以下の選択肢を検討してください。
  - IAM ロールを作成するときに、このロールの目的を端的に示すロール名 (**AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role** など) を選択します。これにより、以後、ロールの使用目的がわかりやすくなります。
  - SNS トピック では、トピックまたは AWS リージョンを選択する必要はありません。AWS Chatbot クライアントを **ターゲット** として選択すると、通知ルールの作成プロセスの一環として、必要なすべてのアクセス許可を持つ Amazon SNS トピックが AWS Chatbot クライアント用に作成および設定されます。



2. クライアントの作成プロセスを完了します。通知ルールの作成時に、このクライアントをターゲットとして選択できます。詳細については、「[通知ルールの作成](#)」を参照してください。

#### Note

設定後に Chatbot クライアントから Amazon SNS AWS トピックを削除しないでください。削除すると、Microsoft Teams に通知が送信されなくなります。

## 通知ルールのターゲットの表示

Amazon SNS コンソールではなくデベロッパーツールコンソールを使用して、AWS リージョン内のすべてのリソースのすべての通知ルールターゲットを表示できます。通知ルールのターゲットの詳細を表示することもできます。

通知ルールのターゲットを表示するには (コンソール)

1. <https://console.aws.amazon.com/codesuite/settings/notifications> で AWS デベロッパーツールコンソールを開きます。
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。
3. 通知ルールターゲットで、現在サインイン AWS アカウント AWS リージョンしている ので通知ルールが使用するターゲットのリストを確認します。セレクタを使用して AWS リージョンを変更します。ターゲットのステータスが [Unreachable (到達不能)] と表示された場合は、調査が必要になる場合があります。詳細については、「[トラブルシューティング](#)」を参照してください。

通知ルールのターゲットを一覧表示するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、list-targets コマンドを実行して、指定した AWS リージョンのすべての通知ルールのターゲットを一覧表示します。

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

2. 成功すると、このコマンドは、次のような AWS リージョン内の各通知ルールの ID と ARN を返します。

```
{
```

```
"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"
  }
]
```

## 通知ルールのターゲットの追加または削除

通知ルールを編集して、通知を送信する先のターゲットを変更できます。デベロッパーツールコンソールまたは `awscli` を使用して、通知ルールのターゲットを変更できます。

通知ルールのターゲットを変更するには (コンソール)

1. <https://console.aws.amazon.com/codesuite/settings/notifications> で AWS デベロッパーツールコンソールを開きます。
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。
3. 通知ルール で、現在サインイン AWS リージョンしている の AWS アカウントでリソース用に設定されたルールのリストを確認します。セレクタを使用して AWS リージョンを変更します。
4. ルールを選択し、[編集] を選択します。
5. [Targets (ターゲット)] で、次のいずれかの操作を行います。
  - 別のターゲットを追加するには、ターゲットの追加 を選択し、リストから追加する Amazon SNS トピックまたは AWS Chatbot (Slack) または AWS Chatbot (Microsoft Teams) クライア

ントを選択します。[Create SNS topic (SNS トピックを作成する)] を選択してトピックを作成し、ターゲットとして追加することもできます。1 つの通知ルールに最大 10 個のターゲットを設定できます。

- ターゲットを削除するには、削除するターゲットの横にある [Remove target (ターゲットの削除)] を選択します。

## 6. [送信] を選択します。

### 通知ルールにターゲットを追加するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、subscribe コマンドを実行してターゲットを追加します。例えば、次のコマンドは、通知ルールのターゲットとして Amazon SNS トピックを追加します。

```
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. 成功すると、次に示すように、更新された通知ルールの ARN がコマンドから返されます。

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

### 通知ルールからターゲットを削除するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、unsubscribe コマンドを実行してターゲットを削除します。例えば、次のコマンドは、通知ルールのターゲットとしての Amazon SNS トピックを削除します。

```
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. 成功すると、次に示すように、更新された通知ルールの ARN および削除されたターゲットに関する情報がコマンドから返されます。

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

以下も参照してください。

- [通知ルールの編集](#)
- [通知ルールの通知の有効化または無効化](#)

## 通知ルールのターゲットの削除

ターゲットが不要になった場合は、削除できます。リソースには通知ルールのターゲットを 10 個しか設定できないため、不要なターゲットを削除することで、その空いたスペースに他の必要なターゲットを追加できます。

### Note

通知ルールのターゲットを削除すると、それをターゲットとして使用するよう設定されているすべての通知ルールからターゲットが削除されます。ただし、ターゲット自体は削除されません。

通知ルールのターゲットを削除するには (コンソール)

1. <https://console.aws.amazon.com/codesuite/settings/notifications> で AWS デベロッパーツールコンソールを開きます。
2. ナビゲーションバーで、[Settings (設定)] を展開し、[Notifications rules (通知ルール)] を選択します。
3. 通知ルールターゲットで、現在サインイン AWS リージョンしているの AWS アカウントでリソース用に設定されたターゲットのリストを確認します。セレクトタを使用して AWS リージョンを変更します。
4. 通知ルールのターゲットを選択し、[削除] を選択します。
5. 「**delete**」と入力後、[削除] を選択します。

## 通知ルールのターゲットを削除するには (AWS CLI)

1. ターミナルまたはコマンドプロンプトで、delete-target コマンドを実行します。実行する際にターゲットの ARN を指定します。例えば、次のコマンドは、Amazon SNS トピックを使用するターゲットを削除します。

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

2. 成功すると、コマンドは何も返しません。失敗すると、コマンドはエラーを返します。最も一般的なエラーは、トピックが 1 つ以上の通知ルールのターゲットになっている場合です。

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

--force-unsubscribe-all パラメータを使用すると、そのトピックをターゲットとして使用するよう設定されているすべての通知ルールからターゲットを削除できます。さらにターゲット自体も削除できます。

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

## 通知と AWS Chatbot との統合の設定

AWS Chatbot は、DevOps やソフトウェア開発チームが Amazon Chime チャットルーム、Slack チャンネル、Microsoft Team チャンネルを使用し、AWS クラウド 内の運用イベントをモニタリングして対応できるようにする AWS のサービスです。通知ルールのターゲットと AWS Chatbot との統合を設定すると、選択した Amazon Chime ルーム、Slack チャンネル、または Microsoft Teams チャンネルにイベントに関する通知を表示できます。詳細については、「[AWS Chatbot ドキュメント](#)」を参照してください。

AWS Chatbot との統合を設定する前に、通知ルールとルールのターゲットを設定する必要があります。詳細については、[セットアップ](#) および [通知ルールの作成](#) を参照してください。また、AWS Chatbot で Slack チャンネル、Microsoft Teams チャンネル、または Amazon Chime チャットルームも設定する必要があります。詳細については、これらのサービスのドキュメントを参照してください。

### トピック

- [Slack チャンネルの AWS Chatbot クライアントの設定](#)

- [Microsoft Teams チャンネルの AWS Chatbot クライアントの設定](#)
- [Slack または Amazon Chime のクライアントの手動設定](#)

## Slack チャンネルの AWS Chatbot クライアントの設定

AWS Chatbot クライアントをターゲットとして使用する通知ルールを作成できます。Slack チャンネルのクライアントを作成すると、このクライアントを通知ルールの作成ワークフローでターゲットとして直接使用できます。これは、Slack チャンネルに表示される通知を設定する最も簡単な方法です。

ターゲットとして使用する AWS Chatbot クライアントを Slack で作成するには

1. 「AWS Chatbot 管理者ガイド」の「[AWS Chatbot を Slack で設定する](#)」の手順に従ってください。この場合、通知との統合を最適化するために以下の選択肢を検討してください。
  - IAM ロールを作成するときに、このロールの目的を端的に示すロール名 (**AWSCodeStarNotifications-Chatbot-Slack-Role** など) を選択します。これにより、以後、ロールの使用目的がわかりやすくなります。
  - [SNS topics] では、トピックや AWS リージョンを選択する必要はありません。AWS Chatbot クライアントを **ターゲット** として選択すると、通知ルールの作成プロセスの一環として、すべての必要なアクセス許可を持つ Amazon SNS トピックが AWS Chatbot クライアントとして作成および設定されます。
2. クライアントの作成プロセスを完了します。通知ルールの作成時に、このクライアントをターゲットとして選択できます。詳細については、「[通知ルールの作成](#)」を参照してください。

### Note

Amazon SNS トピックの設定後は、そのトピックを AWS Chatbot クライアントから削除しないでください。削除すると、Slack に通知が送信されなくなります。

## Microsoft Teams チャンネルの AWS Chatbot クライアントの設定

AWS Chatbot クライアントをターゲットとして使用する通知ルールを作成できます。Microsoft Teams チャンネルのクライアントを作成すると、このクライアントを通知ルールの作成ワークフローでターゲットとして直接使用できます。これは、Microsoft Teams チャンネルに表示される通知を設定する最も簡単な方法です。

ターゲットとして使用する AWS Chatbot クライアントを Microsoft Teams で作成するには

1. 「AWS Chatbot 管理者ガイド」の「[AWS Chatbot を Microsoft Teams で設定する](#)」の手順に従ってください。この場合、通知との統合を最適化するために以下の選択肢を検討してください。
  - IAM ロールを作成するときに、このロールの目的を端的に示すロール名 (**AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role** など) を選択します。これにより、以後、ロールの使用目的がわかりやすくなります。
  - [SNS topics] では、トピックや AWS リージョンを選択する必要はありません。AWS Chatbot クライアントを [ターゲット](#) として選択すると、通知ルールの作成プロセスの一環として、すべての必要なアクセス許可を持つ Amazon SNS トピックが AWS Chatbot クライアントとして作成および設定されます。
2. クライアントの作成プロセスを完了します。通知ルールの作成時に、このクライアントをターゲットとして選択できます。詳細については、「[通知ルールの作成](#)」を参照してください。

#### Note

Amazon SNS トピックの設定後は、そのトピックを AWS Chatbot クライアントから削除しないでください。削除すると、Microsoft Teams に通知が送信されなくなります。

## Slack または Amazon Chime のクライアントの手動設定

Slack や Amazon Chime と通知との統合を直接作成することを選択できます。これは、Amazon Chime チャットルームへの通知を設定するための唯一の方法です。この統合を手動で設定する場合は、通知ルールのターゲットとして以前に設定した Amazon SNS トピックを使用する AWS Chatbot クライアントを作成します。

通知と AWS Chatbot や Slack とを手動で統合するには

1. AWS デベロッパーツールコンソールは、次の URL で開きます。<https://console.aws.amazon.com/codesuite/settings/notifications>
2. [Settings (設定)]、[Notification rules (通知ルール)] の順に選択します。
3. [通知ルールのターゲット] で、ターゲットを検索してコピーします。



**Note**

そのターゲットと同じ Amazon SNS トピックを使用する通知ルールを複数設定できません。これはメッセージングを統合するのに役立ちますが、サブスクリプションリストが 1 つの通知ルールまたはリソースを対象としている場合、意図しない結果が生じることがあります。

4. AWS Chatbot コンソールは、次の URL で開きます。 <https://console.aws.amazon.com/opsworks/>
5. [Configure new client]、[Slack] の順に選択します。
6. [Configure] (設定) を選択します。
7. Slack ワークスペースにサインインします。
8. 選択内容を確認するメッセージが表示されたら、[Allow (許可)] を選択します。
9. [Configure new channel] を選択します。
10. [Configuration details] で、[Configuration name] にクライアント名を入力します。これは、通知ルールの作成時に AWS Chatbot (Slack) ターゲットタイプの使用可能なターゲットのリストに表示される名前です。
11. [Configure Slack Channel] (Slack チャンネルの設定) の [Channel type] (チャンネルタイプ) で、統合するチャンネルのタイプに応じて [Public] (パブリック) または [Private] (プライベート) を選択します。
  - [Public channel (パブリックチャンネル)] で、Slack チャンネルの名前をリストから選択します。
  - [Private channel ID (プライベートチャンネル ID)] に、チャンネルコードまたは URL を入力します。
12. [IAM permissions] (IAM アクセス許可) の [Role] (ロール) で、[Create an IAM role using a template] (テンプレートを使用して IAM ロールを作成する) を選択します。[ポリシーテンプレート] で、[通知のアクセス許可] を選択します。[ロール名] に、このロールの名前 ( **AWSCodeStarNotifications-Chatbot-Slack-Role** など ) を入力します。[ポリシーテンプレート] で、[通知のアクセス許可] を選択します。
13. [SNS topics] (SNS トピック) の [SNS Region] (SNS リージョン) で、通知ルールのターゲットを作成した AWS リージョンを選択します。[SNS topics] で、通知ルールのターゲットとして設定した Amazon SNS トピックの名前を選択します。



**Note**

このステップは、このクライアントをターゲットとして使用する通知ルールを作成する場合は必要ありません。

14. [Configure] (設定) を選択します。

**Note**

プライベートチャンネルとの統合を設定した場合、そのチャンネルに通知が表示されるには AWS Chatbot をチャンネルに招待する必要があります。詳細については、「[AWS Chatbot ドキュメント](#)」を参照してください。

15. (オプション) 統合をテストするには、ターゲットとして Amazon SNS トピックを使用するように設定された通知ルールのイベントタイプに対応するリソースを変更します。例えば、プルリクエストに対してコメントが作成されたときに通知を送信するように設定された通知ルールがある場合は、プルリクエストにコメントし、ブラウザで Slack チャンネルを監視して、通知がいつ表示されるかを確認します。

通知を AWS Chatbot および Amazon Chime と統合するには

1. AWS デベロッパーツールコンソールは、次の URL で開きます。<https://console.aws.amazon.com/codesuite/settings/notifications>
2. [Settings (設定)]、[Notification rules (通知ルール)] の順に選択します。
3. [通知ルールのターゲット] で、ターゲットを検索してコピーします。

**Note**

そのターゲットと同じ Amazon SNS トピックを使用する通知ルールを複数設定できます。これはメッセージングを統合するのに役立ちますが、サブスクリプションリストが1つの通知ルールまたはリソース用である場合、意図しない結果が生じることがあります。

4. Amazon Chime で、統合用に設定するチャットルームを開きます。
5. 右上の歯車アイコンを選択して、[Manage webhooks] を選択します。

6. [Manage webhooks (ウェブフックの管理)] ダイアログボックスで [新規] を選択し、ウェブフックの名前を入力して [作成] を選択します。
7. Webhook が表示されることを確認し、[Copy webhook URL (Webhook URL のコピー)] を選択します。
8. AWS Chatbot コンソールは、次の URL で開きます。 <https://console.aws.amazon.com/opsworks/>
9. [Configure new client] (新しいクライアントを設定)、[Amazon Chime] の順に選択します。
10. [Configuration details] で、[Configuration name] にクライアント名を入力します。
11. [Webhook URL] で、URL を貼り付けます。[Webhook description (Webhook の説明)] に、オプションの説明を入力します。
12. [IAM permissions] (IAM アクセス許可) の [Role] (ロール) で、[Create an IAM role using a template] (テンプレートを使用して IAM ロールを作成する) を選択します。[ポリシーテンプレート] で、[通知のアクセス許可] を選択します。[ロール名] に、このロールの名前 ( **AWSCodeStarNotifications-Chatbot-Chime-Role** など ) を入力します。
13. [SNS topics] (SNS トピック) の [SNS Region] (SNS リージョン) で、通知ルールのターゲットを作成した AWS リージョンを選択します。[SNS topics (SNS トピック)] で、通知ルールのターゲットとして設定した Amazon SNS トピックの名前を選択します。
14. [Configure] (設定) を選択します。
15. (オプション) 統合をテストするには、ターゲットとして Amazon SNS トピックを使用するように設定された通知ルールのイベントタイプに対応するリソースを変更します。例えば、プルリクエストに対してコメントが作成されたときに通知を送信するように設定された通知ルールがある場合は、プルリクエストにコメントし、Amazon Chime チャットルームを監視して通知がいつ表示されるかを確認します。

## AWS CloudTrail を使用した AWS CodeStar Notifications API コールのログ記録

AWS CodeStar Notifications は、ユーザー、ロール、または AWS のサービスによって実行されるアクションを記録するサービス AWS CloudTrail と統合されています。CloudTrail は、のすべての API コールをイベントとしてキャプチャします。キャプチャされた呼び出しには、デベロッパーツールコンソールからの呼び出しと、AWS CodeStar Notifications API オペレーションへのコードの呼び出しが含まれます。証跡を作成する場合は、通知のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます 追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (イベント履歴) で最新のイベントを表示できます。CloudTrail で収集された

情報を使用して、AWS CodeStar Notifications に対するリクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

詳細については、[AWS CloudTrailユーザーガイド](#)を参照してください。

## CloudTrail での AWS CodeStar Notifications 情報

CloudTrail は、アカウント作成時に AWS アカウント で有効になります。AWS CodeStar Notifications でアクティビティが発生すると、そのアクティビティは [Event history] (イベント履歴) の他の AWS サービスイベントと共に CloudTrail イベントに記録されます。最近のイベントは、AWS アカウント で表示、検索、ダウンロードできます。詳細については、「[Viewing events with CloudTrail event history](#)」(CloudTrail イベント履歴でのイベントの表示) を参照してください。

AWS CodeStar Notifications のイベントを含む、AWS アカウント のイベントの継続的な記録については、証跡を作成します。追跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョンに適用されます。証跡は、AWS パーティションのすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS のサービスを設定できます。詳細については、次を参照してください。

- [「追跡を作成するための概要」](#)
- [CloudTrail がサポートされているサービスと統合](#)
- [CloudTrail の Amazon SNS 通知の設定](#)
- [複数のリージョンから CloudTrail ログファイルを受け取るおよび複数のアカウントから CloudTrail ログファイルを受け取る](#)

すべての AWS CodeStar Notifications アクションは CloudTrail が記録します。

これらのアクションは「[AWS CodeStar Notifications API Reference](#)」

(AWS CodeStar Notifications API リファレンス) で説明されています。例え

ば、CreateNotificationRule、Subscribe、ListEventTypesの各アクションを呼び出すと、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストが、ルート認証情報と AWS Identity and Access Management (IAM) ユーザー認証情報のどちらを使用して送信されたか。

- リクエストがロールまたはフェデレーティッドユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが、別の AWS のサービスによって送信されたかどうか。

詳細については、「[CloudTrail userIdentity 要素](#)」をご参照ください。

## ログファイルエントリの理解

追跡は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルには、単一か複数のログエントリがあります。イベントはあらゆるソースからの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストのパラメータなどの情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

次の例は、CreateNotificationRule アクションと Subscribe アクションの両方を含む通知ルールの作成を示す CloudTrail ログエントリを示しています。

### Note

通知ログファイルエントリの一部のイベントは、サービスにリンクされたロール `AWSServiceRoleForCodeStarNotifications` から送信される場合があります。

```
{
  "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"  
}
```

## トラブルシューティング

以下の情報は、通知で発生する一般的な問題のトラブルシューティングに役立つ場合があります。

### トピック

- [リソースに対する通知ルールを作成しようとする、アクセス許可エラーが表示されます](#)
- [通知ルールを表示できません](#)
- [通知ルールを作成できません](#)
- [アクセスできないリソースに関する通知が届きます](#)
- [Amazon SNS の通知が届きません](#)
- [イベントに関する重複した通知が届きます](#)
- [通知ターゲットのステータスが到達不能と表示される理由を教えてください](#)
- [通知とリソースのクォータを引き上げることはできますか](#)

リソースに対する通知ルールを作成しようとする、アクセス許可エラーが表示されます

アクセス許可が十分であることを確認してください。詳細については、「[アイデンティティベースポリシーの例](#)」を参照してください。

### 通知ルールを表示できません

問題: デベロッパーツールコンソールで、[設定] から [通知] を選択すると、アクセス許可エラーが表示されます。

解決方法: 通知を表示するために必要なアクセス許可がない可能性があります。CodeCommit や CodePipeline などの AWS デベロッパーツールサービスのほとんどの管理ポリシーには通知のアクセス許可が含まれていますが、現在通知をサポートしていないサービスには通知を表示するアクセス

許可は含まれていません。または、通知の表示を許可しないカスタムポリシーを IAM ユーザーまたはロールに適用することもできます。詳細については、「[アイデンティティベースポリシーの例](#)」を参照してください。

## 通知ルールを作成できません

通知ルールの作成に必要なアクセス許可を持っていない可能性があります。詳細については、「[アイデンティティベースポリシーの例](#)」を参照してください。

## アクセスできないリソースに関する通知が届きます

通知ルールを作成してターゲットを追加したときに、受取人がリソースにアクセスできるかどうかは通知機能によって検証されません。アクセスできないリソースに関する通知が届く場合があります。ターゲットのサブスクリプションリストから自分自身を削除できない場合は、削除を依頼してください。

## Amazon SNS の通知が届きません

Amazon SNS トピックの問題のトラブルシューティングを行うには、以下を確認します。

- Amazon SNS トピックが通知ルールと同じ AWS リージョンに作成されていることを確認します。
- E メールエイリアスが正しいトピックにサブスクライブされていること、およびサブスクリプションを確認済みであることを確認します。詳細については、「[Amazon SNS トピックにエンドポイントをサブスクライブする](#)」を参照してください。
- 該当するトピックに通知をプッシュすることを AWS CodeStar Notifications に許可するようにトピックポリシーが編集されていることを確認します。トピックポリシーには、次のようなステートメントを含める必要があります。

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

詳細については、「[通知用に Amazon SNS トピックを設定する](#)」を参照してください。

## イベントに関する重複した通知が届きます

複数の通知を受信する最も一般的な理由は以下のとおりです。

- 同じイベントタイプを含む複数の通知ルールをリソースに設定し、これらのルールのターゲットとして複数の Amazon SNS トピックにサブスクライブしている。この問題を解決するには、いずれかのトピックからサブスクリプションを解除するか、通知ルールを編集して重複を削除します。
- 1 つ以上の通知ルールのターゲットが AWS Chatbot と統合されており、E メールを受信トレイと Slack チャンネル、Microsoft Teams チャンネル、または Amazon Chime チャットルームで通知を受信しています。この問題を解決するには、ルールのターゲットである Amazon SNS トピックから E メールアドレスのサブスクリプションを解除し、Slack チャンネル、Microsoft Teams チャンネル、または Amazon Chime チャットルームを使用して通知を確認することを検討します。

## 通知ターゲットのステータスが到達不能と表示される理由を教えてください

ターゲットのステータスには、[Active (アクティブ)] と [Unreachable (到達不能)] の 2 つがあります。[到達不能] は、ターゲットに送信された通知が未到着であることを示します。通知はそのターゲットに引き続き送信され、到着すると、ステータスが [Active (アクティブ)] にリセットされます。

通知ルールのターゲットは、次のいずれかの理由で使用不能になる場合があります。

- リソース (Amazon SNS トピックまたは AWS Chatbot クライアント) が削除された。通知ルールの別のターゲットを選択した。
- Amazon SNS トピックが暗号化されており、暗号化されたトピックに必要なポリシーが見つからないか、AWS KMS キーが削除されている。詳細については、「[通知用に Amazon SNS トピックを設定する](#)」を参照してください。
- 通知に必要なポリシーが Amazon SNS トピックに存在しない。トピックにポリシーがない場合、通知を Amazon SNS トピックに送信することはできません。詳細については、「[通知用に Amazon SNS トピックを設定する](#)」を参照してください。



- ターゲット (Amazon SNS または AWS Chatbot) のサポートサービスに問題が発生している可能性がある。

## 通知とリソースのクォータを引き上げることはできますか

現在、クォータを変更することはできません。「[通知のクォータ](#)」を参照してください。

## 通知のクォータ

次の表に、デベロッパーツールコンソールでの通知のクォータ (制限) を一覧表示します。変更できる制限の詳細については、「[AWS のサービスクォータ](#)」を参照してください。

リソース	デフォルトの制限
AWS アカウントの通知ルールの最大数	1,000
通知ルールのターゲットの最大数	10
リソースの通知ルールの最大数	10

## 接続とは?

デベロッパーツールコンソールの接続機能を使用して、などのリソース AWS CodePipeline を外部コードリポジトリに接続 AWS できます。この機能には、独自の API である [AWS CodeConnections API リファレンス](#) があります。各接続は、などのサードパーティーリポジトリに接続するためにサービスに付与 AWS できるリソースです BitBucket。例えば、サードパーティーのコードリポジトリにコード変更が行われたときにパイプラインをトリガー CodePipeline するように、に接続を追加できます。各接続には名前が付けられ、接続を参照するために使用される一意の Amazon Resource Name (ARN) に関連付けられます。

### Important

サービス名 AWS CodeStar Connections の名前が変更されました。以前の名前空間 codestar-connections で作成されたリソースは引き続きサポートされます。

## 接続では何ができますか？

接続を使用して、サードパーティープロバイダーのリソースを次のデベロッパーツールの AWS リソースと統合できます。

- Bitbucket などのサードパーティープロバイダーに接続し、 などの AWS リソースとのソース統合としてサードパーティー接続を使用します CodePipeline。
- サードパーティープロバイダー CodePipeline の CodeBuild ビルドプロジェクト、 CodeDeploy アプリケーション、パイプラインで、 リソース間の接続へのアクセスを统一的に管理します。
- スタックテンプレートの接続 ARN は、 CodeDeploy の CodeBuild ビルドプロジェクト、 アプリケーション、パイプラインに使用します。保存されたシークレットやパラメータを参照 CodePipeline する必要はありません。

## どのサードパーティープロバイダーの接続を作成できますか？

接続では、 AWS リソースを次のサードパーティーリポジトリに関連付けることができます。

- Bitbucket Cloud
- GitHub.com
- GitHub エンタープライズクラウド
- GitHub エンタープライズサーバー
- GitLab.com

### Important

の接続サポート GitLab には、バージョン 15.x 以降が含まれています。

- GitLab セルフマネージドインストール (Enterprise Edition または Community Edition の場合 )

接続ワークフローの概要については、「[接続を作成または更新するワークフロー](#)」を参照してください。

などのクラウドプロバイダータイプの接続を作成する手順は、 GitHub Enterprise Server などのインストール済みプロバイダータイプの手順 GitHubとは異なります。プロバイダーのタイプ別に接続を作成するハイレベルの手順については、「[接続の使用](#)」を参照してください。

**Note**

欧州 (ミラノ) で接続を使用するには AWS リージョン、以下を実行する必要があります。

1. リージョン固有のアプリをインストールする
2. リージョンを有効にする

このリージョン固有のアプリで、欧州 (ミラノ) リージョンの接続をサポートします。サードパーティープロバイダーのサイトで公開されているアプリであり、他のリージョンの接続をサポートする既存のアプリとは別のものです。このアプリをインストールすることで、このリージョンでのみサービスとデータを共有することをサードパーティープロバイダーに許可します。アプリをアンインストールすることでいつでもアクセス許可を取り消すことができます。

リージョンを有効にしない限り、サービスはデータを処理または保存しません。このリージョンを有効にすることで、データを処理および保存するアクセス許可をサービスに付与したことになります。

リージョンが有効になっていなくても、リージョン固有のアプリがインストールされたままであれば、サードパーティープロバイダーはお客様のデータをサービスと共有できます。したがって、リージョンを無効にしたら、必ずアプリをアンインストールしてください。詳細については、「[リージョンの有効化](#)」を参照してください。

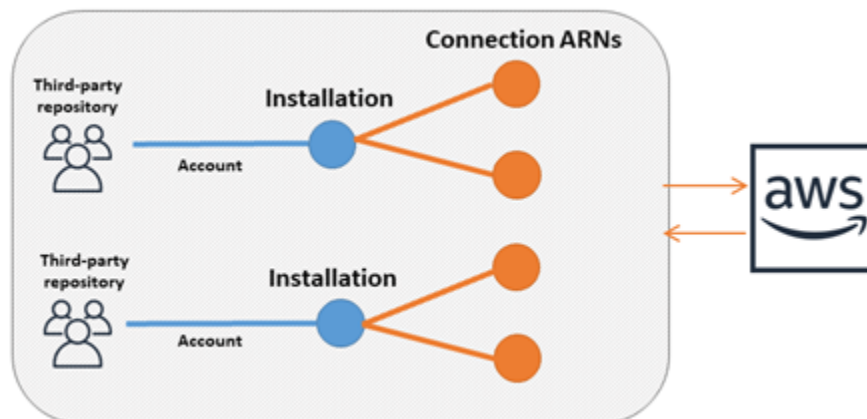
## 接続と AWS のサービス 統合するもの

接続を使用して、サードパーティーのリポジトリを他の AWS のサービスと統合できます。接続のサービス統合を確認するには、「[を製品やサービスと統合する AWS CodeConnections](#)」を参照してください。

## 接続はどのように機能しますか？

接続を作成する前に、サードパーティーアカウントで AWS 認証アプリケーションをインストールするか、そのアプリケーションへのアクセス権を提供する必要があります。接続をインストールした後、このインストールを使用するように更新できます。接続を作成すると、サードパーティーアカウントの AWS リソースへのアクセスを許可します。これにより、リソースに代わって、接続がサードパーティーアカウントのソースリポジトリなどのコンテンツにアクセスできるようになります。その後、その接続を他のと共有 AWS のサービスとして、リソース間で安全な OAuth 接続を提供できます。

Enterprise Server などの GitHubインストール済みプロバイダータイプへの接続を作成する場合は、まず `aws` を使用してホストリソースを作成します AWS Management Console。



接続は、それらを作成する AWS アカウント によって所有されます。接続は、接続 ID を含む ARN によって識別されます。接続 ID は、変更または再マッピングできない UUID です。接続を削除して再確立すると、新しい接続 ID が作成されるため、新しい接続 ARN が作成されます。つまり、接続 ARN が再利用されることはありません。

新しく作成された接続が Pending 状態です。接続のセットアップを完了し、接続を Pending 状態から Available 状態に移行するには、サードパーティーのハンドシェイク ( OAuthフロー ) プロセスが必要です。これが完了すると、接続は Available になり、などの AWS サービスで使用できます CodePipeline。

新しく作成されたホストは Pending 状態です。ホストのセットアップを完了し、ホストを Pending 状態から Available 状態に移行するには、サードパーティーの登録プロセスが必要です。これが完了すると、ホストは Available で、インストール済みプロバイダータイプへの接続に使用できます。

接続ワークフローの概要については、「[接続を作成または更新するワークフロー](#)」を参照してください。インストール済みプロバイダー用のホスト作成ワークフローの概要については、「[ホストを作成または更新するワークフロー](#)」を参照してください。プロバイダーのタイプ別に接続を作成するハイレベルの手順については、「[接続の使用](#)」を参照してください。

## のグローバルリソース AWS CodeConnections

接続はグローバルリソースです。つまり、リソースがすべての AWS リージョンにレプリケートされます。

接続 ARN 形式には作成されたリージョン名が反映されますが、リソースはリージョンに制約されません。接続リソースが作成されたリージョンは、接続リソースデータの更新が制御されるリージョンです。接続リソースデータの更新を制御する API 操作の例として、接続の作成、インストールの更新、接続の削除、接続のタグ付けなどがあります。

接続のホストリソースは、グローバルに利用可能なリソースではありません。ホストリソースは、リソースを作成したリージョンでのみ使用します。

- 接続は 1 回作成するだけで済みます。その後、任意の AWS リージョンで使用できます。
- 接続が作成されたリージョンに問題がある場合、接続リソースデータを制御する API は影響を受けますが、他のすべてのリージョンで接続を正常に使用できます。
- コンソールまたは CLI で接続リソースをリストすると、すべてのリージョンでアカウントに関連付けられているすべての接続リソースが一覧表示されます。
- コンソールまたは CLI でホストリソースをリストすると、リストには、選択したリージョンのアカウントに関連付けられたホストリソースだけが表示されます。
- 関連するホストリソースとの接続がリストされている場合、または CLI で一覧表示されている場合、設定されている CLI リージョンに関係なく、出力はホスト ARN を返します。

## ホストを作成または更新するワークフロー

インストール済みプロバイダタイプの接続を作成するときは、最初にホストを作成します。

ホストの各状態は以下のとおりです。

- Pending - pending ホストは作成済みのホストで、使用する前に設定 (available に移行) する必要があります。
- Available - available ホストを使用することも、接続に渡すこともできます。

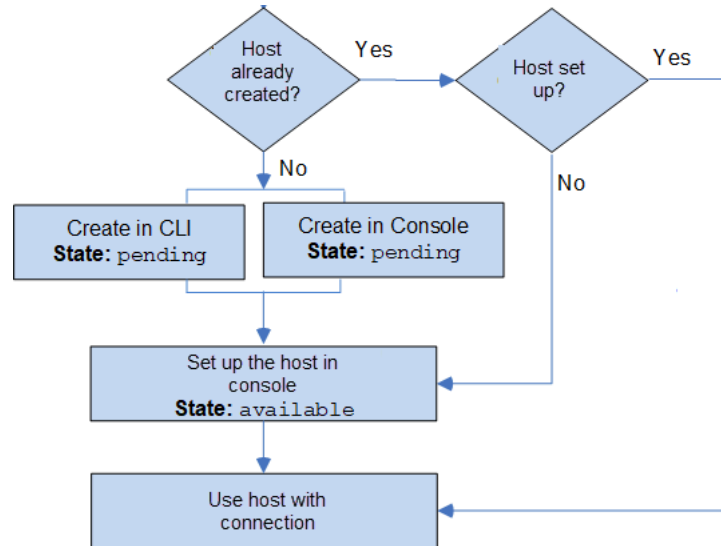
ワークフロー: CLI、SDK、または AWS CloudFormation を使用したホストの作成または更新

[CreateHost](#) API を使用して、AWS Command Line Interface (AWS CLI)、SDK、または [awscli](#) を使用してホストを作成します AWS CloudFormation。作成後、ホストは pending の状態になります。コンソールの [セットアップ] オプションを使用して、プロセスを完了します。

ワークフロー: コンソールを使用したホストの作成または更新

Enterprise Server や GitLab セルフマネージドなどの GitHubインストール済みプロバイダータイプへの接続を作成する場合は、まずホストを作成します。Bitbucket などのクラウドプロバイダーのタイプに接続する場合は、ホストの作成をスキップして、接続の作成を続行します。

コンソールを使用してホストを設定し、ステータスを pending から available に変更します。



### 接続を作成または更新するワークフロー

接続を作成するときは、サードパーティープロバイダーと認証ハンドシェイクをするためのインストールを作成、あるいは既存のインストールを使用します。

接続には、以下のステータスがあります。

- Pending - A pending 接続は、使用する前に完了 ( available に移動 ) する必要があります。
- Available - アカウント内の他のリソースやユーザーに available 接続を使用または渡すことができます。
- Error - error 状態の接続は自動的に再試行されます。 available になるまで使用できません。

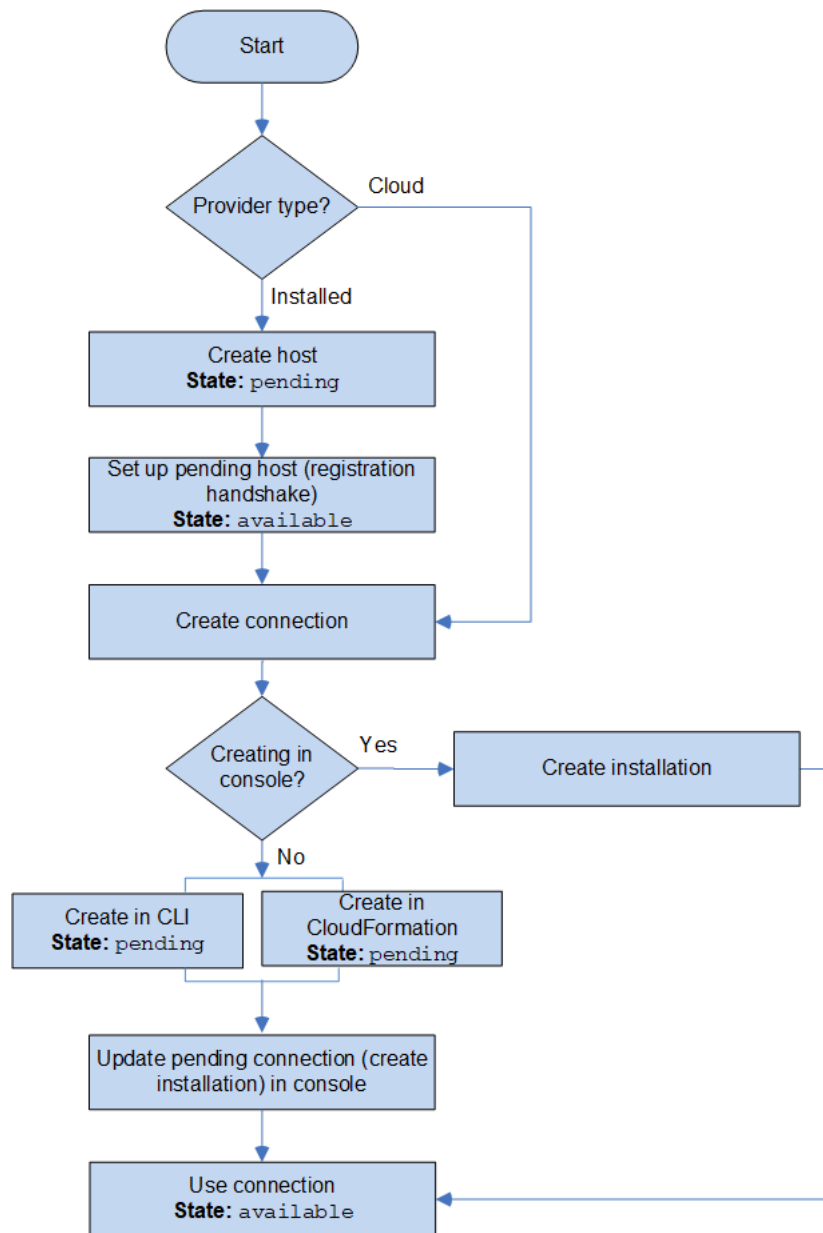
### ワークフロー: CLI、SDK、AWS CloudFormationを使用した接続の作成または更新

[CreateConnection](#) API を使用して、AWS Command Line Interface ( AWS CLI )、SDK、またはを使用して接続を作成します AWS CloudFormation。作成後、接続は pending の状態になります。コンソールの [保留中の接続のセットアップ] オプションを使用して、プロセスを完了します。インストールを作成するか、接続に既存のインストールを使用するかを確認するメッセージがコンソールに表示されます。次に、コンソールでハンドシェイクを完了し、[接続の完了] を選択して、接続を available の状態に移行します。

ワークフロー: コンソールとの接続の作成または更新

Enterprise Server などの GitHubインストール済みプロバイダータイプへの接続を作成する場合は、まずホストを作成します。Bitbucket などのクラウドプロバイダーのタイプに接続する場合は、ホストの作成をスキップして、接続の作成を続行します。

コンソールを使用して接続を作成または更新するには、コンソール CodePipeline の編集アクションページを使用してサードパーティープロバイダーを選択します。コンソールでは、インストールを作成するか、既存のインストールを使用して接続を作成するように求められます。次に、接続の作成を求められます。コンソールがハンドシェイクを完了し、自動的に pending の状態から available の状態に移行します。





## 接続を開始するにはどうしたらいいですか？

使用を開始するには、次のいくつかのトピックが役立ちます。

- [接続の概念](#)について学びます。
- [必要なリソース](#)をセットアップして、接続の操作を開始します。
- [最初の接続](#)を開始し、それらをリソースに接続します。

## 接続概念

概念と用語を理解すれば、接続機能の設定と使用が容易になります。デベロッパーツールコンソールの接続機能を使用する際に知っておかなければならないいくつかの概念を次に示します。

### インストール

サードパーティーアカウントの AWS アプリのインスタンス。AWS CodeStar Connector アプリをインストールすると、AWS からサードパーティーのアカウント内のリソースにアクセスできます。インストールは、サードパーティープロバイダーのウェブサイト以外では編集できません。

### connection

サードパーティーのソースリポジトリを他の AWS サービスに接続するために使用される AWS リソース。

### サードパーティーのリポジトリ

AWS 以外のサービスまたは会社が提供するリポジトリ。例えば、BitBucket リポジトリはサードパーティーのリポジトリです。

### プロバイダーのタイプ

接続先のサードパーティーソースリポジトリを提供するサービスまたは会社。AWS リソースを外部のプロバイダータイプに接続します。そのソースリポジトリがネットワークおよびインフラストラクチャにインストールされているプロバイダータイプが、インストール済プロバイダータイプです。例えば、GitHub Enterprise Server は、インストール済プロバイダータイプの 1 つです。

### ホスト

サードパーティープロバイダーがインストールされているインフラストラクチャを表すリソース。接続は、ホストを使用して、GitHub Enterprise Server などのサードパーティープロバイダが



インストールされているサーバーを表します。そのプロバイダータイプへのすべての接続に対して1つのホストを作成します。

**Note**

コンソールを使用して GitHub Enterprise Server への接続を作成すると、コンソールがホストリソースを作成します。これは、コンソールの処理の一部です。

## AWS CodeConnections サポートされているプロバイダーとバージョン

この章では、がサポートする AWS CodeConnectionsプロバイダーとバージョンについて説明します。

### トピック

- [Bitbucket でサポートされるプロバイダタイプ](#)
- [GitHub および GitHub Enterprise Cloud でサポートされているプロバイダータイプ](#)
- [GitHub Enterprise Server でサポートされているプロバイダータイプとバージョン](#)
- [GitLab.com でサポートされているプロバイダータイプ](#)
- [GitLab セルフマネージドでサポートされているプロバイダータイプ](#)

### Bitbucket でサポートされるプロバイダタイプ

接続アプリは、Atlassian Bitbucket Cloud で使用できます。

Bitbucket サーバーなど、インストールされている Bitbucket プロバイダーのタイプはサポートされていません。

### GitHub および GitHub Enterprise Cloud でサポートされているプロバイダータイプ

接続アプリケーションは、GitHub および GitHub Enterprise Cloud で使用できます。

### GitHub Enterprise Server でサポートされているプロバイダータイプとバージョン

接続アプリは、サポートされているバージョンの GitHub Enterprise Server で使用できます。サポートされているバージョンのリストについては、「<https://enterprise.github.com/releases/>」を参照してください。

**⚠ Important**

AWS CodeConnections は、非推奨の GitHub Enterprise Server バージョンをサポートしていません。例えば、AWS CodeConnections リリースの既知の問題により、は GitHub Enterprise Server バージョン 2.22.0 をサポートしていません。接続するには、バージョン 2.22.1 または入手可能な最新のバージョンにアップグレードします。

## GitLab.com でサポートされているプロバイダタイプ

GitLab.com で接続を使用できます。詳細については、「[への接続を作成する GitLab](#)」を参照してください。

**⚠ Important**

の接続サポート GitLab には、バージョン 15.x 以降が含まれています。

## GitLab セルフマネージドでサポートされているプロバイダタイプ

GitLab セルフマネージドインストール (Enterprise Edition または Community Edition の場合) で接続を使用できます。詳細については、「[セルフマネージドへの接続 GitLabを作成する](#)」を参照してください。

## を製品やサービスと統合する AWS CodeConnections

AWS CodeConnections は、多数の AWS サービス、パートナー製品およびサービスと統合されています。以下のセクションの情報は、使用している製品やサービスと統合するための接続の設定に役立ちます。

このサービスを利用する際に役立つ関連リソースは次のとおりです。

### トピック

- [Amazon CodeGuru Reviewer](#)
- [Amazon CodeWhisperer](#)
- [Amazon SageMaker](#)
- [AWS App Runner](#)

- [AWS CloudFormation](#)
- [AWS CodeBuild](#)
- [AWS CodePipeline](#)
- [AWS CodeStar](#)
- [Service Catalog](#)
- [AWS Proton](#)

## Amazon CodeGuru Reviewer

[CodeGuru Reviewer](#) は、リポジトリコードをモニタリングするためのサービスです。接続を使用して、レビューするコードがあるサードパーティーのリポジトリを関連付けることができます。GitHub リポジトリ内のソースコードをモニタリングして、コードを改善するレコメンデーションを作成できるように CodeGuru Reviewer を設定するチュートリアルについては、「[Amazon CodeGuru Reviewer ユーザーガイド](#)」の「[チュートリアル: GitHub リポジトリ内のソースコードをモニタリングする](#)」を参照してください。

## Amazon CodeWhisperer

[Amazon CodeWhisperer](#) は、リポジトリの code. CodeWhisperer reviews コードを確認し、コードのレコメンデーションをリアルタイムで提供するためのサービスです。接続を使用してデータソースにアクセスする [CodeWhisperer カスタマイズ](#) を設定する手順については、「[Amazon ユーザーガイド](#)」の「[カスタマイズの作成](#)」を参照してください。 CodeWhisperer

## Amazon SageMaker

[Amazon SageMaker](#) は、機械学習言語モデルを構築、トレーニング、デプロイするためのサービスです。GitHub リポジトリへの接続を設定するチュートリアルについては、[Amazon SageMaker デベロッパーガイド SageMaker の「サードパーティーの Git リポジトリを使用した MLOps プロジェクトのチュートリアル」](#)を参照してください。

## AWS App Runner

[AWS App Runner](#) は、AWS クラウドで、ソースコードまたはコンテナイメージから、スケラブルでセキュアなウェブアプリケーションに迅速でシンプルな、費用対効果の高い方法で直接デプロイできるサービスです。App Runner の自動統合および配信パイプラインを使用して、リポジトリからアプリケーションコードをデプロイできます。接続を使用して、プライベート GitHub リポジトリから

App Runner サービスにソースコードをデプロイできます。詳細については、AWS App Runner デベロッパーガイドの「[ソースコードのリポジトリプロバイダー](#)」を参照してください。

## AWS CloudFormation

[AWS CloudFormation](#) は、AWS リソースのモデル化とセットアップに役立つサービスです。これにより、リソースの管理に費やす時間を減らし、で実行されるアプリケーションに集中する時間を増やすことができます AWS。必要なすべての AWS リソース (Amazon EC2 インスタンスや Amazon RDS DB インスタンスなど) を記述するテンプレートを作成し、CloudFormation がそれらのリソースのプロビジョニングと設定を行います。

で Git 同期との接続を使用して CloudFormation、Git リポジトリをモニタリングする同期設定を作成します。スタックデプロイに Git 同期を使用する手順のチュートリアルについては、AWS CloudFormation ユーザーガイドの [CloudFormation「Git 同期」の使用](#) を参照してください。

の詳細については CloudFormation、CloudFormation「[コマンドラインインターフェイスユーザーガイド](#)」の [CloudFormation「拡張機能を発行するためのアカウントの登録](#)」を参照してください。

## AWS CodeBuild

[AWS CodeBuild](#) は、コードを構築およびテストするためのサービスです。は、独自のビルドサーバーをプロビジョニング、管理、スケーリングする必要性 CodeBuild を排除し、一般的なプログラミング言語とビルドツール用のパッケージ済みのビルド環境を提供します。への接続 CodeBuild でを使用する方法の詳細については GitLab、「[ユーザーガイド](#)」の「[GitLab接続](#)AWS CodeBuild」を参照してください。

## AWS CodePipeline

[CodePipeline](#) は、ソフトウェアのリリースに必要なステップをモデル化、視覚化、自動化するために使用できる継続的デリバリーサービスです。接続を使用して、CodePipeline ソースアクション用のサードパーティーリポジトリを設定できます。

詳細はこちら:

- CodePipeline アクションのSourceConnectionsアクション設定リファレンスページを参照してください。設定パラメータと JSON/YAML スニペットの例を表示するには、「[AWS CodePipeline ユーザーガイドCodeStarSourceConnection](#)」の「」を参照してください。
- サードパーティーのソースリポジトリを使用してパイプラインを作成する「[開始方法](#)」チュートリアルを表示するには、「[接続の使用開始](#)」を参照してください。

## AWS CodeStar

[AWS CodeStar](#) は、でソフトウェア開発プロジェクトを作成、管理、操作するためのクラウドベースのサービスです AWS。AWS CodeStar プロジェクトを使用すると、でアプリケーションを迅速に開発、構築 AWS、デプロイできます。接続を使用して、AWS CodeStar プロジェクトのパイプライン用にサードパーティのリポジトリを設定できます。リポジトリへの接続を使用して AWS CodeStar プロジェクトを作成するチュートリアルについては、[ユーザーガイドの「リポジトリへのリンク」](#)を作成する GitHubAWS CodeStar」を参照してください。

## Service Catalog

[Service Catalog](#) を使用すると、組織はでの使用が承認された製品のカatalogを作成および管理できます AWS。

、GitHub GitHub エンタープライズ、などの外部リポジトリプロバイダーと間の接続を承認する AWS アカウントと、接続により BitBucket、Service Catalog 製品をサードパーティのリポジトリで管理されるテンプレートファイルに同期できます。

詳細については、[「Service Catalog ユーザーガイド」の GitHub 「、GitHub エンタープライズ、または Bitbucket からのテンプレートファイルへの Service Catalog 製品の同期」](#)を参照してください。

## AWS Proton

[AWS Proton](#) は、クラウドインフラストラクチャにデプロイするためのクラウドベースのサービスです。接続を使用して、AWS Protonのテンプレートのリソース用のサードパーティリポジトリへのリンクを作成できます。詳細については、AWS Proton ユーザーガイドの[「リポジトリのリンクを作成する」](#)を参照してください。

## 接続のセットアップ

このセクションのタスクを完了して、デベロッパーツールコンソールで接続機能の作成と使用するためのセットアップを行います。

### トピック

- [にサインアップする AWS](#)
- [接続を作成するアクセス許可を持つポリシーの作成と適用](#)

## にサインアップする AWS

### にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

### にサインアップするには AWS アカウント

1. <https://portal.aws.amazon.com/billing/signup> を開きます。
2. オンラインの手順に従います。

サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

にサインアップすると AWS アカウント、AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用して [ルートユーザーアクセスが必要なタスク](#) を実行してください。

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。 <https://aws.amazon.com/> の [アカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

### 管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、 を保護し AWS アカウントのルートユーザー、 を有効にして AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないように管理ユーザーを作成します。

### のセキュリティ保護 AWS アカウントのルートユーザー

1. ルートユーザーを選択し、AWS アカウント E メールアドレスを入力して、アカウント所有者 [AWS Management Console](#) として にサインインします。次のページでパスワードを入力します。

ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの「[ルートユーザーとしてサインインする](#)」を参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

手順については、「IAM [ユーザーガイド](#)」の AWS アカウント「[ルートユーザーの仮想 MFA デバイスを有効にする \(コンソール\)](#)」を参照してください。

## 管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

手順については、「AWS IAM Identity Center ユーザーガイド」の「[AWS IAM Identity Centerの有効化](#)」を参照してください。

2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

を ID ソース IAM アイデンティティセンターディレクトリとして使用する方法的チュートリアルについては、「[ユーザーガイド](#)」の「[デフォルトでユーザーアクセス IAM アイデンティティセンターディレクトリを設定するAWS IAM Identity Center](#)」を参照してください。

## 管理アクセス権を持つユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

IAM Identity Center ユーザーを使用してサインインする方法については、「AWS サインイン [ユーザーガイド](#)」の AWS「[アクセスポータルへのサインイン](#)」を参照してください。

## 追加のユーザーにアクセス権を割り当てる

1. IAM アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従ったアクセス許可セットを作成します。

手順については、「AWS IAM Identity Center ユーザーガイド」の「[権限設定を作成する](#)」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

手順については、「AWS IAM Identity Center ユーザーガイド」の「[グループの参加](#)」を参照してください。

## 接続を作成するアクセス許可を持つポリシーの作成と適用

JSON ポリシーエディタでポリシーを作成するには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/iam/> で IAM コンソールを開きます。
2. 左側のナビゲーションペインで、[ポリシー] を選択します。

初めて [ポリシー] を選択する場合には、[管理ポリシーによるこそ] ページが表示されます。[今すぐ始める] を選択します。

3. ページの上部で、[ポリシーを作成] を選択します。
4. [ポリシーエディタ] セクションで、[JSON] オプションを選択します。
5. 次の JSON ポリシードキュメントを入力します。

```
{
  "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. [次へ] をクリックします。



**Note**

いつでも [Visual] と [JSON] エディタオプションを切り替えることができます。ただし、[Visual] エディタで [次] に変更または選択した場合、IAM はポリシーを再構成して visual エディタに合わせて最適化することがあります。詳細については、「IAM ユーザーガイド」の「[ポリシーの再構成](#)」を参照してください。

7. [確認と作成] ページで、作成するポリシーの [ポリシー名] と [説明] (オプション) を入力します。[このポリシーで定義されているアクセス許可] を確認して、ポリシーによって付与されたアクセス許可を確認します。
8. [ポリシーの作成] をクリックして、新しいポリシーを保存します。

## 接続の使用開始

接続を開始する最も簡単な方法は、サードパーティーのソースリポジトリを AWS リソースに関連付ける接続を設定することです。パイプラインをなどの AWS ソースに接続する CodeCommit 場合は、ソースアクションとして接続します。ただし、外部リポジトリがある場合は、接続を作成して、リポジトリをパイプラインに関連付ける必要があります。このチュートリアルでは、Bitbucket リポジトリと自分のパイプラインとの接続を設定します。

このセクションでは、接続を使用します。

- AWS CodePipeline: これらのステップでは、パイプラインソースとして Bitbucket リポジトリを使用してパイプラインを作成します。
- [Amazon CodeGuru Reviewer](#): 次に、Bitbucket リポジトリを CodeGuru Reviewer のフィードバックおよび分析ツールに関連付けます。

### トピック

- [前提条件](#)
- [ステップ 1: ソースファイルを編集する](#)
- [ステップ 2: パイプラインを作成する](#)
- [ステップ 3: リポジトリを CodeGuru レビューワーに関連付ける](#)

## 前提条件

開始する前に、「[セットアップ](#)」のステップを完了します。また、AWS サービスに接続し、接続で認証を管理できるようにするサードパーティーのソースリポジトリも必要です。例えば、Bitbucket リポジトリをソースリポジトリと統合する AWS サービスに接続できます。

- Bitbucket アカウントを使用して Bitbucket リポジトリを作成します。
- Bitbucket 認証情報を準備します。を使用して接続 AWS Management Console を設定すると、Bitbucket 認証情報を使用してサインインするように求められます。

## ステップ 1: ソースファイルを編集する

Bitbucket リポジトリを作成すると、デフォルトの README.md ファイルが含まれます。このファイルを編集します。

1. Bitbucket リポジトリにログインし、[Source] (送信元) を選択します。
2. README.md ファイルを選択し、次にページの上部の [Edit] (編集) を選択します。既存のテキストを削除し、次のテキストを追加します。

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

3. [Commit] (コミット) を選択します。

README.md ファイルがリポジトリのルートレベルにあることを確認してください。

## ステップ 2: パイプラインを作成する

このセクションでは、次のアクションを使用してパイプラインを作成します。

- Bitbucket リポジトリとアクションへの接続を持つソースステージ。
- ビルドアクションを含む AWS CodeBuild ビルドステージ。

ウィザードを使用してパイプラインを作成するには

1. <https://console.aws.amazon.com/codepipeline/> で CodePipeline コンソールにサインインします。
2. [ようこそ] ページ、[開始方法] ページ、または [パイプライン] ページで、[パイプラインの作成] を選択します。

3. [ステップ 1: パイプラインの設定を選択する] の [パイプライン名] に「**MyBitbucketPipeline**」と入力します。
4. [サービスロール] で、[New service role (新しいサービスロール)] を選択します。

**Note**

代わりに既存の CodePipeline サービスロールを使用する場合は、サービスロールポリシーに `codeconnections:UseConnection` IAM アクセス許可を追加していることを確認してください。CodePipeline サービスロールの手順については、「[サービスロールにアクセス許可を追加する CodePipeline](#)」を参照してください。

5. [詳細設定] では、デフォルト値のままにします。アーティファクトストアで、[Default location] (デフォルトの場所)を選択し、パイプライン用に選択したリージョン内のパイプラインのデフォルトのアーティファクトストア (デフォルトとして指定された Amazon S3 アーティファクトバケットなど) を使用します。

**Note**

これはソースコードのソースバケットではありません。パイプラインのアーティファクトストアです。パイプラインごとに S3 バケットなどの個別のアーティファクトストアが必要です。

[次へ] をクリックします。

6. ステップ2 : [Add source stage] (ソースステージの追加) ページで、ソースステージを追加します。
  - a. [Source provider] (ソースプロバイダー) で、[Bitbucket] を選択します。
  - b. [Connection] (接続) で、[Connect to Bitbucket (Bitbucket に接続)] を選択します。
  - c. [Connect to Bitbucket] (Bitbucket に接続) ページの [Connection name] (接続名) に、作成する接続の名前を入力します。この名前は、後でこの接続を識別するのに役立ちます。

[Bitbucket apps] (Bitbucket アプリ) で、[Install a new app(新しいアプリをインストールする)] を選択します。

- d. アプリのインストールページで、AWS CodeStar アプリが Bitbucket アカウントに接続しようとしていることを示すメッセージが表示されます。[アクセス権の付与] を選択します。接

続を承認すると、Bitbucket のリポジトリが検出され、AWS リソースに関連付けることができます。

- e. 新規インストールの接続 ID が表示されます。[Complete connection (接続の完了)] を選択します。コンソール CodePipelineに戻ります。
- f. [リポジトリ名] で、Bitbucket リポジトリの名前を選択します。
- g. ブランチ名で、リポジトリのブランチを選択します。
- h. [ソースコードの変更時にパイプラインを開始する] オプションが選択されていることを確認します。
- i. 出力アーティファクト形式で、デフォルト CodePipeline のいずれかを選択します。
  - パイプラインのアーティファクトにデフォルトの zip 形式を使用するには、CodePipeline デフォルトを選択します。
  - [完全クローン] を選択して、パイプライン内のアーティファクトのリポジトリに関する Git メタデータを含めます。これは アクションでのみサポートされます CodeBuild。

[次へ] をクリックします。

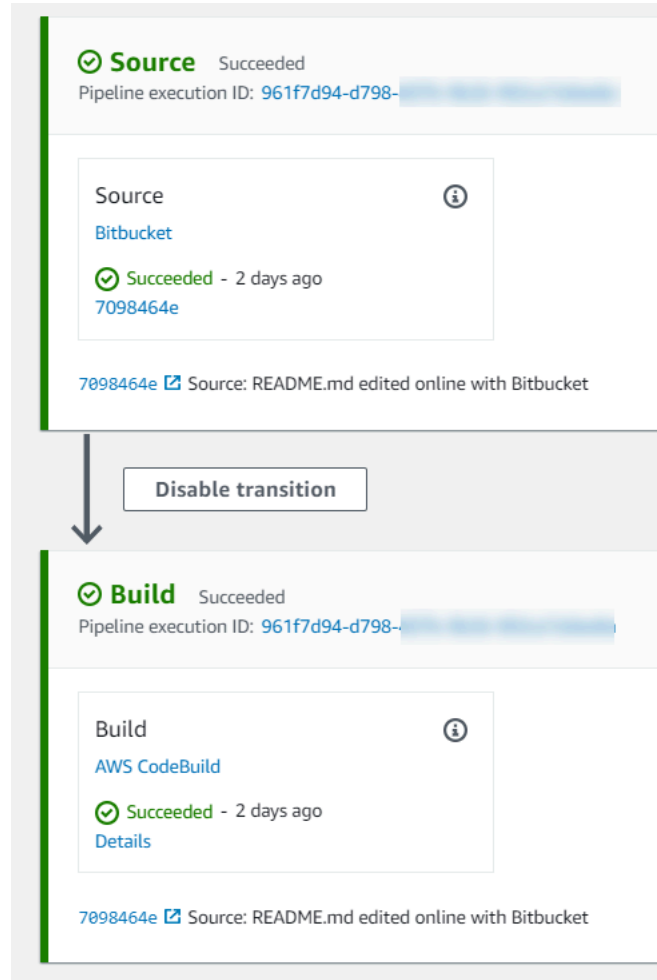
7. [Add build stage (ビルドステージの追加)] で、ビルドステージを追加します。
  - a. [ビルドプロバイダ] で、[AWS CodeBuild] を選択します。[リージョン] がデフォルトでパイプラインリージョンになることを許可します。
  - b. [プロジェクトを作成] を選択します。
  - c. [プロジェクト名] に、このビルドプロジェクトの名前を入力します。
  - d. [環境イメージ] で、[Managed image (マネージド型イメージ)] を選択します。[Operating system] で、[Ubuntu] を選択します。
  - e. [ランタイム] で、[Standard (標準)] を選択します。[イメージ] で、[aws/codebuild/standard:5.0] を選択します。
  - f. [サービスロール] で、[New service role (新しいサービスロール)] を選択します。
  - g. [Buildspec] の Build specifications (ビルド仕様) で、[Insert build commands] (ビルドコマンドの挿入) を選択します。Switch to editor([1]エディタに切り替え)を選択し、Build commands (ビルドコマンド)に以下を貼り付けます。

```
version: 0.2

phases:
  install:
```



## 10. パイプラインが正常に作成されると、パイプラインが実行されます。



## 11. ビルドが成功した段階で、[詳細]を選択します。

実行の詳細で、CodeBuild ビルド出力を表示します。README.md ファイルの内容は、コマンドで次のよう出力されます。

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

```
35 [Container] 2020/06/05 19:14:51 Running command cat README.md
36 This is a Bitbucket repository!
37 [Container] 2020/06/05 19:14:51 Phase complete: PRE_BUILD State: SUCCEEDED
38 [Container] 2020/06/05 19:14:51 Phase context status code: Message:
39 [Container] 2020/06/05 19:14:51 Entering phase BUILD
40 [Container] 2020/06/05 19:14:51 Phase complete: BUILD State: SUCCEEDED
41 [Container] 2020/06/05 19:14:51 Phase context status code: Message:
42 [Container] 2020/06/05 19:14:51 Entering phase POST_BUILD
43 [Container] 2020/06/05 19:14:51 Phase complete: POST_BUILD State: SUCCEEDED
44 [Container] 2020/06/05 19:14:51 Phase context status code: Message:
```

## ステップ 3: リポジトリを CodeGuru レビューワーに関連付ける

接続を作成したら、その接続を同じアカウントのすべての AWS リソースに使用できます。例えば、パイプラインの CodePipeline ソースアクションと Reviewer のリポジトリコミット分析に同じ Bitbucket CodeGuru 接続を使用できます。

1. CodeGuru Reviewer コンソールにサインインします。
2. CodeGuru レビューワー で、リポジトリ の関連付け を選択します。  
  
1 ページのウィザードが開きます。
3. [Select source provider] (ソースプロバイダーの選択) で、[Bitbucket] を選択します。
4. 「Bitbucket に接続する ( を使用 AWS CodeConnections )」で、パイプライン用に作成した接続を選択します。
5. [Repository location] (リポジトリの場所) で、Bitbucket リポジトリの名前を選択し、Associate (関連付け) を選択します。

コードレビューの設定を続行できます。詳細については、[「Amazon CodeGuru Reviewer ユーザーガイド」の「Bitbucket に接続してリポジトリを Reviewer に関連付ける」](#)を参照してください。 CodeGuru

## 接続の使用

接続は、AWS リソースを外部コードリポジトリに接続するために使用する構成です。各接続は、Bitbucket などのサードパーティーリポジトリに接続する AWS CodePipeline などのサービスに付与できるリソースです。例えば、サードパーティーのコードリポジトリにコード変更が行われたときにパイプラインをトリガー CodePipeline するように、に接続を追加できます。AWS リソースを GitHub Enterprise Server などのインストール済みプロバイダータイプに接続することもできます。

GitHub Enterprise Server などのインストール済みプロバイダータイプへの接続を作成する場合、コンソールによってホストが作成されます。ホストは、プロバイダがインストールされているサーバーを表すために作成するリソースです。詳細については、[「ホストの使用」](#)を参照してください。

接続を作成するときは、コンソールのウィザードを使用して接続アプリをサードパーティープロバイダーにインストールし、新しい接続に関連付けます。アプリをインストール済みである場合は、それを使用できます。

**Note**

欧州 (ミラノ) で接続を使用するには AWS リージョン、以下を実行する必要があります。

1. リージョン固有のアプリをインストールする
2. リージョンを有効にする

このリージョン固有のアプリで、欧州 (ミラノ) リージョンの接続をサポートします。サードパーティープロバイダーのサイトで公開されているアプリであり、他のリージョンの接続をサポートする既存のアプリとは別のものです。このアプリをインストールすることで、このリージョンでのみサービスとデータを共有することをサードパーティープロバイダーに許可します。アプリをアンインストールすることでいつでもアクセス許可を取り消すことができます。

リージョンを有効にしない限り、サービスはデータを処理または保存しません。このリージョンを有効にすることで、データを処理および保存するアクセス許可をサービスに付与したことになります。

リージョンが有効になっていなくても、リージョン固有のアプリがインストールされたままであれば、サードパーティープロバイダーはお客様のデータをサービスと共有できます。したがって、リージョンを無効にしたなら、必ずアプリをアンインストールしてください。詳細については、「[リージョンの有効化](#)」を参照してください。

接続の詳細については、[AWS CodeConnections 「API リファレンス」](#) を参照してください。Bitbucket の CodePipeline ソースアクションの詳細については、AWS CodePipeline 「ユーザーガイド [CodestarConnectionSource](#)」の「」を参照してください。

接続を使用するために必要なアクセス許可を持つ AWS Identity and Access Management (IAM) ユーザーまたはロールにポリシーを作成またはアタッチするには、「」を参照してください。[AWS CodeConnections アクセス許可リファレンス](#)。CodePipeline サービスロールが作成された時期によっては、をサポートするようにアクセス許可を更新する必要がある場合があります AWS CodeConnections。手順については、AWS CodePipeline User Guide の「[Update the service role](#)」を参照してください。

**トピック**

- [接続を作成する](#)
- [Bitbucket への接続を作成する](#)
- [への接続を作成する GitHub](#)



- [GitHub Enterprise Server への接続を作成する](#)
- [への接続を作成する GitLab](#)
- [セルフマネージドへの接続 GitLabを作成する](#)
- [保留中の接続の更新](#)
- [接続を一覧表示する](#)
- [接続を削除](#)
- [タグ接続リソース](#)
- [接続の詳細の表示](#)

## 接続を作成する

次のサードパーティーのプロバイダーのタイプへの接続を作成できます。

- Bitbucket への接続を作成するには、「[Bitbucket への接続を作成する](#)」を参照してください。
- GitHub または GitHub Enterprise Cloud への接続を作成するには、「」を参照してください [への接続を作成する GitHub](#)。
- ホストリソースの作成を含む GitHub Enterprise Server への接続を作成するには、「」を参照してください [GitHub Enterprise Server への接続を作成する](#)。
- への接続を作成するには、GitLab 「」を参照してください [への接続を作成する GitLab](#)。

### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN で接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

## Bitbucket への接続を作成する

AWS Management Console または AWS Command Line Interface (AWS CLI) を使用して、`bitbucket.org` でホストされているリポジトリへの接続を作成できます。

開始する前に:

- Bitbucket で、アカウントを作成しておく必要があります。

- bitbucket.org で、コードリポジトリを作成しておく必要があります。

#### Note

Bitbucket Cloudリポジトリへの接続を作成できます。Bitbucket サーバーなど、インストールされている Bitbucket プロバイダーのタイプはサポートされていません。[AWS CodeConnections サポートされているプロバイダーとバージョン](#) を参照してください。

#### Note

接続は、接続の作成に使用されたアカウントで所有するリポジトリへのアクセスだけを提供します。

アプリケーションを Bitbucket ワークスペースにインストールする場合は、「ワークスペースを管理する」アクセス許可が必要です。アクセス許可がないと、アプリケーションをインストールするオプションは表示されません。

## トピック

- [Bitbucket \(コンソール\) への接続を作成する](#)
- [Bitbucket \(CLI\) への接続を作成する](#)

## Bitbucket (コンソール) への接続を作成する

コンソールを使用して Bitbucket への接続を作成できます。

#### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に codestar-connections を持つリソースのみが作成されます。ARN で接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

## ステップ 1: 接続の作成

1. にサインインし AWS Management Console、 で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。

2. 選択[設定] > [接続] を選択してから、[接続を作成する]。
3. Bitbucket リポジトリへの接続を作成するには、[Select a provider] (プロバイダーを選択する) で、[Bitbucket] を選択します。[接続名] に、作成する接続の名前を入力します。[Connect to Bitbucket] (Bitbucket に接続) を選択し、ステップ 2 に進みます。

Developer Tools > Connections > Create connection

## Create a connection [Info](#)

**Select a provider**

Bitbucket  GitHub  GitHub Enterprise Server

**Create Bitbucket connection**

Connection name

[Connect to Bitbucket](#)

### ステップ 2: Bitbucket に接続する

1. [Connect to Bitbucket] 設定ページに、接続名が表示されます。

[Bitbucket apps] (Bitbucket アプリ) で、アプリのインストールを選択するか、アプリを作成するために [Install a new app] (新しいアプリをインストールする) を選択します。

#### Note

アプリケーションは、Bitbucket ワークスペースまたはアカウントごとに 1 回だけインストールします。Bitbucket アプリを既にインストールしている場合は、それを選択してこのセクションの最後のステップに移動します。

Connect to Bitbucket

Bitbucket connection settings [Info](#)

Connection name

a-connection

Bitbucket apps

Bitbucket apps create a link for your connection with Bitbucket. To start, install a new app and save this connection.

or

2. Bitbucket のログインページが表示されたら、認証情報を使用してログインし、続行を選択します。
3. アプリのインストールページで、AWS CodeStar アプリが Bitbucket アカウントに接続しようとしていることを示すメッセージが表示されます。

Bitbucket ワークスペースを使用している場合は、[Authorize for] (承認対象) オプションをそのワークスペースに変更します。管理者権限のあるワークスペースのみが表示されます。

[アクセス権の付与] を選択します。



### AWS CodeStar requests access

This app is hosted at <https://codestar-connections.webhooks.aws>

- Read your account information
- Read your repositories and their pull requests
- Administer your repositories
- Read and modify your repositories

Authorize for

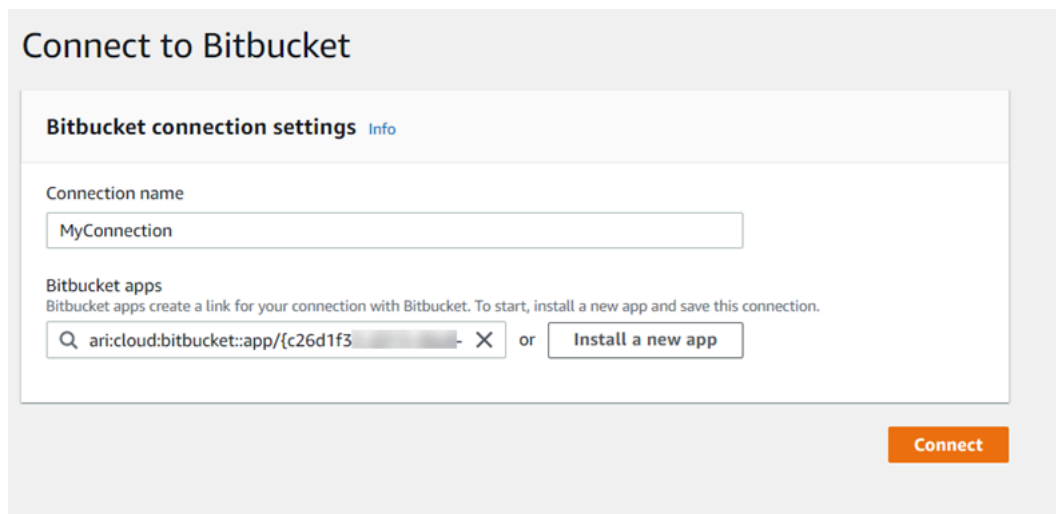
#### Allow AWS CodeStar to do this?

This 3rd party vendor has not provided a privacy policy or terms of use.

Atlassian's Privacy Policy is not applicable to the use of this App.

**Grant access** Cancel

4. Bitbucketアプリには、新規インストールの接続 ID が表示されます。[接続]を選択します。作成された接続が接続リストに表示されます。



## Bitbucket (CLI) への接続を作成する

AWS Command Line Interface ( AWS CLI) を使用して接続を作成できます。

これを行うには、create-connection コマンドを使用します。

### ⚠ Important

AWS CLI または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDINGステータスです。CLI または の接続を作成したら AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを にしますAVAILABLE。

Bitbucket への接続を作成するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して create-connection コマンド AWS CLI を実行し、接続--connection-nameに --provider-typeと を指定します。この例では、サードパーティープロバイダー名は Bitbucket で、指定された接続名は MyConnection です。

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

成功した場合、このコマンドは次のような接続 ARN 情報を返します。

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

2. コンソールを使用して接続を完了します。詳細については、「[保留中の接続の更新](#)」を参照してください。

## への接続を作成する GitHub

AWS Management Console または AWS Command Line Interface ( AWS CLI) を使用して、への接続を作成できます GitHub。

開始する前に:

- でアカウントを作成しておく必要があります GitHub。
- サードパーティーのコードリポジトリを予め作成しておく必要があります。

### Note

接続を作成するには、GitHub 組織の所有者である必要があります。組織のリポジトリでない場合、ユーザーがリポジトリの所有者である必要があります。

## トピック

- [への接続 GitHubを作成する \(コンソール\)](#)
- [\(CLI\) への接続 GitHubを作成する](#)

## への接続 GitHubを作成する (コンソール)

コンソールを使用して、への接続を作成できます GitHub。

### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN で接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

1. にサインインし AWS Management Console、でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. 選択[設定] > [接続] を選択してから、[接続を作成する]。
3. GitHub または GitHub Enterprise Cloud リポジトリへの接続を作成するには、「プロバイダーの選択」で、「」を選択しますGitHub。[接続名] に、作成する接続の名前を入力します。に接続を選択し GitHub、ステップ 2 に進みます。

Create a connection [Info](#)

Select a provider

Bitbucket  GitHub  GitHub Enterprise Server

Create GitHub App connection

Connection name

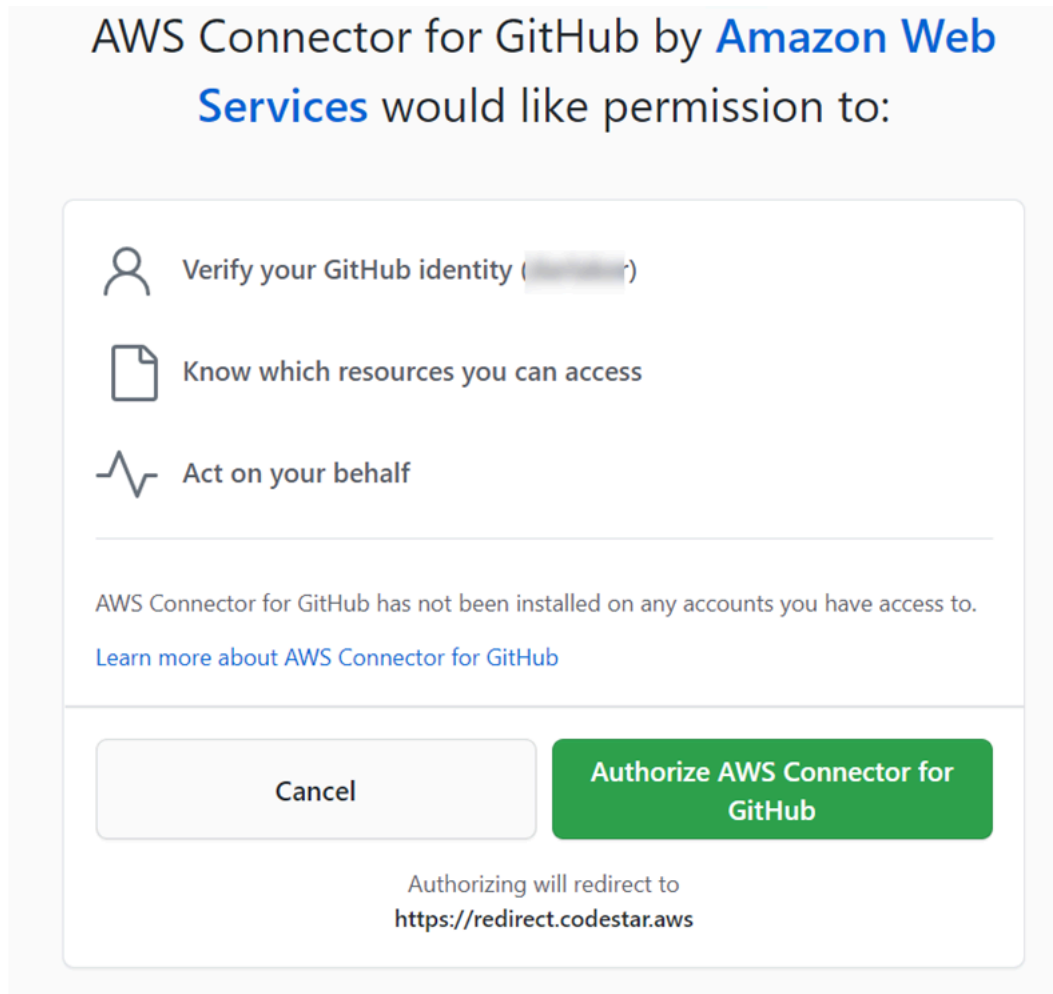
githubc-connection

Connect to GitHub

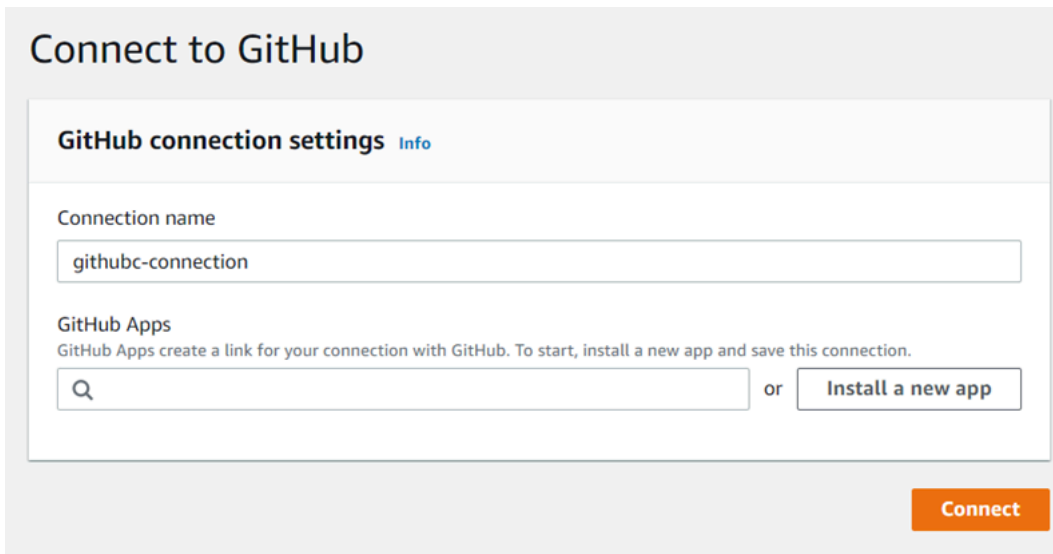
への接続を作成するには GitHub

1. GitHub 接続設定 で、接続名が接続名 に表示されます。[Connect to GitHub ( に接続)] を選択します。アクセス要求のページが表示されます。





2. の AWS コネクタを承認を選択します GitHub。接続ページが表示され、GitHub アプリフィールドが表示されます。

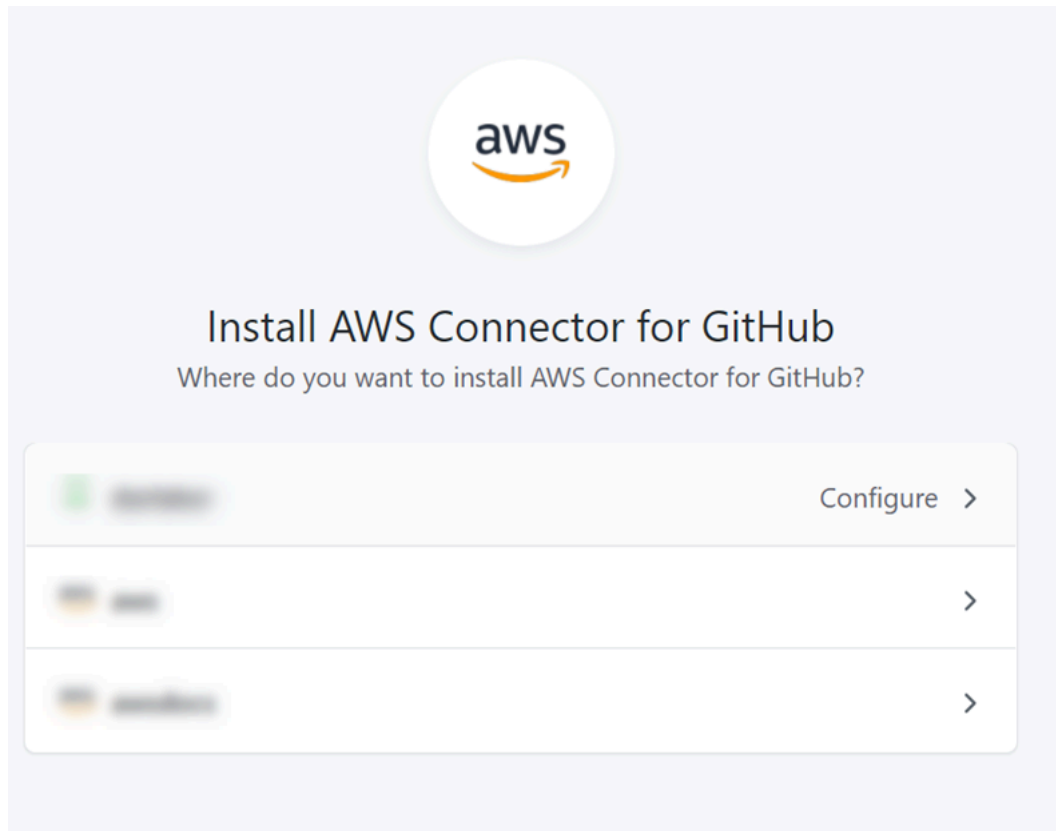


3. GitHub アプリで、アプリのインストールを選択するか、新しいアプリのインストールを選択して作成します。

**Note**

特定のプロバイダーへのすべての接続に対してアプリを1つインストールします。AWS Connector for GitHub App を既にインストールしている場合は、それを選択してこのステップをスキップします。

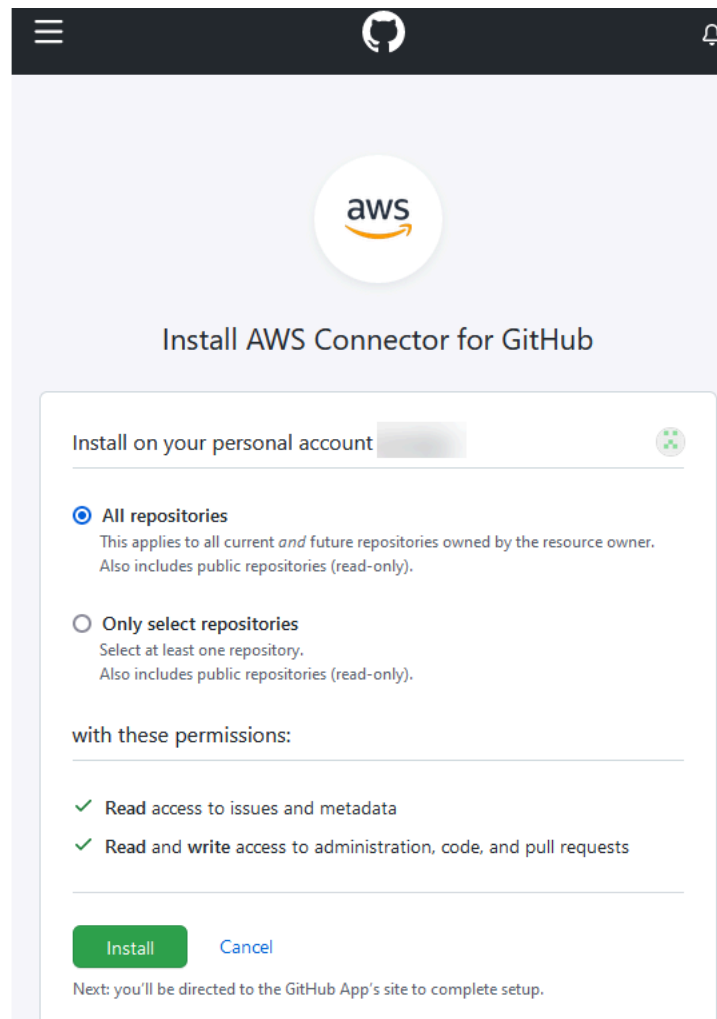
4. 「用AWS コネクタのインストール GitHub」ページで、アプリをインストールするアカウントを選択します。



**Note**

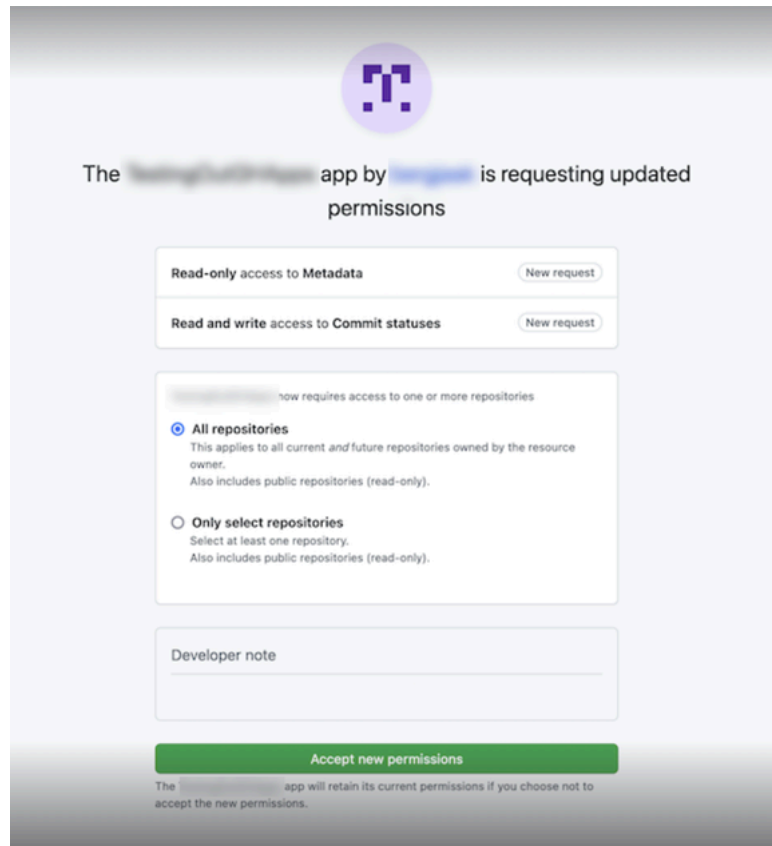
アプリは GitHub アカウントごとに1回だけインストールします。アプリケーションをインストール済みである場合は、[Configure] (設定) を選択してアプリのインストールの変更ページに進むか、戻るボタンでコンソールに戻ることができます。

5. 「用AWS コネクタのインストール GitHub」ページで、デフォルトのままにして、「のインストール」を選択します。



このステップの後、更新されたアクセス許可ページが に表示される場合があります GitHub。

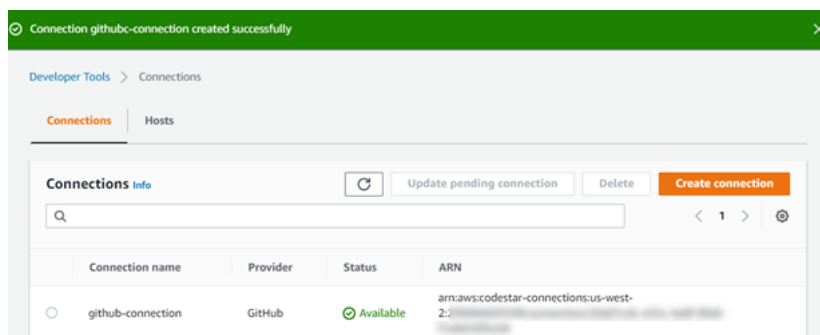
6. Connector for GitHub App の AWS アクセス許可が更新されていることを示すページが表示された場合は、新しいアクセス許可を受け入れる を選択します。



7. Connect to GitHubページに戻ります。新しいインストールの接続 ID が GitHub Apps に表示されます。[接続]を選択します。

### 作成した接続を表示する

- 作成された接続が接続リストに表示されます。



### (CLI) への接続 GitHubを作成する

AWS Command Line Interface ( AWS CLI) を使用して、への接続を作成できます GitHub。

これを行うには、create-connection コマンドを使用します。

### Important

AWS CLI または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDING ステータスです。CLI または との接続を作成したら AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを にします AVAILABLE。

への接続を作成するには GitHub

1. ターミナル (Linux/macOS/Unix) または コマンドプロンプト (Windows) を開きます。を使用して create-connection コマンド AWS CLI を実行し、接続 --connection-name に --provider-type とを指定します。この例では、サードパーティープロバイダー名は GitHub で、指定された接続名は MyConnection です。

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

成功した場合、このコマンドは次のような接続 ARN 情報を返します。

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

2. コンソールを使用して接続を完了します。詳細については、「[保留中の接続の更新](#)」を参照してください。

## GitHub Enterprise Server への接続を作成する

接続を使用して、AWS リソースをサードパーティーのリポジトリに関連付けます。AWS Management Console または AWS Command Line Interface (AWS CLI) を使用して、GitHub Enterprise Server への接続を作成できます。

接続は、アプリケーションのインストール GitHub を許可するために接続の作成中に使用される GitHub Enterprise Server アカウントが所有するリポジトリへのアクセスのみを提供します。

開始する前に:

- GitHub Enterprise Server インスタンスとリポジトリが既に存在している必要があります。
- このセクションに示すように、GitHub アプリケーションを作成し、ホストリソースを作成するには、GitHub Enterprise Server インスタンスの管理者である必要があります。

#### Important

GitHub Enterprise Server のホストを設定すると、ウェブフックイベントデータの VPC エンドポイントが自動的に作成されます。2020 年 11 月 24 日より前にホストを作成し、VPC PrivateLink ウェブフックエンドポイントを使用する場合は、まずホストを削除してから新しいホストを作成する必要があります。

#### トピック

- [Enterprise Server への接続 GitHubを作成する \(コンソール\)](#)
- [Enterprise Server \(CLI\) への接続 GitHubを作成する](#)

#### Enterprise Server への接続 GitHubを作成する (コンソール)

GitHub Enterprise Server 接続を作成するには、GitHub Enterprise Server がインストールされている場所に関する情報を提供し、GitHub Enterprise 認証情報を使用して接続の作成を承認します。

#### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN に接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

#### トピック

- [Enterprise Server 接続を作成する GitHub \(コンソール\)](#)

#### Enterprise Server 接続を作成する GitHub (コンソール)

GitHub Enterprise Server への接続を作成するには、サーバー URL と GitHub Enterprise 認証情報を準備します。

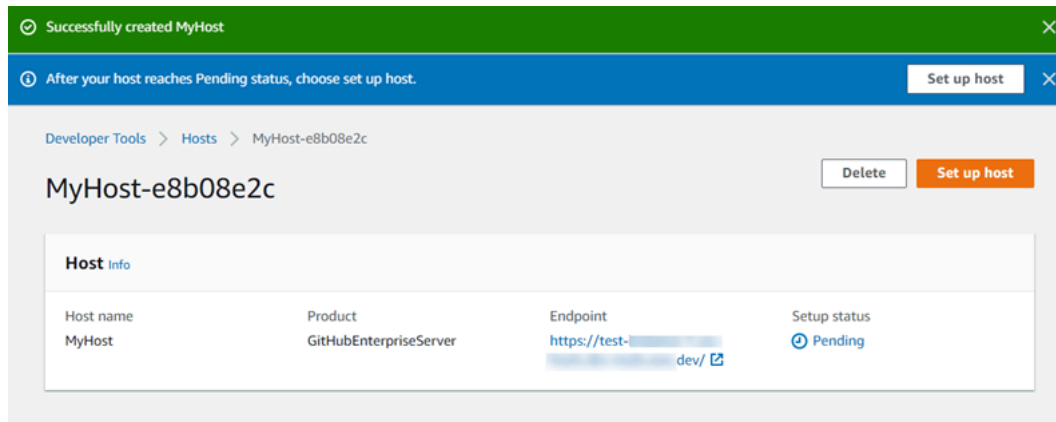
## ホストを作成するには

1. にサインインし AWS Management Console、 で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. [Hosts (ホスト)] タブで、[Create host (ホストの作成)] を選択します。
3. [ホスト名] に、ホストに使用する名前を入力します。
4. [プロバイダーを選択] で、次のいずれかを選択します。
  - GitHub エンタープライズサーバー
  - GitLab セルフマネージド型
5. [URL] に、プロバイダーがインストールされているインフラストラクチャのエンドポイントを入力します。
6. サーバーが Amazon VPC 内に設定されていて、VPC に接続する場合は、Use a VPC (VPC を使用) を選択します。それ以外の場合、[No VPC] を選択します。
7. Amazon VPC でインスタンスを起動し、VPC に接続する場合は、[Use a VPC] (VPC を使用) をクリックして、以下を完了します。
  - a. [VPC ID] で、VPC ID を選択します。インスタンスがインストールされているインフラストラクチャに VPC を選択するか、VPN または Direct Connect を介してインスタンスにアクセスできる VPC を選択します。
  - b. プライベート VPC を設定していて、非公開認証局を使用して TLS 検証を実行するようにインスタンスを設定している場合は、[TLS 証明書] に証明書 ID を入力します。TLS 証明書の値は 証明書のパブリックキーです。
8. [Create host] (ホストの作成) を選択します。
9. ホストの詳細ページが表示されたら、ホストの作成に伴ってホストのステータスが変化します。

### Note

ホスト設定に VPC 設定が含まれている場合は、ホストネットワークコンポーネントのプロビジョニングに数分間かかります。

ホストのステータスが Pending (保留中) になるのを待ってから、セットアップを完了します。詳細については、「[保留中のホストをセットアップする](#)」を参照してください。



## ステップ 2: GitHub Enterprise Server への接続を作成する (コンソール)

1. にサインイン AWS Management Console し、 でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. 選択[設定] > [接続] を選択してから、[接続を作成する]。
3. インストールされている GitHub Enterprise Server リポジトリへの接続を作成するには、GitHub Enterprise Server を選択します。

### Enterprise Server GitHub に接続する

1. [Connection name] (接続名) に、接続の名前を入力します。



Developer Tools > Connections > Create connection

## Create a connection Info

**Select a provider**

Bitbucket  GitHub  GitHub Enterprise Server

**Connection Settings Info**

**Connection name**  
Give your connection a name.

**URL**  
The endpoint of the server to connect to.

Use a VPC  
If your GitHub Enterprise Server is only accessible in a VPC, configure details here. Otherwise, skip this step.  
Complete these steps in the same AWS Region as your VPC.

Cancel **Connect to GitHub Enterprise Server**

2. [URL] に、サーバーのエンドポイントを入力します。

**Note**

指定された URL が接続用の Enterprise Server のセットアップ GitHub に既に使用されている場合は、そのエンドポイント用に以前に作成されたホストリソース ARN を選択するように求められます。

3. (オプション) Amazon VPC でサーバーを起動し、VPC に接続する場合は、[VPC を使用] を選択して、以下を完了します。
  - a. [VPC ID] で、VPC ID を選択します。GitHub エンタープライズサーバーインスタンスがインストールされているインフラストラクチャの VPC、または VPN または Direct Connect を介してエンタープライズサーバーインスタンスにアクセスできる GitHub VPC を必ず選択してください。
  - b. [サブネット ID] で、[Add] を選択します。このフィールドで、ホストに使用するサブネット ID を選択します。最大 10 個のサブネットを選択できます。

Enterprise Server インスタンスがインストールされているインフラストラクチャのサブネット、または VPN または Direct Connect GitHub を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるサブネットを必ず選択してください。

- c. [Security group IDs] (セキュリティグループ ID) で、[Add] (追加) を選択します。このフィールドで、ホストに使用するセキュリティグループを選択します。最大 10 個のセキュリティグループを選択できます。

GitHub Enterprise Server インスタンスがインストールされているインフラストラクチャのセキュリティグループ、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるセキュリティグループを必ず選択してください。

- d. プライベート VPC が設定されていて、非パブリック認証機関を使用して TLS 検証を実行するように GitHub Enterprise Server インスタンスを設定している場合は、TLS 証明書に証明書 ID を入力します。TLS 証明書の値は、証明書のパブリックキーである必要があります。

VPC ID  
Choose the VPC in which your GitHub Enterprise Server is configured.

**Subnet IDs**  
Choose the subnet or subnets for the VPC in which your GitHub Enterprise Server is configured.

Subnet ID

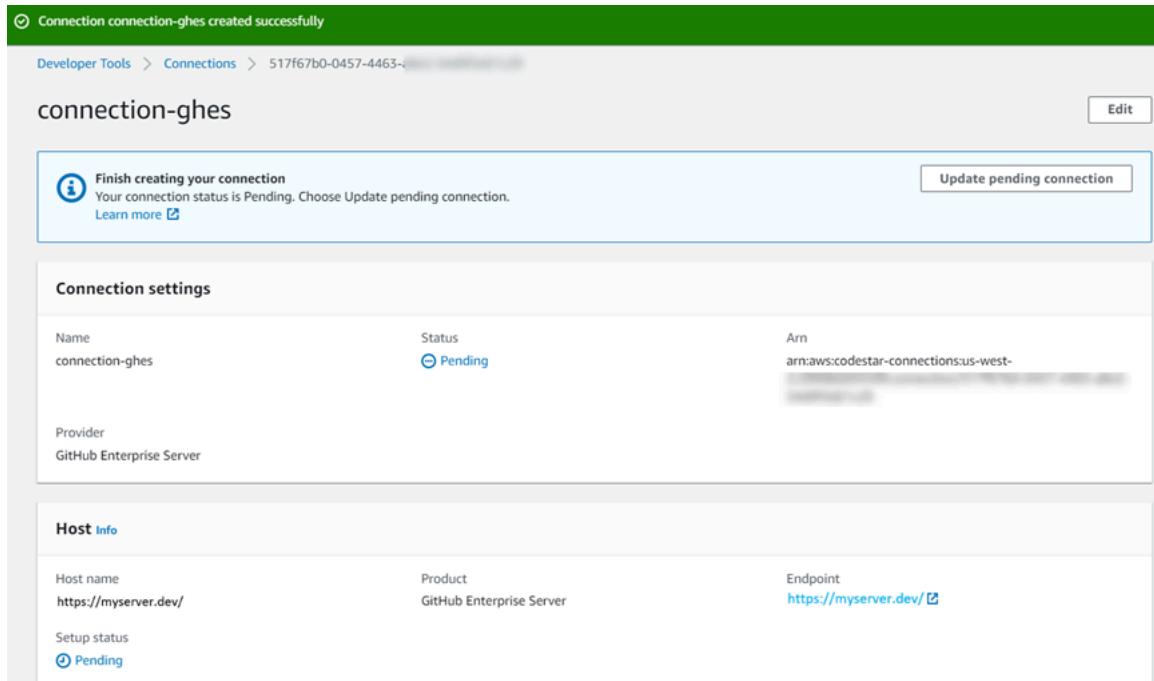
**Security group IDs**  
Choose the security group or groups for the VPC in which your GitHub Enterprise Server is configured.

Security group ID

**TLS certificate - optional**  
If you have a private certificate authority behind a VPC or you are using a self-signed certificate paste the TLS certificate here.

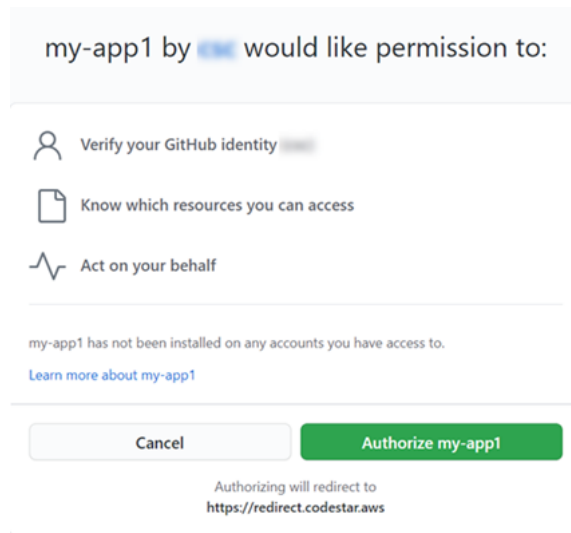
4. GitHub エンタープライズサーバーに接続を選択します。作成された接続は、Pending (保留中) のステータスで表示されます。指定したサーバ情報との接続用に、ホストリソースが作成されます。ホスト名には、URL が使用されます。
5. 保留中の接続の更新を選択します。



6. プロンプトが表示されたら、GitHub エンタープライズログインページで、エンタープライズ認証情報を使用してサインインします GitHub。
7. GitHub アプリの作成ページで、アプリの名前を選択します。

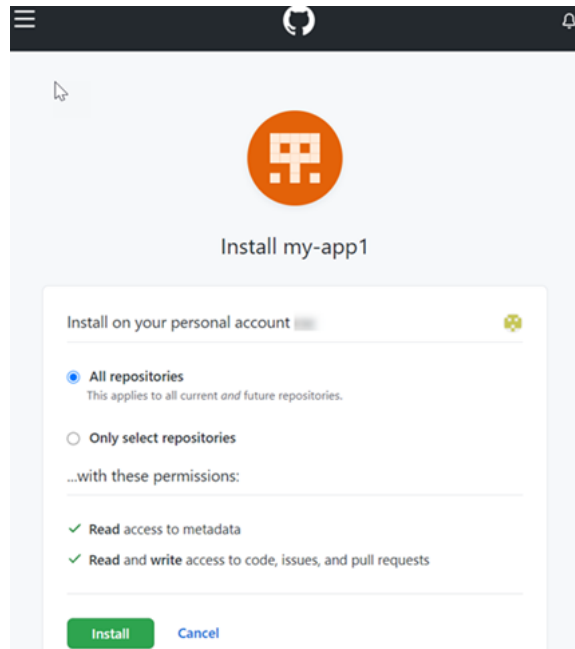


8. GitHub 認証ページで、「<app-name> の承認」を選択します。



9. アプリのインストールページで、コネクタアプリをインストールする準備ができていることを示すメッセージが表示されます。複数の組織がある場合は、アプリをインストールする組織を選択するように求められる場合があります。

アプリをインストールするリポジトリ設定を選択します。[Install] (インストール) を選択します。



10. 接続ページには、作成された接続が Available (使用可能) ステータスで表示されます。

### Enterprise Server (CLI) への接続 GitHubを作成する

AWS Command Line Interface ( AWS CLI) を使用して接続を作成できます。

これを行うには、create-host および create-connection コマンドを使用します。

#### **⚠ Important**

AWS CLI または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDINGステータスです。CLI または の接続を作成したら AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを にしますAVAILABLE。

### ステップ 1: GitHub Enterprise Server のホストを作成するには (CLI)

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-host コマンドを実行し、接続に --name、--provider-type、および --provider-endpointを指定します。この例では、サードパーティープロバイダー名は GitHubEnterpriseServer で、エンドポイントは my-instance.dev です。

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

成功した場合、このコマンドは次のようなホストの Amazon リソースネーム (ARN) 情報を返します。

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

この手順の後、ホストのステータスは PENDING になります。

2. コンソールでホストのセットアップを完了し、ホストのステータスを Available に移行します。詳細については、「[保留中のホストをセットアップする](#)」を参照してください。

ステップ 2: コンソールで保留中のホストを設定するには

1. にサインイン AWS Management Console し、でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. コンソールでホストのセットアップを完了し、ホストのステータスを Available に移行します。 [保留中のホストをセットアップする](#) を参照してください。

ステップ 3: Enterprise Server (CLI) GitHub の接続を作成するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-connection コマンドを実行し、接続 --connection-name に --host-arn とを指定します。

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

成功した場合、このコマンドは次のような接続 ARN 情報を返します。

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

2. コンソールを使用して、保留中の接続を設定します。詳細については、「[保留中の接続の更新](#)」を参照してください。

## ステップ 4: コンソールで GitHub Enterprise Server の接続を完了するには

1. にサインイン AWS Management Console し、 でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. コンソールを使用して、保留中の接続を設定し、接続のステータスを Available に移行します。詳細については、「[保留中の接続の更新](#)」を参照してください。

## への接続を作成する GitLab

AWS Management Console または AWS Command Line Interface ( AWS CLI) を使用して、gitlab.com でホストされているリポジトリへの接続を作成できます。

### Note

でこの接続のインストールを承認することで GitLab、データを処理するためのアクセス許可を当社のサービスに付与し、アプリケーションをアンインストールすることでいつでもアクセス許可を取り消すことができます。

開始する前に:

- でアカウントを作成しておく必要があります GitLab。

### Note

Connections は、接続の作成と承認に使用されたアカウント用のアクセスだけを提供しません。

### Note

で所有者ロールを持つ接続を作成し GitLab、その接続を などのリソースを含むリポジトリで使用できます CodePipeline。グループ内のリポジトリでは、グループの所有者である必要はありません。

## トピック

- [への接続 GitLabを作成する \(コンソール\)](#)

- [への接続 GitLabを作成する \(CLI\)](#)

## への接続 GitLabを作成する (コンソール)

コンソールを使用して接続を作成できます。

### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN に接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

### ステップ 1: 接続の作成

1. にサインインし AWS Management Console、 で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. [設定] を選択して、次に [接続] を選択します。[Create connection] (接続の作成) を選択します。
3. GitLab リポジトリへの接続を作成するには、「プロバイダーの選択」で、「」を選択します GitLab。[接続名] に、作成する接続の名前を入力します。 に接続 GitLabを選択します。



Developer Tools > Connections > Create connection

## Create a connection Info

### Select a provider

Bitbucket

GitHub

GitHub Enterprise Server

GitLab

### Create GitLab connection Info

Connection name

▶ **Tags - optional**

**Connect to GitLab**

4. のサインインページ GitLab が表示されたら、認証情報を使用してログインし、「サインイン」を選択します。
5. 認証ページに、GitLab アカウントにアクセスするための接続の承認を要求するメッセージが表示されます。

[承認] を選択します。

## Authorize **codestar-connections** to use your account?

An application called **codestar-connections** is requesting access to your GitLab account. This application was created by **Amazon AWS**. Please note that this application is not provided by GitLab and you should verify its authenticity before allowing access.

This application will be able to:

- **Access the authenticated user's API**  
Grants complete read/write access to the API, including all groups and projects, the container registry, and the package registry.
- **Read the authenticated user's personal information**  
Grants read-only access to the authenticated user's profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under /users.
- **Read Api**  
Grants read access to the API, including all groups and projects, the container registry, and the package registry.
- **Allows read-only access to the repository**  
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.
- **Allows read-write access to the repository**  
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).

6. ブラウザは接続コンソールページに戻ります。GitLab 「接続の作成」で、新しい接続が接続名に表示されます。
7. に接続 GitLabを選択します。

接続が正常に作成されると、成功バナーが表示されます。接続の詳細は、[接続設定] ページに表示されます。

## への接続 GitLabを作成する (CLI)

AWS Command Line Interface ( AWS CLI) を使用して接続を作成できます。

これを行うには、create-connection コマンドを使用します。

### ⚠ Important

AWS CLI または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDINGステータスです。CLI または の接続を作成したら AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを にしますAVAILABLE。

## への接続を作成するには GitLab

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-connection コマンドを実行し、接続--connection-nameに --provider-typeと を指定します。この例では、サードパーティープロバイダー名は GitLab で、指定された接続名は MyConnection です。

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

成功した場合、このコマンドは次のような接続 ARN 情報を返します。

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

2. コンソールを使用して接続を完了します。詳細については、「[保留中の接続の更新](#)」を参照してください。

## セルフマネージドへの接続 GitLabを作成する

セルフマネージドインストールを使用して、GitLab Enterprise Edition または GitLab Community Edition の接続を作成できます。

AWS Management Console または AWS Command Line Interface ( AWS CLI) を使用して、GitLab セルフマネージド用の接続とホストを作成できます。

### Note

この接続アプリケーションを GitLab セルフマネージドで承認することで、データを処理するためのアクセス許可を当社のサービスに付与し、アプリケーションをアンインストールすることでいつでもアクセス許可を取り消すことができます。

GitLab セルフマネージドへの接続を作成する前に、次の手順で説明するように、接続に使用するホストを作成する必要があります。インストール済みプロバイダー用のホスト作成ワークフローの概要については、「[ホストを作成または更新するワークフロー](#)」を参照してください。

オプションで VPC を使用してホストを設定できます。ホストリソース用のネットワークおよび VPC 設定の詳細については、「[\(オプション\) 前提条件: 接続用のネットワーク設定または Amazon VPC 設定](#)」および「[ホストの VPC 設定のトラブルシューティング](#)」を参照してください。

開始する前に:

- アカウントを既に作成 GitLab し、セルフマネージドインストールで GitLab Enterprise Edition または GitLab Community Edition を持っている必要があります。詳細については、[https://docs.gitlab.com/ee/subscriptions/self\\_managed/](https://docs.gitlab.com/ee/subscriptions/self_managed/) を参照してください。

### Note

Connections は、接続の作成と承認に使用されたアカウント用のアクセスだけを提供します。

**Note**

で所有者ロールを持つリポジトリへの接続を作成し GitLab、その接続を などのリソースとともに使用できます CodePipeline。グループ内のリポジトリでは、グループの所有者である必要はありません。

- スコープダウンアクセス許可のみを持つ GitLab 個人アクセストークン (PAT) を既に作成しておく必要があります。API。詳細については、[https://docs.gitlab.com/ee/user/profile/personal\\_access\\_tokens.html](https://docs.gitlab.com/ee/user/profile/personal_access_tokens.html) を参照してください。PAT を作成して使用するには、管理者である必要があります。

**Note**

PAT はホストの認証に使用され、それ以外の方法で保存または接続に使用されることはありません。ホストを設定するには、一時的な PAT を作成し、ホストを設定した後に PAT を削除できます。

## トピック

- [GitLab セルフマネージドへの接続を作成する \(コンソール\)](#)
- [セルフマネージド \(CLI\) への接続 GitLabを作成する](#)

## GitLab セルフマネージドへの接続を作成する (コンソール)

以下のステップを使用して、ホストと、コンソールで GitLab セルフマネージドへの接続を作成します。VPC でホストをセットアップする際の考慮事項については、「[\(オプション\) 前提条件: 接続用のネットワーク設定または Amazon VPC 設定](#)」を参照してください。

**Note**

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN に接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

**Note**

1 つの GitLab セルフマネージド型インストール用のホストを作成し、そのホストへの 1 つ以上の GitLab セルフマネージド型接続を管理できます。

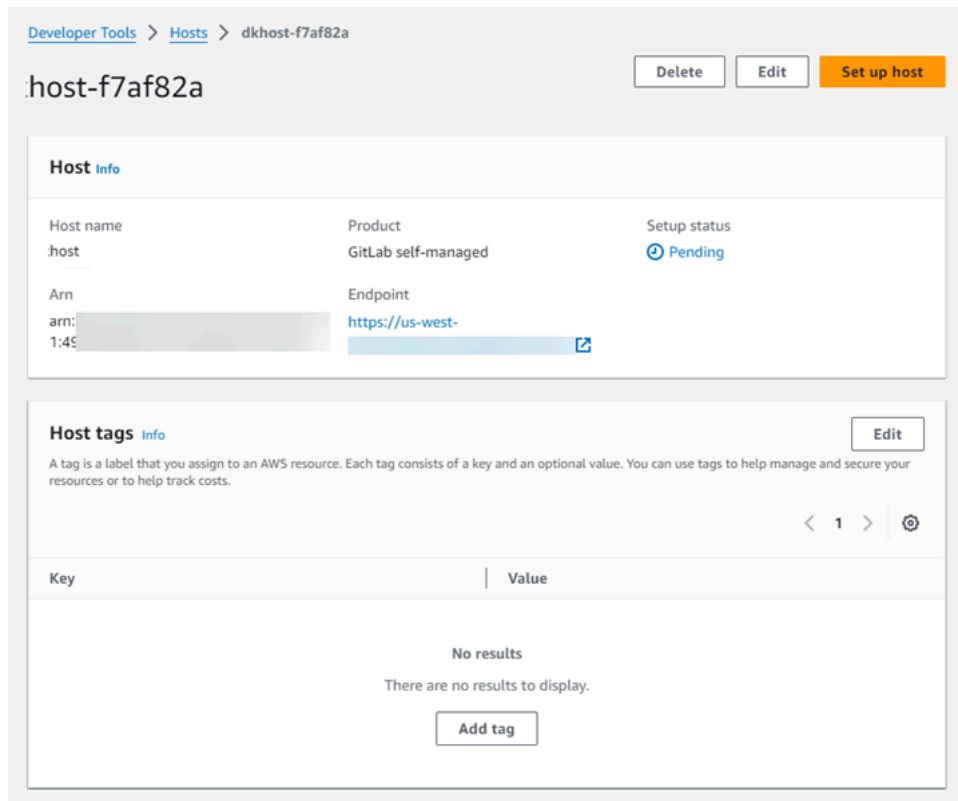
**ステップ 1: ホストを作成する**

1. にサインインし AWS Management Console、 で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. [Hosts (ホスト)] タブで、[Create host (ホストの作成)] を選択します。
3. [ホスト名] に、ホストに使用する名前を入力します。
4. 「プロバイダーの選択」で、GitLab セルフマネージド を選択します。
5. [URL] に、プロバイダーがインストールされているインフラストラクチャのエンドポイントを入力します。
6. サーバーが Amazon VPC 内に設定されていて、VPC に接続する場合は、Use a VPC (VPC を使用) を選択します。それ以外の場合、[No VPC] を選択します。
7. (オプション) Amazon VPC でホストを起動し、VPC に接続する場合は、[VPC を使用] を選択して、以下を完了します。
  - a. [VPC ID] で、VPC ID を選択します。ホストがインストールされているインフラストラクチャに VPC を選択するか、VPN または Direct Connect を介してインスタンスにアクセスできる VPC を選択します。
  - b. プライベート VPC を設定していて、非公開認証局を使用して TLS 検証を実行するようにホストを設定している場合は、[TLS 証明書] に証明書 ID を入力します。TLS 証明書の値は証明書のパブリックキーです。
8. [Create host] (ホストの作成) を選択します。
9. ホストの詳細ページが表示されたら、ホストの作成に伴ってホストのステータスが変化します。

**Note**

ホスト設定に VPC 設定が含まれている場合は、ホストネットワークコンポーネントのプロビジョニングに数分間かかります。

ホストのステータスがPending (保留中) になるのを待ってから、セットアップを完了します。詳細については、「[保留中のホストをセットアップする](#)」を参照してください。



The screenshot shows the AWS Developer Tools console for a host named 'host-f7af82a'. The breadcrumb navigation is 'Developer Tools > Hosts > dkhost-f7af82a'. At the top right, there are buttons for 'Delete', 'Edit', and 'Set up host'. The host name 'host-f7af82a' is displayed. Below this is a 'Host Info' section with a table:

Host name	Product	Setup status
host	GitLab self-managed	Pending

Below the table, the 'Arn' is partially visible as 'arn:1:4...' and the 'Endpoint' is 'https://us-west-...'. There is also a 'Host tags Info' section with an 'Edit' button and a table with columns 'Key' and 'Value'. The table is empty, showing 'No results' and 'There are no results to display.' with an 'Add tag' button.

## ステップ 2: 保留中のホストを設定する

1. [ホストをセットアップ] を選択します。
2. [**host\_name** のセットアップ] ページが表示されます。「個人用アクセストークンの提供」で、スコープダウンされたアクセス許可のみを持つ GitLab PAT を指定します。api

### Note

PAT を作成して使用できるのは管理者のみです。

## Set up myhostgl

### Provide personal access token

To set up GitLab self-managed, provide your personal access token from GitLab. The personal access token is required to have the following scoped-down permissions only: api.

Cancel

Continue

- ホストが正常に登録されると、ホストの詳細ページが表示され、ホストのステータスが Available (使用可能) になります。

myhostgl-5

Delete

Edit

Set up host

### Host Info

Host name

myhostgl

Product

GitLab self-managed

Setup status

✔ Available

Arn

[Redacted Arn]

Endpoint

[Redacted Endpoint]

### Host tags Info

Edit

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help manage and secure your resources or to help track costs.

< 1 >





### ステップ 3: 接続を作成する

1. にサインインし AWS Management Console、 で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. [設定] を選択して、次に [接続] を選択します。[Create connection] (接続の作成) を選択します。
3. GitLab リポジトリへの接続を作成するには、「プロバイダーの選択」で、GitLab セルフマネージドを選択します。[接続名] に、作成する接続の名前を入力します。

Developer Tools > Connections > Create connection

## Create a connection Info

**Select a provider**

Bitbucket  GitHub  GitHub Enterprise Server

GitLab  GitLab self-managed

**Connection Settings Info**

**Connection name**  
Give your connection a name.

**URL**  
The endpoint of the server to connect to.

**Use a VPC**  
If your GitLab self-managed is only accessible in a VPC, configure details here.  
Otherwise, skip this step.  
Complete these steps in the same AWS Region as your VPC.

**VPC ID**  
Choose the VPC in which your GitLab self-managed is configured.

**Subnet ID**  
Choose the subnet in which your GitLab self-managed is configured.

4. [URL] に、サーバーのエンドポイントを入力します。
5. Amazon VPC でサーバーを起動し、VPC に接続する場合は、[Use a VPC] (VPC を使用) をクリックして、以下を完了します。
  - a. [VPC ID] で、VPC ID を選択します。ホストがインストールされているインフラストラクチャに VPC を選択するか、VPN または Direct Connect を介してホストにアクセスできる VPC を選択します。
  - b. [サブネット ID] で、[Add] を選択します。このフィールドで、ホストに使用するサブネット ID を選択します。最大 10 個のサブネットを選択できます。

ホストがインストールされているインフラストラクチャにサブネットを選択するか、VPN または Direct Connect を介してインストールされたホストにアクセスできるサブネットを選択します。

- c. [Security group IDs] (セキュリティグループ ID) で、[Add] (追加) を選択します。このフィールドで、ホストに使用するセキュリティグループを選択します。最大 10 個のセキュリティグループを選択できます。

ホストがインストールされているインフラストラクチャにセキュリティグループを選択するか、VPN または Direct Connect を介してインストールされたホストにアクセスできるセキュリティグループを選択します。

- d. プライベート VPC を設定していて、非公開認証局を使用して TLS 検証を実行するようにホストを設定している場合は、[TLS 証明書] に証明書 ID を入力します。TLS 証明書の値は、証明書のパブリックキーである必要があります。
6. GitLab セルフマネージド に接続する を選択します。作成された接続は、Pending (保留中) のステータスで表示されます。指定したサーバ情報との接続用に、ホストリソースが作成されます。ホスト名には、URL が使用されます。
  7. 保留中の接続の更新を選択します。
  8. のサインインページ GitLab が表示されたら、認証情報を使用してログインし、「 でサインイン」を選択します。
  9. 認証ページに、GitLab アカウントにアクセスするための接続の承認を要求するメッセージが表示されます。  
[承認] を選択します。
  10. ブラウザは接続コンソールページに戻ります。GitLab 「接続の作成」で、新しい接続が接続名に表示されます。
  11. GitLab セルフマネージド に接続する を選択します。

接続が正常に作成されると、成功バナーが表示されます。接続の詳細は、[接続設定] ページに表示されます。

## セルフマネージド (CLI) への接続 GitLabを作成する

AWS Command Line Interface ( AWS CLI) を使用して、セルフマネージド型の GitLabホストと接続を作成できます。

これを行うには、create-host および create-connection コマンドを使用します。

**⚠ Important**

AWS CLI または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDING ステータスです。CLI または の接続を作成したら AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを にします AVAILABLE。

**ステップ 1: GitLab セルフマネージド型のホストを作成するには (CLI)**

1. ターミナル (Linux/macOS/Unix) または コマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-host コマンドを実行し、接続に --name、--provider-type、および --provider-endpoint を指定します。この例では、サードパーティープロバイダー名は GitLabSelfManaged で、エンドポイントは my-instance.dev です。

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

成功した場合、このコマンドは次のようなホストの Amazon リソースネーム (ARN) 情報を返します。

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

この手順の後、ホストのステータスは PENDING になります。

2. コンソールを使用してホストのセットアップを完了し、次のステップでホストのステータスを Available に移行します。

**ステップ 2: コンソールで保留中のホストを設定するには**

1. にサインイン AWS Management Console し、 でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. コンソールでホストのセットアップを完了し、ホストのステータスを Available に移行します。 [保留中のホストをセットアップする](#) を参照してください。

### ステップ 3: GitLab セルフマネージド型の接続を作成するには (CLI)

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-connection コマンドを実行し、接続--connection-nameに --host-arnとを指定します。

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

成功した場合、このコマンドは次のような接続 ARN 情報を返します。

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

2. 次のステップでコンソールを使用して、保留中の接続を設定します。

### ステップ 4: コンソールで GitLab セルフマネージドの接続を完了するには

1. にサインイン AWS Management Console し、 でデベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。
2. コンソールを使用して、保留中の接続を設定し、接続のステータスを Available に移行します。詳細については、「[保留中の接続の更新](#)」を参照してください。

### 保留中の接続の更新

AWS Command Line Interface ( AWS CLI) または を介して作成された接続 AWS CloudFormation は、デフォルトで PENDINGステータスです。AWS CLI または との接続を作成したら AWS CloudFormation、コンソールを使用して接続を更新し、ステータスを にしますAVAILABLE。

#### Note

保留中の接続を更新するには、コンソールを使用する必要があります。AWS CLIを使用して保留中の接続を更新できません。

コンソールを初めて使用してサードパーティープロバイダーに新しい接続を追加するときは、接続に関連付けられたインストールを使用して、サードパーティープロバイダと OAuth ハンドシェイクを完了する必要があります。

デベロッパーツールコンソールを使用して、保留中の接続を完了できます。

接続を完了するには

1. で AWS デベロッパーツールコンソールを開きます <https://console.aws.amazon.com/codesuite/settings/connections>。

2. [設定] > [接続] を選択します。

AWS アカウントに関連付けられているすべての接続の名前が表示されます。

3. [Name (名前)] で、更新する保留中の接続の名前を選択します。

Pending (保留中) ステータスの接続を選択すると、Update connection (接続の更新) が有効になります。

4. [保留中の接続の更新]を選択します。

5. [Connect to Bitbucket] (Bitbucket に接続) ページの [Connection name] (接続名) で、接続名を確認します。

[Bitbucket apps] (Bitbucket アプリ) で、アプリのインストールを選択するか、[Install a new app] (新しいアプリをインストールする) を選択してアプリを作成します。

Connect to Bitbucket

Bitbucket connection settings [Info](#)

Connection name

a-connection

Bitbucket apps

Bitbucket apps create a link for your connection with Bitbucket. To start, install a new app and save this connection.

or

6. アプリのインストールページで、AWS CodeStar アプリが Bitbucket アカウントに接続しようとしていることを示すメッセージが表示されます。[アクセス権の付与] を選択します。



### AWS CodeStar requests access

This app is hosted at <https://codestar-connections.webhooks.aws>

- Read your account information
- Read your repositories and their pull requests
- Administer your repositories
- Read and modify your repositories

Authorize for

#### Allow AWS CodeStar to do this?

This 3rd party vendor has not provided a privacy policy or terms of use.

Atlassian's Privacy Policy is not applicable to the use of this App.

**Grant access** Cancel

7. 新規インストールの接続 ID が表示されます。[Complete connection (接続の完了)] を選択します。

## 接続を一覧表示する

開発者用ツールコンソールまたは AWS Command Line Interface (AWS CLI) 内の `list-connections` コマンドを使用して、アカウント内の接続のリストを表示できます。

### 接続を一覧表示する (コンソール)

接続を一覧表示するには

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [設定] > [接続] を選択します。

### 3. 接続の名前、ステータス、および ARN を表示します。

#### 接続を一覧表示する (CLI)

を使用して AWS CLI、サードパーティーのコードリポジトリへの接続を一覧表示できます。GitHub Enterprise Server への接続など、ホストリソースに関連付けられた接続の場合、出力はさらにホスト ARN を返します。

これを行うには、list-connections コマンドを使用します。

#### 接続を一覧表示するには

- ターミナル (Linux、macOS または Unix) または コマンドプロンプト (Windows) を開き、AWS CLI を使用して list-connections コマンドを実行します。

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

このコマンドで、以下の出力が返ります。

```
{
  "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"
}
```

## 接続を削除

デベロッパーツールコンソールまたは AWS Command Line Interface (AWS CLI) の `delete-connection` コマンドを使用して、接続を削除できます。

### トピック

- [接続を削除する \(コンソール\)](#)
- [接続を削除する \(CLI\)](#)

### 接続を削除する (コンソール)

接続を削除するには

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [設定] > [接続] を選択します。
3. [Connection name (接続名)] で、削除する接続の名前を選択します。
4. [削除] を選択します。
5. フィールドに「**delete**」と入力して確認し、[Delete (削除)] を選択します。

#### Important

このアクションを元に戻すことはできません。

### 接続を削除する (CLI)

AWS Command Line Interface (AWS CLI) を使用して接続を削除できます。

これを行うには、`delete-connection` コマンドを使用します。

#### Important

コマンドを実行すると、接続は削除されます。確認のダイアログボックスは表示されません。新しい接続を作成することはできますが、Amazon リソースネーム (ARN) は再利用されません。



## 接続を削除するには

- ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して delete-connection コマンド AWS CLI を実行し、削除する接続の ARN を指定します。

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

このコマンドは何も返しません。

## タグ接続リソース

タグは、AWS リソース AWS に割り当てるカスタム属性ラベルです。各 AWS タグには 2 つの部分があります。

- タグキー (例: CostCenter、Environment、または Project)。タグキーでは、大文字と小文字が区別されます。
- タグ値と呼ばれるオプションのフィールド (111122223333、Production、チーム名など)。タグ値を省略すると、空の文字列を使用した場合と同じになります。タグキーと同様に、タグ値では大文字と小文字が区別されます。

これらは共にキーと値のペアと呼ばれます。

コンソールまたは CLI を使用して、リソースのタグ付けをします。

AWS では、次のリソースタイプにタグを付けることができます CodeConnections。

- 接続
- [ホスト]

このステップでは、の最新バージョンが既にインストールされているか、最新バージョンに AWS CLI 更新されていることを前提としています。詳細については、「AWS Command Line Interface ユーザーガイド」の「[AWS CLIのインストール](#)」を参照してください。

タグを使用してリソースを識別、整理、追跡するだけでなく、AWS Identity and Access Management (IAM) ポリシーのタグを使用して、リソースを表示および操作できるユーザーを制御できます。タグベースのアクセスポリシーの例については、「[タグを使用して AWS CodeConnections リソースへのアクセスを制御する](#)」を参照してください。

## トピック

- [リソースのタグ付け \(コンソール\)](#)
- [タグリソース \(CLI\)](#)

## リソースのタグ付け (コンソール)

コンソールを使用して、接続リソースにタグを追加、更新、または削除できます。

## トピック

- [接続リソースにタグを追加する \(コンソール\)](#)
- [接続リソース \(コンソール\) のタグを表示する](#)
- [接続リソース \(コンソール\) のタグを編集する](#)
- [接続リソースからのタグを削除する \(コンソール\)](#)

## 接続リソースにタグを追加する (コンソール)

コンソールを使用して、既存の接続またはホストにタグを追加します。

### Note

Enterprise Server などの GitHubインストール済みプロバイダーの接続を作成し、ホストリソースも作成されると、作成時のタグは接続にのみ追加されます。これにより、ホストを新しい接続で再利用する場合は、ホストに個別にタグを付けることができます。ホストにタグを追加するには、次の手順に従います。

## 接続にタグを追加するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Connections (接続)] タブを選択します。
3. 編集する接続を選択します。接続設定のページが表示されます。
4. [Connection tags] (接続タグ) で、[Edit] (編集) を選択します。[Edit Connection tags] (接続タグの編集) ページが表示されます。

- [Key] フィールドと [Value] フィールドに、追加するタグのセットごとにキーペアを入力します。 ([値] フィールドはオプションです。) 例えば、[キー] では、「**Project**」と入力します。 [値] には「**ProjectA**」と入力します。

**Edit Connection tags**

**Connection tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help manage and secure your resources or to help track costs.

Key  Value - optional

- (オプション) [タグを追加] をクリックして行を追加し、さらにタグを入力します。
- [送信] を選択します。タグは、接続の設定の下に表示されます。

ホストにタグを追加するには

- コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
- [Settings] (設定) で、[Connections] (接続) を選択します。[Hosts] (ホスト) タブを選択します。
- 編集するホストを選択します。ホスト設定のページが表示されます。
- [Host tags] (ホストタグ) で、[Edit] (編集) を選択します。[Hosts Tag] (ホストタグ) ページが表示されます。
- [Key] フィールドと [Value] フィールドに、追加するタグのセットごとにキーペアを入力します。 ([値] フィールドはオプションです。) 例えば、[キー] では、「**Project**」と入力します。 [値] には「**ProjectA**」と入力します。

**Edit Host tags**

**Host tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help manage and secure your resources or to help track costs.

Key  Value - optional

6. (オプション) [Add tag] (タグの追加) を選択して行を追加し、さらにホストのタグを入力します。
7. [送信] を選択します。タグは、ホストの設定の下に表示されます。

### 接続リソース (コンソール) のタグを表示する

コンソールを使用して、既存のリソースのタグを表示できます。

#### 接続のタグを表示するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Connections (接続)] タブを選択します。
3. 表示する接続を選択します。接続設定のページが表示されます。
4. [Connection tags] で、[Key] 列と [Value] 列下の接続のタグを表示します。

#### ホストのタグを表示するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Hosts] (ホスト) タブを選択します。
3. 表示するホストを選択します。
4. [Host tags] で、[Key] 列と [Value] 列下のホストのタグを表示します。

### 接続リソース (コンソール) のタグを編集する

コンソールを使用して、接続リソースに追加されたタグを編集します。

#### 接続のタグを編集するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Connections (接続)] タブを選択します。
3. 編集する接続を選択します。接続設定のページが表示されます。
4. [Connection tags] (接続タグ) で、[Edit] (編集) を選択します。[Connection tags] (接続タグ) ページが表示されます。

5. [キー] フィールドと [値] フィールドに、必要に応じて各フィールドの値を更新します。例えば、**Project** キーの場合は、[Value] で、**ProjectA** を **ProjectB** に変更します。
6. [送信] を選択します。

#### ホストのタグを編集するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Hosts] (ホスト) タブを選択します。
3. 編集するホストを選択します。ホスト設定のページが表示されます。
4. [Host tags] (ホストタグ) で、[Edit] (編集) を選択します。[Hosts Tag] (ホストタグ) ページが表示されます。
5. [キー] フィールドと [値] フィールドに、必要に応じて各フィールドの値を更新します。例えば、**Project** キーの場合は、[Value] で、**ProjectA** を **ProjectB** に変更します。
6. [送信] を選択します。

#### 接続リソースからのタグを削除する (コンソール)

コンソールを使用して、接続リソースからタグを削除できます。関連付けられているリソースからタグを削除すると、そのタグが削除されます。

#### 接続のタグを削除するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Connections (接続)] タブを選択します。
3. 編集する接続を選択します。接続設定のページが表示されます。
4. [Connection tags] (接続タグ) で、[Edit] (編集) を選択します。[Connection tags] (接続タグ) ページが表示されます。
5. 削除する各タグのキーと値の横にある [Remove tag] を選択します。
6. [送信] を選択します。

#### ホストのタグを削除するには

1. コンソールにサインインします。ナビゲーションパネルから [Settings (設定)] を選択します。
2. [Settings] (設定) で、[Connections] (接続) を選択します。[Hosts] (ホスト) タブを選択します。

3. 編集するホストを選択します。ホスト設定のページが表示されます。
4. [Host tags] (ホストタグ) で、[Edit] (編集) を選択します。[Hosts Tag] (ホストタグ) ページが表示されます。
5. 削除する各タグのキーと値の横にある [Remove tag] を選択します。
6. [送信] を選択します。

## タグリソース (CLI)

CLI を使用して、接続リソースのタグを表示、追加、更新、または削除できます。

### トピック

- [接続リソースにタグを追加する \(CLI\)](#)
- [接続リソース \(CLI\) のタグを表示する](#)
- [接続リソース \(CLI\) のタグを編集する](#)
- [接続リソース \(CLI\) からのタグを削除する](#)

## 接続リソースにタグを追加する (CLI)

を使用して AWS CLI、接続内のリソースにタグを付けることができます。

ターミナルまたはコマンドラインで、タグを追加するリソースの Amazon リソース名前 (ARN)、および追加するタグのキーと値を指定して tag-resource コマンドを実行します。複数のタグを追加できます。

### 接続にタグを追加するには

1. リソースの ARN を取得します。[接続を一覧表示する](#) に示されている list-connections コマンドを使用して、接続ARNを取得します。
2. ターミナルまたはコマンドラインで、tag-resource コマンドを実行します。

例えば、次のコマンドを使用して、接続に 2 つのタグ、*Project* という名前のタグキーに *ProjectA* のタグ値、および *true* というタグ値 *ReadOnly* を持つ という名前のタグキーをタグ付けします。

```
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
```

成功した場合、このコマンドは何も返しません。

ホストにタグを追加するには

1. リソースの ARN を取得します。[ホストを一覧表示](#) に示されている list-hosts コマンドを使用して、ホスト ARN を取得します。
2. ターミナルまたはコマンドラインで、tag-resource コマンドを実行します。

例えば、次のコマンドを使用して、2 つのタグ、Project という名前のタグキー、*Project ProjectA*、および *true* というタグ値 *IscontainerBased* を持つ という名前のタグキーでホストにタグ付けします。

```
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
```

成功した場合、このコマンドは何も返しません。

接続リソース (CLI) のタグを表示する

を使用して AWS CLI、接続リソースの AWS タグを表示できます。タグが追加されていない場合、返されるリストは空になります。を使用する list-tags-for-resource コマンドを使用して、接続またはホストに追加されたタグを表示します。

接続のタグを表示するには

1. リソースの ARN を取得します。[接続を一覧表示する](#) に示されている list-connections コマンドを使用して、接続ARNを取得します。
2. ターミナルまたはコマンドラインで、list-tags-for-resource コマンドを実行します。例えば、接続のタグキーとタグ値の一覧を表示するには、次のコマンドを使用します。

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

このコマンドは、リソースに関連付けられているタグを返します。この例は、接続に対して返される 2 つのキーと値のペアを示しています。

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

ホストのタグを表示するには

1. リソースの ARN を取得します。[ホストを一覧表示](#) に示されている list-hosts コマンドを使用して、ホスト ARN を取得します。
2. ターミナルまたはコマンドラインで、list-tags-for-resource コマンドを実行します。例えば、ホストのタグキーとタグ値の一覧を表示するには、次のコマンドを使用します。

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

このコマンドは、リソースに関連付けられているタグを返します。この例は、ホストに対して返される 2 つのキーと値のペアを示しています。

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



## 接続リソース (CLI) のタグを編集する

を使用して AWS CLI、リソースのタグを編集できます。既存のキーの値を変更したり、別のキーを追加できます。

ターミナルまたはコマンドラインで、タグを更新するリソースの ARN を指定して、tag-resource コマンドを実行し、更新するタグキーとタグ値を指定します。

タグを編集すると、指定されていないタグキーは保持されますが、同じキーで新しい値を持つものはすべて更新されます。edit コマンドで追加された新しいキーは、新しいキーと値のペアとして追加されます。

接続のタグを編集するには

1. リソースの ARN を取得します。[接続を一覧表示する](#) に示されている list-connections コマンドを使用して、接続ARNを取得します。
2. ターミナルまたはコマンドラインで、tag-resource コマンドを実行します。

この例では、キーの値 Project が ProjectB に変更されています。

```
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
```

成功した場合、このコマンドは何も返しません。接続に関連付けられているタグを確認するには、list-tags-for-resource コマンドを実行します。

ホストのタグを編集するには

1. リソースの ARN を取得します。[ホストを一覧表示](#) に示されている list-hosts コマンドを使用して、ホスト ARN を取得します。
2. ターミナルまたはコマンドラインで、tag-resource コマンドを実行します。

この例では、キーの値 Project が ProjectB に変更されています。

```
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
```

成功した場合、このコマンドは何も返しません。ホストに関連付けられているタグを確認するには、`list-tags-for-resource` コマンドを実行します。

### 接続リソース (CLI) からのタグを削除する

を使用してリソースからタグ AWS CLI を削除するには、次の手順に従います。関連付けられているリソースからタグを削除すると、そのタグが削除されます。

#### Note

接続リソースを削除すると、削除されたリソースからすべてのタグの関連付けが削除されます。接続リソースを削除する前に、タグを削除する必要はありません。

ターミナルまたはコマンドラインで、タグを削除するリソースの ARN と削除するタグのタグキーを指定して、`untag-resource` コマンドを実行します。例えば、タグキー *Project* および *ReadOnly* の接続で複数のタグを削除するには *ReadOnly*、次のコマンドを使用します。

```
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
```

成功した場合、このコマンドは何も返しません。リソースに関連付けられているタグを確認するには、`list-tags-for-resource` コマンドを実行します。出力は、すべてのタグが削除されたことを示しています。

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

### 接続の詳細の表示

デベロッパーツールコンソールまたは AWS Command Line Interface (AWS CLI) の `get-connection` コマンドを使用して、接続の詳細を表示できます。を使用するには AWS CLI、の最新バージョンがインストールされているか、最新バージョンに AWS CLI 更新されている必要があります。詳細については、「AWS Command Line Interface ユーザーガイド」の「[AWS CLIのインストール](#)」を参照してください。

## 接続を表示するには (コンソール)

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [設定] > [接続] を選択します。
3. 表示する接続の横にあるボタンを選択して、[View details] をクリックします。
4. 接続に関する次の情報が表示されます。
  - 接続名。
  - 接続のプロバイダータイプ。
  - 接続ステータス。
  - 接続 ARN。
  - Enterprise Server などの GitHubインストール済みプロバイダー用に接続が作成された場合、接続に関連付けられたホスト情報。
  - Enterprise Server などの GitHubインストール済みプロバイダー用に接続が作成された場合、接続のホストに関連付けられたエンドポイント情報。
5. 接続のステータスが Pending (保留中) のときに接続を完了するには、保留中の接続の更新を選択します。詳細については、「[Update a pending connection](#)」を参照してください。

## 接続を表示するには (CLI)

- ターミナルまたはコマンドラインで、get-connection コマンドを実行します。例えば、arn:aws:codestar-connections:us-west-2:*account\_id*:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f ARN 値を持つ接続の詳細を表示するには、次のコマンドを使用します。

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

コマンドが成功すると、このコマンドから接続情報が返されます。

Bitbucket 接続の出力例 :

```
{
  "Connection": {
    "ConnectionName": "MyConnection",
```

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

GitHub 接続の出力例 :

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

GitHub Enterprise Server 接続の出力例 :

```
{
  "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:codeconnections:us-west-2:account_id:host/sdfsdf-
EXAMPLE"
  }
}
```

## ホストの使用

インストール済みプロバイダータイプ (GitHub Enterprise Server など) への接続を作成するには、まず AWS Management Console を使用してホストを作成します。ホストは、プロバイダーがインス

トールされているインフラストラクチャを表すために作成するリソースです。次に、そのホストを使用して接続を作成します。詳細については、「[接続の使用](#)」を参照してください。

例えば、接続用のホストを作成して、インフラストラクチャを表すためにプロバイダーのサードパーティーアプリを登録できるようにします。プロバイダータイプに対してホストを1つ作成します。そのプロバイダータイプへのすべての接続がそのホストを使用します。

コンソールを使用してインストール済みプロバイダータイプ (GitHub Enterprise Server など) への接続を作成すると、コンソールがホストリソースを作成します。

## トピック

- [ホストを作成する](#)
- [保留中のホストをセットアップする](#)
- [ホストを一覧表示](#)
- [ホストを編集する](#)
- [ホストを削除する](#)
- [ホストの詳細の表示](#)

## ホストを作成する

AWS Management Console または AWS Command Line Interface (AWS CLI) を使用して、インフラストラクチャにインストールされているサードパーティーのコードリポジトリへの接続を作成できます。例えば、Amazon EC2 インスタンスで Enterprise Server を仮想マシンとして実行しているとします。GitHub Enterprise Server への接続を作成する前に、接続に使用するホストを作成します。

インストール済みプロバイダー用のホスト作成ワークフローの概要については、「[ホストを作成または更新するワークフロー](#)」を参照してください。

開始する前に:

- (オプション) VPC を使用してホストを作成する場合は、ネットワークまたは仮想プライベートクラウド (VPC) をあらかじめ作成しておく必要があります。
- インスタンスをあらかじめ作成しておく必要があります。VPC に接続するときは、ホストを VPC で起動しておく必要があります。

**Note**

各 VPC は、一度に 1 つのホストにのみ関連付けることができます。

オプションで VPC を使用してホストを設定できます。ホストリソース用のネットワークおよび VPC 設定の詳細については、「[\(オプション\) 前提条件: 接続用のネットワーク設定または Amazon VPC 設定](#)」および「[ホストの VPC 設定のトラブルシューティング](#)」を参照してください。

コンソールを使用してホストと GitHub Enterprise Server への接続を作成するには、「」を参照してください[Enterprise Server 接続を作成する GitHub \(コンソール\)](#)。コンソールでホストが作成されます。

コンソールを使用してホストと GitLab セルフマネージドへの接続を作成するには、「」を参照してください[セルフマネージドへの接続 GitLabを作成する](#)。コンソールでホストが作成されます。

(オプション) 前提条件: 接続用のネットワーク設定または Amazon VPC 設定

インフラストラクチャにネットワーク接続が設定されている場合は、このセクションをスキップできます。

ホストに VPC でのみアクセスできる場合は、続行する前に、これらの VPC 要件に従ってください。

### VPC の要件

オプションで VPC を使用してホストを作成することもできます。以下は、インストール用に設定した VPC に応じた、一般的な VPC 要件を示します。

- パブリックサブネットとプライベートサブネットを使用してパブリック VPC を構成できます。優先 CIDR ブロックまたはサブネットがない場合は、AWS アカウント にデフォルトの VPC を使用できます。
- プライベート VPC が設定されていて、非パブリック認証機関を使用して TLS 検証を実行するように GitHub Enterprise Server インスタンスを設定している場合は、ホストリソースの TLS 証明書を提供する必要があります。
- 接続によってホストが作成されると、ウェブフック用の VPC エンドポイント (PrivateLink) が自動的に作成されます。詳細については、「[AWS CodeConnections およびインターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)」を参照してください。

## • セキュリティグループの設定

- ホストの作成中に使用されるセキュリティグループには、ネットワークインターフェイスが GitHub Enterprise Server インスタンスに接続できるようにするインバウンドルールとアウトバウンドルールが必要です。
- GitHub Enterprise Server インスタンスにアタッチされたセキュリティグループ (ホスト設定の一部ではない) には、接続によって作成されたネットワークインターフェイスからのインバウンドおよびアウトバウンドのアクセスが必要です。
- VPC サブネットは、リージョン内の異なるアベイラビリティーゾーンに存在している必要があります。アベイラビリティーゾーンとは、他のアベイラビリティーゾーンで発生した障害から切り離すために作られた場所です。各サブネットが完全に 1 つのアベイラビリティーゾーン内に含まれている必要があります、1 つのサブネットが複数のゾーンに、またがることはできません。

VPC とサブネットの使用方法の詳細については、Amazon VPC ユーザーガイドの「[IPv4 用の VPC とサブネットのサイズ設定](#)」を参照してください。

## ホストセットアップ用に提供する VPC 情報

次のステップで接続用のホストリソースを作成するときは、以下を提供する必要があります。

- VPC ID: GitHub Enterprise Server インスタンスがインストールされているサーバーの VPC、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできる VPC の ID。
- サブネット ID IDs : GitHub Enterprise Server インスタンスがインストールされているサーバーのサブネットの ID、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるサブネット。
- セキュリティグループ : GitHub Enterprise Server インスタンスがインストールされているサーバーのセキュリティグループ、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるセキュリティグループ。
- エンドポイント: サーバーエンドポイントを準備して、次のステップに進みます。

VPC またはホスト接続のトラブルシューティングなどの詳細については、「[ホストの VPC 設定のトラブルシューティング](#)」を参照してください。

## アクセス許可の要件

ホスト作成プロセスの一環として、はユーザーに代わってネットワークリソース AWS CodeConnections を作成し、VPC 接続を容易にします。これには、 がホストからデータをクエリ

AWS CodeConnections するためのネットワークインターフェイス、PrivateLinkVPC エンドポイント、またはホストがウェブフック経由で接続にイベントデータを送信するためのネットワークインターフェイスが含まれます。これらのネットワークリソースを作成できるようにするには、ホストを作成するロールに次のアクセス許可があることを確認してください。

```
ec2:CreateNetworkInterface
ec2:CreateTags
ec2:DescribeDhcpOptions
ec2:DescribeNetworkInterfaces
ec2:DescribeSubnets
ec2>DeleteNetworkInterface
ec2:DescribeVpcs
ec2:CreateVpcEndpoint
ec2>DeleteVpcEndpoints
ec2:DescribeVpcEndpoints
```

VPC 内のアクセス許可またはホスト接続のトラブルシューティングの詳細については、「[ホストの VPC 設定のトラブルシューティング](#)」を参照してください。

ウェブフック VPC エンドポイントの詳細については、「[AWS CodeConnections およびインターフェイス VPC エンドポイント \(AWS PrivateLink \)](#)」を参照してください。

## トピック

- [接続用のホストを作成する \(コンソール\)](#)
- [接続用のホストを作成する \(CLI\)](#)

### 接続用のホストを作成する (コンソール)

GitHub Enterprise Server や GitLabセルフマネージドなどのインストール用の接続では、ホストを使用して、サードパーティープロバイダーがインストールされているインフラストラクチャのエンドポイントを表します。

#### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN に接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。



VPC でホストをセットアップする際の考慮事項については、「[セルフマネージドへの接続 GitLabを作成する](#)」を参照してください。

コンソールを使用してホストと GitHub Enterprise Server への接続を作成するには、「」を参照してください[Enterprise Server 接続を作成する GitHub \(コンソール\)](#)。コンソールでホストが作成されます。

コンソールを使用してホストと GitLab セルフマネージドへの接続を作成するには、「」を参照してください[セルフマネージドへの接続 GitLabを作成する](#)。コンソールでホストが作成されます。

#### Note

ホストは、GitHub Enterprise Server または GitLab セルフマネージドアカウントごとに 1 回のみ作成します。特定の GitHub Enterprise Server または GitLab セルフマネージドアカウントへのすべての接続は、同じホストを使用します。

### 接続用のホストを作成する (CLI)

AWS Command Line Interface (AWS CLI) を使用して、インストールされた接続用のホストを作成できます。

#### Note

ホストは、GitHub Enterprise Server アカウントごとに 1 回のみ作成します。特定の GitHub Enterprise Server アカウントへのすべての接続は、同じホストを使用します。

ホストを使用して、サードパーティーのプロバイダがインストールされているインフラストラクチャのエンドポイントを表します。CLI を使用してホストを作成するには、`create-host` コマンドを実行します。ホストの作成が完了すると、ホストのステータスが Pending (保留中) になります。次に、ホストを設定して、ホストのステータスが Available (使用可能) に移行します。ホストが使用可能になったら、接続を作成する手順を完了します。

#### Important

を通じて作成されたホスト AWS CLI は、デフォルトで Pending ステータスです。CLI でホストを作成後、コンソールでホストを設定し、ステータスを Available にします。

コンソールを使用してホストと GitHub Enterprise Server への接続を作成するには、「」を参照してください[Enterprise Server 接続を作成する GitHub \(コンソール\)](#)。コンソールでホストが作成されます。

コンソールを使用してホストと GitLab セルフマネージドへの接続を作成するには、「」を参照してください[セルフマネージドへの接続 GitLabを作成する](#)。コンソールでホストが作成されます。

## 保留中のホストをセットアップする

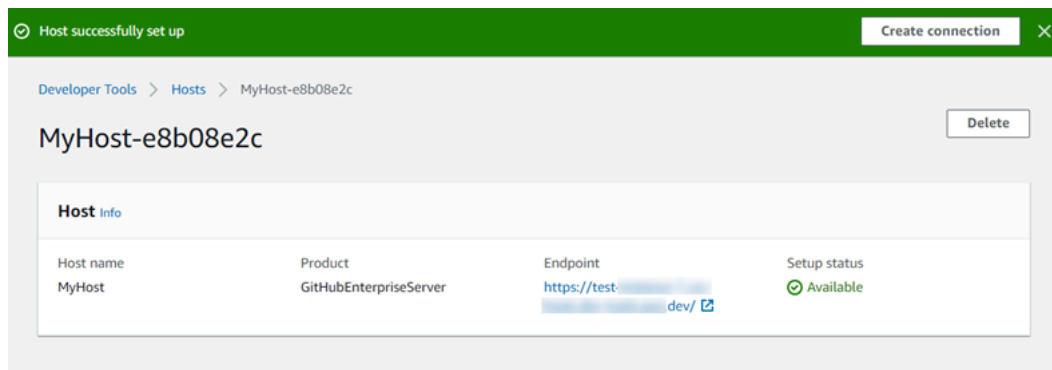
AWS Command Line Interface (AWS CLI) または SDK を使用して作成された接続のステータスは、デフォルトで Pending になります。AWS CLI または SDK を使用して接続を作成した後、コンソールで接続を編集し、ステータスを Available に変更します。

予めホストを作成しておく必要があります。詳細については、「[Create a host](#)」を参照してください。

保留中のホストをセットアップするには

ホストが作成されると、ステータスが Pending (保留中) になります。ホストを Pending から Available に移行するには、次の手順を実行します。このプロセスは、サードパーティープロバイダーとのハンドシェイクを実行し、ホストに AWS 接続アプリケーションを登録します。

1. AWS デベロッパーツールコンソールでホストのステータスが [Pending] (保留中) になった後、[Set up host] (ホストをセットアップ) を選択します。
2. GitLab セルフマネージド用のホストを作成する場合は、[セットアップ] ページが表示されます。[個人アクセストークンの提供] で、GitLab PAT に、api というスコープダウンされたアクセス許可のみを指定します。
3. GitHub Enterprise Server ログインページなどのサードパーティーのインストール済みプロバイダーのログインページでプロンプトが表示されたら、アカウントの認証情報を使用してログインします。
4. アプリのインストールページの [GitHub App name] (GitHub アプリ名) に、ホストにインストールするアプリの名前を入力します。Create GitHub App (GitHub アプリの作成) を選択します。
5. ホストが正常に登録されると、ホストの詳細ページが表示され、ホストのステータスが Available (使用可能) になります。



6. ホストが使用可能になった後も、接続の作成を続行できます。成功バナーで、[Create connection] (接続を作成する) を選択します。[[Create a connection](#)] (接続を作成する) の手順を完了します。

## ホストを一覧表示

開発者用ツールコンソールまたは AWS Command Line Interface (AWS CLI) 内の `list-connections` コマンドを使用して、アカウント内の接続のリストを表示できます。

### ホストを一覧表示 (コンソール)

ホストを一覧表示するには

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [Hosts] (ホスト) タブを選択します。ホストの名前、ステータス、および ARN を表示します。

### ホストを一覧表示 (CLI)

を使用して AWS CLI、インストールされているサードパーティープロバイダー接続のホストを一覧表示できます。

これを行うには、`list-hosts` コマンドを使用します。

ホストを一覧表示するには

- ターミナル (Linux、macOS または Unix) または コマンドプロンプト (Windows) を開き、AWS CLI を使用して `list-hosts` コマンドを実行します。

```
aws codeconnections list-hosts
```

このコマンドで、以下の出力が返ります。

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

## ホストを編集する

Pending ステータスのホストのホスト設定を編集できます。ホスト名、URL、または VPC 設定を編集できます。

同じ URL を複数のホストに使用することはできません。

### Note

VPC でホストをセットアップする際の考慮事項については、「[\(オプション\) 前提条件: 接続用のネットワーク設定または Amazon VPC 設定](#)」を参照してください。

ホストを編集するには

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [設定] > [接続] を選択します。
3. [Hosts] (ホスト) タブを選択します。

AWS アカウントに関連付けられ、選択した AWS リージョンで作成されたホストが表示されます。

4. ホスト名を編集するには、[Name] (名前) に新しい値を入力します

5. ホストエンドポイントを編集するには、[URL] に新しい値を入力します。
6. ホスト VPC 設定を編集するには、[VPC ID] に新しい値を入力します。
7. [Edit host] (ホストを編集) を選択します。
8. 更新された設定が表示されます。[Set up Pending host] (保留中のホストの設定) を選択します。

## ホストを削除する

デベロッパーツールコンソールまたは AWS Command Line Interface (AWS CLI) の delete-host コマンドを使用して、ホストを削除できます。

### トピック

- [ホストの削除 \(コンソール\)](#)
- [ホストの削除 \(CLI\)](#)

### ホストの削除 (コンソール)

ホストを削除するには

1. <https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
2. [Hosts] (ホスト) タブを選択します。[Name] (名前) で、削除するホストの名前を選択します。
3. [削除] を選択します。
4. フィールドに「**delete**」と入力して確認し、[Delete (削除)] を選択します。

#### Important

このアクションを元に戻すことはできません。

### ホストの削除 (CLI)

AWS Command Line Interface (AWS CLI) を使用してホストを削除できます。

これを行うには、delete-host コマンドを使用します。

### ⚠ Important

ホストを削除する前に、ホストに関連付けられたすべての接続を削除する必要があります。コマンドを実行すると、ホストは削除されます。確認のダイアログボックスは表示されません。

### ホストを削除するには

- ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して delete-host コマンド AWS CLI を実行し、削除するホストの Amazon リソースネーム (ARN) を指定します。

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

このコマンドは何も返しません。

### ホストの詳細の表示

デベロッパーツールコンソールまたは AWS Command Line Interface (AWS CLI) の get-host コマンドを使用して、ホストの詳細を表示します。

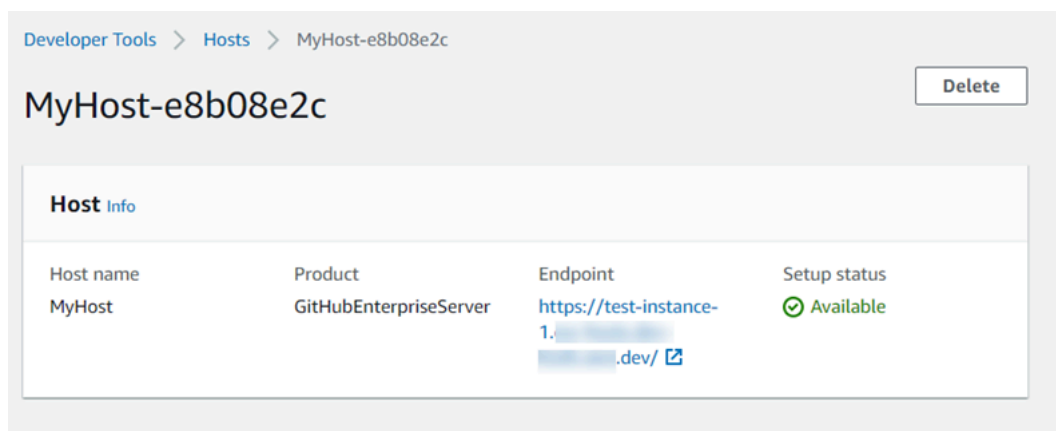
### ホストの詳細を表示するには (コンソール)

- AWS Management Console にサインインして、<https://console.aws.amazon.com/codesuite/settings/connections> でデベロッパーツールコンソールを開きます。
- [設定] > [接続] を選択して、次に [ホスト] タブを選択します。
- 表示するホストの横にあるボタンを選択して、[View event details] (イベント詳細を表示) をクリックします。
- ホストに関する次の情報が表示されます。
  - ホスト名。
  - 接続のプロバイダータイプ。
  - プロバイダーがインストールされているインフラストラクチャのエンドポイント。

- ホストの設定ステータス。接続の準備が整ったホストのステータスは、Available (使用可能) になります。ホストは作成されたが、セットアップが完了しなかった場合は、ホストのステータスが異なる可能性があります。

以下のステータスがあります。

- PENDING - ホストは、作成を完了し、ホストにプロバイダーアプリを登録してセットアップを開始する準備ができています。
- AVIAL - ホストは、作成とセットアップを完了し、接続で使用できます。
- ERROR - ホストの作成または登録中にエラーが発生しました。
- VPC\_CONFIG\_VPC\_INITIALIZING - ホストの VPC 設定を作成中です。
- VPC\_CONFIG\_VPC\_FAILED\_INITIALIZATION - ホストの VPC 設定が検出され、エラーが発生して失敗しました。
- VPC\_CONFIG\_VPC\_AVAILABLE - ホストの VPC 設定はセットアップが完了し、使用可能です。
- VPC\_CONFIG\_VPC\_DELETING - ホストの VPC 設定を削除中です。



5. ホストを削除するには、[Delete] (削除) を選択します。
6. ホストのステータスが Pending (保留中) の場合、セットアップを完了するにはホストの設定を選択します。詳細については、[Set up a pending host \(保留中のホストをセットアップする\)](#) を参照してください。

ホストの詳細を表示するには ( CLI )

- ターミナル (Linux、macOS または Unix) またはコマンドプロンプト (Windows) を開き、AWS CLI を使用して `get-host` コマンドを実行し、詳細を表示するホストの Amazon リソースネーム (ARN) を指定します。

```
aws codeconnections get-host --host-arn arn:aws:codeconnections:us-west-2:account_id:host/My-Host-28aef605
```

このコマンドで、以下の出力が返ります。

```
{
  "Name": "MyHost",
  "Status": "AVAILABLE",
  "ProviderType": "GitHubEnterpriseServer",
  "ProviderEndpoint": "https://test-instance-1.dev/"
}
```

## リンクされたリポジトリの同期設定を操作する

では AWS CodeConnections、接続を使用して、Bitbucket Cloud、GitHub Enterprise Server、GitHub、などのサードパーティリポジトリに AWS リソースを関連付けます。GitLab、CFN\_STACK\_SYNC 同期タイプを使用すると、同期設定を作成できます。これにより、AWS Git リポジトリのコンテンツを同期して、指定された AWS resource. AWS CloudFormation integrates を接続と更新できます。これにより、Git 同期を使用して、同期するリンクされたリポジトリ内のテンプレートとパラメータファイルを管理できます。

接続を作成したら、接続 CLI または AWS CloudFormation コンソールを使用してリポジトリリンクと同期設定を作成できます。

- リポジトリリンク: リポジトリリンクは、接続と外部の Git リポジトリとの関連付けを作成します。リポジトリリンクにより、Git 同期は指定された Git リポジトリ内のファイルへの変更をモニタリングして同期できます。
- 同期設定: 同期設定を使用して Git リポジトリのコンテンツを同期し、指定された AWS リソースを更新します。

詳細については、[AWS CodeConnections 「API リファレンス」](#) を参照してください。



AWS CloudFormation コンソールを使用して AWS CloudFormation スタックの同期設定を作成するチュートリアルについては、「CloudFormation ユーザーガイド」の[AWS CloudFormation 「Git 同期の使用」](#)を参照してください。

## トピック

- [リポジトリリンクを操作する](#)
- [同期設定を使用する](#)

## リポジトリリンクを操作する

リポジトリリンクは、接続と外部の Git リポジトリとの関連付けを作成します。リポジトリリンクを使用すると、Git 同期は指定された Git リポジトリ内のファイルへの変更をモニタリングし、AWS CloudFormation スタックに同期できます。

リポジトリリンクの詳細については、[AWS CodeConnections 「API リファレンス」](#)を参照してください。

## トピック

- [レポジトリリンクを作成する](#)
- [レポジトリリンクを更新する](#)
- [リポジトリリンクを一覧表示する](#)
- [リポジトリリンクを削除する](#)
- [リポジトリリンクの詳細を表示する](#)

## レポジトリリンクを作成する

AWS Command Line Interface ( AWS CLI) の create-repository-link コマンドを使用して、接続と同期する外部リポジトリ間のリンクを作成できます。

リポジトリリンクを作成する前に、などのサードパーティープロバイダーで外部リポジトリを作成しておく必要があります GitHub。

## レポジトリリンクを作成するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して create-repository-link コマンドを実行します。関連する接続の ARN、所有者 ID、およびリポジトリ名を指定します。

```
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. このコマンドで、以下の出力が返ります。

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

## レポジトリリンクを更新する

AWS Command Line Interface ( AWS CLI) の `update-repository-link` コマンドを使用して、指定されたリポジトリリンクを更新できます。

リポジトリリンクの次の情報を更新できます。

- `--connection-arn`
- `--owner-id`
- `--repository-name`

リポジトリに関連付けられている接続を変更したいときに、リポジトリリンクを更新できます。別の接続を使用するには、接続 ARN を指定する必要があります。接続 ARN を表示する手順については、「[接続の詳細を表示する](#)」を参照してください。

レポジトリリンクを更新するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して `update-repository-link` コマンド AWS CLI を実行し、リポジトリリンク用に更新する値を指定し

ます。例えば、以下のコマンドはリポジトリリンク ID に関連付けられた接続を更新します。新しい接続 ARN を `--connection` パラメータで指定します。

```
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-adeb-4ceaefcb2167
```

2. このコマンドで、以下の出力が返ります。

```
{
  "RepositoryLinkInfo": {
    "ConnectionArn": "arn:aws:codestar-connections:us-
east-1:account_id:connection/aEXAMPLE-f055-4843-adeb-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": []
  }
}
```

## リポジトリリンクを一覧表示する

AWS Command Line Interface (AWS CLI) の `list-repository-links` コマンドを使用して、アカウントのリポジトリリンクを一覧表示できます。

リポジトリリンクを一覧表示するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `list-repository-links` コマンドを実行します。

```
aws codeconnections list-repository-links
```

2. このコマンドで、以下の出力が返ります。

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

## リポジトリリンクを削除する

AWS Command Line Interface ( AWS CLI) の `delete-repository-link` コマンドを使用して、リポジトリリンクを削除できます。

リポジトリリンクを削除する前に、リポジトリリンクに関連付けられた同期設定をすべて削除する必要があります。

### Important

コマンドを実行すると、レポジトリリンクは削除されます。確認のダイアログボックスは表示されません。新しいレポジトリリンクを作成することはできますが、Amazon リソースネーム (ARN) は再利用されません。

## リポジトリリンクを削除するには

- ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して `delete-repository-link` コマンド AWS CLI を実行し、削除するリポジトリリンクの ID を指定します。

```
aws codeconnections delete-repository-link --repository-link-id  
6053346f-8a33-4edb-9397-10394b695173
```

このコマンドは何も返しません。

## リポジトリリンクの詳細を表示する

AWS Command Line Interface ( AWS CLI) の `get-repository-link` コマンドを使用して、リポジトリリンクの詳細を表示できます。

リポジトリリンクの詳細を表示するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `get-repository-link` コマンドを実行し、リポジトリリンク ID を指定します。

```
aws codestar-connections get-repository-link --repository-link-id
6053346f-8a33-4edb-9397-10394b695173
```

2. このコマンドで、以下の出力が返ります。

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

## 同期設定を使用する

同期設定により、指定したリポジトリと接続が関連付けられます。同期設定を使用して Git リポジトリのコンテンツを同期し、指定された AWS リソースを更新します。

接続の詳細については、[AWS CodeConnections 「API リファレンス」](#) を参照してください。

### トピック

- [同期設定を作成する](#)
- [同期設定を更新する](#)

- [同期設定を一覧表示する](#)
- [同期設定を削除する](#)
- [同期設定の詳細を表示する](#)

## 同期設定を作成する

AWS Command Line Interface ( AWS CLI) の `create-repository-link` コマンドを使用して、接続と同期する外部リポジトリ間のリンクを作成できます。

同期設定を作成する前に、接続とサードパーティーのリポジトリとの間にリポジトリリンクを作成しておく必要があります。

## 同期設定を作成するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `create-repository-link` コマンドを実行します。関連する接続の ARN、所有者 ID、およびリポジトリ名を指定します。次のコマンドは、AWS CloudFormation内のリソースの同期タイプを使用して同期設定を作成します。また、リポジトリ内のリポジトリブランチと設定ファイルも指定します。この例では、リソースは `mystack` という名前のスタックです。

```
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. このコマンドで、以下の出力が返ります。

```
{
  "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"
  }
}
```

## 同期設定を更新する

AWS Command Line Interface (AWS CLI) の `update-sync-configuration` コマンドを使用して、指定された同期設定を更新できます。

同期設定に関する次の情報を更新できます。

- `--branch`
- `--config-file`
- `--repository-link-id`
- `--resource-name`
- `--role-arn`

### 同期設定を更新するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。を使用して `update-sync-configuration` コマンド AWS CLI を実行し、リソース名と同期タイプとともに更新する値を指定します。例えば、次のコマンドは、同期設定に関連付けられているブランチ名を `--branch` パラメータで更新します。

```
aws codeconnections update-sync-configuration --sync-type CFN_STACK_SYNC --
resource-name mystack --branch feature-branch
```

2. このコマンドで、以下の出力が返ります。

```
{
  "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"
  }
}
```

## 同期設定を一覧表示する

AWS Command Line Interface (AWS CLI) の `list-sync-configurations` のコマンドを使用して、アカウントのリポジトリリンクを一覧表示できます。

リポジトリリンクを一覧表示するには

1. ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `list-sync-configurations` コマンドを実行し、同期タイプとリポジトリリンク ID を指定します。

```
aws codeconnections list-sync-configurations --repository-link-id
6053346f-8a33-4edb-9397-10394b695173 --sync-type CFN_STACK_SYNC
```

2. このコマンドで、以下の出力が返ります。

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

## 同期設定を削除する

AWS Command Line Interface (AWS CLI) の `delete-sync-configuration` コマンドを使用して、同期設定を削除できます。



**⚠ Important**

コマンドを実行すると、同期設定は削除されます。確認のダイアログボックスは表示されません。新しい同期設定を作成することはできますが、Amazon リソースネーム (ARN) は再利用されません。

## 同期設定を削除するには

- ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `delete-sync-configuration` コマンドを実行し、削除する同期設定の同期タイプとリソース名を指定します。

```
aws codeconnections delete-sync-configuration --sync-type CFN_STACK_SYNC --resource-name mystack
```

このコマンドは何も返しません。

## 同期設定の詳細を表示する

AWS Command Line Interface (AWS CLI) の `get-sync-configuration` コマンドを使用して、同期設定の詳細を表示できます。

## 同期設定の詳細を表示するには

- ターミナル (Linux/macOS/Unix) またはコマンドプロンプト (Windows) を開きます。AWS CLI を使用して `get-sync-configuration` コマンドを実行し、リポジトリリンク ID を指定します。

```
aws codeconnections get-sync-configuration --sync-type CFN_STACK_SYNC --resource-name mystack
```

- このコマンドで、以下の出力が返ります。

```
{
  "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"
}
}
```

## を使用した AWS CodeConnections API コールのログ記録 AWS CloudTrail

AWS CodeConnections は、ユーザー AWS CloudTrail、ロール、または AWS のサービスによって実行されたアクションを記録するサービスであると統合されています。は、通知のすべての API コールをイベントとして CloudTrail キャプチャします。キャプチャされた呼び出しには、開発者向けツールコンソールからの呼び出しと、AWS CodeConnections API オペレーションへのコードの呼び出しが含まれます。

証跡を作成する場合は、通知の CloudTrail イベントなど、Amazon Simple Storage Service (Amazon S3) バケットへのイベントの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールのイベント履歴で最新のイベントを表示できます。によって収集された情報を使用して CloudTrail、に対して行われたリクエスト AWS CodeConnections、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

詳細については、『[AWS CloudTrail ユーザーガイド](#)』を参照してください。

### AWS CodeConnections の情報 CloudTrail

CloudTrail AWS アカウントを作成すると、がアカウントで有効になります。でアクティビティが発生すると AWS CodeConnections、そのアクティビティは CloudTrail イベント履歴の他の AWS サービスイベントとともにイベントに記録されます。AWS アカウントで最近のイベントを表示、検索、ダウンロードできます。詳細については、「[AWS CloudTrail ユーザーガイド](#)」の「[イベント履歴を含む CloudTrail イベントの表示](#)」を参照してください。

のイベントなど、AWS アカウント内のイベントの継続的な記録については AWS CodeConnections、証跡を作成します。証跡により CloudTrail、はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョンに適用されます。証跡は、AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをより詳細に分析し、それに基づいて行動するように他の AWS サービスを設定できます。

詳細については、『AWS CloudTrail ユーザーガイド:』の以下のトピックを参照してください。

- 証跡作成の概要
- [CloudTrail がサポートするサービスと統合](#)
- [の Amazon SNS 通知の設定 CloudTrail](#)
- [複数のリージョンからの CloudTrail ログファイルの受信](#)
- [複数のアカウントからの CloudTrail ログファイルの受信](#)

すべての AWS CodeConnections アクションは、[AWS CodeConnections API リファレンス](#)によってログに記録され、[AWS CodeConnections API リファレンス](#)に記載されています。例えば、`GetConnection` アクションを呼び出す `DeleteConnection` と `CreateConnection`、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます:

- リクエストが、ルートと他の IAM 認証情報のどちらを使用して送信されたか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、[CloudTrail userIdentity 要素](#)を参照してください。

## ログファイルエントリについて

証跡は、指定した Amazon S3 バケットにイベントをログファイルとして配信できるようにする設定です。CloudTrail ログファイルには 1 つ以上のログエントリが含まれます。イベントは任意のソースからの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどに関する情報が含まれます。CloudTrail ログファイルはパブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

## CreateConnection の例

次の例は、`CreateConnection` アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、CreateHostアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、CreateSyncConfigurationアクションを示す CloudTrail ログエントリを示しています。

```

{
  "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 の例

次の例は、DeleteConnectionアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、DeleteHostアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、DeleteSyncConfigurationアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、GetConnectionアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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"
        }
      }
    },
    "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"
}
}
```

## GetHost の例

次の例は、GetHostアクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、GetRepositoryLinkアクションを示す CloudTrail ログエントリを示しています。

```
{
    "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-adeb-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 の例

次の例は、[GetRepositorySyncStatus](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[GetResourceSyncStatus](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[GetSyncBlockerSummary](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[GetSyncConfiguration](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[ListConnections](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 boto-core/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 の例

次の例は、[ListHosts](#)アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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"
        }
      }
    },
    "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 の例

次の例は、[ListRepositoryLinks](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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-adeb-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 の例

次の例は、[ListRepositorySyncDefinitions](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[ListSyncConfigurations](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[ListTagsForResource](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[TagResource](#)アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[UntagResource](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[UpdateHost](#)アクションを示す CloudTrail ログエントリを示しています。

```
"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 の例

次の例は、[UpdateRepositoryLink](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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-adeb-4ceaefcb2167",
    "repositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173"
},
"responseElements": {
    "repositoryLinkInfo": {
        "connectionArn": "arn:aws:codeconnections:us-
east-1:123456789012:connection/7df263cc-f055-4843-adeb-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 の例

次の例は、[UpdateSyncBlocker](#)アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 の例

次の例は、[UpdateSyncConfiguration](#) アクションを示す CloudTrail ログエントリを示しています。

```
{
  "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 およびインターフェイス VPC エンドポイント (AWS PrivateLink )

VPC と の間にプライベート接続を確立するには、インターフェイス VPC エンドポイント AWS CodeConnections を作成します。インターフェイスエンドポイントは、インターネットゲートウェイ [AWS PrivateLink](#)、NAT デバイス、VPN 接続、AWS Direct Connect 接続のいずれも必要とせずに AWS CodeConnections APIs にプライベートにアクセスできるテクノロジーである を利用しています。VPC 内のインスタンスは、パブリック IP アドレスがなくても AWS CodeConnections APIs と間のトラフィック AWS CodeConnections が Amazon ネットワークを離れないためです。

各インターフェイスエンドポイントは、サブネット内の 1 つ以上の [Elastic Network Interface](#) によって表されます。

詳細については、「Amazon [VPC ユーザーガイド](#)」の「[インターフェイス VPC エンドポイント \(AWS PrivateLink \)](#)」を参照してください。

### AWS CodeConnections VPC エンドポイントに関する考慮事項

のインターフェイス VPC エンドポイントを設定する前に AWS CodeConnections、「Amazon VPC ユーザーガイド」の「[インターフェイスエンドポイント](#)」を確認してください。

AWS CodeConnections は、VPC からのすべての API アクションの呼び出しをサポートします。

VPC エンドポイントはすべての AWS CodeConnections リージョンでサポートされています。

## VPC エンドポイントの概念

VPC エンドポイントの主な概念は次のとおりです。

### VPC エンドポイント

サービスへのプライベート接続を可能にする VPC 内のエン트리ポイント。VPC エンドポイントのさまざまなタイプを次に示します。サポートされるサービスにより要求される VPC エンドポイントのタイプを作成します。

- [AWS CodeConnections アクションの VPC エンドポイント](#)
- [AWS CodeConnections ウェブフックの VPC エンドポイント](#)

### AWS PrivateLink

VPC とサービスの間でプライベート接続を提供するテクノロジー。

## AWS CodeConnections アクションの VPC エンドポイント

AWS CodeConnections サービスの VPC エンドポイントを管理できます。

アクション用の AWS CodeConnections インターフェイス VPC エンドポイントの作成

Amazon VPC コンソールまたは AWS Command Line Interface () を使用して、AWS CodeConnections サービスの VPC エンドポイントを作成できます。AWS CLI。詳細については、Amazon VPC ユーザーガイドの [インターフェイスエンドポイントの作成](#) を参照してください。

VPC との接続の使用を開始するには、のインターフェイス VPC エンドポイントを作成します。AWS CodeConnections。の VPC エンドポイントを作成するときは AWS CodeConnections、AWS サービスを選択し、サービス名 で以下を選択します。

- `com.amazonaws.region.codestar-connections.api` : このオプションは AWS CodeConnections、API オペレーション用の VPC エンドポイントを作成します。例えば、ユーザーが AWS CLI、AWS CodeConnections API、または AWS SDKs を使用して、`CreateConnection`、などのオペレーション AWS CodeConnections で操作する場合は `ListConnections`、このオプションを選択します `CreateHost`。

DNS 名を有効にするオプションでは、エンドポイントにプライベート DNS を選択した場合、などのリージョンのデフォルト DNS 名 AWS CodeConnections を使用してに API リクエストを実行できません `codestar-connections.us-east-1.amazonaws.com`。

#### Important

プライベート DNS は、AWS サービスおよび AWS Marketplace パートナーサービス用に作成されたエンドポイントに対してデフォルトで有効になっています。

詳細については、「Amazon VPC ユーザーガイド」の「[インターフェイスエンドポイントを介したサービスへのアクセス](#)」を参照してください。

#### アクション用の AWS CodeConnections VPC エンドポイントポリシーの作成

VPC エンドポイントには、AWS CodeConnectionsへのアクセスを制御するエンドポイントポリシーをアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- このアクションを実行できるリソース。

詳細については、「Amazon VPC ユーザーガイド」の「[VPC エンドポイントでサービスへのアクセスを制御する](#)」を参照してください。

#### Note

`com.amazonaws.region.codestar-connections.webhooks` エンドポイントは、ポリシーをサポートしていません。

#### 例: AWS CodeConnections アクションの VPC エンドポイントポリシー

のエンドポイントポリシーの例を次に示します AWS CodeConnections。このポリシーは、エンドポイントにアタッチされると、すべてのリソースのすべてのプリンシパルに対して、リストされた AWS CodeConnections アクションへのアクセスを許可します。

```
{
```

```
"Statement": [  
  {  
    "Sid": "GetConnectionOnly",  
    "Principal": "*",  
    "Action": [  
      "codestar-connections:GetConnection"  
    ],  
    "Effect": "Allow",  
    "Resource": "*"   
  }  
]
```

## AWS CodeConnections ウェブフックの VPC エンドポイント

AWS CodeConnections は、VPC 設定でホストを作成または削除するときに、ウェブフックエンドポイントを作成します。エンドポイント名は `com.amazonaws.region.codestar-connections.webhooks` です。

GitHub ウェブフック用の VPC エンドポイントを使用すると、ホストはウェブフック経由で Amazon ネットワーク経由で統合 AWS サービスにイベントデータを送信できます。

### Important

GitHub Enterprise Server のホストを設定すると、AWS CodeConnections はウェブフックイベントデータ用の VPC エンドポイントを作成します。2020 年 11 月 24 日より前にホストを作成し、VPC PrivateLink ウェブフックエンドポイントを使用する場合は、まずホストを[削除](#)してから新しいホストを[作成](#)する必要があります。

AWS CodeConnections は、これらのエンドポイントのライフサイクルを管理します。エンドポイントを削除するには、対応するホストリソースを削除する必要があります。

### ホストの AWS CodeConnections ウェブフックエンドポイントの使用方法

ウェブフックエンドポイントは、サードパーティーのリポジトリからのウェブフックが AWS CodeConnections 処理のために送信される場所です。ウェブフックでは、顧客のアクションを説明します。git push を実行すると、ウェブフックエンドポイントはプロバイダーからプッシュの詳細を示すウェブフックを受信します。例えば、AWS CodeConnections に通知 CodePipeline してパイプラインを開始できます。

Bitbucket などのクラウドプロバイダーや VPC を使用しない GitHub Enterprise Server ホストの場合、プロバイダーは Amazon ネットワークが使用されていない AWS CodeConnections 場所にウェブフックを送信しているため、ウェブフック VPC エンドポイントは適用されません。

## 接続のトラブルシューティング

以下の情報は、AWS CodeBuild AWS CodeDeploy、およびのリソースへの接続に関する一般的な問題のトラブルシューティングに役立ちます AWS CodePipeline。

### トピック

- [接続を作成できません](#)
- [接続を作成または完了しようとする、アクセス許可エラーが表示される](#)
- [接続を使用しようとする、アクセス許可エラーが表示されます](#)
- [接続が使用可能な状態でないか、または保留中ではなくなりました](#)
- [接続の GitClone アクセス許可を追加する](#)
- [ホストが使用可能な状態ではありません](#)
- [接続エラーのあるホストのトラブルシューティング](#)
- [ホストへの接続を作成できません](#)
- [ホストの VPC 設定のトラブルシューティング](#)
- [Enterprise Server 接続のウェブフック VPC エンドポイント \(PrivateLink\) GitHub のトラブルシューティング](#)
- [2020 年 11 月 24 日以前に作成されたホストのトラブルシューティング](#)
- [GitHub リポジトリの接続を作成できない](#)
- [GitHub Enterprise Server 接続アプリのアクセス許可を編集する](#)
- [への接続時に接続エラー：GitHub「問題が発生しました。ブラウザで Cookie が有効になっていることを確認してください」または「組織の所有者が GitHub アプリをインストールする必要があります」](#)
- [IAM ポリシーでは、リソースの接続サービスプレフィックスを更新する必要がある場合があります](#)
- [接続の制限を引き上げることはできますか](#)

## 接続を作成できません

接続を作成するためのアクセス許可がない可能性があります。詳細については、「[「のアクセス許可と例 AWS CodeConnections」](#)」を参照してください。

## 接続を作成または完了しようとする、アクセス許可エラーが表示される

CodePipeline コンソールで接続を作成または表示しようとする、次のエラーメッセージが返されることがあります。

User: *username* is not authorized to perform: *permission* on resource: *connection-ARN*

このメッセージが表示された場合は、アクセス許可が十分であることを確認してください。

AWS Command Line Interface ( AWS CLI) または AWS Management Console で接続を作成および表示するアクセス許可は、コンソールで接続を作成および完了するために必要なアクセス許可の一部にすぎません。単に接続を表示、編集、または作成してから保留中の接続を完了するために必要なアクセス許可は、特定のタスクだけを実行する必要があるユーザーを対象に絞り込む必要があります。詳細については、「[「のアクセス許可と例 AWS CodeConnections」](#)」を参照してください。

## 接続を使用しようとする、アクセス許可エラーが表示されます

アクセス許可を一覧表示、取得、作成するためのアクセス許可がある場合でも、CodePipeline コンソールで接続を使用しようとする、次のエラーメッセージの 1 つまたは両方が返されることがあります。

You have failed to authenticate your account.(アカウントの認証に失敗しました。)

ユーザー: *username* is not authorized to perform: *codestar-connections:UseConnection* on resource: *connection-ARN*

これが発生した場合、アクセス許可が十分であることを確認してください。

プロバイダーの場所で使用可能なリポジトリをリストするなど、接続を使用するためのアクセス許可があることを確認してください。 詳細については、「[「のアクセス許可と例 AWS CodeConnections」](#)」を参照してください。

## 接続が使用可能な状態でないか、または保留中ではなくなりました

接続が使用可能な状態ではないというメッセージがコンソールに表示される場合は、[Complete connection] (完全な接続) を選択します。

接続を完了することを選択し、接続が保留状態ではないというメッセージが表示された場合は、接続がすでに使用可能な状態になっているため、要求をキャンセルできます。

## 接続の GitClone アクセス許可を追加する

ソースアクションと CodeBuild アクションで AWS CodeStar 接続を使用する場合、入力アーティファクトをビルドに渡す方法は 2 つあります。

- デフォルト: ソースアクションは、ダウンロードするコード CodeBuild を含む zip ファイルを生成します。
- Git クローン: ソースコードは、直接ビルド環境にダウンロードできます。

Git クローンモードでは、作業中の Git リポジトリとしてソースコードを操作することができます。このモードを使用するには、接続を使用するためのアクセス許可を CodeBuild 環境に付与する必要があります。

CodeBuild サービスロールポリシーにアクセス許可を追加するには、CodeBuild サービスロールにアタッチするカスタマー管理ポリシーを作成します。次の手順では、UseConnection のアクセス許可が action フィールドに指定され、接続 Amazon Resource Name (ARN) が Resource フィールドに指定されたポリシーを作成します。

コンソールを使用して UseConnection アクセス許可を追加するには

1. パイプラインの接続 ARN を確認するには、パイプラインを開き、ソースアクションの (i) のアイコンを選択します。設定ペインが開き、接続 ARN が の横に表示されます ConnectionArn。接続 ARN を CodeBuild サービスロールポリシーに追加します。
2. CodeBuild サービスロールを検索するには、パイプラインで使用されているビルドプロジェクトを開き、ビルドの詳細タブに移動します。
3. [Environment] (環境) セクションで、[Service role] (サービスロール) リンクを選択します。これにより、AWS Identity and Access Management (IAM) コンソールが開き、接続へのアクセスを許可する新しいポリシーを追加できます。
4. IAM コンソールで [ポリシーのアタッチ] を選択し、[ポリシーの作成] を選択します。

次のサンプルポリシーテンプレートを使用します。次の例に示すように、Resource フィールドに接続 ARN を追加します。

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



```
"Statement": [  
  {  
    "Effect": "Allow",  
    "Action": "codestar-connections:UseConnection",  
    "Resource": "insert connection ARN here"  
  }  
]  
}
```

[JSON] タブで、ポリシーを貼り付けます。

- [ポリシーの確認] を選択します。ポリシーの名前 (例: **connection-permissions**) を入力し、[ポリシーの作成] を選択します。
- サービスロール Attach Permissions (アクセス許可のアタッチ) ページに戻り、ポリシーリストを更新して、作成したポリシーを選択します。[ポリシーのアタッチ] を選択します。

## ホストが使用可能な状態ではありません

ホストが Available 状態ではないというメッセージがコンソールに表示される場合は、[Set up host] (ホストのセットアップ) を選択します。

ホスト作成の最初のステップにより、作成されたホストは Pending 状態になります。ホストを Available 状態に移行するには、コンソールでホストをセットアップすることを選択する必要があります。詳細については、「[保留中のホストをセットアップする](#)」を参照してください。

### Note

AWS CLI を使用して Pending ホストをセットアップすることはできません。

## 接続エラーのあるホストのトラブルシューティング

基盤となる GitHub アプリケーションを削除または変更すると、接続とホストがエラー状態になる可能性があります。エラー状態のホストと接続はリカバリできず、ホストを再作成する必要があります。

- アプリの pem キーの変更、アプリ名の変更 (最初の作成後) などのアクションにより、ホストと関連するすべての接続がエラー状態になります。

コンソールまたは CLI がホストまたは Error 状態のホストに関連する接続を返す場合は、次の手順を実行する必要がある場合があります。

- ホストリソースを削除して再作成し、ホスト登録アプリを再インストールします。詳細については、「[ホストを作成する](#)」を参照してください。

## ホストへの接続を作成できません

接続またはホストを作成するには、次の条件が必要です。

- ホストは AVAILABLE 状態である必要があります。詳細については、次を参照してください。
- 接続はホストと同じリージョンで作成する必要があります。

## ホストの VPC 設定のトラブルシューティング

ホストリソースを作成するときは、GitHub Enterprise Server インスタンスがインストールされているインフラストラクチャのネットワーク接続または VPC 情報を提供する必要があります。ホストの VPC またはサブネット設定をトラブルシューティングするには、ここに示す VPC 情報の例を参考にしてください。

### Note

このセクションでは、Amazon VPC 内の GitHub Enterprise Server ホスト設定に関連するトラブルシューティングを行います。VPC (PrivateLink) のウェブフックエンドポイントを使用するように設定された接続に関連するトラブルシューティングについては、「」を参照してください[Enterprise Server 接続のウェブフック VPC エンドポイント \(PrivateLink\) GitHub のトラブルシューティング](#)。

この例では、次のプロセスを使用して、GitHub Enterprise Server インスタンスをインストールする VPC とサーバーを設定します。

1. VPC を作成します。詳細については、「<https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#Create-VPC>」を参照してください。
2. VPC にサブネットを作成する 詳細については、「<https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#AddSubnet>」を参照してください。

3. VPC でインスタンスを起動する 詳細については、「[https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#VPC\\_Launch\\_Instance](https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#VPC_Launch_Instance)」を参照してください。

### Note

各 VPC は、一度に 1 つのホスト (GitHub Enterprise Server インスタンス) にのみ関連付けることができます。

次の図は、GitHub Enterprise AMI を使用して起動された EC2 インスタンスを示しています。

The screenshot displays the AWS Management Console interface for an EC2 instance. The instance is named 'GitHub Enterprise', has an Instance ID of 'i-0b4441c7242dfd867', and is running in the 'us-east-2b' availability zone. The instance type is 'm5.xlarge'. The console shows various details including the Instance ID, Instance state (running), Instance type (m5.xlarge), Finding (Opt-in to AWS Compute Optimizer for recommendations), Private DNS (ip-...us-east-2.compute.internal), Private IPs, Secondary private IPs, VPC ID (vpc-a04993cb), Subnet ID (subnet-75350e0f), Network interfaces (eth0), IAM role (ghe-EC2InstanceRole-1OHLRWYXR1RHR), Public DNS (IPv4) (ec2-...us-east-2.compute.amazonaws.com), IPv4 Public IP, IPv6 IPs, Elastic IPs, Availability zone (us-east-2b), Security groups (ghe-InstanceSecurityGroup-11EZ3GYA4DVN6), Scheduled events (No scheduled events), AMI ID (GitHub Enterprise Server 2.20.9), Platform details (Linux/UNIX), Usage operation (RunInstances), and Source/dest. check (True).

Enterprise Server 接続に VPC GitHub を使用する場合は、ホストの設定時にインフラストラクチャに以下を指定する必要があります。

- VPC ID: Enterprise Server インスタンスがインストールされているサーバーの VPC GitHub、または VPN または Direct Connect を介してインストールされた Enterprise Server インスタンスにアクセスできる VPC。
- サブネット ID : Enterprise Server IDs インスタンスがインストールされているサーバーのサブネット、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるサブネット。 GitHub
- セキュリティグループ: GitHub Enterprise Server インスタンスがインストールされているサーバーのセキュリティグループ、または VPN または Direct Connect を介してインストールされた GitHub Enterprise Server インスタンスにアクセスできるセキュリティグループ。

- エンドポイント: サーバーエンドポイントを準備して、次のステップに進みます。

VPC とサブネットの使用方法の詳細については、Amazon VPC ユーザーガイドの「[IPv4 用の VPC とサブネットのサイズ設定](#)」を参照してください。

## トピック

- [保留状態のホストを取得できません](#)
- [利用可能な状態でホストを取得できません](#)
- [接続/ホストが動作していて、現在動作を停止しています](#)
- [ネットワークインターフェイスを削除できません](#)

## 保留状態のホストを取得できません

ホストが VPC\_CONFIG\_FAILED\_INTENTIONAL\_TERMINATION の状態になった場合、ホスト用に選択した VPC、サブネット、またはセキュリティグループに問題がある可能性があります。

- VPC、サブネット、セキュリティグループは、すべて、ホストを作成するアカウントに属している必要があります。
- サブネットとセキュリティグループは、選択した VPC に属している必要があります。
- 提供される各サブネットは、異なるアベイラビリティーゾーンに存在する必要があります。
- ホストを作成するユーザーには、次の IAM アクセス許可が必要です。

```
ec2:CreateNetworkInterface
ec2:CreateTags
ec2:DescribeDhcpOptions
ec2:DescribeNetworkInterfaces
ec2:DescribeSubnets
ec2>DeleteNetworkInterface
ec2:DescribeVpcs
ec2:CreateVpcEndpoint
ec2>DeleteVpcEndpoints
ec2:DescribeVpcEndpoints
```

## 利用可能な状態でホストを取得できません

ホストの CodeConnections アプリ設定を完了できない場合は、VPC 設定または GitHub Enterprise Server インスタンスに問題がある可能性があります。

- パブリック認証機関を使用していない場合は、GitHub エンタープライズインスタンスで使用される TLS 証明書をホストに提供する必要があります。TLS 証明書の値は、証明書のパブリックキーである必要があります。
- GitHub アプリケーションを作成するには、GitHub Enterprise Server インスタンスの管理者である必要があります。

接続/ホストが動作していて、現在動作を停止しています

接続/ホストが以前に動作していて、現在動作していない場合は、VPC の設定変更または GitHub アプリケーションの変更が原因である可能性があります。以下をチェックしてください:

- 接続用に作成したホストリソースにアタッチされたセキュリティグループが変更されるか、GitHub Enterprise Server へのアクセス権がなくなりました。GitHub Enterprise Server インスタンスに接続できるセキュリティグループ CodeConnections が重要です。
- DNS サーバーの IP が最近変更されました。これを確認するには、接続用に作成したホストリソースで指定されている VPC にアタッチされている DHCP オプションをチェックします。最近 AmazonProvidedDNS からカスタム DNS サーバーに移動した場合、または新しいカスタム DNS サーバーの使用を開始した場合、ホスト/接続は機能しなくなることに注意してください。これを修正するには、既存のホストを削除して再作成してください。これにより、最新の DNS 設定がデータベースに保存されます。
- ネットワーク ACLs の設定が変更され、GitHub Enterprise Server インフラストラクチャが配置されているサブネットへの HTTP 接続が許可されなくなりました。
- GitHub Enterprise Server 上の CodeConnections アプリの設定が変更されました。URLs やアプリシークレットなどの設定を変更すると、インストールされている GitHub Enterprise Server インスタンスと 間の接続が切断される可能性があります CodeConnections。

ネットワークインターフェイスを削除できません

ネットワークインターフェイスを検出できない場合は、次の点を確認してください。

- によって作成されたネットワークインターフェイスは、ホストを削除することによってのみ削除 CodeConnections できます。ユーザーが手動で削除することはできません。
- アクセス許可を持っている必要があります。

```
ec2:DescribeNetworkInterfaces
ec2>DeleteNetworkInterface
```

## Enterprise Server 接続のウェブフック VPC エンドポイント (PrivateLink) GitHub のトラブルシューティング

VPC 設定でホストを作成すると、Webhook VPC エンドポイントが自動的に作成されます。

### Note

このセクションは、VPC () のウェブフックエンドポイントを使用するように設定された接続に関連するトラブルシューティングに使用しますPrivateLink。Amazon VPC 内の GitHub Enterprise Server ホスト設定に関連するトラブルシューティングについては、「」を参照してください[ホストの VPC 設定のトラブルシューティング](#)。

インストールされたプロバイダータイプへの接続を作成し、サーバーが VPC 内で設定されていることを指定した場合、はホスト AWS CodeConnections を作成し、ウェブフック用の VPC エンドポイント (PrivateLink) が自動的に作成されます。これにより、ホストはウェブフック経由で Amazon ネットワーク経由で統合 AWS サービスにイベントデータを送信できます。詳細については、「[AWS CodeConnections およびインターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)」を参照してください。

### トピック

- [ウェブフックVPC エンドポイントを削除できません](#)

#### ウェブフックVPC エンドポイントを削除できません

AWS CodeConnections は、ホストのウェブフック VPC エンドポイントのライフサイクルを管理します。エンドポイントを削除する場合は、対応するホストリソースを削除して、削除する必要があります。

- によって作成されたウェブフック VPC エンドポイント (PrivateLink) は、ホスト[を削除](#)することによってのみ削除 CodeConnections できます。手動で削除することはできません。
- アクセス許可を持っている必要があります。

```
ec2:DescribeNetworkInterfaces
ec2:DeleteNetworkInterface
```

## 2020 年 11 月 24 日以前に作成されたホストのトラブルシューティング

2020 年 11 月 24 日現在、ガホストを AWS CodeConnections セットアップすると、追加の VPC エンドポイント (PrivateLink) サポートが設定されます。この更新の前に作成したホストについては、このトラブルシューティングのセクションを使用してください。

詳細については、「[AWS CodeConnections およびインターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)」を参照してください。

### トピック

- [2020 年 11 月 24 日より前に作成されたホストがあり、ウェブフックに VPC エンドポイント \(PrivateLink\) を使用したい](#)
- [利用可能な状態 \( VPC エラー \) のホストを取得できません](#)

2020 年 11 月 24 日より前に作成されたホストがあり、ウェブフックに VPC エンドポイント (PrivateLink) を使用したい

GitHub Enterprise Server 用にホストを設定すると、ウェブフックエンドポイントが自動的に作成されます。接続で VPC PrivateLink ウェブフックエンドポイントを使用するようになりました。2020 年 11 月 24 日より前にホストを作成し、VPC PrivateLink ウェブフックエンドポイントを使用する場合は、まずホストを[削除](#)してから新しいホストを[作成](#)する必要があります。

利用可能な状態 ( VPC エラー ) のホストを取得できません

ホストが 2020 年 11 月 24 日より前に作成されており、ホストの CodeConnections アプリケーションセットアップを完了できない場合、VPC 設定または GitHub Enterprise Server インスタンスに問題がある可能性があります。

エンタープライズサーバーインスタンスが GitHub ウェブフックの送信ネットワークトラフィックを送信できるように GitHub、VPC には NAT ゲートウェイ (またはアウトバウンドインターネットアクセス) が必要です。

### GitHub リポジトリの接続を作成できない

問題:

GitHub リポジトリへの接続は AWS Connector for を使用するため GitHub、接続を作成するには、リポジトリに対する組織所有者のアクセス許可または管理者のアクセス許可が必要です。



解決方法：GitHub リポジトリのアクセス許可レベルについては、<https://docs.github.com/en/free-pro-team@latest/github/setting-up-and-managing-organizations-and-teams/permission-levels-for-an-organization> を参照してください。

## GitHub Enterprise Server 接続アプリのアクセス許可を編集する

2020 年 12 月 23 日以前に GitHub Enterprise Server 用のアプリケーションをインストールした場合は、組織のメンバーに読み取り専用アクセス許可をアプリケーションに付与する必要がある場合があります。GitHub アプリ所有者の場合は、以下の手順に従って、ホストの作成時にインストールされたアプリのアクセス許可を編集します。

### Note

GitHub Enterprise Server インスタンスでこれらのステップを完了し、GitHub アプリ所有者である必要があります。

1. GitHub Enterprise Server で、プロフィール写真のドロップダウンオプションから、**設定** を選択します。
2. **デベロッパー設定** を選択し、**GitHubアプリ** を選択します。
3. アプリの一覧で、接続するアプリの名前を選択し、**[Permissions and events]** (アクセス許可とイベント) 設定画面に表示されます。
4. **[Organization permissions]** (組織のアクセス許可) の **[Members]** (メンバー) で、**[Access]** (アクセス) ドロップダウンから **[Read-only]** (読み取り専用) を選択します。

### Organization permissions

#### Members ⓘ

Organization members and teams.

Access: Read-only ▼

#### Administration ⓘ

Manage access to an organization.

#### Webhooks ⓘ

Manage the post-receive hooks for an organization.

#### Plan ⓘ

View an organization's plan.

Access: No access ▼

5. **[Add a note to users]** (新しいクライアントを設定) で、更新の理由の説明を追加します。**[変更を保存]** を選択します。



への接続時に接続エラー：GitHub「問題が発生しました。ブラウザで Cookie が有効になっていることを確認してください」または「組織の所有者が GitHub アプリをインストールする必要があります」

問題:

GitHub リポジトリの接続を作成するには、GitHub 組織の所有者である必要があります。組織のリポジトリではない場合、ユーザーがリポジトリの所有者である必要があります。接続の作成者が組織の所有者以外である場合、組織の所有者へのリクエストが作成され、次のエラーのいずれかが表示されます。

問題が発生しました。ブラウザで Cookie が有効になっていることを確認してください

または

組織の所有者は GitHub アプリをインストールする必要があります

解決方法：組織内のリポジトリの場合 GitHub、組織所有者は GitHub リポジトリへの接続を作成する必要があります。組織のリポジトリでない場合、ユーザーがリポジトリの所有者である必要があります。

IAM ポリシーでは、リソースの接続サービスプレフィックスを更新する必要がある場合があります

2024 年 3 月 29 日に、サービスの名前が AWS CodeStar Connections からに変更されました AWS CodeConnections。2024 年 5 月 1 日以降、コンソールはリソース ARN codeconnections にとの接続を作成します。両方のサービスプレフィックスを持つリソースは、コンソールに引き続き表示されます。コンソールを使用して作成されたリソースのサービスプレフィックスは になりま ずcodeconnections。新しい SDK/CLI リソースは、リソース ARN のコード接続を使用して作成 されます。作成されたリソースには、自動的に新しいサービスプレフィックスが付けられます。

以下は、 で作成されるリソースです AWS CodeConnections。

- 接続
- [ホスト]

問題:

ARN で codestar-connections を使用して作成されたリソースは、リソース ARN の新しいサービスプレフィックスに自動的に名前が変更されません。新しいリソースを作成すると、接続サービスのプ

レフィックスを持つリソースが作成されます。ただし、codestar-connections サービスプレフィックスを持つ IAM ポリシーは、新しいサービスプレフィックスを持つリソースでは機能しません。

解決方法：リソースのアクセスまたはアクセス許可の問題を回避するには、次のアクションを実行します。

- 新しいサービスプレフィックスの IAM ポリシーを更新します。そうしないと、名前を変更または作成したリソースは IAM ポリシーを使用できません。
- コンソールまたは CLI/CDK/CFN を使用して、新しいサービスプレフィックスのリソースを更新します。

必要に応じて、ポリシー内のアクション、リソース、および条件を更新します。次の例では、両方のサービスプレフィックスの Resource フィールドが更新されています。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "codeconnections:UseConnection"
    ],
    "Resource": [
      "arn:aws:codestar-connections:*:*:connection/*",
      "arn:aws:codeconnections:*:*:connection/*"
    ]
  }
}
```

## 接続の制限を引き上げることはできますか

で特定の制限の引き上げをリクエストできます CodeConnections。詳細については、「[接続のクォータ](#)」を参照してください。

## 接続のクォータ

次の表に、デベロッパーツールコンソールでの接続のクォータ（制限）を示します。

この表のクォータは AWS リージョン ごとに適用され、引き上げることができます。引き上げをリクエストするには、[サポートセンターコンソール](#)を使用します。AWS リージョンの情報と変更可能なクォータについては、「[AWS のサービスクォータ](#)」を参照してください。

**Note**

欧州 (ミラノ) AWS リージョンを使用する前に、このリージョンを有効にする必要があります。詳細については、「[リージョンの有効化](#)」を参照してください。

リソース	デフォルトの制限
AWS アカウントあたりの接続の最大数	250

このテーブルのクォータは固定されており、変更できません。

リソース	デフォルトの制限
接続名の最大文字数	32 文字
AWS アカウントあたりのホストの最大数	50
リポジトリリンクの最大数	100
AWS CloudFormation スタック同期設定の最大数	100
リポジトリリンクあたりの同期設定の最大数	100
ブランチあたりの同期設定の最大数	50

## 許可リストに追加する IP アドレス

IP フィルタリングを実装するか、Amazon EC2 インスタンスで特定の IP アドレスを許可する場合は、以下の IP アドレスを許可リストに追加します。これにより、GitHub や Bitbucket などのプロバイダーへの接続が可能になります。

次の表に、デベロッパーツールコンソールの接続用の IP アドレスを AWS リージョン別に一覧表示します。

**Note**

欧州 (ミラノ) リージョンの場合、このリージョンを使用する前にリージョンを有効にする必要があります。詳細については、「[リージョンの有効化](#)」を参照してください。

リージョン	IP アドレス
米国西部 (オレゴン) (us-west-2)	35.160.210.199、54.71.206.108、54.71.36.205
米国東部 (バージニア北部) (us-east-1)	3.216.216.90、3.216.243.220、3.217.241.85
欧州 (アイルランド) (eu-west-1)	34.242.64.82、52.18.37.201、54.77.75.62
米国東部 (オハイオ) (us-east-2)	18.217.188.190、18.218.158.91、18.220.4.80
アジアパシフィック (シンガポール) (ap-south-east-1)	18.138.171.151、18.139.22.70、3.1.157.176
アジアパシフィック (シドニー) (ap-south-east-2)	13.236.59.253、52.64.166.86、54.206.1.112
アジアパシフィック (東京) (ap-northeast-1)	52.196.132.231、54.95.133.227、18.181.13.91
ヨーロッパ (フランクフルト) (eu-central-1)	18.196.145.164、3.121.252.59、52.59.104.195
アジアパシフィック (ソウル) (ap-northeast-2)	13.125.8.239、13.209.223.177、3.37.200.23
アジアパシフィック (ムンバイ) (ap-south-1)	13.234.199.152、13.235.29.220、35.154.230.124
南米 (サンパウロ) (sa-east-1)	18.229.77.26、54.233.226.52、54.233.207.69
カナダ (中部) (ca-central-1)	15.222.219.210、35.182.166.138、99.79.111.198
ヨーロッパ (ロンドン) (eu-west-2)	3.9.97.205、35.177.150.185、35.177.200.225
米国西部 (北カリフォルニア) (us-west-1)	52.52.16.175、52.8.63.87

リージョン	IP アドレス
欧州 (パリ) (eu-west-3)	35.181.127.138、35.181.145.22、35.181.20.200
欧州 (ストックホルム) (eu-north-1)	13.48.66.148、13.48.8.79、13.53.78.182
欧州 (ミラノ) (eu-south-1)	18.102.28.105、18.102.35.130、18.102.8.116
AWS GovCloud (米国東部)	18.252.168.157、18.252.207.77、18.253.185.119

# デベロッパーツールコンソールの機能のセキュリティ

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャからメリットを得られます。

セキュリティは、AWS とユーザーの間で共有される責任です。[責任共有モデル](#)では、これをクラウドのセキュリティおよびクラウド内のセキュリティとして説明しています。

- **クラウドのセキュリティ** — クラウドで AWS サービスを実行するインフラストラクチャを保護する責任 AWS は AWS にあります。AWS また、は、安全に使用できるサービスも提供します。[AWS コンプライアンスプログラム](#)の一環として、サードパーティーの監査が定期的にセキュリティの有効性をテストおよび検証しています。AWS CodeStar 通知およびに適用されるコンプライアンスプログラムの詳細については AWS CodeConnections、[AWS 「コンプライアンスプログラムによる対象範囲内のサービス」](#)を参照してください。
- **クラウドのセキュリティ** — お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、AWS CodeStar 通知と を使用する際の責任共有モデルの適用方法を理解するのに役立ちます AWS CodeConnections。以下のトピックでは、セキュリティおよびコンプライアンスの目的 AWS CodeConnections を達成するために AWS CodeStar 通知と を設定する方法を示します。また、AWS CodeStar 通知と AWS CodeConnections リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても説明します。

デベロッパーツールコンソールにおけるサービスのセキュリティについては、以下を参照してください。

- [CodeBuild セキュリティ](#)
- [CodeCommit セキュリティ](#)
- [CodeDeploy セキュリティ](#)
- [CodePipeline セキュリティ](#)

## 通知の内容とセキュリティについて

通知は、設定した通知ルールのターゲットにサブスクライブしているユーザーにリソースに関する情報を提供します。これには、リポジトリのコンテンツ、ビルドのステータス、デプロイのステータス、パイプラインの実行など、デベロッパーツールのリソースに関する情報が含まれます。

例えば、コミットまたはプルリクエストに関するコメントを含める CodeCommit ように、 のリポジトリの通知ルールを設定できます。その場合、このルールに応答して送信される通知には、そのコメントで参照されているコード行が含まれる場合があります。同様に、 でビルドプロジェクトの通知ルールを設定 CodeBuild して、ビルドの状態とフェーズの成功または失敗を含めることができます。このルールに応答して送信される通知には、該当する情報が含まれます。

パイプラインの通知ルールを設定 CodePipeline して、手動承認に関する情報を含めることができます。そのルールに応答して送信される通知には、その承認を提供する人の名前が含まれる場合があります。 でアプリケーションの通知ルールを設定 CodeDeploy して、デプロイの成功を示すことができます。そのルールに応答して送信される通知には、デプロイターゲットに関する情報が含まれる場合があります。

通知には、ビルドのステータス、コメントのあるコード行、デプロイのステータス、パイプラインの承認など、プロジェクト固有の情報が含まれます。プロジェクトのセキュリティを確保するために、通知ルールのターゲットと、ターゲットとして指定された Amazon SNS トピックの受信者のリストの両方を定期的に確認してください。さらに、イベントに反応して送信される通知の内容は、基盤となるサービスに機能が追加されると、変わる場合があります。この変更は、既存の通知ルールへの予告なしに発生する可能性があります。通知メッセージの内容を定期的に確認して、送信内容と送信先のユーザーを確認してください。

通知ルールで使用できるイベントタイプの詳細については、「[通知の概念](#)」を参照してください。

通知に含まれる詳細を、イベントに含まれるもののみに制限するように選択できます。これは、ベーシック詳細タイプと呼ばれます。これらのイベントには、Amazon EventBridge および Amazon CloudWatch Events に送信される情報とまったく同じ情報が含まれます。

などのデベロッパーツールコンソールサービスは、イベントで利用できるもの以外に、通知メッセージにイベントタイプの一部またはすべてに関する情報を追加することを選択する CodeCommit 場合があります。この補足情報は、現在のイベントタイプを強化、または将来のイベントタイプを補足するためにいつでも追加できます。[Full (完全)] 詳細タイプを選択して、イベントに関する補足情報 (使用可能な場合) を通知に含めることができます。詳細については、「[詳細タイプ](#)」を参照してください。

## AWS CodeStar 通知とでのデータ保護 AWS CodeConnections

責任 AWS [共有モデル](#)、AWS CodeStar 通知およびのデータ保護に適用されます AWS CodeConnections。このモデルで説明されているように、AWS はすべての を実行するグローバル インフラストラクチャを保護する責任があります AWS クラウド。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。また、使用する AWS のサービスのセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「[データプライバシーのよくある質問](#)」を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された記事「[AWS 責任共有モデルおよび GDPR](#)」を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須であり TLS 1.3 がお勧めです。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。
- AWS 暗号化ソリューションと、内のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度なマネージドセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-2 検証済みの暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-2](#)」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報は、タグ、または名前フィールドなどの自由形式のテキストフィールドに配置しないことを強くお勧めします。これは、コンソール、API、AWS CodeConnections または AWS CLI SDK を使用して AWS CodeStar 通知 やその他の AWS のサービスを使用する場合も同様です。AWS SDKs 名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。



# AWS CodeStar Notifications と の Identity and Access Management AWS CodeConnections

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、誰を認証 (サインイン) し、誰に AWS CodeStar 通知と AWS CodeConnections リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

## Note

新しいサービスプレフィックスで作成されたリソースのアクションcodeconnectionsを使用できます。新しいサービスプレフィックスでリソースを作成すると、リソース ARN codeconnectionsで が使用されます。codestar-connections サービスプレフィックスのアクションとリソースは引き続き使用できます。IAM ポリシーでリソースを指定する場合、サービスプレフィックスはリソースのプレフィックスと一致する必要があります。

## トピック

- [対象者](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [デベロッパーツールコンソールの機能と IAM との連携方法](#)
- [AWS CodeConnections アクセス許可リファレンス](#)
- [アイデンティティベースポリシーの例](#)
- [タグを使用して AWS CodeConnections リソースへのアクセスを制御する](#)
- [コンソールでの通知と接続の使用](#)
- [ユーザーが自分の許可を表示できるようにする](#)
- [AWS CodeStar 通知と AWS CodeConnectionsアイデンティティとアクセスのトラブルシューティング](#)
- [AWS CodeStar 通知のサービスにリンクされたロールの使用](#)
- [AWS CodeConnectionsのサービスにリンクされたロールの使用](#)
- [AWS の マネージドポリシー AWS CodeConnections](#)

## 対象者

AWS Identity and Access Management (IAM) の用途は、AWS CodeStar 通知と で行う作業によって異なります AWS CodeConnections。

サービスユーザー – AWS CodeStar 通知と AWS CodeConnections サービスを使用してジョブを実行する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AWS CodeStar 通知と AWS CodeConnections 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解しておく、管理者に適切な許可をリクエストするうえで役立ちます。AWS CodeStar 通知 および の機能にアクセスできない場合は AWS CodeConnections、「」を参照してください[AWS CodeStar 通知と AWS CodeConnections アイデンティティとアクセスのトラブルシューティング](#)。

サービス管理者 – 社内の AWS CodeStar 通知と AWS CodeConnections リソースを担当している場合は、通常、AWS CodeStar 通知と へのフルアクセスがあります AWS CodeConnections。サービスユーザーがどの AWS CodeStar 通知、AWS CodeConnections 機能、リソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で AWS CodeStar 通知と で IAM を使用方法の詳細については AWS CodeConnections、「」を参照してください[デベロッパーツールコンソールの機能と IAM との連携方法](#)。

IAM 管理者 – IAM 管理者は、AWS CodeStar 通知と へのアクセスを管理するポリシーの作成方法の詳細について確認する場合があります AWS CodeConnections。IAM で使用できる AWS CodeStar 通知と AWS CodeConnections アイデンティティベースのポリシーの例を表示するには、「」を参照してください [アイデンティティベースポリシーの例](#)。

## アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証 ( にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーテッド ID AWS として にサインインできます。AWS IAM Identity Center (IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーテッド ID の例です。フェデレーテッドアイデンティティとしてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることとなります。

ユーザーのタイプに応じて、AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「ユーザーガイド」の「[にサインインする方法 AWS アカウント](#) AWS サインイン」を参照してください。

AWS プログラムでにアクセスする場合、は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM [ユーザーガイドの API AWS リクエスト](#)の署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させることをお勧めします。詳細については、『AWS IAM Identity Center ユーザーガイド』の「[Multi-factor authentication](#)」(多要素認証) および『IAM ユーザーガイド』の「[AWSにおける多要素認証 \(MFA\) の使用](#)」を参照してください。

## AWS アカウントのルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての およびリソースへの AWS のサービス 完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、IAM ユーザーガイドの「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

## IAM ユーザーとグループ

[IAM ユーザー](#)は、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM ユーザーガイドの「[長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できま

す。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する権限を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、『IAM ユーザーガイド』の「[IAM ユーザー \(ロールではなく\) の作成が適している場合](#)」を参照してください。

## IAM ロール

[IAM ロール](#)は、特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、[IAM ロール](#)を一時的に引き受けることができます。ロールを引き受けるには、または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「[IAM ロールの使用](#)」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス - フェデレーテッドアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーテッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、『IAM ユーザーガイド』の「[サードパーティーアイデンティティプロバイダー向けロールの作成](#)」を参照してください。IAM アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。権限セットの詳細については、『AWS IAM Identity Center ユーザーガイド』の「[権限セット](#)」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、(ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、

『IAM ユーザーガイド』の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。

- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの権限、サービスロール、またはサービスにリンクされたロールを使用してこれを行う場合があります。
- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの「[AWS のサービスにアクセス許可を委任するロールの作成](#)」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールの権限を表示できますが、編集することはできません。
- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、『IAM ユーザーガイド』の「[Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して権限を付与する](#)」を参照してください。



IAM ロールと IAM ユーザーのどちらを使用するかについては、『IAM ユーザーガイド』の「[IAM ユーザーではなく\) IAM ロールをいつ作成したら良いのか?](#)」を参照してください。

## ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、AWS ID またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義するオブジェクトです。は、プリンシパル(ユーザー、ルートユーザー、またはロールセッション) AWS がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメント AWS としてに保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「IAM ユーザーガイド」の「[JSON ポリシー概要](#)」を参照してください。

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの権限を定義します。例えば、iam:GetRole アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、AWS Management Console、AWS CLI または AWS API からロール情報を取得できます。

### アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースのポリシーを作成する方法については、IAM ユーザーガイドの「[IAM ポリシーの作成](#)」を参照してください。

アイデンティティベースポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、『IAM ユーザーガイド』の「[マネージドポリシーとインラインポリシーの比較](#)」を参照してください。

## デベロッパーツールコンソールの機能と IAM との連携方法

IAM を使用してデベロッパーツールコンソールの機能へのアクセスを管理する前に、どの IAM 機能を使用できるかを理解する必要があります。通知やその他の AWS のサービスが IAM と連携する方法の概要を把握するには、「IAM ユーザーガイド」の「IAM [AWS と連携する のサービス](#)」を参照してください。

### トピック

- [デベロッパーツールコンソールにおける通知のアイデンティティベースのポリシー](#)
- [AWS CodeStar 通知と AWS CodeConnections リソースベースのポリシー](#)
- [タグに基づく認可](#)
- [IAM ロール](#)

### デベロッパーツールコンソールにおける通知のアイデンティティベースのポリシー

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、アクションを許可または拒否する条件を指定できます。AWS CodeStar 通知とは、特定のアクション、リソース、および条件キー AWS CodeConnections をサポートします。JSON ポリシーで使用するすべての要素については、「IAM ユーザーガイド」の「[IAM JSON ポリシーエレメントのリファレンス](#)」を参照してください。

### アクション

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件でアクションを実行できるかということです。

JSON ポリシーの Action 要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない権限のみのアクションなど、いくつかの例外があります。また、ポリシーに複数アクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

このアクションは、関連付けられたオペレーションを実行するための権限を付与するポリシーで使用されます。

デベロッパーツールコンソールでの通知のポリシーアクションは、アクションの前にプレフィックス `codestar-notifications` and `codeconnections` を使用します。例えば、アカウント

ト内のすべての通知ルールを表示するアクセス許可をユーザーに付与するには、そのユーザーのポリシーに `codestar-notifications:ListNotificationRules` アクションを含めます。ポリシーステートメントには、`Action` または `NotAction` 要素を含める必要があります。AWS CodeStar 通知 および は、このサービスで実行できるタスクを記述する独自のアクションのセット `AWS CodeConnections` を定義します。

1 つのステートメントで複数の AWS CodeStar 通知アクションを指定するには、次のようにカンマで区切ります。

```
"Action": [  
    "codestar-notifications:action1",  
    "codestar-notifications:action2"
```

1 つのステートメントで複数の `AWS CodeConnections` アクションを指定するには、次のようにカンマで区切ります。

```
"Action": [  
    "codeconnections:action1",  
    "codeconnections:action2"
```

ワイルドカード `*` を使用して複数のアクションを指定することができます。例えば、`List` という単語で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "codestar-notifications:List*"
```

AWS CodeStar Notifications API アクションには以下が含まれます。

- `CreateNotificationRule`
- `DeleteNotificationRule`
- `DeleteTarget`
- `DescribeNotificationRule`
- `ListEventTypes`
- `ListNotificationRules`
- `ListTagsForResource`
- `ListTargets`



- `Subscribe`
- `TagResource`
- `Unsubscribe`
- `UntagResource`
- `UpdateNotificationRule`

AWS CodeConnections API アクションには以下が含まれます。

- `CreateConnection`
- `DeleteConnection`
- `GetConnection`
- `ListConnections`
- `ListTagsForResource`
- `TagResource`
- `UntagResource`

認証ハンドシェイクを完了する AWS CodeConnections には、で次のアクセス許可のみのアクションが必要です。

- `GetIndividualAccessToken`
- `GetInstallationUrl`
- `ListInstallationTargets`
- `StartOAuthHandshake`
- `UpdateConnectionInstallation`

接続 AWS CodeConnections を使用するには、で次のアクセス許可のみのアクションが必要です。

- `UseConnection`

サービスに接続 AWS CodeConnections を渡すには、で次のアクセス許可のみのアクションが必要です。

- `PassConnection`

AWS CodeStar 通知と AWS CodeConnections アクションのリストを確認するには、「IAM ユーザーガイド」の[AWS CodeStar 「通知で定義されるアクション」](#) および「[で定義されるアクション AWS CodeConnections](#)」を参照してください。

## リソース

AWS CodeStar 通知 および は、ポリシーでのリソース ARNs指定をサポート AWS CodeConnections していません。

## 条件キー

AWS CodeStar 通知と独自の条件キーのセット AWS CodeConnections を定義し、一部のグローバル条件キーの使用もサポートします。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

すべての AWS CodeStar 通知アクションは、codestar-notifications:NotificationsForResource条件キーをサポートします。詳細については、「[アイデンティティベースポリシーの例](#)」を参照してください。

AWS CodeConnections は、IAM ポリシーの Condition要素で使用できる以下の条件キーを定義します。これらのキーを使用して、ポリシーステートメントが適用される条件をさらに絞り込むことができます。詳細については、「[AWS CodeConnections アクセス許可リファレンス](#)」を参照してください。

条件キー	説明
codeconnections:BranchName	サードパーティーリポジトリのブランチ名でアクセスをフィルタリングします
codeconnections:FullRepositoryId	リクエストで渡されたりポジトリによるアクセスをフィルタリングします。特定のリポジトリにアクセスするための UseConnection リクエストにのみ適用します
codeconnections:InstallationId	接続の更新に使用されるサードパーティー ID (Bitbucket アプリのインストール ID など) でアクセスをフィルタリングします。接続を作成するために使用できるサードパーティー製アプリのインストールを制限できます。

条件キー	説明
codeconnections:OwnerId	サードパーティープロバイダーの所有者またはアカウント ID でアクセスをフィルタリングします
codeconnections:PassedToService	プリンシパルが接続を渡すことができるサービスでアクセスをフィルタリングします
codeconnections:ProviderAction	ListRepositories など、UseConnection リクエストのプロバイダーアクションでアクセスをフィルタリングします。
codeconnections:ProviderPermissionsRequired	サードパーティープロバイダーのアクセス許可のタイプでアクセスをフィルタリングします
codeconnections:ProviderType	リクエストで渡されたサードパーティープロバイダーのタイプによってアクセスをフィルタリングします。
codeconnections:ProviderTypeFilter	結果をフィルタリングするために使用されるサードパーティープロバイダーのタイプによってアクセスをフィルタリングします。
codeconnections:RepositoryName	サードパーティーのリポジトリ名でアクセスをフィルタリングします

## 例

AWS CodeStar 通知と AWS CodeConnections アイデンティティベースのポリシーの例を表示するには、「」を参照してください [アイデンティティベースポリシーの例](#)。

## AWS CodeStar 通知と AWS CodeConnections リソースベースのポリシー

AWS CodeStar 通知 および AWS CodeConnections は、リソースベースのポリシーをサポートしていません。

## タグに基づく認可

タグは、AWS CodeStar 通知と AWS CodeConnections リソースにアタッチすることも、リクエストで渡すこともできます。タグに基づいてアクセスを管理するには、`codestar-notifications` and `codeconnections:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの [条件要素](#) でタグ情報を提供します。タグ付け戦略の詳細については、「[AWS リソースのタグ付け](#)」を参照してください。AWS CodeStar 通知と AWS CodeConnections リソースのタグ付けの詳細については、「」を参照してください [タグ接続リソース](#)。

リソースのタグに基づいてリソースへのアクセスを制限するためのアイデンティティベースのポリシーの例を表示するには、「[タグを使用して AWS CodeConnections リソースへのアクセスを制御する](#)」を参照してください。

## IAM ロール

[IAM ロール](#) は、特定のアクセス許可を持つ AWS アカウント内のエンティティです。

### 一時的な認証情報を使用する

一時的な認証情報を使用して、フェデレーションでのサインイン、IAM ロールの引き受け、またはクロスアカウントロールの引き受けを行うことができます。一時的なセキュリティ認証情報を取得するには、[AssumeRole](#) や などの AWS STS API オペレーションを呼び出します [GetFederationToken](#)。

AWS CodeStar 通知とは、一時的な認証情報の使用 AWS CodeConnections をサポートします。

### サービスリンクロール

[サービスにリンクされたロール](#) を使用すると、AWS サービスが他のサービスのリソースにアクセスして、ユーザーに代わってアクションを実行できます。サービスリンクロールは IAM アカウント内に表示され、サービスによって所有されます。IAM 管理者は、サービスリンクロールの許可を表示できますが、編集することはできません。

AWS CodeStar 通知は、サービスにリンクされたロールをサポートします。AWS CodeStar 通知と AWS CodeConnections サービスにリンクされたロールの作成または管理の詳細については、「」を参照してください [AWS CodeStar 通知のサービスにリンクされたロールの使用](#)。

CodeConnections は、サービスにリンクされたロールをサポートしていません。

## AWS CodeConnections アクセス許可リファレンス

次の表は、各 AWS CodeConnections API オペレーション、アクセス許可を付与できる対応するアクション、およびアクセス許可の付与に使用するリソース ARN の形式を示しています。AWS CodeConnections APIs は、その API で許可されるアクションの範囲に基づいてテーブルにグループ化されます。IAM アイデンティティ (アイデンティティベースのポリシー) にアタッチできるアクセス許可ポリシーを作成する際、参照してください。

アクセス許可ポリシーを作成するときに、ポリシーの Action フィールドでアクションを指定します。ポリシーの Resource フィールドで、ワイルドカード文字 (\*) を使用して、または使用せずに、ARN としてリソース値を指定します。

接続ポリシーで条件を示すには、ここで説明され、[条件キー](#) に一覧表示されている条件キーを使用します。AWS 全体の条件キーを使用することもできます。AWS 全体のキーの完全なリストについては、「IAM ユーザーガイド」の「[使用可能なキー](#)」を参照してください。

アクションを指定するには、API オペレーション名 (例えば、codeconnections や codeconnections:ListConnections) の前に codeconnections:CreateConnection プレフィックスを使用します。

### ワイルドカードの使用

複数のアクションまたはリソースを指定するには、ARN でワイルドカード文字 (\*) を使用します。例えば、codeconnections:\* はすべての AWS CodeConnections アクションを指定し、という単語で始まるすべての AWS CodeConnections アクション codeconnections:Get\* を指定します。Get。次の例では、MyConnection で始まる名前すべてのリソースへのアクセスを許可します。

```
arn:aws:codeconnections:us-west-2:account-ID:connection/*
```

次のテーブルに示されている ## リソースでのみワイルドカードを使用できます。ワイルドカードを *region* または *account-id* リソースで使用することはできません。ワイルドカードの詳細については、IAM ユーザーガイドの [IAM ID](#) を参照してください。

### トピック

- [接続を管理するアクセス許可](#)
- [ホストを管理するためのアクセス許可](#)
- [接続を完了するためのアクセス許可](#)
- [ホスト設定のアクセス許可](#)

- [サービスに接続を渡す](#)
- [接続の使用](#)
- [ProviderAction でサポートされるアクセスタイプ](#)
- [接続リソースにタグ付けするためにサポートされているアクセス許可](#)
- [リポジトリリンクに接続を渡す](#)
- [リポジトリリンクでサポートされる条件キー](#)

## 接続を管理するアクセス許可

AWS CLI または SDK を使用して接続を表示、作成、または削除するように指定されたロールまたはユーザーには、以下に制限されたアクセス許可が必要です。

### Note

次のアクセス許可のみでは、コンソールでの接続を完了または使用することはできません。[接続を完了するためのアクセス許可](#) でアクセス許可を追加する必要があります。

```
codeconnections:CreateConnection
codeconnections>DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
```

AWS CodeStar 接続を管理するためのアクションの通知と AWS CodeConnections 必要なアクセス許可

### CreateConnection

アクション:codeconnections:CreateConnection

CLI またはコンソールを使用して接続を作成するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### DeleteConnection

アクション:codeconnections>DeleteConnection

CLI またはコンソールを使用して接続を削除するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### GetConnection

アクション:codeconnections:GetConnection

CLI またはコンソールを使用して接続の詳細を表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### ListConnections

アクション:codeconnections>ListConnections

CLI またはコンソールを使用してアカウント内のすべての接続を一覧表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

これらのオペレーションでは、次の条件キーがサポートされます。

アクション	条件キー
codeconnections:CreateConnection	codeconnections:ProviderType
codeconnections>DeleteConnection	該当なし
codeconnections:GetConnection	該当なし
codeconnections>ListConnections	codeconnections:ProviderTypeFilter

## ホストを管理するためのアクセス許可

ホストを表示、作成、または削除するために AWS CLI または SDK を使用するよう指定されたロールまたはユーザーには、以下に制限されたアクセス許可が必要です。

**Note**

次のアクセス許可のみでは、ホストでの接続を完了または使用することはできません。[ホスト設定のアクセス許可](#) でアクセス許可を追加する必要があります。

```
codeconnections:CreateHost
codeconnections>DeleteHost
codeconnections:GetHost
codeconnections:ListHosts
```

AWS CodeStar ホストを管理するためのアクションの通知と AWS CodeConnections 必要なアクセス許可

**CreateHost**

アクション:codeconnections:CreateHost

CLI またはコンソールを使用してホストを作成するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

**DeleteHost**

アクション:codeconnections>DeleteHost

CLI またはコンソールを使用してホストを削除するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

**GetHost**

アクション:codeconnections:GetHost

CLI またはコンソールを使用してホストの詳細を表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

**ListHosts**

アクション:codeconnections:ListHosts

CLI またはコンソールを使用してアカウント内のすべてのホストを一覧表示するために必要です。



リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

これらのオペレーションでは、次の条件キーがサポートされます。

アクション	条件キー
codeconnections:CreateHost	codeconnections:ProviderType
codeconnections>DeleteHost	該当なし
codeconnections:GetHost	該当なし
codeconnections>ListHosts	codeconnections:ProviderTypeFilter

### 接続を完了するためのアクセス許可

コンソールで接続を管理するように指定されたロールまたはユーザーは、コンソールで接続を完了し、インストールを作成するために必要なアクセス許可を持っている必要があります。これには、プロバイダーへのハンドシェイクの許可と、使用する接続用のインストールの作成が含まれます。上記のアクセス許可に加えて、次のアクセス許可を使用します。

ブラウザベースのハンドシェイクを実行する際に、コンソールは、次の IAM オペレーションを使用します。

ListInstallationTargets、GetInstallationUrl、StartOAuthHandshake、UpdateConnection は IAM ポリシーアクセス許可です。API アクションではありません。

```
codeconnections:GetIndividualAccessToken
codeconnections:GetInstallationUrl
codeconnections>ListInstallationTargets
codeconnections:StartOAuthHandshake
codeconnections:UpdateConnectionInstallation
```

これに基づいて、コンソールで接続を使用、作成、更新、または削除するには、次のアクセス許可が必要です。

```
codeconnections:CreateConnection
codeconnections>DeleteConnection
```

```
codeconnections:GetConnection
codeconnections:ListConnections
codeconnections:UseConnection
codeconnections:ListInstallationTargets
codeconnections:GetInstallationUrl
codeconnections:StartOAuthHandshake
codeconnections:UpdateConnectionInstallation
codeconnections:GetIndividualAccessToken
```

AWS CodeConnections 接続を完了するためのアクションに必要なアクセス許可

### GetIndividualAccessToken

アクション:codeconnections:GetIndividualAccessToken

コンソールを使用して接続を完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### GetInstallationUrl

アクション:codeconnections:GetInstallationUrl

コンソールを使用して接続を完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### ListInstallationTargets

アクション:codeconnections:ListInstallationTargets

コンソールを使用して接続を完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### StartOAuthHandshake

アクション:codeconnections:StartOAuthHandshake

コンソールを使用して接続を完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### UpdateConnectionInstallation

アクション:codeconnections:UpdateConnectionInstallation

コンソールを使用して接続を完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

これらのオペレーションでは、次の条件キーがサポートされます。

アクション	条件キー
codeconnections:GetIndividualAccessToken	codeconnections:ProviderType
codeconnections:GetInstallationUrl	codeconnections:ProviderType
codeconnections:ListInstallationTargets	該当なし
codeconnections:StartOAuthHandshake	codeconnections:ProviderType
codeconnections:UpdateConnectionInstallation	codeconnections:InstallationId

### ホスト設定のアクセス許可

コンソールで接続を管理するように指定されたロールまたはユーザーは、コンソールでホストをセットアップするために必要なアクセス許可が必要です。これには、プロバイダーへのハンドシェイクの

許可とホストアプリのインストールが含まれます。上記のホストのアクセス許可に加えて、次のアクセス許可を使用します。

ブラウザベースのホスト登録を実行するときに、次の IAM オペレーションがコンソールで使用されます。RegisterAppCode および StartAppRegistrationHandshake は IAM ポリシーのアクセス許可です。API アクションではありません。

```
codeconnections:RegisterAppCode
codeconnections:StartAppRegistrationHandshake
```

これに基づき、以下のアクセス許可を使用して、コンソールでホストを必要とする接続 (インストール済プロバイダタイプなど) を使用、作成、更新、または削除します。

```
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 ホストセットアップを完了するためのアクションに必要なアクセス許可

### RegisterAppCode

アクション:codeconnections:RegisterAppCode

コンソールを使用してホストのセットアップを完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

### StartAppRegistrationHandshake

アクション:codeconnections:StartAppRegistrationHandshake

コンソールを使用してホストのセットアップを完了するために必要です。これは単なる IAM ポリシーのアクセス許可であり、API アクションではありません。

リソース:arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

これらのオペレーションでは、次の条件キーがサポートされます。

## サービスに接続を渡す

サービスに接続を渡す際 (例えば、パイプラインを作成または更新するためにパイプライン定義で接続 ARN が提供されるなど)、ユーザーには codeconnections:PassConnection のアクセス許可が必要です。

AWS CodeConnections 接続を渡すために必要なアクセス許可

PassConnection

アクション:codeconnections:PassConnection

サービスに接続を渡すために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

このオペレーションでは、次の条件キーもサポートされます。

- codeconnections:PassedToService

条件キーでサポートされる値

キー	有効なアクションプロバイダー
codeconnections:PassedToService	<ul style="list-style-type: none"><li>• codeguru-reviewer</li><li>• codepipeline.amazonaws.com</li><li>• proton.amazonaws.com</li></ul>

## 接続の使用

のようなサービスが接続 CodePipeline を使用する場合、サービスロールには特定の接続に対する アクセスcodeconnections:UseConnection許可が必要です。

コンソールで接続を管理するには、ユーザーポリシーに codeconnections:UseConnection アクセス許可が必要です。

AWS CodeConnections 接続を使用するために必要なアクション

### UseConnection

アクション:codeconnections:UseConnection

接続を使用するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

このオペレーションでは、次の条件キーもサポートされます。

- codeconnections:BranchName
- codeconnections:FullRepositoryId
- codeconnections:OwnerId
- codeconnections:ProviderAction
- codeconnections:ProviderPermissionsRequired
- codeconnections:RepositoryName

### 条件キーでサポートされる値

キー	有効なアクションプロバイダー
codeconnections:FullRepositoryId	ユーザー名とリポジトリ名 (my-owner/my-repository など)。接続を使用して特定のリポジトリにアクセスする場合のみサポートされます。
codeconnections:ProviderPermissionsRequired	read_only または read_write

キー	有効なアクションプロバイダー
<code>codeconnections:ProviderAction</code>	<p><code>GetBranch</code> , <code>ListRepositories</code> , <code>ListOwners</code> , <code>ListBranches</code> , <code>StartUploadArchiveToS3</code> , <code>GitPush</code>, <code>GitPull</code>, <code>GetUploadArchiveToS3Status</code> , <code>CreatePullRequestDiffComment</code> , <code>GetPullRequest</code> , <code>ListBranchCommits</code> , <code>ListCommitFiles</code> , <code>ListPullRequestComments</code> , <code>ListPullRequestCommits</code> .</p> <p>詳細については、次のセクションをご覧ください。</p>

一部の機能に必要な条件キーは、時間の経過とともに変化する可能性があります。アクセスコントロールの要件で、異なるアクセス許可が必要でない限り、`codeconnections:UseConnection` を使用して接続へのアクセスを制御することをお勧めします。

## ProviderAction でサポートされるアクセスタイプ

AWS サービスで接続を使用すると、ソースコードプロバイダーに対して API コールが行われます。例えば、`https://api.bitbucket.org/2.0/repositories/username` API をコールすることによって、サービスは、Bitbucket 接続のリポジトリを一覧表示できます。

ProviderAction 条件キーを使用すると、プロバイダのどの API をコールすることができるかを制限できます。API パスは動的に生成される場合があります、パスはプロバイダーによって異なるため、ProviderAction 値は API の URL ではなく抽象アクション名にマッピングされます。これにより、接続のプロバイダーの種類に関係なく、同じ効果を持つポリシーを書くことができます。

サポートされている各 ProviderAction 値に対して許可されるアクセスタイプは次のとおりです。以下は IAM ポリシーアクセス許可です。API アクションではありません。

### AWS CodeConnections でサポートされているアクセスタイプ **ProviderAction**

`GetBranch`

アクション:`codeconnections:GetBranch`

ブランチの最新のコミットなど、ブランチに関する情報にアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### ListRepositories

アクション:codeconnections>ListRepositories

所有者に属する公開および非公開リポジトリのリストにアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### ListOwners

アクション:codeconnections>ListOwners

接続がアクセスできる所有者のリストにアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### ListBranches

アクション:codeconnections>ListBranches

指定したリポジトリに存在するブランチのリストにアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### StartUploadArchiveToS3

アクション:codeconnections:StartUploadArchiveToS3

ソースコードを読み取り、Amazon S3 にアップロードするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

### GitPush

アクション:codeconnections:GitPush

Git を使用してリポジトリに書き込むために必要です。



リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### GitPull

アクション:codeconnections:GitPull

Git を使用してリポジトリから読み込むために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### GetUploadArchiveToS3Status

アクション:codeconnections:GetUploadArchiveToS3Status

StartUploadArchiveToS3 で始まるエラーメッセージを含む、アップロードのステータスにアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### CreatePullRequestDiffComment

アクション:codeconnections:CreatePullRequestDiffComment

プルリクエストのコメントにアクセスするために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### GetPullRequest

アクション:codeconnections:GetPullRequest

リポジトリのプルリクエストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### ListBranchCommits

アクション:codeconnections>ListBranchCommits

リポジトリブランチのコミットのリストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### ListCommitFiles

アクション:codeconnections>ListCommitFiles

コミットのファイルのリストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### ListPullRequestComments

アクション:codeconnections>ListPullRequestComments

プルリクエストのコメントのリストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

#### ListPullRequestCommits

アクション:codeconnections>ListPullRequestCommits

プルリクエストのコミットのリストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*

接続リソースにタグ付けするためにサポートされているアクセス許可

次の IAM オペレーションは、接続リソースをタグ付けするときに使用されます。

```
codeconnections:ListTagsForResource
codeconnections:TagResource
codeconnections:UntagResource
```

AWS CodeConnections 接続リソースのタグ付けに必要なアクション

#### ListTagsForResource

アクション:codeconnections>ListTagsForResource

接続リソースに関連付けられているタグのリストを表示するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

#### TagResource

アクション:codeconnections:TagResource

接続リソースにタグを付けるために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

#### UntagResource

アクション:codeconnections:UntagResource

接続リソースからタグを解除するために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codeconnections:*region*:*account-id*:host/*host-id*

## リポジトリリンクに接続を渡す

同期設定でリポジトリリンクを提供する場合、ユーザーにはリポジトリリンク ARN/リソースに対する codeconnections:PassRepository アクセス許可が必要です。

AWS CodeConnections 接続を渡すために必要なアクセス許可

#### PassRepository

アクション:codeconnections:PassRepository

リポジトリリンクを同期設定に渡すために必要です。

リソース:arn:aws:codeconnections:*region*:*account-id*:repository-link/*repository-link-id*

このオペレーションでは、次の条件キーもサポートされます。

- codeconnections:PassedToService

### 条件キーでサポートされる値

キー	有効なアクションプロバイダー
codeconnections:PassedToService	<ul style="list-style-type: none"> <li>cloudformation.sync.codeconnections.amazonaws.com</li> </ul>

### リポジトリリンクでサポートされる条件キー

リポジトリリンクと同期設定リソースの操作は、以下の条件キーでサポートされています。

- codeconnections:Branch

リクエストで渡されたブランチ名でアクセスをフィルタリングします

### 条件キーでサポートされるアクション

キー	有効値
codeconnections:Branch	<p>以下のアクションが、この条件キーに対してサポートされています。</p> <ul style="list-style-type: none"> <li>CreateSyncConfiguration</li> <li>UpdateSyncConfiguration</li> <li>GetRepositorySyncStatus</li> </ul>

## アイデンティティベースポリシーの例

デフォルトでは、、、または のマネージドポリシーのいずれか AWS CodePipeline が適用されている IAM ユーザーとロールには AWS CodeCommit AWS CodeBuild AWS CodeDeploy、それらのポリシーの意図に沿った接続、通知、および通知ルールに対するアクセス許可があります。例えば、フルアクセスポリシー (、AWSCodeCommitFullAccess、、または AWSCodeBuildAdminAccessAWSCodePipeline\_FullAccess) のいずれかが適用されている IAM ユーザーまたはロールはAWSCodeDeployFullAccess、それらのサービスのリソース用に作成された通知および通知ルールにもフルアクセスできます。

他の IAM ユーザーおよびロールには、AWS CodeStar 通知と AWS CodeConnectionsリソースを作成または変更するアクセス許可はありません。また、、AWS Management Console AWS CLI、ま

または AWS API を使用してタスクを実行することはできません。IAM 管理者は、必要な指定されたリソースに対して API オペレーションを実行するためのアクセス許可をユーザーとロールに付与する IAM ポリシーを作成する必要があります。続いて、管理者はそれらの権限が必要な IAM ユーザーまたはグループにそのポリシーをアタッチする必要があります。

## AWS CodeStar 通知のアクセス許可と例

次のポリシーステートメントと例は、AWS CodeStar 通知の管理に役立ちます。

### フルアクセスマネージドポリシーの通知に関連するアクセス許可

AWSCodeCommitFullAccess、AWSCodeBuildAdminAccess、AWSCodeDeployFullAccess、AWSCodePipeline\_FullAccess 管理ポリシーには、デベロッパーツールコンソールの通知へのフルアクセスを許可する以下のステートメントが含まれています。これらの管理ポリシーのいずれかが適用されたユーザーは、通知の Amazon SNS トピックの作成と管理、トピックに対するユーザーのサブスクライブとサブスクライブ解除、通知ルールのターゲットとして選択するトピックの一覧表示を行うこともできます。

#### Note

管理ポリシーでは、条件キー `codestar-notifications:NotificationsForResource` はサービスのリソースタイプに固有の値を持ちます。例えば、のフルアクセスポリシーでは `CodeCommit`、値は `arn:aws:codecommit:*` です。

```
{
  "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": "*"
  }
}
```

読み取り専用マネージドポリシーの通知に関連するアクセス許可

AWSCodeCommitReadOnlyAccess、AWSCodeBuildReadOnlyAccess、AWSCodeDeployReadOnlyAccess、  
AWSCodePipeline\_ReadOnlyAccess管理ポリシーには、通知への読み取り専用アクセスを許可する

以下のステートメントが含まれています。例えば、デベロッパーツールコンソールでリソースの通知を表示することはできますが、リソースを作成、管理、サブスクライブすることはできません。

#### Note

管理ポリシーでは、条件キー `codestar-notifications:NotificationsForResource` はサービスのリソースタイプに固有の値を持ちます。例えば、のフルアクセスポリシーでは `CodeCommit`、値は `arn:aws:codecommit:*` です。

```
{
  "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": "*"
}
```

#### その他の管理ポリシーの通知に関連するアクセス許可

`AWSCodeCommitPowerUser`、`AWSCodeBuildDeveloperAccess`、および `AWSCodeBuildDeveloperAccess` 管理ポリシーには、これらの管理ポリシーのいずれかが適用されたデベロッパーが通知を作成、編集、サブスクライブできるようにする以下のステートメントが含まれています。通知ルールを削除したり、リソースのタグを管理したりすることはできません。

**Note**

管理ポリシーでは、条件キー `codestar-notifications:NotificationsForResource` はサービスのリソースタイプに固有の値を持ちます。例えば、のフルアクセスポリシーでは `CodeCommit`、値は `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": "*"
}
```

### 例: AWS CodeStar 通知を管理するための管理者レベルのポリシー

この例では、AWS CodeStar ユーザーが通知ルールの詳細を確認し、通知ルール、ターゲット、イベントタイプを一覧表示できるように、AWS アカウントの IAM ユーザーに通知へのフルアクセスを付与します。また、通知ルールの追加、更新、および削除をユーザーに許可します。これはフルアクセスポリシーであり、AWSCodeBuildAdminAccess、AWSCodeCommitFullAccessAWSCodeDeployFullAccessおよび AWSCodePipeline\_FullAccess 管理ポリシーの一部として含まれる通知アクセス許可に相当します。これらの管理ポリシーと同様に、この種のポリシーステートメントは、AWS アカウント全体の通知および通知ルールへの完全な管理アクセスを必要とする IAM ユーザー、グループ、またはロールにのみアタッチする必要があります。

#### Note

このポリシーには、許可として CreateNotificationRule が含まれています。このポリシーが IAM ユーザーまたはロールに適用されているユーザーは、そのユーザーがそれらのリソース自体にアクセスできない場合でも、AWS アカウントの AWS CodeStar Notifications でサポートされているすべてのリソースタイプの通知ルールを作成できます。例えば、このポリシーを持つユーザーは、それ自体にアクセスする CodeCommit アクセス許可を持たない CodeCommit リポジトリの通知ルールを作成できます。

```
{
  "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": "*"
}
]
}

```

### 例: AWS CodeStar 通知を使用するための寄稿者レベルのポリシー

この例では、AWS CodeStar 通知の作成やサブスクライブなど、通知 day-to-day の使用に対するアクセス権を付与しますが、通知ルールやターゲットの削除など、より破壊的なアクションに対するアクセス権は付与しません。これは、AWSCodeBuildDeveloperAccess、AWSCodeDeployDeveloperAccessおよびAWSCodeCommitPowerUser管理ポリシーで提供されるアクセスと同等です。

#### Note

このポリシーには、許可として CreateNotificationRule が含まれています。このポリシーが IAM ユーザーまたはロールに適用されているユーザーは、そのユーザーがそれらのリソース自体にアクセスできない場合でも、AWS アカウントの AWS CodeStar Notifications でサポートされているすべてのリソースタイプの通知ルールを作成できます。例えば、このポリシーを持つユーザーは、それ自体にアクセスする CodeCommitアクセス許可を持たない CodeCommit リポジトリの通知ルールを作成できます。

```

{
  "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": "*"
}
]
}

```

### 例: AWS CodeStar 通知を使用するための read-only-level ポリシー

次の例では、アカウントの IAM ユーザーに対して、AWS アカウントで通知ルール、ターゲット、およびイベントタイプへの読み取り専用アクセスを付与します。この例は、これらの項目の表示を許可するポリシーの作成方法を示しています。これは、AWSCodeBuildReadOnlyAccess、AWSCodeCommitReadOnlyAWSCodePipeline\_ReadOnlyAccess マネージドポリシーの一部として含まれるアクセス許可と同等です。

```

{
  "Version": "2012-10-17",
  "Id": "CodeNotification__ReadOnly",
  "Statement": [
    {
      "Sid": "Reads_API_Access",
      "Effect": "Allow",
      "Action": [
        "CodeNotification:DescribeNotificationRule",
        "CodeNotification:ListNotificationRules",
        "CodeNotification:ListTargets",
        "CodeNotification:ListEventTypes"
      ],
      "Resource": "*"
    }
  ]
}

```


### のアクセス許可と例 AWS CodeConnections

以下のポリシーステートメントと例は、AWS CodeConnectionsの管理に役立ちます。

これらの JSON ポリシードキュメント例を使用して IAM の ID ベースのポリシーを作成する方法については、[IAM ユーザーガイド](#)の「JSON タブでのポリシーの作成」を参照してください。

例: CLI AWS CodeConnections で を作成し、コンソールで表示するためのポリシー

AWS CLI または SDK を使用して接続を表示、作成、タグ付け、または削除するように指定されたロールまたはユーザーには、以下に制限されたアクセス許可が必要です。

 Note

次のアクセス許可のみでは、コンソールでの接続を完了することはできません。次のセクションでアクセス許可を追加する必要があります。

コンソールを使用して、使用可能な接続の一覧を表示し、タグを表示し、接続を使用するには、次のポリシーを使用します。

```
{
  "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": "*"
    }
  ]
}
```

例: コンソール AWS CodeConnections で を作成するためのポリシー

コンソールで接続を管理するように指定されたロールまたはユーザーは、コンソールで接続を完了し、インストールを作成するために必要なアクセス許可を持っている必要があります。これに

は、プロバイダーへのハンドシェイクの許可と、使用する接続用のインストールの作成が含まれます。UseConnection もまたコンソールで接続を使用するために追加する必要があります。コンソールで接続を表示、使用、作成、タグ付け、または削除するには、次のポリシーを使用します。

```
{
  "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",
        "codeconnections:TagResource",
        "codeconnections:ListTagsForResource",
        "codeconnections:UntagResource"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

例: を管理するための管理者レベルのポリシー AWS CodeConnections

この例では、AWS アカウントの IAM ユーザーに へのフルアクセスを付与して、CodeConnections ユーザーが接続を追加、更新、削除できるようにします。これはフルアクセスポリシーであり、AWSCodePipeline\_FullAccess管理ポリシーに相当します。その管理ポリシーと同様に、この種のポリシーステートメントは、AWS アカウント全体の接続への完全な管理アクセスを必要とする IAM ユーザー、グループ、またはロールにのみアタッチする必要があります。

```
{
  "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": "*"
}
]
```

#### 例: を使用するための寄稿者レベルのポリシー AWS CodeConnections

この例では、接続の詳細の作成や表示など CodeConnections、day-to-day の使用に対するアクセス権を付与しますが、接続の削除など、より破壊的なアクションに対するアクセス権は付与しません。

```
{
  "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": "*"
}
]
}

```

#### 例: を使用するための read-only-level ポリシー AWS CodeConnections

この例では、アカウントの IAM ユーザーに、アカウントの接続への読み取り専用アクセスを付与します AWS。この例は、これらの項目の表示を許可するポリシーの作成方法を示しています。

```

{
  "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": "*"
    }
  ]
}

```

#### 例: 指定されたりリポジトリ AWS CodeConnections で を使用するためのスコープダウンポリシー

次の例では、お客様は CodeBuild サービスロールが指定された Bitbucket リポジトリにアクセスすることを希望しています。CodeBuild サービスロールのポリシー :

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "codeconnections:UseConnection"
    ]
  }
}

```

```

    ],
    "Resource": "arn:aws:codeconnections:us-west-2:connection:3dee99b9-172f-4ebe-
a257-722365a39557",
    "Condition": {"ForAllValues:StringEquals": {"codeconnections:FullRepositoryId":
"myrepoowner/myreponame"}}
  }
}

```

### 例: との接続を使用するポリシー CodePipeline

次の例では、管理者はユーザーが との接続を使用することを望んでいます CodePipeline。ユーザーにアタッチされたポリシー:

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "codeconnections:PassConnection"
    ],
    "Resource": "arn:aws:codeconnections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
    "Condition": {"ForAllValues:StringEquals": {"codeconnections:PassedToService":
"codepipeline.amazonaws.com"}}
  }
}

```

### 例: で Bitbucket 読み取りオペレーションのサービス CodeBuild ロールを使用する AWS CodeConnections

次の例では、お客様は、リポジトリに関係なく、CodeBuild サービスロールが Bitbucket で読み取りオペレーションを実行したいと考えています。CodeBuild サービスロールのポリシー:

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "codeconnections:UseConnection"
    ],
    "Resource": "arn:aws:codeconnections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",

```



```
"Condition": {"ForAllValues:StringEquals":
{"codeconnections:ProviderPermissionsRequired": "read_only"}}
}
}
```

例: CodeBuild でサービスロールがオペレーションを実行できないように制限する AWS CodeConnections

次の例では、お客様は CodeBuild サービスロールが のようなオペレーションを実行できないようにしたいと考えています CreateRepository。 CodeBuild サービスロールのポリシー：

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "codeconnections:UseConnection"
    ],
    "Resource": "arn:aws:codeconnections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
    "Condition": {"ForAllValues:StringNotEquals":
{"codeconnections:ProviderAction": "CreateRepository"}}
  }
}
```

## タグを使用して AWS CodeConnections リソースへのアクセスを制御する

タグは、リソースにアタッチするか、タグ付けをサポートするサービスへのリクエストに渡すことができます。では AWS CodeConnections、リソースにタグを付けることができ、一部のアクションにタグを含めることができます。IAM ポリシーを作成するときに、タグ条件キーを使用して以下をコントロールできます。

- どのユーザーがパイプラインリソースに対してアクションを実行できるか (リソースに既に付けられているタグに基づいて)。
- どのタグをアクションのリクエストで渡すことができるか。
- リクエストで特定のタグキーを使用できるかどうか。

次の例は、 CodeConnections ユーザーの ポリシー AWS でタグ条件を指定する方法を示しています。

### Example 1: リクエストのタグに基づいてアクションを許可する

次のポリシーは、で AWS 接続を作成するアクセス許可をユーザーに付与します  
CodeConnections。

これを行うには、リクエストに指定されているタグ Project の値が ProjectA である場合に、CreateConnection アクションと TagResource アクションを許可します。(この aws:RequestTag 条件キーを使用して、IAM リクエストで渡すことができるタグをコントロールします)。aws:TagKeys 条件は、タグキーの大文字と小文字を区別します。

```
{
  "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: リソースタグに基づいてアクションを制限する

次のポリシーは、のリソースに対してアクションを実行し、情報を取得するアクセス許可をユーザーに付与します AWS CodeConnections。

これを行うには、パイプラインに含まれているタグ Project の値が ProjectA である場合に、特定のアクションを許可します。(この aws:RequestTag 条件キーを使用して、IAM リクエストで渡すことができるタグをコントロールします)。aws:TagKeys 条件は、タグキーの大文字と小文字を区別します。

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

## コンソールでの通知と接続の使用

通知エクスペリエンスは CodeBuild、CodeCommit、CodeDeploy、コンソール、および CodePipeline 設定ナビゲーションバー自体のデベロッパーツールコンソールに組み込まれています。コンソールで通知にアクセスするには、それらのサービスにいずれかの管理ポリシーを適用するか、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、AWS アカウントの AWS CodeStar 通知と AWS CodeConnections リソースの詳細を一覧表示および表示できます。最小限必要な許可よりも厳しく制限されたアイデンティティベースポリシーを作成すると、そのポリシーを添付したエンティティ (IAM ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。これらのコンソールへのアクセス AWS CodePipelineを含む AWS CodeCommit、AWS CodeBuild、AWS CodeDeploy、およびへのアクセス許可の付与の詳細については、以下のトピックを参照してください。

- CodeBuild: [でのアイデンティティベースのポリシー CodeBuildの使用](#)
- CodeCommit: [でのアイデンティティベースのポリシー CodeCommitの使用](#)
- AWS CodeDeploy: [のアイデンティティとアクセスの管理 AWS CodeDeploy](#)
- CodePipeline: [IAM ポリシーによるアクセスコントロール](#)

AWS CodeStar 通知には AWS 管理ポリシーはありません。通知機能へのアクセスを提供するには、上記のいずれかのサービスに対する管理ポリシーの 1 つを適用するか、ユーザーまたはエンティティに付与するアクセス許可のレベルでポリシーを作成してから、これらのアクセス許可が必要なユーザー、グループ、またはロールにそれらのポリシーをアタッチする必要があります。詳細については、次の例を参照してください。

- [例: AWS CodeStar 通知を管理するための管理者レベルのポリシー](#)
- [例: AWS CodeStar 通知を使用するための寄稿者レベルのポリシー](#)
- [例: AWS CodeStar 通知を使用するための read-only-level ポリシー](#)

AWS CodeConnections には AWS マネージドポリシーはありません。[接続を完了するためのアクセス許可](#) で詳しく説明している許可など、アクセスの許可や許可の組み合わせを使用します。

詳細については、次を参照してください。

- [例: を管理するための管理者レベルのポリシー AWS CodeConnections](#)
- [例: を使用するための寄稿者レベルのポリシー AWS CodeConnections](#)
- [例: を使用するための read-only-level ポリシー AWS CodeConnections](#)

AWS CLI または AWS API のみ呼び出すユーザーには、コンソールのアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

## ユーザーが自分の許可を表示できるようにする

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
  "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": "*"
  }
]
```

## AWS CodeStar 通知と AWS CodeConnections アイデンティティとアクセスのトラブルシューティング

次の情報は、通知と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

### トピック

- [管理者として通知へのアクセスを他のユーザーに許可したい](#)
- [Amazon SNS トピックを作成して通知ルールのターゲットとして追加したが、イベントに関する E メールが届かない](#)
- [自分の AWS アカウント以外のユーザーに自分の AWS CodeStar 通知と AWS CodeConnections リソースへのアクセスを許可したい](#)

## 管理者として通知へのアクセスを他のユーザーに許可したい

AWS CodeStar Notifications および へのアクセスを他のユーザーに許可するには AWS CodeConnections、アクセスを必要とする人またはアプリケーションの IAM エンティティ (ユーザーまたはロール) を作成する必要があります。ユーザーまたはアプリケーションは、そのエンティティの認証情報を使用して AWS にアクセスします。次に、AWS CodeStar 通知と で正しいアクセス許可を付与するポリシーをエンティティにアタッチする必要があります AWS CodeConnections。

すぐにスタートするには、「IAM ユーザーガイド」の「[IAM が委任した初期のユーザーおよびグループの作成](#)」を参照してください。

AWS CodeStar 通知固有の情報については、「」を参照してください [AWS CodeStar 通知のアクセス許可と例](#)。

## Amazon SNS トピックを作成して通知ルールのターゲットとして追加したが、イベントに関する E メールが届かない

イベントに関する通知を受信するには、通知ルールのターゲットとして有効な Amazon SNS トピックがサブスクライブされていること、および E メールアドレスが Amazon SNS トピックにサブスクライブされていることが必要です。Amazon SNS トピックの問題のトラブルシューティングを行うには、以下を確認します。

- Amazon SNS トピックが通知ルールと同じ AWS リージョンにあることを確認します。
- E メールエイリアスが正しいトピックにサブスクライブされていること、およびサブスクリプションを確認済みであることを確認します。詳細については、「[Amazon SNS トピックにエンドポイントをサブスクライブする](#)」を参照してください。
- AWS CodeStar 通知がそのトピックに通知をプッシュできるようにトピックポリシーが変更されていることを確認します。トピックポリシーには、次のようなステートメントを含める必要があります。

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

詳細については、「[セットアップ](#)」を参照してください。

## 自分の AWS アカウント以外のユーザーに自分の AWS CodeStar 通知と AWS CodeConnections リソースへのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- AWS CodeStar Notifications と [これらの機能 AWS CodeConnections をサポートしているかどうかを確認するには、「](#)」を参照してください [デベロッパーツールコンソールの機能と IAM との連携方法](#)。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、[IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」](#)を参照してください。
- リソースへのアクセスをサードパーティー に提供する方法については AWS アカウント、IAM ユーザーガイドの [「サードパーティー AWS アカウント が所有する へのアクセスを提供する」](#)を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、『IAM ユーザーガイド』の [「外部で認証されたユーザー \(ID フェデレーション\) へのアクセス権限」](#)を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の [「IAM ロールとリソースベースのポリシーとの相違点」](#)を参照してください。

## AWS CodeStar 通知のサービスにリンクされたロールの使用

AWS CodeStar 通知では、AWS Identity and Access Management (IAM) [サービスにリンクされたロール](#)を使用します。サービスにリンクされたロールは、AWS CodeStar 通知に直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは AWS CodeStar Notifications によって事前定義されており、ユーザーに代わってサービスから他の AWS のサービスを呼び出すために必要なすべてのアクセス許可が含まれています。このロールは、通知ルールを初めて作成したときに自動的に作成されます。ユーザーがロールを作成する必要はありません。

サービスにリンクされたロールを使用すると、手動でアクセス許可を追加する必要がないため、AWS CodeStar 通知の設定が簡単になります。AWS CodeStar 通知はサービスにリンクされたロールのアクセス許可を定義し、特に定義されている場合を除き、AWS CodeStar 通知のみがそのロールを引き受けることができます。定義したアクセス許可には、信頼ポリシーと許可ポリシーが含まれます。この許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスにリンクされたロールを削除するには、まず関連するリソースを削除する必要があります。これにより、リソースへのアクセス許可を誤って削除することがなくなるため、AWS CodeStar 通知リソースが保護されます。

サービスにリンクされたロールをサポートしているその他のサービスの詳細については、「[IAM と連携するAWS のサービス](#)」を参照してください。

### AWS CodeStar Notifications のサービスにリンクされたロールのアクセス許可

AWS CodeStar 通知は、AWSServiceRoleForCodeStarNotifications サービスにリンクされたロールを使用して、ツールチェーンで発生するイベントに関する情報を取得し、指定したターゲットに通知を送信します。

AWSServiceRoleForCodeStarNotifications サービスにリンクされたロールは、次のサービスを信頼してロールを引き受けます。

- `codestar-notifications.amazonaws.com`

ロールのアクセス許可ポリシーは、指定されたリソースに対して以下のアクションを実行することを AWS CodeStar 通知に許可します。

- アクション: `CloudWatch Event rules that are named awscodestar-notifications-`  
\* 上で `PutRule`



- アクション: CloudWatch Event rules that are named `awscodestar-notifications-` \* 上で `DescribeRule`
- アクション: CloudWatch Event rules that are named `awscodestar-notifications-` \* 上で `PutTargets`
- アクション: `CreateTopic` (create Amazon SNS topics for use with AWS CodeStar Notifications with the prefix `CodeStarNotifications-` が対象)
- アクション: all comments on all pull requests in all CodeCommit repositories in the AWS account 上で `GetCommentsForPullRequests`
- アクション: all comments on all commits in all CodeCommit repositories in the AWS account 上で `GetCommentsForComparedCommit`
- アクション: all commits in all CodeCommit repositories in the AWS account 上で `GetDifferences`
- アクション: all comments on all commits in all CodeCommit repositories in the AWS account 上で `GetCommentsForComparedCommit`
- アクション: all commits in all CodeCommit repositories in the AWS account 上で `GetDifferences`
- アクション: all AWS Chatbot clients in the AWS account 上で `DescribeSlackChannelConfigurations`
- アクション: all AWS Chatbot clients in the AWS account 上で `UpdateSlackChannelConfiguration`
- アクション: all actions in all pipelines in the AWS account 上で `ListActionExecutions`
- アクション: all files in all CodeCommit repositories in the AWS account unless otherwise tagged 上で `GetFile`

これらのアクションは、`AWSServiceRoleForCodeStarNotifications` サービスにリンクされたロールのポリシーステートメントで確認できます。

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

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。詳細については、「IAM ユーザーガイド」の「[サービスリンクロール権限](#)」を参照してください。

## AWS CodeStar Notifications のサービスにリンクされたロールの作成

サービスリンクロールを手動で作成する必要はありません。デベロッパーツールコンソール、または AWS CLI または SDKs の `CreateNotificationRule` API を使用して、通知ルールを作成できます。API を直接呼び出すこともできます。使用する方法にかかわらず、サービスにリンクされたロールが作成されます。

このサービスリンクロールを削除した後で再度作成する必要がある場合は、同じ方法でアカウントにロールを再作成できます。デベロッパーツールコンソール、または AWS CLI または SDKs の `CreateNotificationRule` API を使用して、通知ルールを作成できます。API を直接呼び出すこともできます。使用する方法にかかわらず、サービスにリンクされたロールが作成されます。

## AWS CodeStar Notifications のサービスにリンクされたロールの編集

サービスにリンクされたロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、名前を変更することはできません。ただし、IAM を使用してロールの説明を編集することはできます。詳細については、「IAM ユーザーガイド」の「[サービスリンクロールの編集](#)」を参照してください。

## AWS CodeStar Notifications のサービスにリンクされたロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、使用していないエンティティがアクティブにモニタリングされたり、メンテナンスされたりすることがなくなります。削除する前に、サービスにリンクされたロールのリソースをクリーンアップする必要があります。AWS CodeStar 通知の場合、これは AWS アカウントのサービスロールを使用するすべての通知ルールを削除することを意味します。

### Note

リソースを削除しようとしたときに AWS CodeStar Notifications サービスがロールを使用している場合、削除が失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

が使用する AWS CodeStar 通知リソースを削除するには `AWSServiceRoleForCodeStarNotifications`

1. <https://console.aws.amazon.com/codesuite/settings/notifications> で AWS デベロッパーツールコンソールを開きます。

**Note**

通知ルールは、作成された AWS リージョンに適用されます。複数の AWS リージョンに通知ルールがある場合は、リージョンセレクタを使用して変更します AWS リージョン。

2. リストに表示されるすべての通知ルールを選択し、[Delete (削除)] を選択します。
3. 通知ルールを作成したすべての AWS リージョンで、これらのステップを繰り返します。

IAM を使用して、サービスにリンクされたロールを削除するには

IAM コンソール、または AWS Identity and Access Management API を使用して AWS CLI、AWSServiceRoleForCodeStarNotifications サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「[サービスリンクロールの削除](#)」を参照してください。

## AWS CodeStar Notifications サービスにリンクされたロールでサポートされているリージョン

AWS CodeStar 通知では、サービスが利用可能なすべての AWS リージョンで、サービスにリンクされたロールの使用がサポートされています。詳細については、[AWS 「リージョンとエンドポイントと通知AWS CodeStar」](#) を参照してください。

## AWS CodeConnectionsのサービスにリンクされたロールの使用

AWS CodeConnections は AWS Identity and Access Management、(IAM) [サービスにリンクされたロール](#) を使用します。サービスにリンクされたロールは、に直接リンクされた一意のタイプの IAM ロールです AWS CodeConnections。サービスにリンクされたロールは によって事前定義 AWS CodeConnections されており、ユーザーに代わってサービスから他の AWS のサービスを呼び出すために必要なすべてのアクセス許可が含まれています。このロールは、接続を初めて作成するときにお客様用に作成されます。ユーザーがロールを作成する必要はありません。

サービスにリンクされたロールを使用すると、アクセス許可を手動で追加する必要がないため、の設定 AWS CodeConnections が簡単になります。は、サービスにリンクされたロールのアクセス許可 AWS CodeConnections を定義し、特に定義されている場合を除き、のみがそのロールを引き受け AWS CodeConnections ることができます。定義したアクセス許可には、信頼ポリシーと許可ポリシーが含まれます。この許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスにリンクされたロールを削除するには、まず関連するリソースを削除する必要があります。これにより、AWS CodeConnections リソースへのアクセス許可を誤って削除することがなくなるため、リソースが保護されます。

サービスにリンクされたロールをサポートしているその他のサービスの詳細については、「[IAM と連携するAWS のサービス](#)」を参照してください。

#### Note

新しいサービスプレフィックスで作成されたリソースのアクションcodeconnectionsを使用できます。新しいサービスプレフィックスでリソースを作成すると、リソース ARN codeconnectionsでが使用されます。codestar-connections サービスプレフィックスのアクションとリソースは引き続き使用できます。IAM ポリシーでリソースを指定する場合、サービスプレフィックスはリソースのプレフィックスと一致する必要があります。

## のサービスにリンクされたロールのアクセス許可 AWS CodeConnections

AWS CodeConnections は、AWSServiceRoleForGitSync サービスにリンクされたロールを使用して、接続された Git ベースのリポジトリとの Git 同期を使用します。

AWSServiceRoleForGitSync サービスにリンクされたロールは、次のサービスを信頼してロールを引き受けます。

- `repository.sync.codeconnections.amazonaws.com`

という名前のロールアクセス許可ポリシー AWSGitSyncServiceRolePolicy は AWS CodeConnections、 が指定されたリソースに対して次のアクションを実行できるようにします。

- アクション: 外部の Git ベースのリポジトリへの接続を作成し、それらのレポジトリで Git 同期を使用できるようにするアクセス許可を、ユーザーに付与します。

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。詳細については、IAM ユーザーガイドの[サービスにリンクされたロールの許可](#)を参照してください。

## AWS CodeConnectionsのサービスリンクロールの作成

サービスリンクロールを手動で作成する必要はありません。CreateRepositoryLink API を使用して Git 同期プロジェクトのリソースを作成するときに、ロールを作成します。

このサービスリンクロールを削除した後で再度作成する必要がある場合は、同じ方法でアカウントにロールを再作成できます。

## AWS CodeConnectionsのサービスにリンクされたロールの編集

サービスにリンクされたロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、名前を変更することはできません。ただし、IAM を使用してロールの説明を編集することはできます。詳細については、「IAM ユーザーガイド」の「[サービスにリンクされたロールの編集](#)」を参照してください。

## AWS CodeConnectionsのサービスリンクロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、使用していないエンティティがアクティブにモニタリングされたり、メンテナンスされたりすることがなくなります。削除する前に、サービスにリンクされたロールのリソースをクリーンアップする必要があります。つまり、AWS アカウントでサービスロールを使用するすべての接続を削除します。

### Note

リソースを削除しようとしたときに AWS CodeConnections サービスがロールを使用している場合、削除が失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

が使用する AWS CodeConnections リソースを削除するには AWSServiceRoleForGitSync

1. 開発者ツールコンソールを開き、[設定] を選択します。
2. リストに表示されるすべての接続を選択し、[削除] を選択します。
3. 接続を作成したすべての AWS リージョンで、これらの手順を繰り返します。

IAM を使用して、サービスにリンクされたロールを削除するには

IAM コンソール、または AWS Identity and Access Management API を使用して AWS CLI、AWSServiceRoleForGitSync サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「[サービスリンクロールの削除](#)」を参照してください。

## AWS CodeConnections サービスにリンクされたロールでサポートされているリージョン

AWS CodeConnections は、サービスが利用可能なすべての AWS リージョンで、サービスにリンクされたロールの使用をサポートします。詳細については、「[AWS リージョンとエンドポイント](#)」を参照してください。

## AWS の マネージドポリシー AWS CodeConnections

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンポリシーです AWS。AWS 管理ポリシーは、多くの一般的なユースケースにアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケース別に[カスタマー マネージドポリシー](#)を定義して、マネージドポリシーを絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) が更新されます。は、新しい AWS のサービスが起動されるか、既存のサービスで新しい API AWS オペレーションが使用可能になると、AWS 管理ポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

### Note

新しいサービスプレフィックスで作成されたリソースのアクションcodeconnectionsを使用できます。新しいサービスプレフィックスでリソースを作成すると、リソース ARN codeconnectionsで が使用されます。codestar-connections サービスプレフィックスのアクションとリソースは引き続き使用できます。IAM ポリシーでリソースを指定する場合、サービスプレフィックスはリソースのプレフィックスと一致する必要があります。

## AWS マネージドポリシー: AWSGitSyncServiceRolePolicy

IAM エンティティ `AWSGitSyncServiceRolePolicy` に をアタッチすることはできません。このポリシーは、 がユーザーに代わってアクションを実行できるようにするサービスにリンクされたロール `AWS CodeConnections` にアタッチされます。詳細については、「[AWS CodeConnectionsのサービスにリンクされたロールの使用](#)」を参照してください。

このポリシーにより、お客様は Git ベースのリポジトリにアクセスして接続に使用することができます。お客様は `CreateRepositoryLink` API を使用した後にこれらのリソースにアクセスします。

### アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `codeconnections` – ユーザーが外部 Git ベースのリポジトリへの接続を作成できるようにするアクセス許可を付与します。

```
{
  "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 CodeConnectionsAWS 管理ポリシーの更新

このサービスがこれらの変更の追跡を開始した AWS CodeConnections 以降の の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動アラートを受け取るには、AWS CodeConnections [ドキュメント履歴](#) ページの RSS フィードにサブスクライブしてください。

変更	説明	日付
<a href="#">AWSGitSyncServiceRolePolicy</a> - ポリシーを更新	AWS CodeStar Connections サービス名が に変更されました AWS CodeConnections。両方のサービスプレフィックスを含む ARNs を持つリソースのポリシーを更新しました。	2024 年 4 月 26 日
<a href="#">AWSGitSyncServiceRolePolicy</a> - 新しいポリシー	AWS CodeStar Connections にポリシーが追加されました。  接続ユーザーが接続された Git ベースのリポジトリと Git 同期を使用できるようにするアクセス許可を付与します。	2023 年 11 月 26 日
AWS CodeConnections が変更の追跡を開始しました	AWS CodeConnections が AWS マネージドポリシーの変更の追跡を開始しました。	2023 年 11 月 26 日

## AWS CodeStar 通知と のコンプライアンス検証 AWS CodeConnections

特定のコンプライアンスプログラム AWS の対象となるサービスのリストについては、「[コンプライアンスAWS プログラムによる対象範囲内の のサービス](#)」を参照してください。一般的な情報については、「[AWS コンプライアンスプログラム](#)」を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、[AWS 「Artifact でのレポートのダウンロード」](#)を参照してください。

AWS CodeStar 通知を使用する際のお客様のコンプライアンス責任 AWS CodeConnections は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。AWS では、コンプライアンスに役立つ以下のリソースを提供しています。

- [セキュリティとコンプライアンスのクイックスタートガイド](#) – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境を にデプロイする手順について説明します AWS。
- [AWS コンプライアンスリソース](#) – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- [AWS Config](#) – この AWS サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。
- [AWS Security Hub](#) – この AWS サービスは、内のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立ちます。

## AWS CodeStar 通知と の耐障害性 AWS CodeConnections

AWS グローバルインフラストラクチャは、AWS リージョンとアベイラビリティーゾーンを中心に構築されています。AWS リージョンは、低レイテンシー、高スループット、および高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティーゾーンを提供します。アベイラビリティーゾーンでは、アベイラビリティーゾーン間で中断せずに、自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、耐障害性、およびスケーラビリティが優れています。

AWS リージョンとアベイラビリティーゾーンの詳細については、[AWS 「グローバルインフラストラクチャ」](#)を参照してください。

- 通知ルールは、作成された AWS リージョン に固有です。複数の に通知ルールがある場合は AWS リージョン、リージョンセレクタを使用して各 の通知ルールを確認します AWS リージョン。
- AWS CodeStar 通知は、通知ルールのターゲットとして Amazon Simple Notification Service (Amazon SNS) トピックに依存します。Amazon SNS トピックと通知ルールのターゲットに関する情報は、通知ルールを設定した リージョンと異なる AWS リージョンに保存される場合があります。

## AWS CodeStar Notifications および のインフラストラクチャセキュリティ AWS CodeConnections

マネージドサービスの機能である AWS CodeStar 通知 および AWS CodeConnections は、ホワイトペーパー「Amazon Web Services: セキュリティプロセスの概要」に記載されている AWS グローバルネットワークセキュリティの手順で保護されています。 [https://d0.awsstatic.com/whitepapers/Security/AWS\\_Security\\_Whitepaper.pdf](https://d0.awsstatic.com/whitepapers/Security/AWS_Security_Whitepaper.pdf)

AWS が公開した API コールを使用して、AWS CodeStar 通知および ネットワーク AWS CodeConnections 経由でアクセスします。クライアントで Transport Layer Security (TLS) 1.0 以降がサポートされている必要があります。また、Ephemeral Diffie-Hellman (DHE) や Elliptic Curve Ephemeral Diffie-Hellman (ECDHE) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートもクライアントでサポートされている必要があります。最新のシステムは、ほとんどの場合これらのモードをサポートしています。

リクエストは、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットのアクセスキーを使用して署名する必要があります。または、[AWS Security Token Service](#) (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

### リージョンをまたぐ AWS CodeConnections リソース間のトラフィック

接続機能を使用して リソースの接続を有効にする場合、基盤となるサービス AWS リージョン を使用している の AWS リージョン 外部に、リソースが作成されたリージョン以外のリージョンでそのようなリソースへの接続を提供する目的に限り、そのような接続リソースに関連する情報を保存および処理することに同意し、指示します。

詳細については、「[のグローバルリソース AWS CodeConnections](#)」を参照してください。

**Note**

接続機能を使用して、先立って有効にする必要のないリージョンでリソースへの接続を有効にした場合、情報は前述のトピックで詳しく説明したとおりに保存および処理されます。欧州 (ミラノ) リージョンなど、先立って有効にする必要があるリージョンで確立した接続については、当リージョンのその接続に関する情報のみが保存および処理されます。

## 接続の名前変更 - 変更の概要

デベロッパーツールコンソールの接続機能を使用すると、AWS リソースをサードパーティーのソースリポジトリに接続できます。2024 年 3 月 29 日、AWS CodeStar Connections の名前が `codeconnections` に変更されました。以下のセクションでは、名前の変更に伴って変更された機能のさまざまな部分と、リソースが正常に機能し続けるために実行する必要があるアクションについて説明します。

これはすべてを網羅したリストではないことに注意してください。製品の他の部分も変更されましたが、これらの更新が最も関連性があります。

### Note

新しいサービスプレフィックスで作成されたリソースのアクション `codeconnections` を使用できます。新しいサービスプレフィックスでリソースを作成すると、リソース ARN `codeconnections` が使用されます。 `codestar-connections` サービスプレフィックスのアクションとリソースは引き続き使用できます。IAM ポリシーでリソースを指定する場合、サービスプレフィックスはリソースのプレフィックスと一致する必要があります。

### Note

現在、コンソールを使用して接続を作成すると、リソース ARN に `codestar-connections` を持つリソースのみが作成されます。ARN で接続サービスのプレフィックスを持つリソースを作成するには、CLI、SDK、または CFN を使用します。両方のサービスプレフィックスを持つリソースは、引き続きコンソールに表示されます。

## 名前が変更されたサービスプレフィックス

Connections APIs、名前が変更されたサービスプレフィックスを使用します `codeconnections`。

CLI コマンドで新しいプレフィックスを使用するには、バージョン 2 をダウンロードします AWS CLI。以下は、更新されたプレフィックスを持つコマンドの例です。

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

## IAM での名前変更アクション

IAM のアクションでは、次の例に示すように、新しいプレフィックスを使用します。

```
codeconnections:CreateConnection
codeconnections>DeleteConnection
codeconnections:GetConnection
codeconnections:ListConnections
```

## 新しいリソース ARN

作成された Connections リソースには、新しい ARN があります。

```
arn:aws:codeconnections:us-west-2:account-ID:connection/*
```

## 影響を受けるサービスロールポリシー

以下のサービスの場合、サービスロールポリシーはポリシーステートメントで新しいプレフィックスを使用します。既存のサービスロールポリシーを更新して新しいアクセス許可を使用することもできますが、古いプレフィックスで作成されたポリシーは引き続きサポートされます。

- CodePipeline カスタマー管理サービスロールポリシー
- AWS CodeStar サービスロールAWSCodeStarServiceRoleポリシー

## 新しい CloudFormation リソース

AWS CloudFormation 接続にリソースを使用するには、新しいリソースを使用できます。既存のリソースは引き続きサポートされます。

- 新しい[AWS CloudFormation](#)リソースの名前は `AWS::CodeConnections::Connection` です。ユーザーガイドの[AWS::CodeConnections::Connection](#) CloudFormation 「」を参照してください。
- 既存の `AWS::CodeStarConnections::Connection` リソースは引き続きサポートされます。ユーザーガイドの[AWS::CodeStarConnections::Connection](#) CloudFormation 「」を参照してください。

## ドキュメント履歴

以下の表は、デベロッパーツールコンソールの今回のリリースの内容をまとめたものです。

- AWS CodeStar Notifications API バージョン : 2019-10-15
- AWS CodeConnections API バージョン : 2023-12-01

変更	説明	日付
<a href="#">接続の マネージドポリシーの更新 service-linked-role</a>	Git リポジトリと Git 同期を使用するためのサービスにリンクされたロールの マネージドポリシーが、両方のサービスプレフィックスを持つリソースに対して更新されました。詳細については、 <a href="#">「のサービスにリンクされたロールの使用 AWS CodeConnections」</a> および <a href="#">「管理ポリシー」</a> を参照してください。	2024 年 4 月 26 日
<a href="#">AWS CodeStar 接続の名前が変更されました AWS CodeConnections</a>	の紹介。AWS CodeConnectionsのパイプラインなどのリソースと AWS CodePipeline サードパーティーの Git プロバイダー間の接続を作成および管理できます。	2024 年 3 月 29 日
<a href="#">でサポート GitLab される への接続 CodeBuild</a>	への接続を設定するためのサポート CodeBuild が追加されました GitLab。詳細については、 <a href="#">「との製品とサービスの統合 AWS CodeConnections」</a> を参照してください。	2024 年 3 月 27 日

## [GitLab セルフマネージドのサポート](#)

GitLab セルフマネージドとやり取りする AWS リソースの接続とホストの設定のサポートが追加されました。詳細については、[「ホストを作成または更新するワークフロー」](#) および [「セルフマネージドへの接続 GitLabを作成する」](#) を参照してください。

2023 年 12 月 28 日

## [接続用の新しいリポジトリリンクと同期設定](#)

リポジトリリンクの設定と同期設定に関する情報を追加しました。同期設定を使用して Git リポジトリのコンテンツを同期し、AWS CloudFormation スタックリソースを更新します。詳細については、[「リポジトリリンクを操作する」](#) と [「同期設定を使用する」](#) を参照してください。

2023 年 11 月 27 日

## [接続のサポート service-linked-role](#)

Git 同期を Git リポジトリで使用するための接続設定のサポートが追加されました。詳細については、[「のサービスにリンクされたロールの使用 AWS CodeConnections」](#) および [「管理ポリシー」](#) を参照してください。

2023 年 11 月 26 日

## [GitLab グループのサポート](#)

グループと GitLab やり取りする AWS リソースの接続の設定のサポートが追加されました。詳細については、[「接続の作成」](#) および [「への接続の作成 GitLab」](#) を参照してください。

2023 年 9 月 15 日



## [新しい GitLab プロバイダータイプ](#)

への接続を作成できるようになりました GitLab。詳細については、[「接続の作成」](#)および[「への接続の作成 GitLab」](#)を参照してください。

2023 年 8 月 10 日

## [通知ルールの新しいターゲットタイプ](#)

通知ルールのターゲットとして、Microsoft Teams チャンネル用に設定された AWS Chatbot クライアントを選択できるようになりました。詳細については、[「通知ルールの作成」](#)と[「通知ルールのターゲットの使用」](#)を参照してください。

2023 年 5 月 17 日

## [欧州 \(ミラノ\) リージョンで接続が利用可能に](#)

欧州 (ミラノ) リージョンの接続に関する情報を追加しました。詳細については、[「リージョン間の AWS CodeConnections リソース間のトラフィック」](#)を参照してください。

2023 年 5 月 17 日

## [リポジトリのアクセス許可に関する接続エラーのトラブルシューティングを追加](#)

GitHub 組織内のリポジトリへの接続を作成するときは、GitHub 組織の所有者である必要があります。詳細については、[「に接続するときの接続エラー GitHub」](#)を参照してください。

2022 年 8 月 29 日

### [ホストリソースのタグ付けに関する情報を追加](#)

コンソールと CLI を使用して、ホストへのタグ付けができるようになりました。詳細については、「[のリソースにタグを付ける AWS CodeConnections](#)」を参照してください。

2021 年 4 月 19 日

### [接続の VPC エンドポイントのサポート](#)

接続で VPC エンドポイントを使用できます。詳細については、[AWS CodeConnections 「」および「インターフェイス VPC エンドポイント \(AWS PrivateLink \)」](#)を参照してください。

2020 年 11 月 24 日

### [新しい GitHub および GitHub Enterprise Cloud プロバイダータイプ](#)

GitHub および GitHub Enterprise Cloud への接続を作成できるようになりました。詳細については、「[接続の作成](#)」および「[への接続の作成 GitHub](#)」を参照してください。

2020 年 9 月 30 日

## [GitHub Enterprise Server プロバイダータイプとホストリソースを追加](#)

このガイドには、接続のホストリソースに関する情報が追加されました。GitHub Enterprise Server への接続を作成できるようになりました。接続を作成して作業する方法の詳細については、[「Create a connection」](#) および [「Working with hosts」](#) を参照してください。これは、デベロッパーツールコンソールのユーザーガイドで説明されている接続機能を備えた一般公開リリースです。

2020 年 6 月 29 日

## [接続の使用とタグ付けに関する情報を追加](#)

コンソールの接続機能に関する情報が、このガイドに追加されました。概念、開始手順、ポリシーの例を含むアクセス許可に関するリファレンス、接続の作成、表示、およびタグ付けの手順を表示できます。詳細については、[「接続とは」](#)、[「接続の概念」](#)、[「接続の開始方法」](#)、[「接続の作成」](#)、[「 のリソースにタグ AWS CodeConnections を付ける」](#)、[「セキュリティ」](#)、[「接続のクォータ」](#)、[「トラブルシューティング」](#)、[「 AWS CodeConnections での API コール AWS CloudTrail」](#)を参照してください。追加のプロバイダーアクション (アクセス許可のみのアクション) のリストを表示するには、[「 のアクション ProviderType」](#)を参照してください。

2020 年 6 月 28 日

## [通知ルールの新しいターゲットタイプ](#)

通知ルールのターゲットとして、Slack チャネル用に設定された AWS Chatbot クライアントを選択できるようになりました。詳細については、[「通知ルールの作成」](#)と[「通知ルールのターゲットの使用」](#)を参照してください。

2020 年 4 月 2 日

### [追加の AWS CodeCommit イベントに関する通知を追加](#)

プルリクエストの承認に関連するイベントの通知を設定できるようになりました。詳細については、[「リポジトリの通知ルールのイベント」](#)および[「でのプルリクエストの使用 CodeCommit」](#)を参照してください。

2020 年 2 月 10 日

### [2 つの追加 AWS リージョンで利用可能な通知](#)

デベロッパーツールコンソールで、中東 (バーレーン) およびアジアパシフィック (香港) の通知をサポートできるようになりました。詳細については、「」の[AWS CodeStar「通知」](#)を参照してください  
AWS 全般のリファレンス。

2020 年 2 月 5 日

### [暗号化された Amazon SNS トピックのサポートを追加](#)

暗号化された Amazon SNS トピックを通知ターゲットとして使用するためのガイダンスを追加しました。詳細については、「[Configure Amazon SNS topics for notifications](#)」を参照してください。

2020 年 2 月 4 日

### [通知には、のセッションタグ情報を含めることができます CodeCommit](#)

の通知に、セッションタグを使用して、表示名や E メールアドレスなどのユーザー ID 情報を含める CodeCommit ことができるようになりました。詳細については、「[の概念](#)」および「[タグを使用してのアイデンティティ情報を提供する CodeCommit](#)」を参照してください。

2019 年 12 月 19 日

## 初回リリース

これはデベロッパーツールコンソールのユーザーガイドの初回リリースです。

2019年11月5日

# AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「[AWS 用語集](#)」を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。