aws

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

Table of Contents

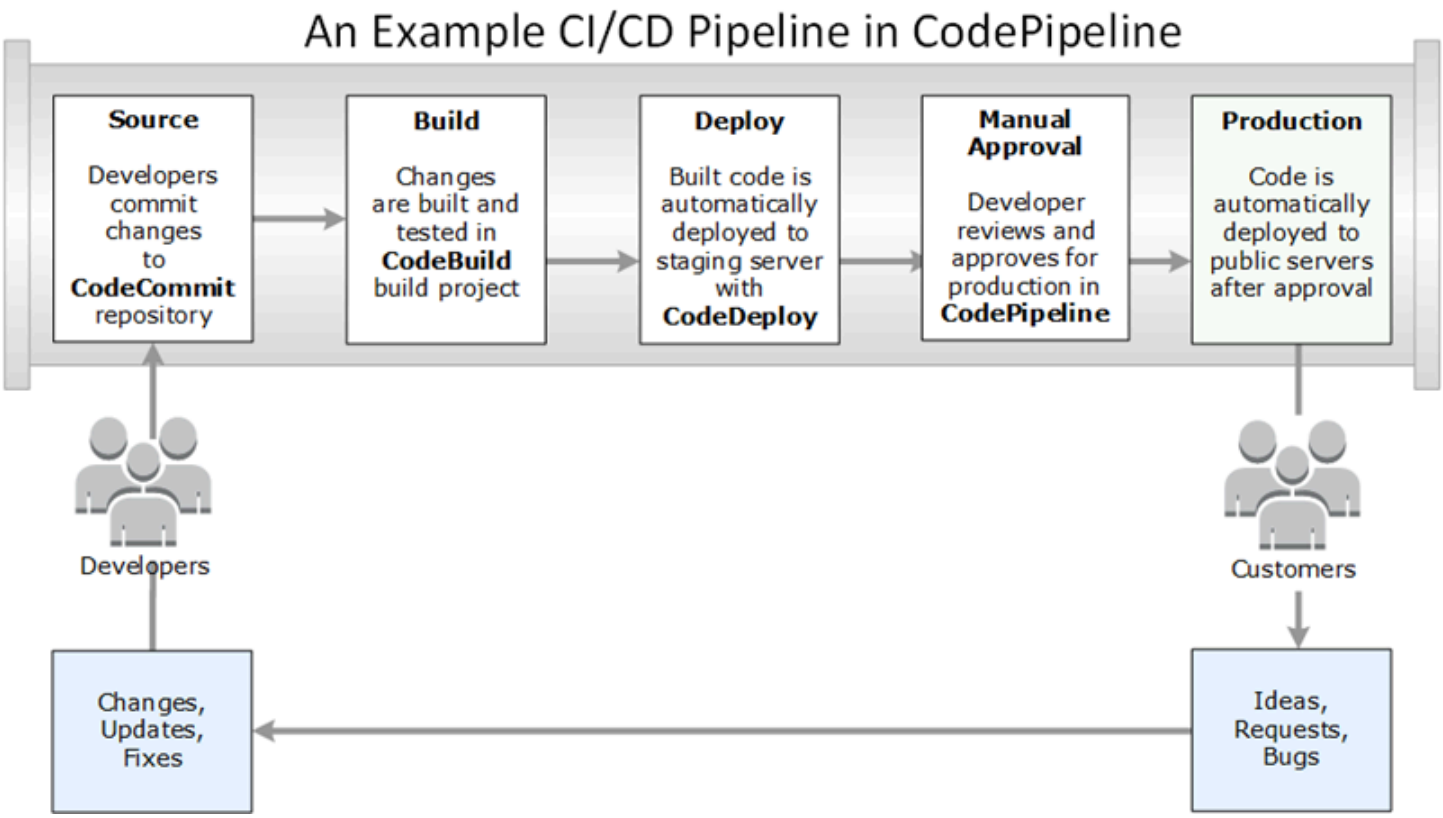
デベロッパーツールコンソールとは

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、個別にまたはまとめて使用し、DevOps、継続的インテグレーション、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して、CodeBuild で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、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メントを自動化する完全マネージド型のデプロイメントサービスです。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すばや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。



この例では、開発者が CodeCommit でリポジトリを作成し、それを使用してコードを開発して、共同作業します。CodeBuild でビルドプロジェクトを作成してコードをビルドおよびテストし、CodeDeploy を使用してテスト環境と実稼働環境にコードをデプロイします。すばやい反復処理が必要なため、CodePipeline でパイプラインを作成し、CodeCommit のリポジトリ内の変更を検出します。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レビューし、問題がない場合、本番用の変更を手動で承認します。CodePipeline は、テストされ承認されたコードを本番稼働用インスタンスにデプロイします。

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。AWS 内や他のサードパーティー製ツールの両方で 、他の製品やサービスとの多くの統合が利用できます。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 と連携した CodeBuild を使用する
- CodeDeploy: 製品およびサービスの統合

- CodePipeline:製品およびサービスの統合

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
- CodeBuild の開始方法、概念
- https://docs.aws.amazon.com/codedeploy/latest/userguide/getting-started-codedeploy.html CodeDeploy の使用開始、プライマリコンポーネント
- CodePipeline の使用開始、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は、AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy および AWS CodePipeline のイベントに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を備えています。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 である AWS CodeStar Connections があります。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証済み接続をセットアップし、他の AWS のサービスと併せて接続リソースを使用できます。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 リージョンに存在している必要があります。

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) 成功 進行中 停止	codebuild-project-build-state-failed codebuild-project-build-state-succeeded codebuild-project-build-state-in-progress codebuild-project-build-state-stopped
ビルドフェーズ	失敗 成功	codebuild-project-build-phase-failure 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) 成功 Started	codedeploy-application-deployment-failed codedeploy-application-deployment-succeeded 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 [Failed] (失敗) キャンセル Started	codepipeline-pipeline-action-execution-succeeded codepipeline-pipeline-action-execution-failed codepipeline-pipeline-action-execution-canceled codepipeline-pipeline-action-execution-started
ステージの実行	Started 成功 再開 Canceled [Failed] (失敗)	codepipeline-pipeline-stage-execution-started codepipeline-pipeline-stage-execution-succeeded codepipeline-pipeline-stage-execution-resumed codepipeline-pipeline-stage-execution-canceled codepipeline-pipeline-stage-execution-failed
パイプラインの実行	[Failed] (失敗) キャンセル Started 再開 成功	codepipeline-pipeline-pipeline-execution-failed codepipeline-pipeline-pipeline-execution-canceled 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 AWS CodePipeline の管理ポリシーが IAM ユーザーまたはロールに適用されている場合、ポリシーによって提供されるロールおよび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を IAM ユーザーまたはロールに適用し、CodeBuild のビルドプロジェクト、CodeCommit のリポジトリ、CodeDeploy のデプロイアプリケーション、または CodePipeline のパイプラインに適用している場合、最初の通知ルールを作成する準備ができています。「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: CodeBuild の開始方法
- CodeCommit: CodeCommit の開始方法
- CodeDeploy: チュートリアル

- CodePipeline: CodePipeline の使用開始

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 ユーザーを使用してサインインするか、サービスおよび通知を作成するサービス (AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy または AWS CodePipeline) にアクセスするアクセス許可を持つロールを使用できます。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。ポリシーの範囲に応じて、ポリシーに適切な通知機能へのアクセスが含まれる、CodeBuild、CodeCommit、CodeDeploy、または CodePipeline の管理ポリシーを使用することもできます。

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 ポリシーエディタを使用してポリシーを作成するには

1. AWS Management Console にサインインして、IAM コンソール (https://console.aws.amazon.com/iam/) 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、[管理ポリシーにようこそ] ページが表示されます。[Get Started] (今すぐ始める) を選択します。
3. ページの上部で、[ポリシーの作成] 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. [Next] (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. [Create Policy] (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。

- 通知ルールを作成する対象のリソース (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) と同じ AWS リージョン に作成されています。
- 2019 年 11 月 5 日より前の CodeCommit の通知を送信するためには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ー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。
- トピックに通知を発行することを AWS CodeStar Notifications に許可するポリシーがあります。

AWS CodeStar Notifications の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には

1. AWS Management Console にサインインし、Amazon SNS コンソール (https://console.aws.amazon.com/sns/v3/home) 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。

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、および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",
        "SNS:Receive"
      ],
      "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. [Save changes] (変更の保存) をクリックします。
6. AWS KMS で暗号化された Amazon SNS トピックを使用して通知を送信するには、以下のステートメントを AWS KMS key のポリシーに追加して、イベントソース (AWS CodeStar Notifications) と暗号化されたトピックの間の互換性も保つ必要があります。AWS リージョン (この例では us-east-2) を、キーが作成された AWS リージョン に置き換えます。

```
{
    "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. AWS Management Console にサインインし、Amazon SNS コンソール (https://console.aws.amazon.com/sns/v3/home) 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 Notifications 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

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. 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。

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. ビルドプロジェクトに移動し、ビルドを開始します。
11. 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。

1. 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。

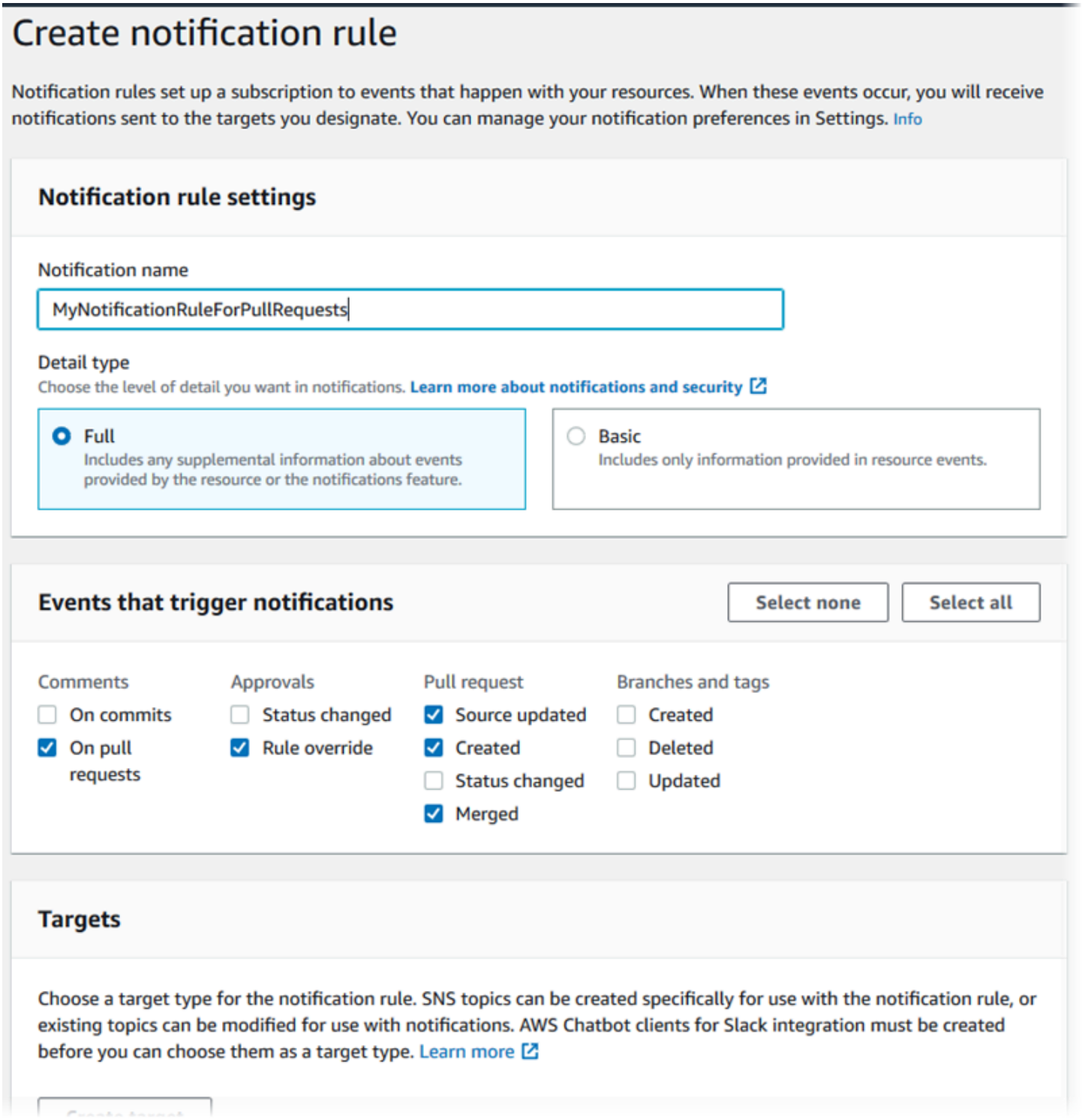
[codestar-notifications-] プレフィックスの後にトピックの名前を入力し、[送信] を選択します。

> **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。



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",
    "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。

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。Amazon SNS トピックを選択するか、Slack チャンネルまたは Microsoft Teams チャネル用に設定された 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。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。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。

デベロッパーツールコンソールまたは AWS CLI を使用して、通知ターゲットを管理できます。コンソールまたは AWS CLI を使用して、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。また、ターゲットとして設定した Amazon SNS トピックと AWS Chatbot と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、Amazon SNS トピックであるか、Slack チャンネルまたは Microsoft Teams チャネル用に設定された AWS Chatbot クライアントです。

クライアントをターゲットとして選択するには、事前に AWS Chatbot クライアントを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 チャネルに通知を送信するため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。

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. AWS Management Console にサインインし、Amazon SNS コンソール (https://console.aws.amazon.com/sns/v3/home) 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、および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",
        "SNS:Receive"
      ],
      "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. [Save changes] (変更の保存) をクリックします。
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 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。

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。

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. 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 rule targets] (通知ルールのターゲット) 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 の通知ルールで使用されてい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、このコマンドは以下のような AWS リージョンの各通知ルールの 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 AWS CLI を使用して、通知ルールのターゲットを変更します。

通知ルールのターゲットを変更するには (コンソール)

1. 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、[ターゲットを追加] を選択し、リストから追加する Amazon SNS トピックを選択するか、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. [Submit] (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に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. 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 rule targets] (通知ルールのターゲット) 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。
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に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 チャネルの設定) の [Channel type] (チャネルタイプ) で、統合するチャ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 (パブリックチャネル)] で、Slack チャネルの名前をリストから選択します。
    - [Private channel ID (プライベートチャネル ID)] に、チャネルコードまたは 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と、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 メールの受信トレイと Slack チャンネル、Microsoft Teams チャネル、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?

開発者ツールコンソールの接続機能を使用して、 AWS AWS CodePipeline 外部のコードリポジトリなどのリソースに接続できます。この機能には、AWS CodeStar 接続 API リファレンスという独自の API があります。各接続は、 AWS BitBucketなどのサードパーティのリポジトリに接続するためのサービスに提供できるリソースです。たとえば、 CodePipeline サードパーティのコードリポジトリでコードが変更されたときにパイプラインがトリガーされるように、接続を追加することができます。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。

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プロバイダに接続し、そのサードパーティConnect AWS CodePipeline をなどのリソースとのソース統合として使用します。

- CodeBuild サードパーティプロバイダのビルドプロジェクト、 CodeDeploy アプリケーション、CodePipeline パイプライン内のリソース全体にわたる接続へのアクセスを統一的に管理します。
- スタックテンプレートの接続 ARN は、保存されたシークレットやパラメータを参照しなくても CodePipeline、 CodeBuild ビルドプロジェクト、 CodeDeployアプリケーション、パイプラインに使用できます。

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?

接続により、 AWS リソースを以下のサードパーティリポジトリに関連付けることができます。

- Bitbucket Cloud
- GitHub
- 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
- GitLab 自己管理型インストール (エンタープライズエディションまたはコミュニティエディション用)

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。

クラウドプロバイダータイプ (など) の接続を作成する手順は GitHub、 GitHub Enterprise Server などのインストール済みプロバイダーの種類の接続を作成する手順とは異なります。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**
>
> ヨーロッパ (ミラノ) で接続を使用するには AWS リージョン、次のことを行う必要があります。
>
> 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。

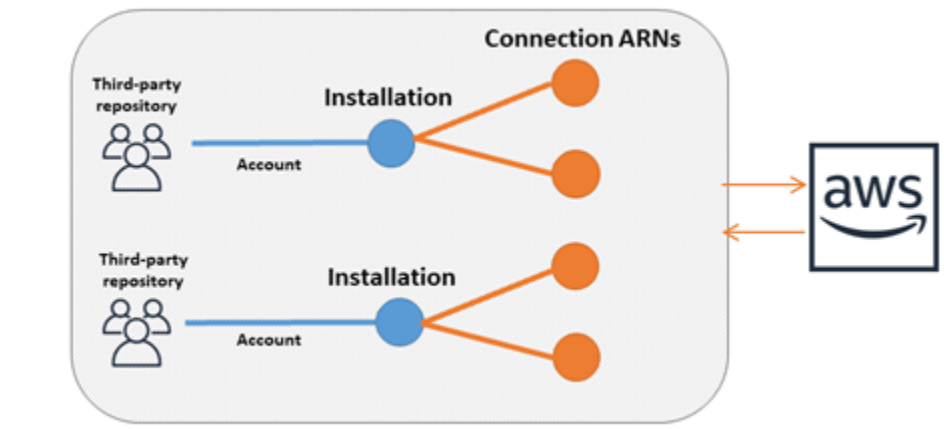
AWS のサービス コネクションと統合できるものは何か？

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、「AWS CodeStar Connections との製品とサービスの統合」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、 AWS コネクションがリソースに代わって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、 AWS のサービス その接続を他のユーザーと共有して、リソース間の安全な OAuth 接続を実現できます。

GitHubEnterprise Server など、インストールされているプロバイダーの種類への接続を作成する場合は、まずを使用してホストリソースを作成します。 AWS Management Console



コネクションは、 AWS アカウント コネクションを作成した人が所有します。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、Available接続は作成され、 AWS などのサービスで使用できるようになります 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。

AWS CodeStar Connections のグローバルリソース

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ー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。

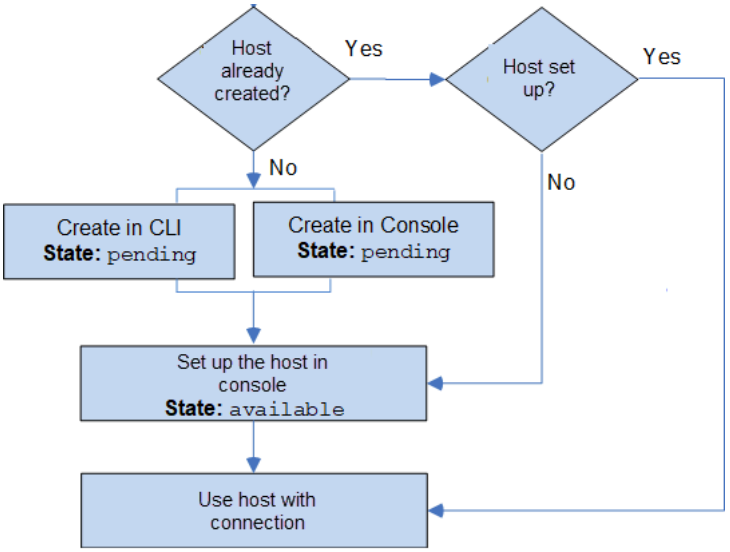
ワークフロー: CLI、SDK、または AWS CloudFormationを使用したホストの作成または更新

CreateHostAPI を使用して AWS Command Line Interface (AWS CLI)、SDK、または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

GitHubEnterprise Server GitLab やセルフマネージドなど、インストール済みの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。

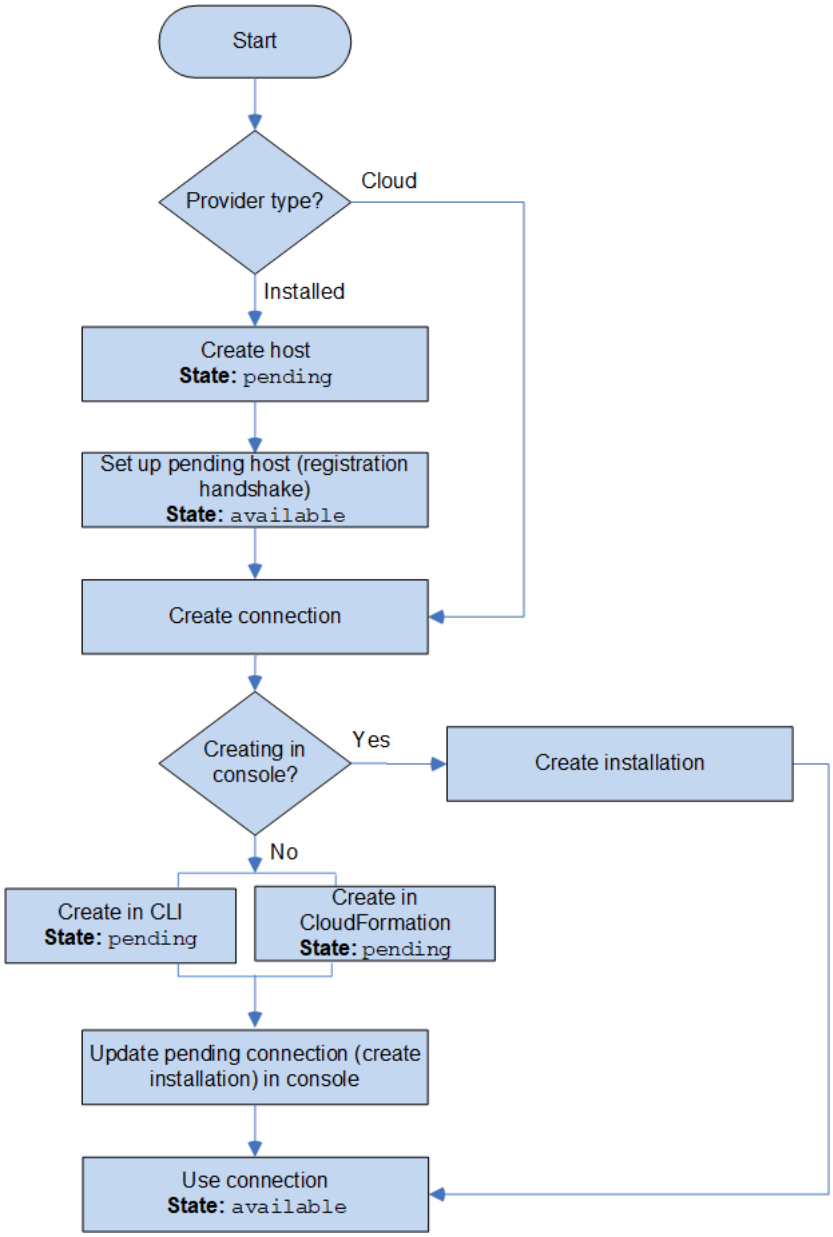
ワークフロー: CLI、SDK、 AWS CloudFormationを使用した接続の作成または更新

CreateConnectionAPI を使用して AWS Command Line Interface (AWS CLI)、SDK、 AWS CloudFormationまたはを使用して接続を作成します。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

GitHubEnterprise 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、 CodePipeline コンソールの編集アクションページを使用してサードパーティ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 CodeStar 接続がサポートするプロバイダーとバージョン

この章では、 AWS CodeStar Connections がサポートする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

AWS CodeStar このアプリはアトラシアン 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。

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

AWS GitHub GitHub アプリケーション用コネクタはエンタープライズクラウドで使用できます。 GitHub

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

AWS CodeStar 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 CodeStar Connections GitHub は廃止予定のエンタープライズサーバーバージョンをサポートしていません。たとえば、リリースには既知の問題があるため、 AWS CodeStar Connections 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。

以下のプロバイダータイプがサポートされています。 GitLab

との接続を使用できます GitLab。詳細については、「への接続を作成します。 GitLab」を参照してください。

GitLab セルフマネージドでサポートされるプロバイダータイプ

GitLab 接続は自己管理型インストール (エンタープライズエディションまたはコミュニティエディション) で使用できます。詳細については、「 GitLabセルフマネージド接続を作成します。」を参照してください。

AWS CodeStar Connections との製品とサービスの統合

AWS CodeStar Connections は多数の AWS のサービス、およびパートナーの製品や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 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。CodeGuru Reviewer を設定して GitHub リポジトリ内のソースコードをモニタリングし、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する方法のチュートリアルについては、Amazon CodeGuru Reviewer ユーザーガイドの「Tutorial: monitor source code in a GitHub repository」を参照してください。

Amazon CodeWhisperer

Amazon CodeWhisperer は、リポジトリコードをレビューするためのサービスです。CodeWhisperer は、コードをレビューし、リアルタイムで推奨コードを提供します。接続を使用してデータソースにアクセスするように CodeWhisperer のカスタマイズを設定する手順については、「Amazon CodeWhisperer 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。

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 開発者ガイド」の「サードパーティの Git リポジトリを使用する SageMaker 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、スケー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およびセットアップに役立つサービスです。リソース管理に割く時間を減らし、AWS で実行するアプリケーションにさらに注力できるようになります。使用するすべての 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すれば、CloudFormation がお客様に代わってこれらのリソースのプロビジョニングや設定を受け持ちます。詳細については、「CloudFormation コマンドラインインターフェイスユーザーガイド」の「アカウントを登録して CloudFormation 拡張機能を発行する」を参照してください。

AWS CodePipeline

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、視覚化、および自動化に使用できる継続的な配信サービスです。接続を使用して、CodePipeline ソースアクション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:

- `CodeStarSourceConnection` アクションについては、CodePipeline アクション設定の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 スニペット例を表示する場合は、AWS CodePipeline ユーザーガイドの「CodeStarSourceConnection」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、AWS でソフトウェア開発プロジェクトを作成、管理、および操作するクラウドベースのサービスです。AWS CodeStar プロジェクトを使用して、AWS でアプリケーションをすばやく開発、構築、およびデプロイすることができます。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。GitHub リポジトリに接続して 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、「AWS CodeStar ユーザーガイド」の「リポジトリへのリンクを作成する」を参照してください。

Service Catalog

Service Catalog により、組織は AWS での使用が承認された製品のカタログを作成および管理できます。

自分の AWS アカウントと外部リポジトリプロバイダー (GitHub、GitHub Enterprise、BitBucket など) との接続を許可すると、その接続により、Service Catalog 製品をサードパーティのリポジトリで管理されているテンプレートファイルに同期できるようになります。

詳細については、「Service Catalog ユーザーガイド」の「Service Catalog 製品を GitHub、GitHub Enterprise、または Bitbucket のテンプレートファイルに同期する」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ー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

- AWS にサインアップする
- 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

AWS にサインアップする

AWS アカウントへのサインアップ

AWS アカウントがない場合は、以下のステップを実行して作成します。

AWS アカウントにサインアップするには

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、電話のキーパッドを使用して検証コードを入力するように求められます。

   AWS アカウントにサインアップすると、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、管理ユーザーに管理アクセスを割り当て、ルートユーザーアクセスが必要なタスクを実行する場合にのみ、ルートユーザーを使用してください。

サインアップ処理が完了すると、AWS からユーザーに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。

管理ユーザーの作成

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

AWS アカウントのルートユーザーをセキュリティで保護する

1. [ルートユーザー] を選択し、AWS アカウント のメールアドレスを入力して、アカウント所有者として AWS Management Console 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 User Guide」の「Signing in as the root user」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 アカウントの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。

管理ユーザーを作成する

1. IAM Identity Center を有効にする

   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、管理ユーザーに管理アクセス権を付与します。

   IAM アイデンティティセンターディレクトリをアイデンティティソースとして使用するチュートリアルについては、「AWS IAM Identity Centerユーザーガイド」の「デフォルト IAM アイデンティティセンターディレクトリでのユーザーアクセスの設定」を参照してください。

管理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 アイデンティティセンターの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 User Guide」の「Signing in to the AWS access portal」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. AWS Management Console にサインインして、IAM コンソール (https://console.aws.amazon.com/iam/) 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、[管理ポリシーにようこそ] 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": [
               "codestar-connections:CreateConnection",
               "codestar-connections:DeleteConnection",
               "codestar-connections:GetConnection",
               "codestar-connections:ListConnections",
               "codestar-connections:GetInstallationUrl",
               "codestar-connections:GetIndividualAccessToken",
               "codestar-connections:ListInstallationTargets",
               "codestar-connections:StartOAuthHandshake",
               "codestar-connections:UpdateConnectionInstallation",
               "codestar-connections:UseConnection"
           ],
           "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

接続の使用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 CodeCommit などの AWS ソースに接続する場合は、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 Reviewer 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 のサービスに接続し、認証を管理するための接続を許可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 の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。AWS Management Console を使用して接続を設定すると、Bitbucket の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。
- AWS CodeBuild ビルドアクションを含むビルドステージ。

ウィザードを使用してパイプラインを作成するには

1. CodePipeline コンソール (http://console.aws.amazon.com/codesuite/codepipeline/home) にサインインします。
2. [ようこそ] ページ、[Getting started] (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**
>
> 既存の CodePipeline サービスロールを代わりに使用する場合は、サービスロールポリシーに対する `codestar-connections:UseConnection` IAM アクセス許可を追加した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、「Add permissions to the the CodePipeline service role」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。

[Next] (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。[Grant access (アクセス権の付与)] を

選択します。接続を許可すると、Bitbucket 上のリポジトリが検出され、AWS リソースに関連付けることを選択できます。

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。CodePipeline コンソールに戻ります。

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、次の [CodePipeline デフォルト] のいずれかを選択します。

- [CodePipeline デフォルト] を選択して、パイプライン内のアーティファクトにデフォルトの zip 形式を使用します。
- [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、CodeBuild アクションでのみサポートされます。

[Next] (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. [Create project] (プロジェクトの作成) 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:
```

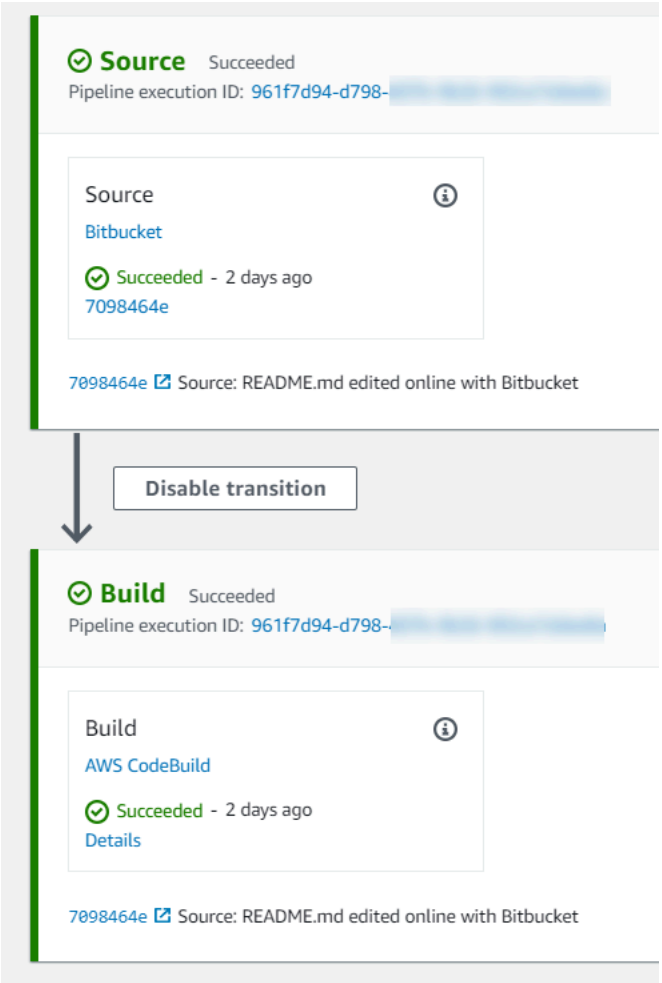
```
  install:
    #If you use the Ubuntu standard image 2.0 or later, you must specify
 runtime-versions.
    #If you specify runtime-versions and use an image other than Ubuntu
 standard image 2.0, the build fails.
    runtime-versions:
      nodejs: 12
      # name: version
    #commands:
      # - command
      # - command
  pre_build:
    commands:
      - ls -lt
      - cat README.md
  # build:
    #commands:
      # - command
      # - command
  #post_build:
    #commands:
      # - command
      # - command
#artifacts:
  #files:
    # - location
    # - location
  #name: $(date +%Y-%m-%d)
  #discard-paths: yes
  #base-directory: location
#cache:
  #paths:
    # - paths
```

h. [Continue to CodePipeline] (CodePipeline に進む) を選択します。CodePipeline コンソールに戻り、ビルドコマンドを使用して設定する CodeBuild プロジェクトが作成されます。ビルドプロジェクトでは、サービスロールを使用して AWS のサービスのアクセス許可を管理します。このステップには数分かかる場合があります。

i. [Next] (次へ) をクリックします。

8. [Step 4: Add deploy stage (ステップ 4: デプロイステージの追加)] ページで、[Skip deploy stage (デプロイステージのスキップ)] を選択し、[スキップ] を選択して警告メッセージを受け入れます。[Next] (次へ) をクリックします。

9. [Step 5: Review (ステップ 5: 確認)] で、[パイプラインの作成] を選択します。
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に出力されます。

```
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 Reviewer に関連付ける

接続を作成すると、同じアカウントのすべての AWS リソースにその接続を使用できます。例えば、パイプラインの CodePipeline ソースアクションと CodeGuru Reviewer のリポジトリコミット分析に同じ Bitbucket 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 Reviewer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 への接続 (AWS CodeStar Connections を使用)] 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 User Guide の「Amazon CodeGuru Reviewer User Guide」を参照してください。

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 AWS CodePipeline などのサードパーティリポジトリへの接続などのサービスに提供できるリソースです。たとえば、 CodePipeline サードパーティのコードリポジトリでコードが変更されたときにパイプラインがトリガーされるように、接続を追加することができます。 AWS リソースを GitHub Enterprise Server などのインストール済みの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の種類への接続を作成する場合、コンソールが自動的にホストを作成します。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。

接続を作成したら、 AWS CodeStar コンソールのウィザードを使用してサードパーティプロバイダーにアプリケーションをインストールし、新しい接続に関連付けます。アプリを既にインストールしている場合は、 AWS CodeStar そのアプリを使用できます。

> **Note**
>
> ヨーロッパ (ミラノ) で接続を利用するには AWS リージョン、以下の条件を満たす必要があります。
>
> 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。

接続について詳しくは、AWS CodeStar 接続 API リファレンスをご覧ください。Bitbucket CodePipeline の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。

AWS CodeStar 接続の使用に必要な権限を持つポリシーを AWS Identity and Access Management (IAM) ユーザーまたはロールに作成またはアタッチする方法については、を参照してください。AWS CodeConnections 権限リファレンス CodePipeline サービスロールを作成した時期によっては、AWS CodeStar 接続をサポートするためにアクセス権限を更新する必要がある場合があります。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
- への接続を作成します。 GitHub

- GitHub エンタープライズサーバーへの接続を作成します。
- への接続を作成します。 GitLab
- GitLabセルフマネージド接続を作成します。
- 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します。 GitHub。
- ホストリソースの作成を含め、 GitHub Enterprise Server への接続を作成するには、を参照してください GitHub エンタープライズサーバーへの接続を作成します。 。
- 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します。 GitLab。

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 CodeStar 接続がサポートす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。

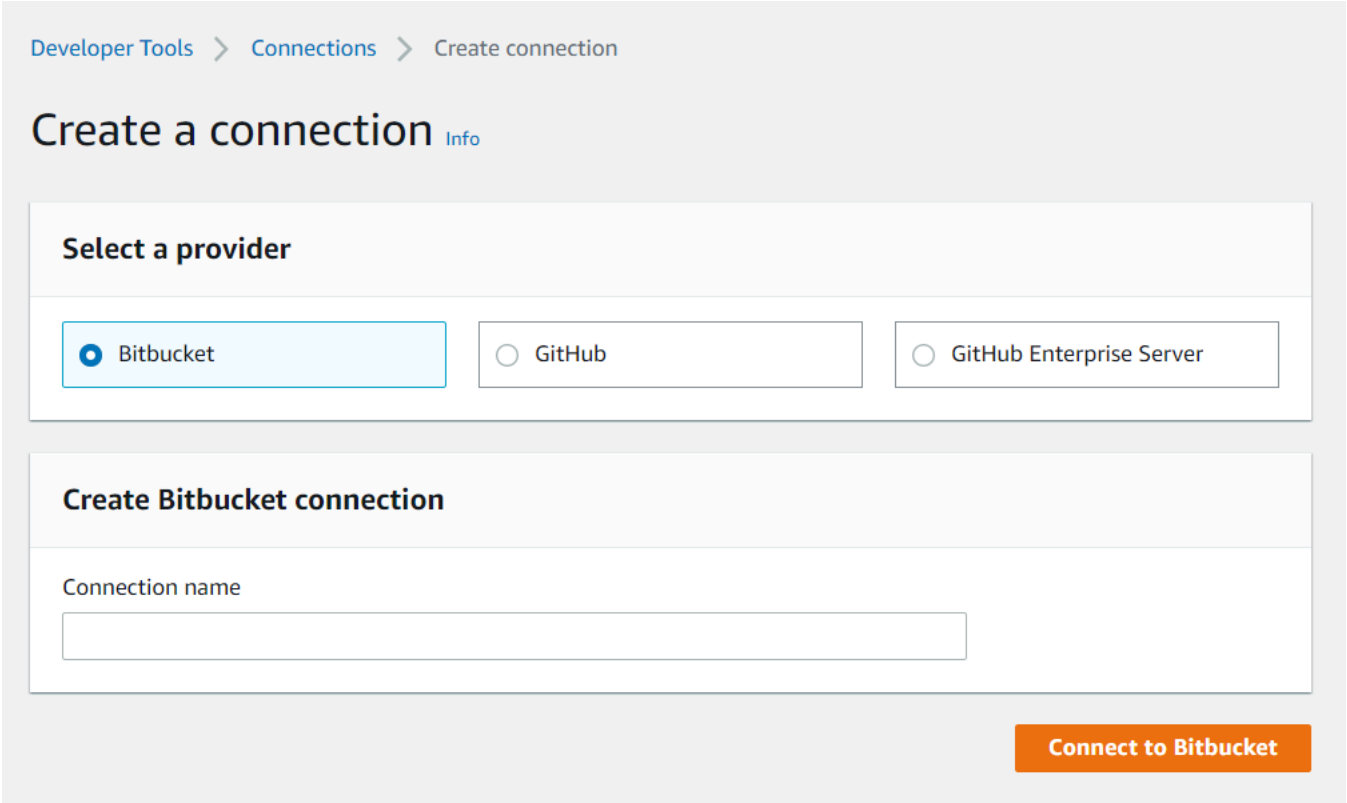
トピック

- 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

ステップ 1: 接続の作成

1. にサインインし AWS Management Console、 AWS で開発者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。

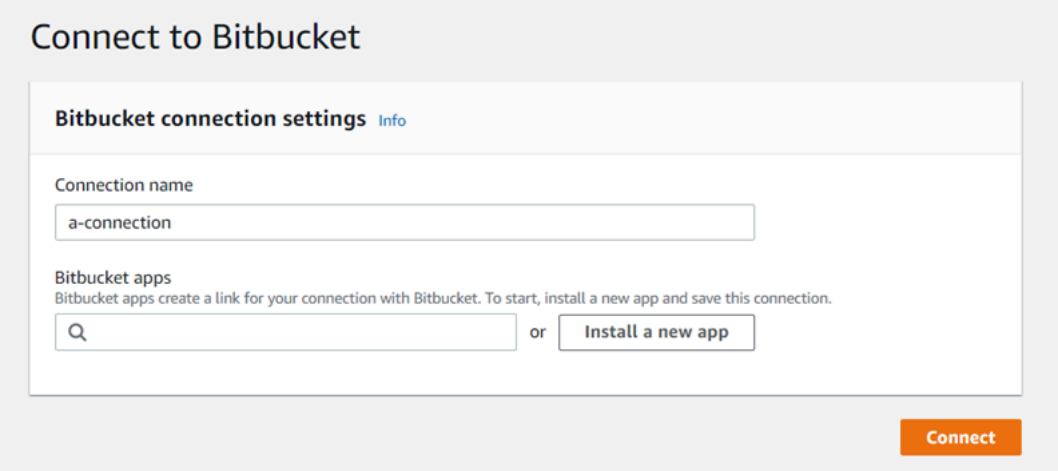


ステップ 2: Bitbucket に接続する

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。

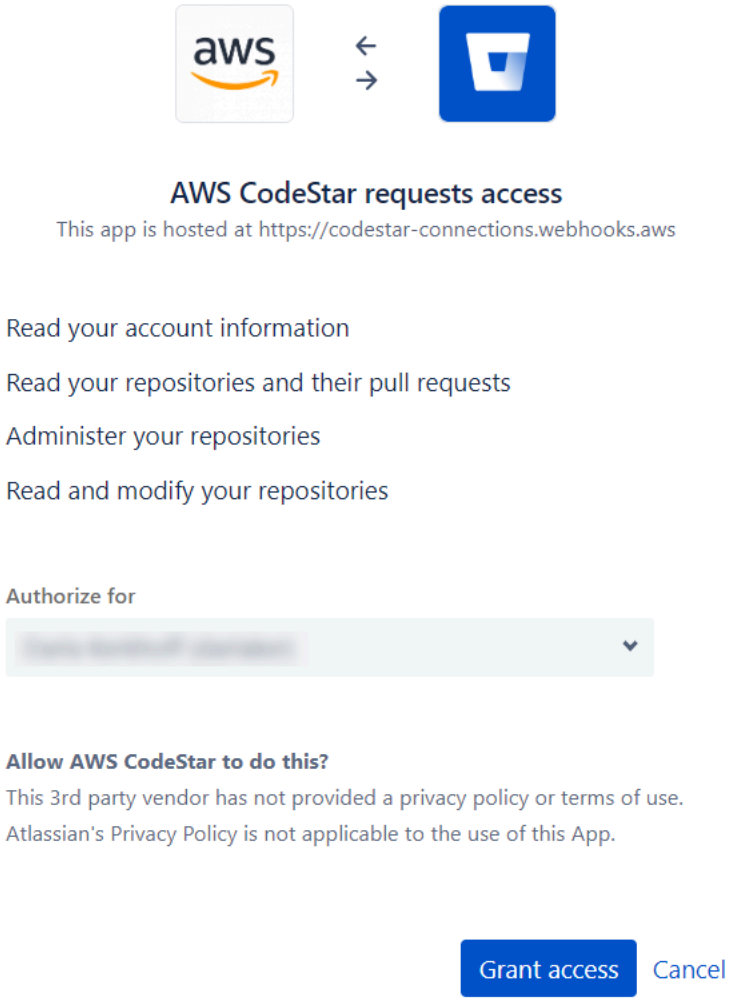
> **Note**
>
> アプリケーションは、Bitbucket ワークスペースまたはアカウントごとに 1 回だけインストールします。Bitbucket アプリを既にインストールしている場合は、それを選択してこのセクションの最後のステップに移動します。



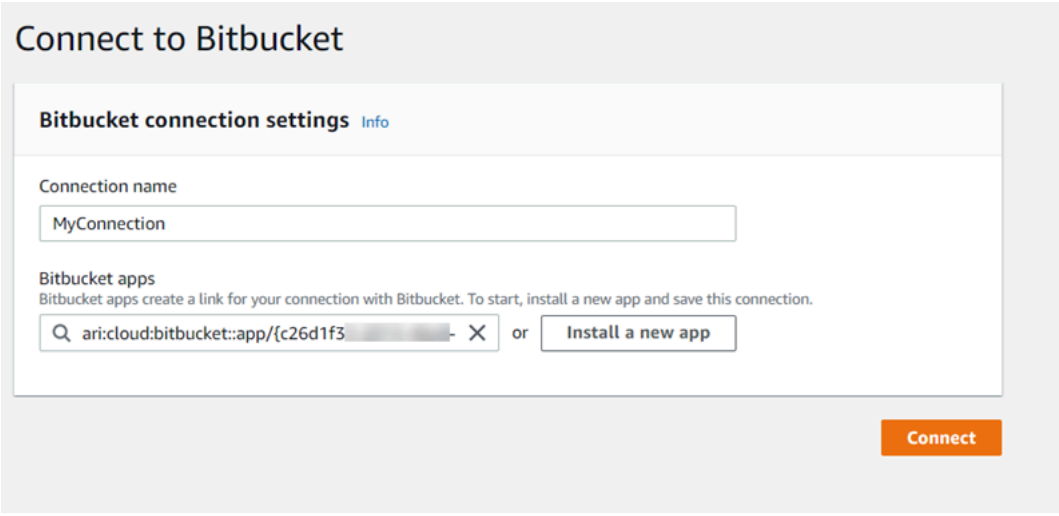
2. Bitbucket のログインページが表示されたら、認証情報を使用してログインし、続行を選択します。
3. アプリのインストールページに、 AWS CodeStar アプリが Bitbucket アカウントに接続しようとしていることを示すメッセージが表示されます。

   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。



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 AWS CloudFormation またはを介して作成された接続は、PENDINGデフォルトではステータスになっています。CLI またはを使用して接続を作成したらAWS CloudFormation、コンソールを使用して接続を編集し、ステータスを作成します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。 AWS CLI create-connectionを使用してコマンドを実行し、--provider-type--connection-name接続用のとを指定します。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 codestar-connections 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:codestar-connections: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。

への接続を作成します。 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。

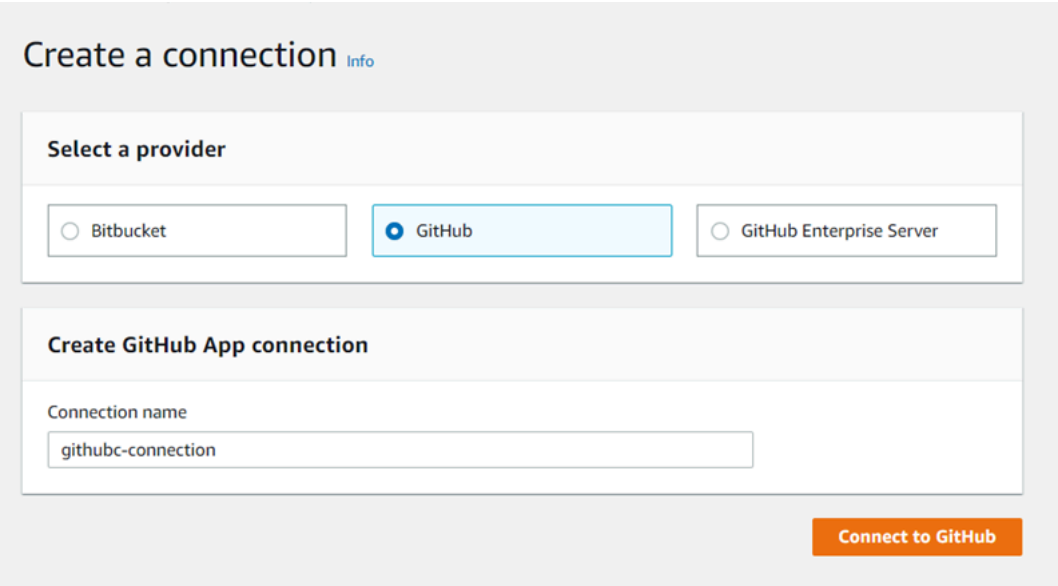
> **Note**
>
> 接続を作成するには、 GitHub 組織のオーナーである必要があります。組織のリポジトリでない場合、ユーザーがリポジトリの所有者である必要があります。

トピック

- GitHub(コンソール) への接続を作成します。
- GitHub(CLI) への接続を作成

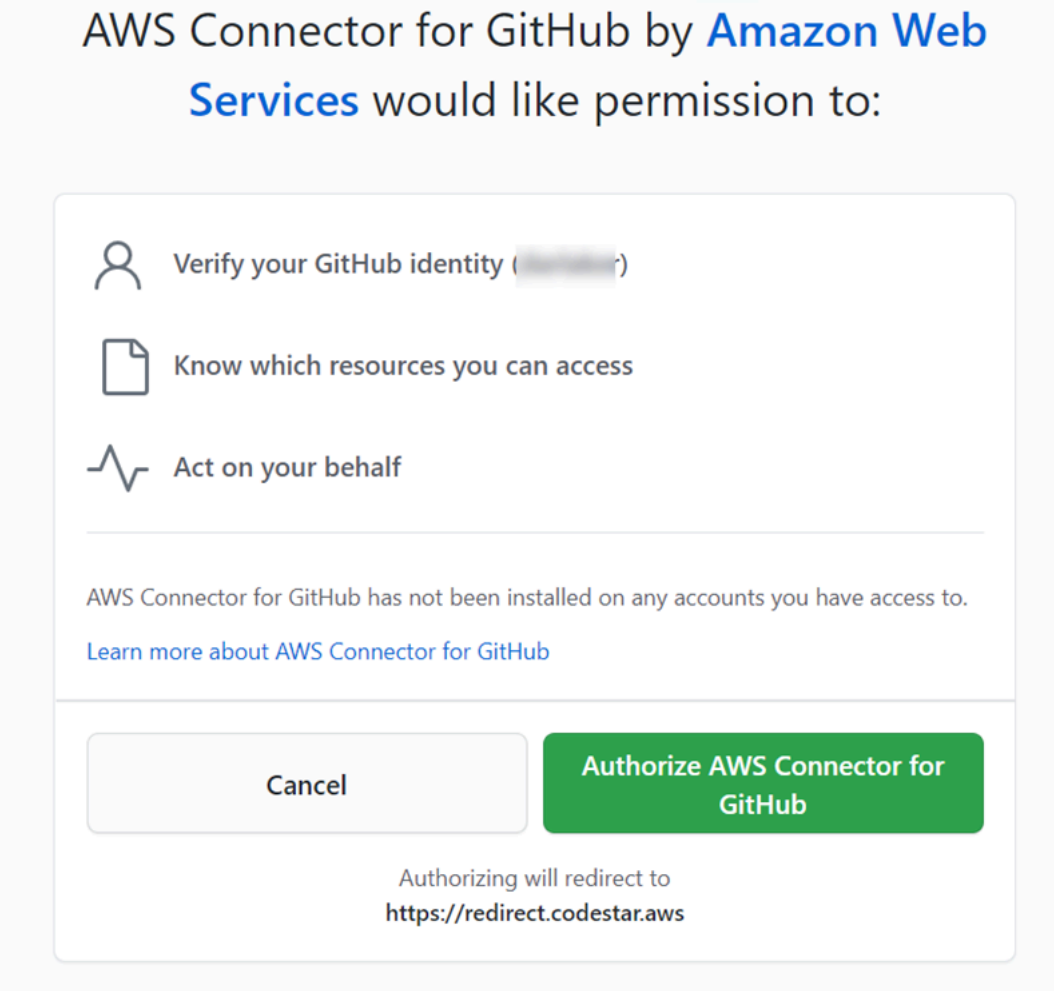
GitHub(コンソール) への接続を作成します。

1. にログインし AWS Management Console、で開発者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。 [Connect 先] を選択し GitHub、ステップ 2 に進みます。

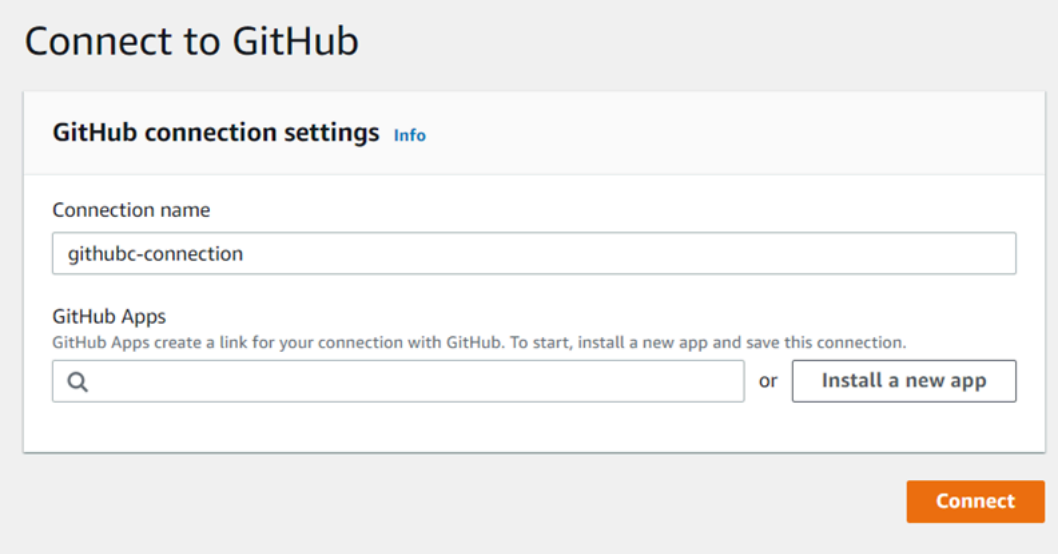


への接続を作成するには GitHub

1. [GitHub 接続設定] の [接続名] に接続名が表示されます。[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。



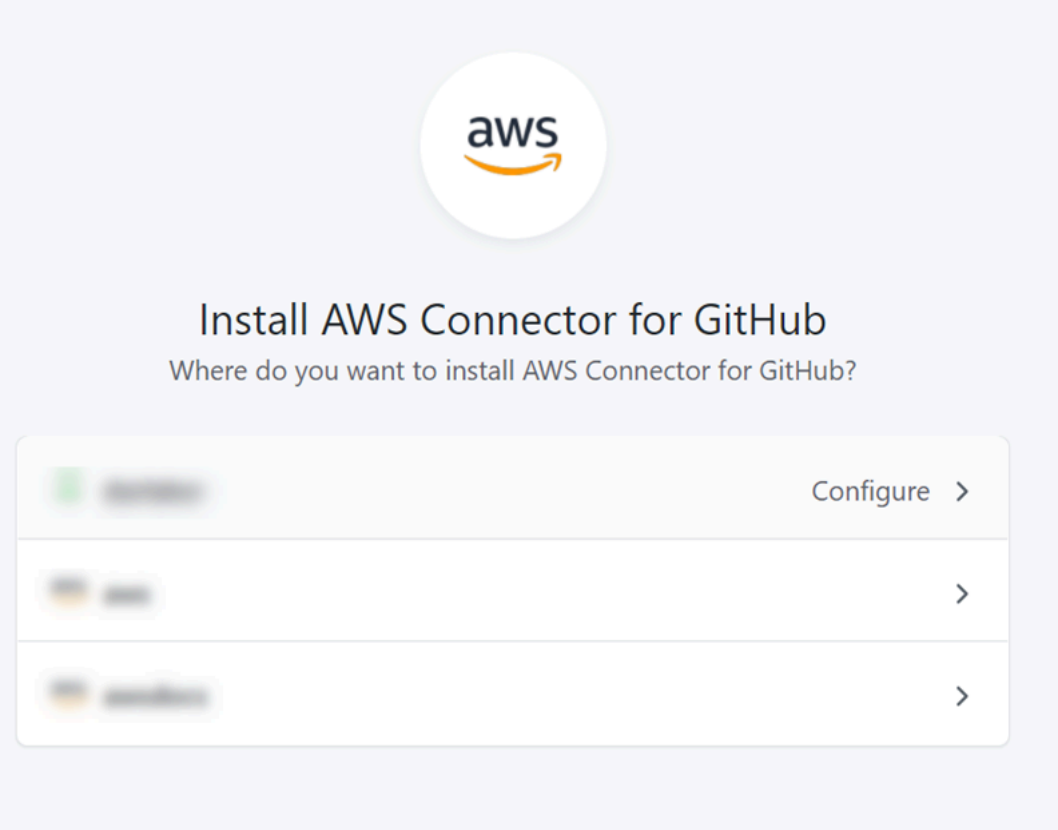
2. には [AWS コネクタを承認] を選択します。 GitHub接続ページに「GitHub Apps」フィールドが表示され、表示されます。



3. 「GitHub アプリ」で、インストールするアプリを選択するか、「新しいアプリをインストール」を選択してアプリを作成します。

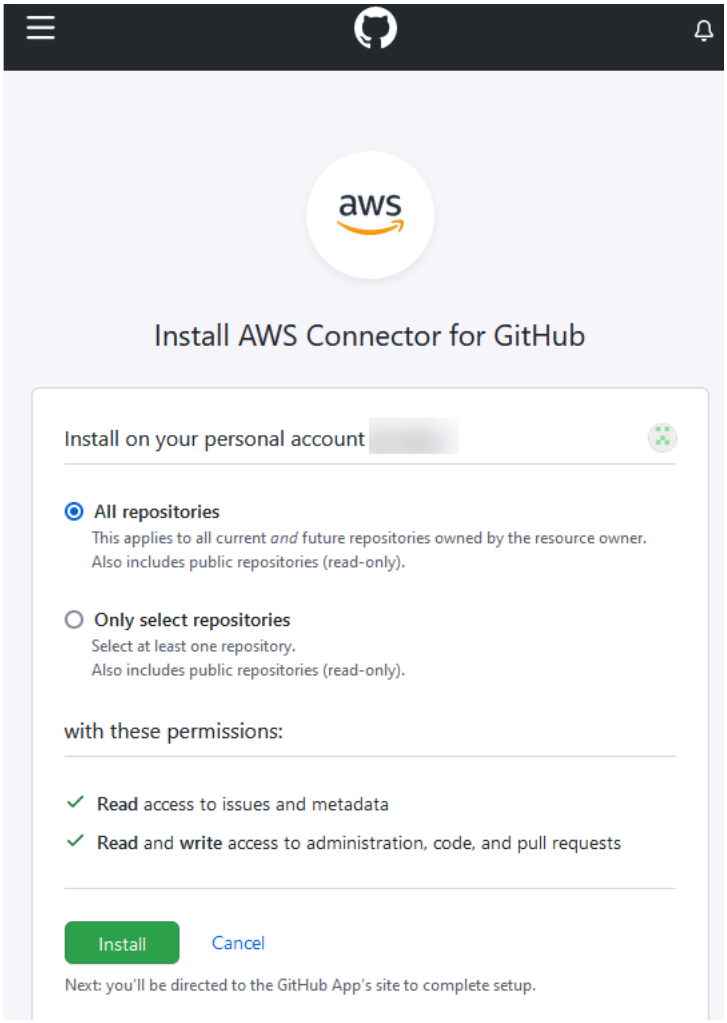
> **Note**
>
> 特定のプロバイダーへのすべての接続に対してアプリを 1 つインストールします。AWS Connector for GitHub app を既にインストールしている場合は、それを選択してこのステップをスキップしてください。

4. 「AWS Connector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。



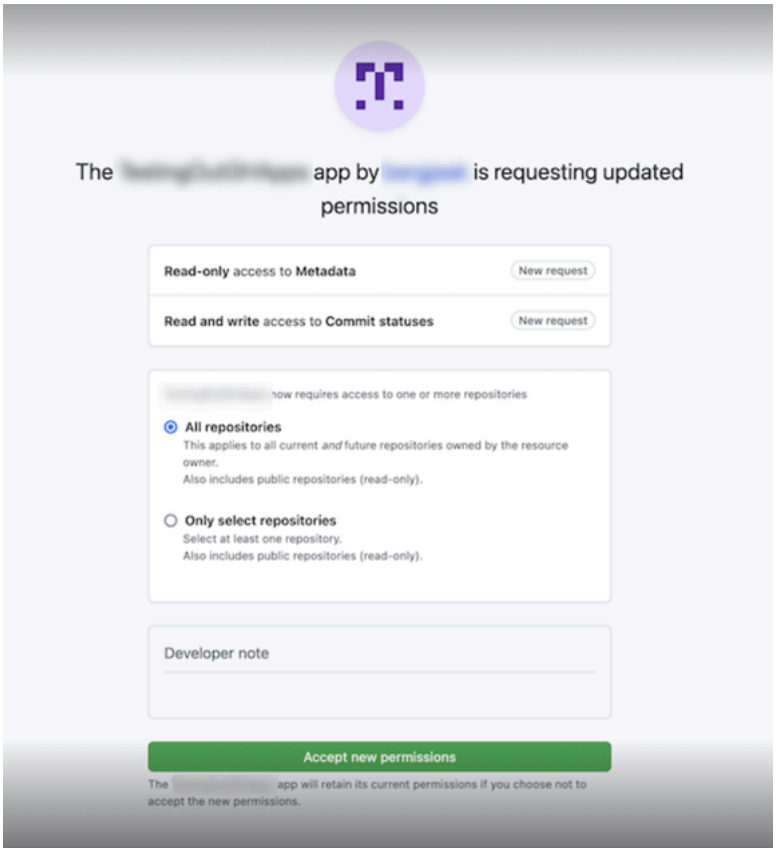
> **Note**
>
> GitHub アプリはアカウントごとに 1 回だけインストールできます。アプリケーションをインストール済みである場合は、[Configure] (設定) を選択してアプリのインストールの変更ページに進むか、戻るボタンでコンソールに戻ることができます。

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



この手順を実行すると、更新された権限ページが表示される場合があります GitHub。

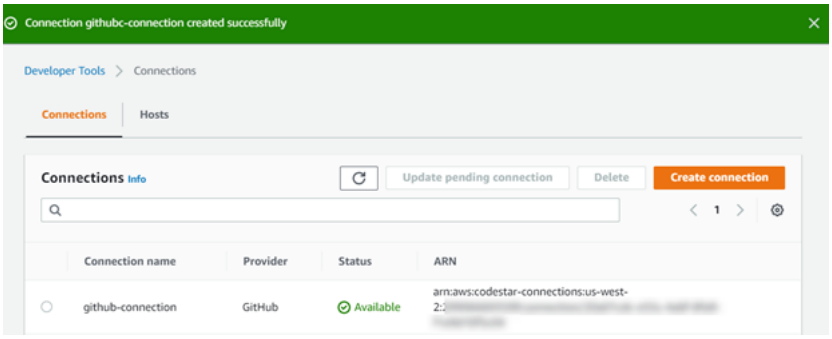
6. AWS Connector for GitHub app の権限が更新されたことを示すページが表示されたら、「新しい権限を承認」を選択します。



7. 「Connect」 GitHub ページに戻ります。新規インストールの接続 ID が [GitHubアプリ] 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。



GitHub(CLI) への接続を作成

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 AWS CloudFormation またはを介して作成された接続は、PENDINGデフォルトではステータスになっています。CLI またはを使用して接続を作成したらAWS CloudFormation、コンソールを使用して接続を編集し、ステータスを作成します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。 AWS CLI create-connectionを使用してコマンドを実行し、--provider-type--connection-name接続用のとを指定します。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 codestar-connections 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:codestar-connections: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 エンタープライズサーバーへの接続を作成します。

接続を使用して、 AWS リソースをサードパーティ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。

接続によって提供されるのは、 GitHub Enterprise Server アカウントが所有するリポジトリへのアクセスだけです。このリポジトリは、接続の作成時にアプリケーションのインストールを承認するために使用されます。 GitHub

開始する前に:

- GitHub Enterprise Server インスタンスとその中にリポジトリがあらかじめ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て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 用のホストを設定すると、Webhook イベントデータ用の 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ていて、VPC PrivateLink Webhook 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

- GitHubエンタープライズサーバー (コンソール) への接続を作成します。
- GitHubエンタープライズサーバー (CLI) への接続の作成

GitHubエンタープライズサーバー (コンソール) への接続を作成します。

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

トピック

- GitHubエンタープライズサーバー接続 (コンソール) を作成します。

GitHubエンタープライズサーバー接続 (コンソール) を作成します。

GitHub エンタープライズサーバーへの接続を作成するには、サーバー URL GitHub とエンタープライズ認証情報を用意してください。

ホストを作成するには

1. にサインインし AWS Management Console、 AWS の開発者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

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] (PC を使用) をクリックして、以下を完了します。

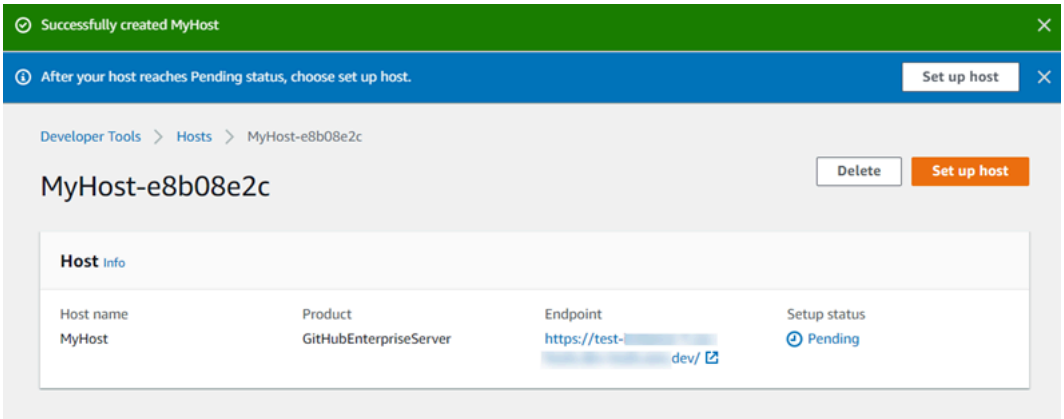
   a. [VPC ID] で、 VPC ID を選択します。インスタンスがインストールされているインフラストラクチャに VPC を選択するか、VPN または Direct Connect を介してインスタンスにアクセスできる VPC を選択します。

   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。

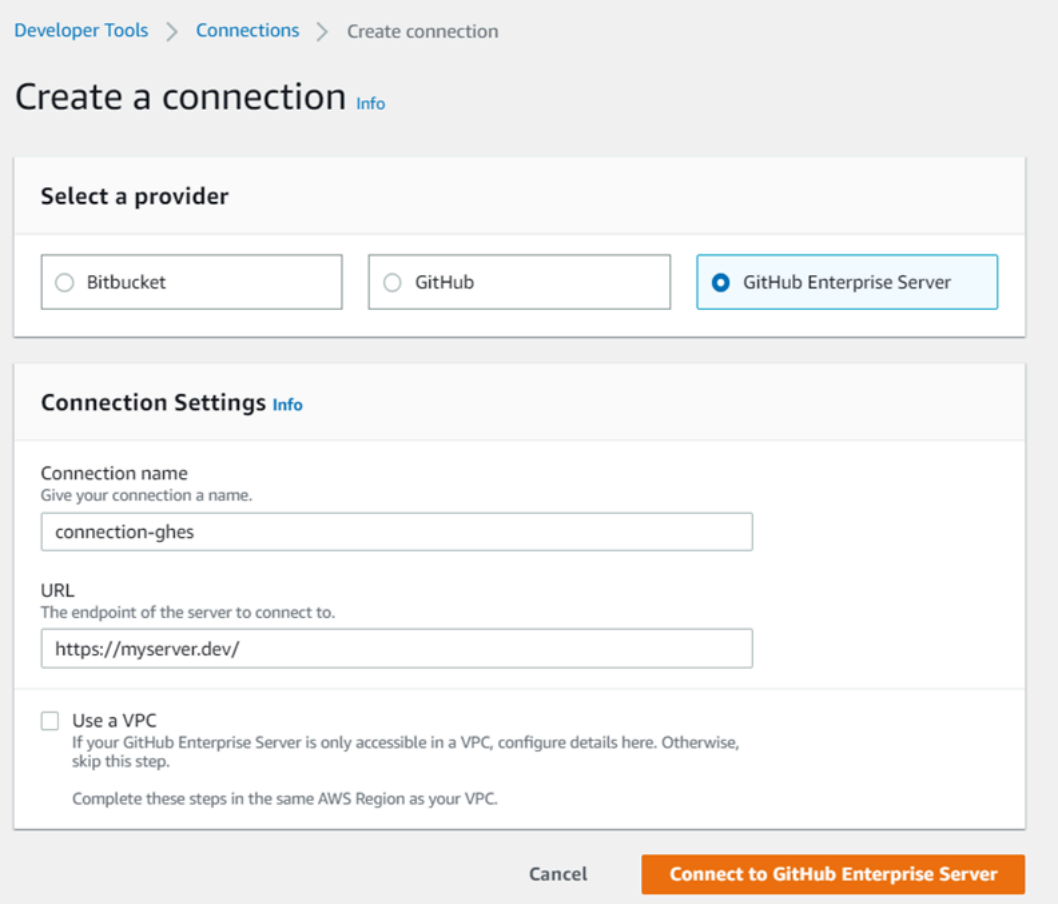


ステップ 2: GitHub エンタープライズサーバー (コンソール) への接続を作成する

1. AWS Management Console にサインインし、の開発者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。

GitHub エンタープライズサーバーにConnect

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



2. [URL] に、サーバーのエンドポイントを入力します。

> **Note**
>
> 提供された URL が接続用の GitHub Enterprise Server の設定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。必ず、 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用の VPC を選択するか、VPN または Direct Connect GitHub 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 VPC を選択してください。

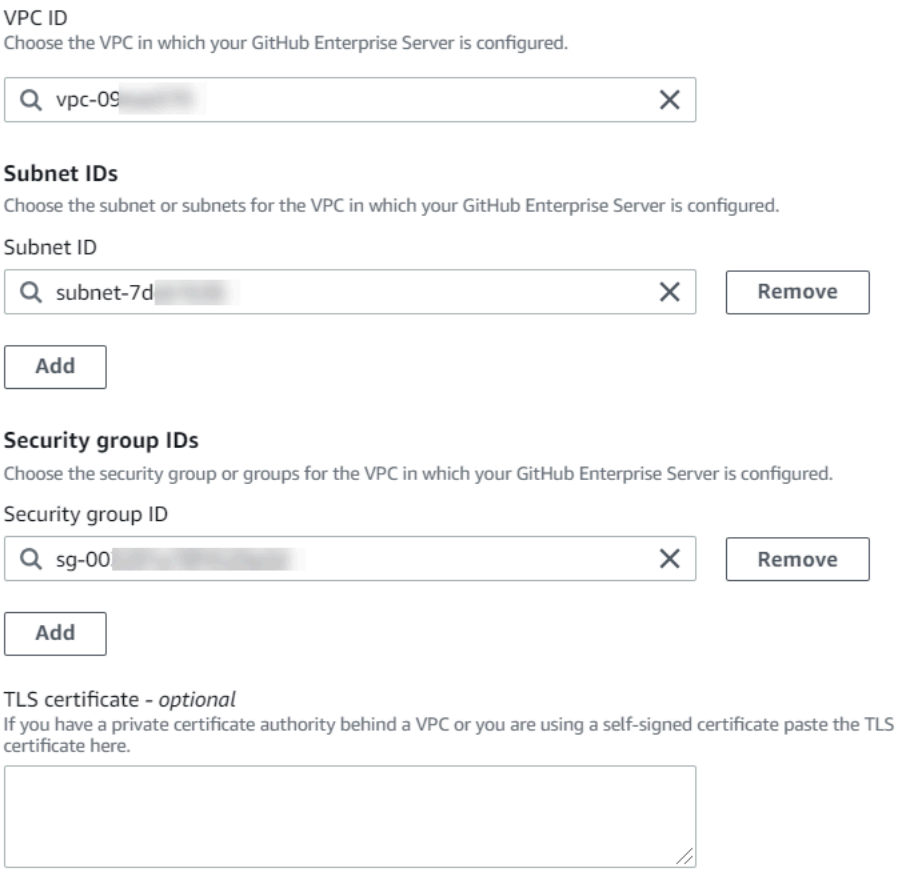
   b. [サブネット ID] で、[Add] を選択します。このフィールドで、ホストに使用するサブネット ID を選択します。最大 10 個のサブネットを選択できます。

必ず、 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サブネット、または VPN または Direct Connect を介してインストールした GitHub Enterprise Server インスタンスにアクセスできるサブネットを選択してください。

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。

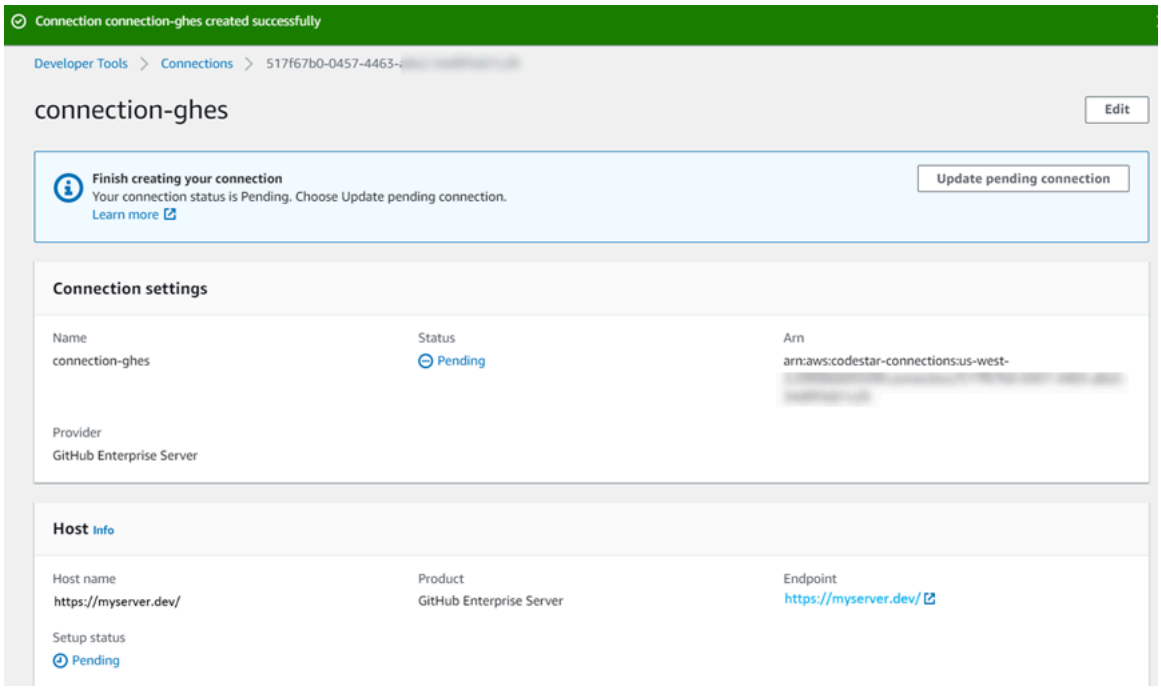
   必ず、 GitHub Enterprise Server インスタンスがインストールされているインフラストラクチャのセキュリティグループを選択するか、インストールした Enterprise Server インスタンスに VPN または Direct Connect 経由でアクセスできるセキュリティグループを選択してください。 GitHub

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 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。



4. [GitHub エンタープライズサーバーにConnect] 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。

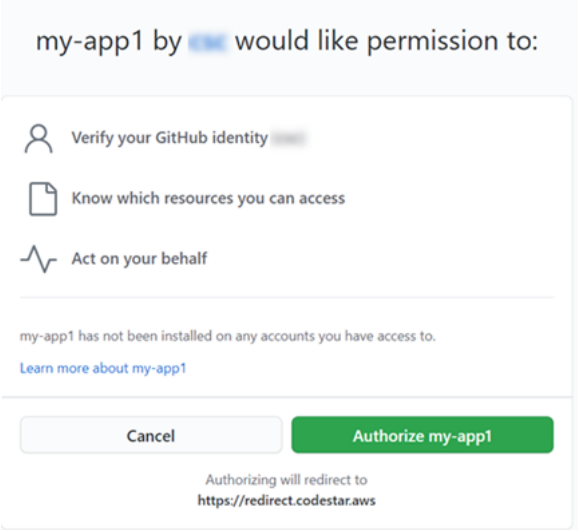
5. 保留中の接続の更新を選択します。



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、 GitHub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。

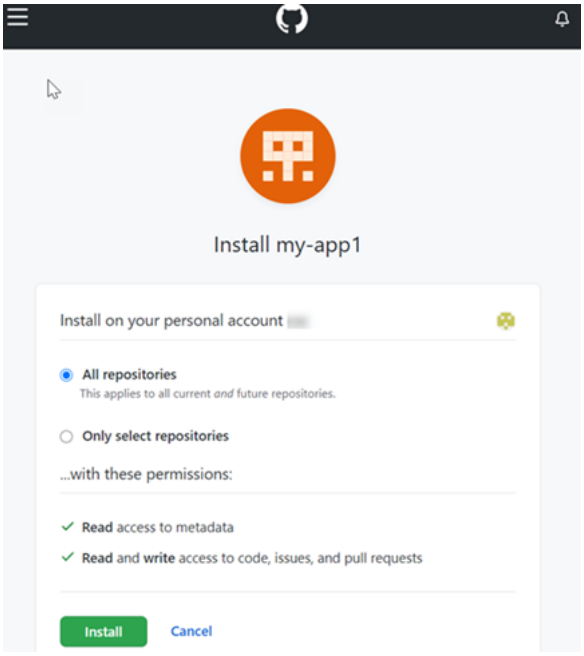
7. 「 GitHub アプリを作成」ページで、アプリの名前を選択します。

8. GitHub <app-name>認証ページで [承認] を選択します。



9. アプリのインストールページに、 AWS CodeStar Connectorアプリをインストールする準備ができた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。

GitHubエンタープライズサーバー (CLI) への接続の作成

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 AWS CloudFormation またはを介して作成された接続は、PENDINGデフォルトではステータスになっています。CLI またはを使用して接続を作成したらAWS CloudFormation、コンソールを使用して接続を編集し、ステータスを作成しますAVAILABLE。

ステップ 1: GitHub エンタープライズサーバー (CLI) 用のホストを作成するには

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。`--provider-endpoint`を使用して接続用に、`--name``--provider-type`、create-hostを指定してコマンドを実行します。 AWS CLI 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 codestar-connections 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:codestar-connections: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. AWS Management Console にサインインし、の開発者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: GitHub エンタープライズサーバー (CLI) 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。 AWS CLI create-connectionを使用してコマンドを実行し、--host-arn--connection-name接続用のとを指定します。

```
aws codestar-connections create-connection --host-arn arn:aws:codestar-
connections: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:codestar-connections: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 への接続を完了するには

1. AWS Management Console にサインインし、で開発者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。

への接続を作成します。 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**
>
> でこの接続インストールを承認することで GitLab、データを処理する権限をサービスに付与したものとみなされます。アプリケーションをアンインストールすることで、いつでもその権限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す。

> **Note**
>
> オーナーロールが割り当てられている場所で接続を作成すると 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。

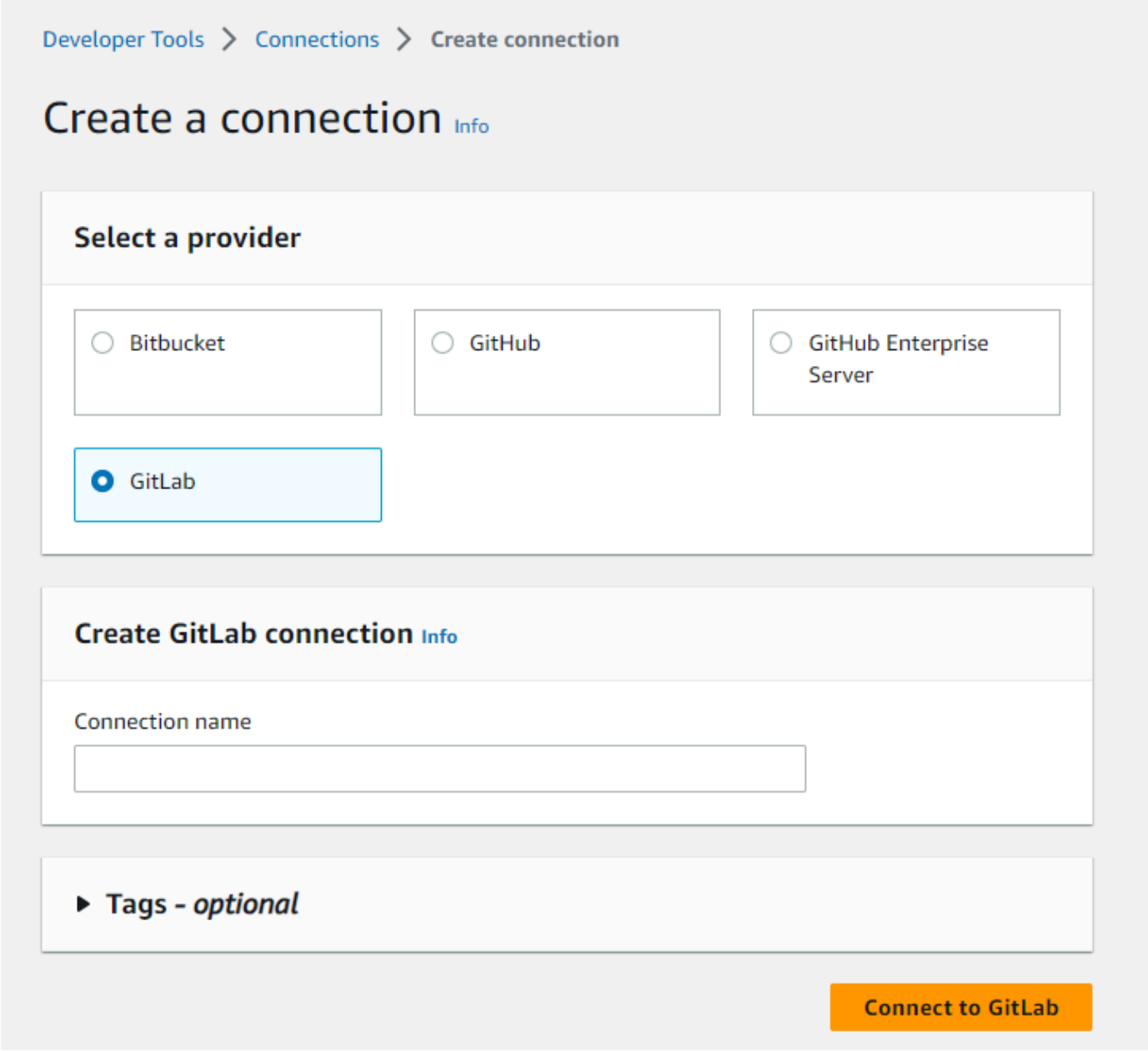
トピック

- GitLab(コンソール) への接続を作成します。
- GitLab(CLI) への接続を作成

GitLab(コンソール) への接続を作成します。

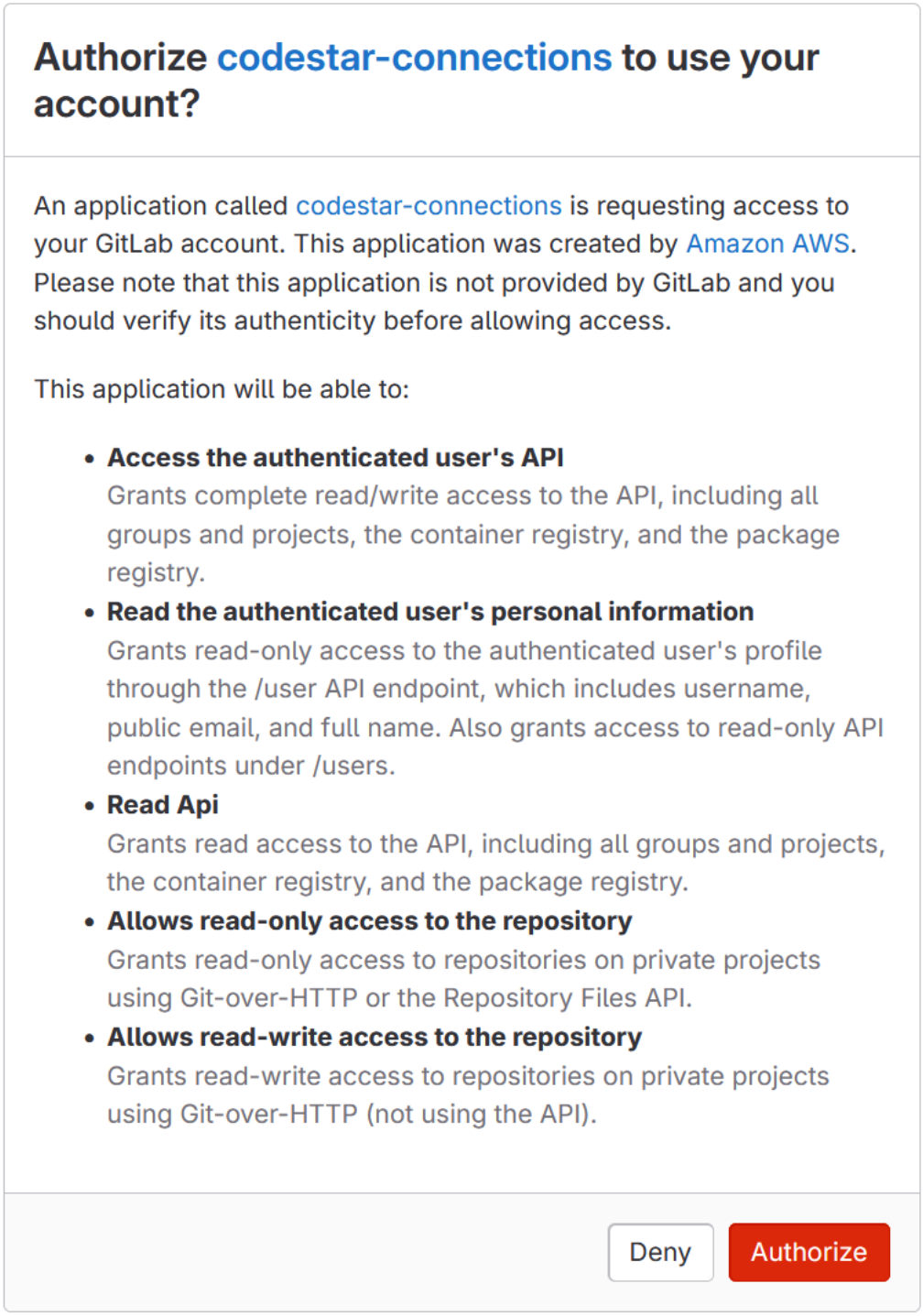
ステップ 1: 接続の作成

1. にログインし AWS Management Console、 AWS で開発者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。 [Connect] を選択します GitLab。



4. GitLab のサイン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、[Sign in] を選択します。
5. 認証ページが開き、アカウントにアクセスするための接続の承認を求めるメッセージが表示されます。 GitLab

   [承認] を選択します。



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] の [接続名] に新しい接続が表示されます。
7. [Connect] を選択します GitLab。

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。

GitLab(CLI) への接続を作成

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 AWS CloudFormation またはを介して作成された接続は、PENDINGデフォルトではステータスになっています。CLI またはを使用して接続を作成したらAWS CloudFormation、コンソールを使用して接続を編集し、ステータスを作成します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 create-connectionを使用してコマンドを実行し、--provider-type--connection-name接続用のとを指定します。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 codestar-connections 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:codestar-connections: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。

GitLabセルフマネージド接続を作成します。

自己管理型インストールでは、 GitLab GitLab エンタープライズエディションまたはコミュニティエディション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 自己管理用の接続とホストを作成できます。

> **Note**
>
> GitLab この接続アプリケーションを自己管理型で承認すると、データを処理する権限をサービスに付与したことになり、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権限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、以下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:

- 自己管理型インストールの GitLab Enterprise Edition または GitLab Community Edition GitLab でアカウントを作成し、所有し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/ 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**
>
> オーナーロールが割り当てられているリポジトリへの接続を作成すると GitLab、CodePipelineその接続をなどのリソースで使用できます。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。

- api GitLab というスコープダウン権限のみを持つ個人アクセストークン (PAT) をすでに作成している必要があります。詳細については、https://docs.gitlab.com/ee/user/profile/personal_access_tokens.html を参照してください。管理者が使用する PAT のみを使用できます。

> **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 (コンソール) への接続を作成します。
- GitLabセルフマネージド (CLI) への接続を作成

GitLab セルフマネージド (コンソール) への接続を作成します。

以下の手順に従って、 GitLab コンソールでホストとセルフマネージド接続を作成します。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**
>
> 1 GitLab つの自己管理型インストール用のホストを作成すると、そのホストへの 1 GitLab つ以上の自己管理接続を管理できます。

ステップ 1: ホストを作成する

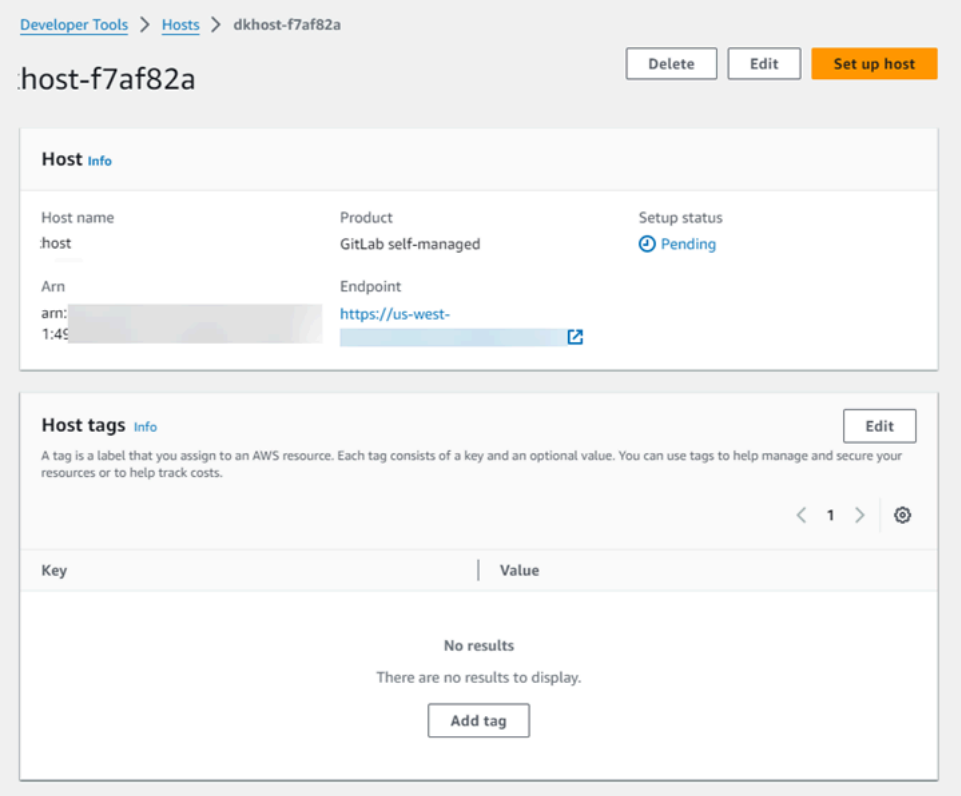
1. にサインインし AWS Management Console、 AWS で開発者ツールコンソールを開きます。https://console.aws.amazon.com/codesuite/settings/connections
2. [Hosts (ホスト)]タブで、[Create host (ホストの作成)]を選択します。
3. [ホスト名]に、ホストに使用する名前を入力します。
4. [プロバイダーの選択] で [GitLabセルフマネージド] を選択します。
5. [URL] に、プロバイダーがインストールされているインフラストラクチャのエンドポイントを入力します。
6. サーバーが 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。

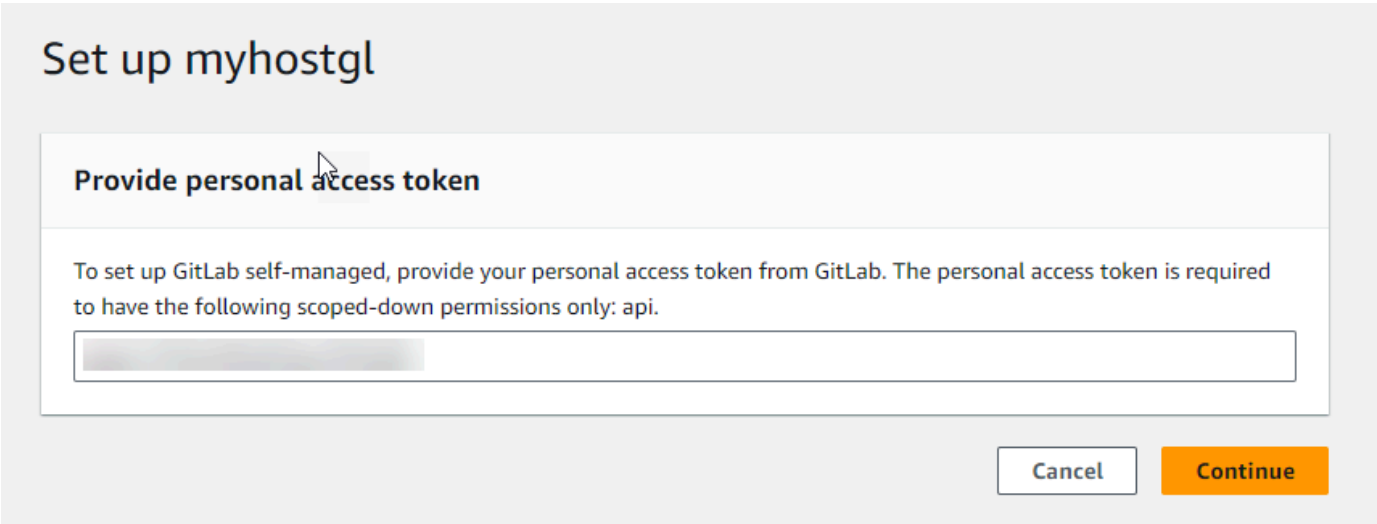
> **Note**
>
> ホスト設定に VPC 設定が含まれている場合は、ホストネットワークコンポーネントのプロビジョニングに数分間かかります。

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。

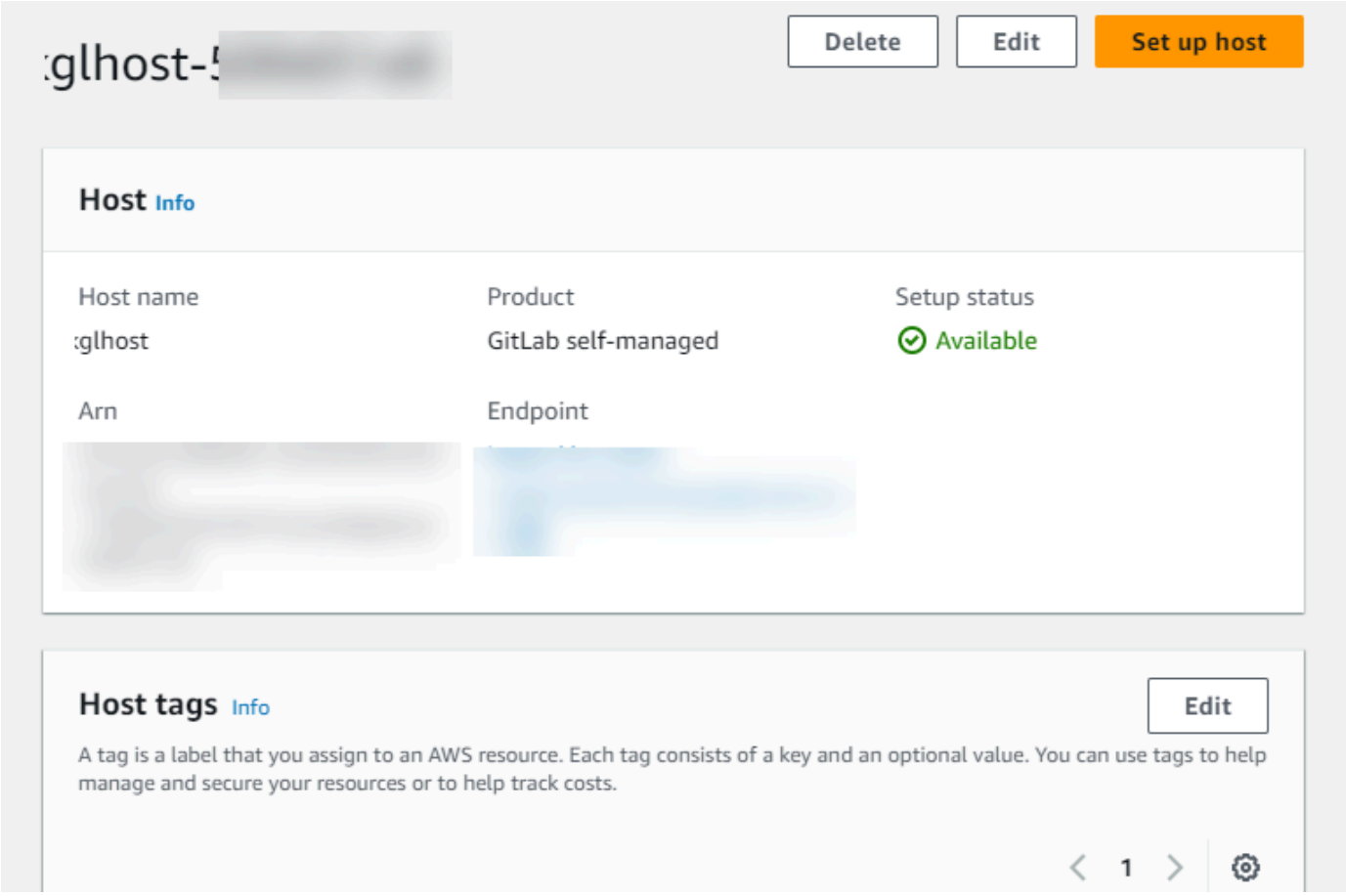


ステップ 2: 保留中のホストを設定する

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権限のみを指定します。

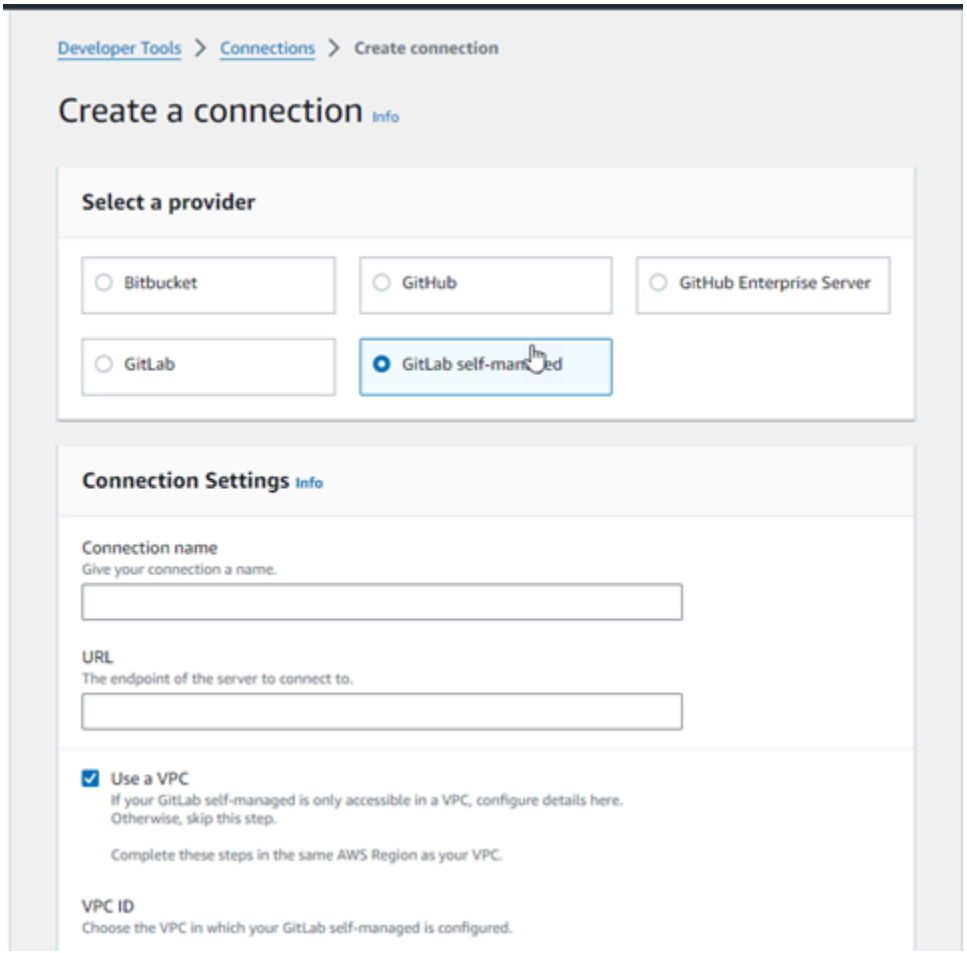


3. ホストが正常に登録されると、ホストの詳細ページが表示され、ホストのステータスが Available (使用可能) になります。



ステップ 3: 接続を作成する

1. にログインし AWS Management Console、 AWS で開発者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、[プロバイダの選択] で [GitLab 自己管理] 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。



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Connect] 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. GitLab のサイン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、[Sign in] を選択します。
9. 認証ページが開き、アカウントにアクセスするための接続の承認を求めるメッセージが表示されます。 GitLab

   [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] の [接続名] に新しい接続が表示されます。
11. [GitLab セルフマネージドにConnect] 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。

GitLabセルフマネージド (CLI) への接続を作成

AWS Command Line Interface (AWS CLI) を使用して、 GitLabセルフマネージド用の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 AWS CloudFormation またはを介して作成された接続は、`PENDING`デフォルトではステータスになっています。CLI またはを使用して接続を作成したら

AWS CloudFormation、コンソールを使用して接続を編集し、ステータスを作成しますAVAILABLE。

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

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 create-hostを使用してコマンドを実行し、接続に合わせて--name--provider-type--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 codestar-connections 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:codestar-connections: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. AWS Management Console にサインインし、の開発者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 (CLI) 用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 create-connectionを使用してコマンドを実行し、--host-arn--connection-name接続用のとを指定します。

```
aws codestar-connections create-connection --host-arn arn:aws:codestar-
connections: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:codestar-connections: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: GitLab コンソールでセルフマネージド接続を完了するには

1. AWS Management Console にログインし、で開発者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) で作成された接続、 AWS CloudFormation PENDINGまたはデフォルトではステータスになっている接続。 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、ステータスを確認します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。

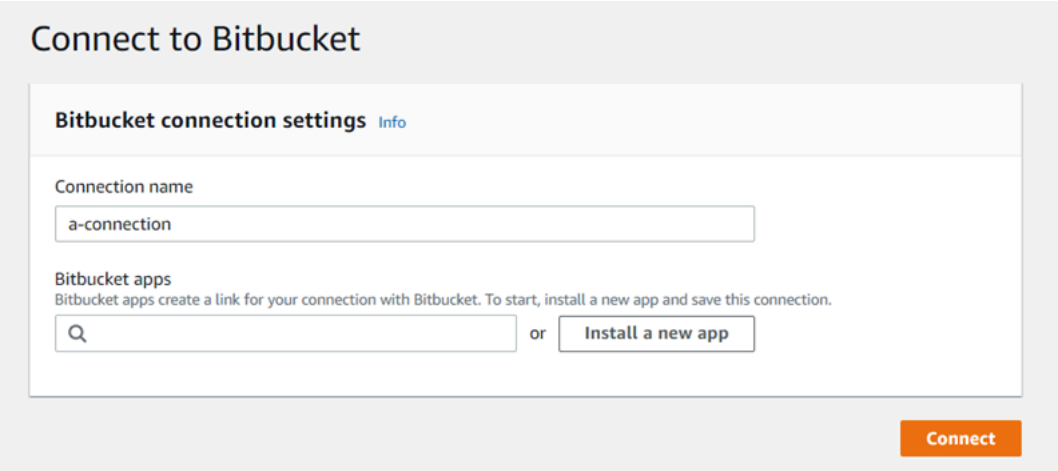
接続を完了するには

1. AWS で開発者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。

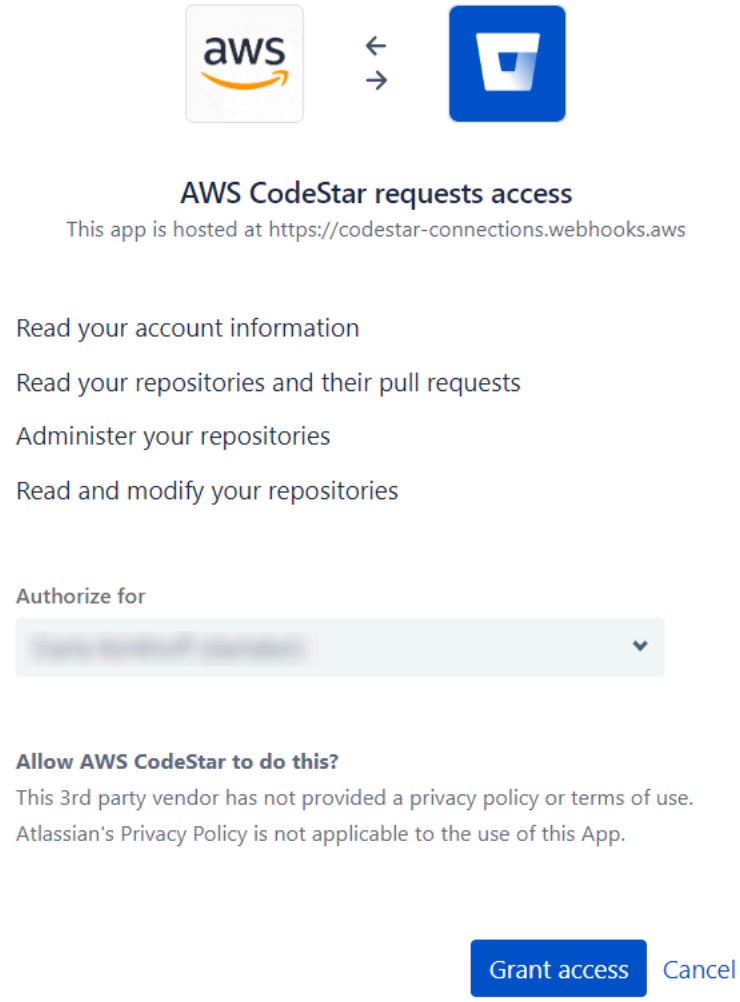
   AWS アカウントに関連付けられているすべての接続の名前が表示されます。
3. [Name (名前)] で、更新する保留中の接続の名前を選択します。

   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。



6. アプリのインストールページに、 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。



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、サードパーティのコードリポジトリへの接続を一覧表示できます。 AWS CLI 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、出力はホストの ARN を追加で返します。

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、を使用してコマンドを実行します。 AWS CLI list-connections

```
aws codestar-connections 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:codestar-connections: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:codestar-connections: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] をクリックします。
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。 AWS CLI delete-connectionを使用してコマンドを実行し、削除する接続の ARN を指定します。

```
aws codestar-connections delete-connection --connection-arn arn:aws:codestar-
connections: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 AWS リソースにユーザーまたは割り当てるカスタム属性ラベルです。 AWS 各タグには次の 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。

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

- 接続
- [ホスト]

これらの手順は、 AWS CLI の最新バージョンが既にインストールされているか、現在のバージョンに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。

タグによるリソースの識別、整理、追跡に加えて、 AWS Identity and Access Management (IAM) ポリシーでタグを使用すると、リソースを閲覧したり操作したり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 CodeStar Connections 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)

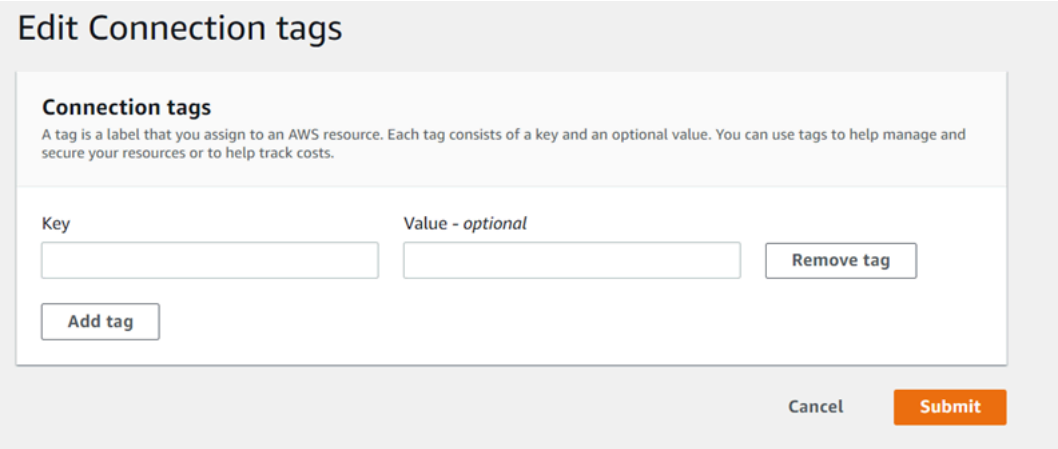
コンソールを使用して、既存の接続またはホストにタグを追加します。

> **Note**
>
> GitHubEnterprise Server などのインストール済みプロバイダーの接続を作成し、ホストリソースも自動的に作成すると、作成中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。

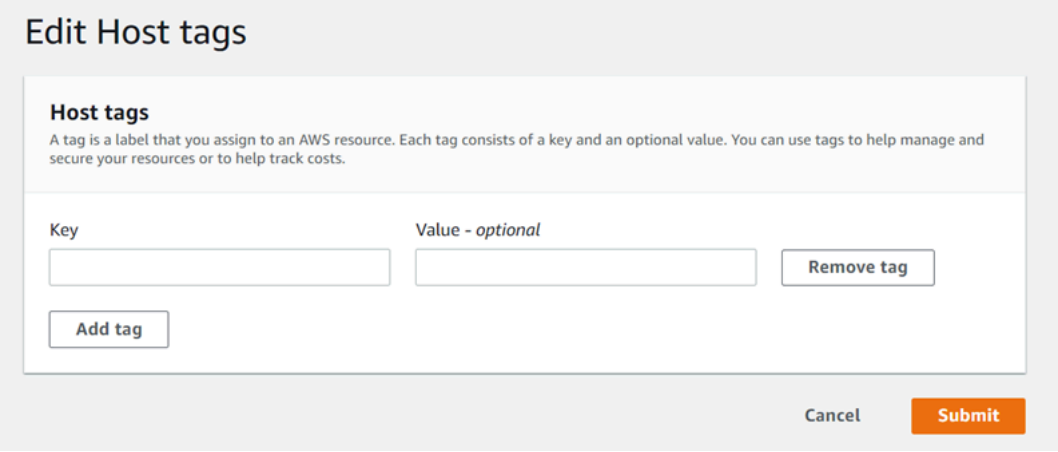
5. [Key] フィールドと [Value] フィールドに、追加するタグのセットごとにキーペアを入力します。([値] フィールドはオプションです。) 例えば、[キー] では、「**Project**」と入力します。[値] には「**ProjectA**」と入力します。



6. (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。
7. [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

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. [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。



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)

AWS CLI を使用して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 リソースネーム (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。

   たとえば、次のコマンドを使用して接続に 2 つのタグをタグ付けします。1 つは *Project* というタグキーに *ProjectA ReadOnly*というタグ値、もう 1 つはタグ値が *true* のタグキーです。

```
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。

   たとえば、次のコマンドを使用して、ホストに 2 つのタグをタグ付けします。1 つは *Project* というタグキーに *ProjectA IscontainerBased*というタグ値、もう 1 つはタグ値が *true* のタグキーです。

   ```
   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

AWS を使用して接続リソースのタグを表示できます。 AWS CLI 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

を使用してリソースのタグを編集できます。 AWS CLI 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

AWS CLI を使用してリソースからタグを削除する手順は、次のとおりです。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。たとえば、*Project* and というタグキーを使用して接続上の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、 AWS CLI の最新バージョンがインストールされているか、現在のバージョンに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。
   - GitHubEnterprise Server などのインストール済みプロバイダに対して接続が作成された場合、その接続に関連するホスト情報。
   - GitHubEnterprise Server などのインストール済みプロバイダに対して接続が作成された場合は、その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 codestar-connections get-connection --connection-arn arn:aws:codestar-connections: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:codestar-connections: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:codestar-connections:us-
west-2:account_id:connection/ebcd4a13-EXAMPLE",
        "ProviderType": "GitHub",
        "OwnerAccountId": "account_id",
        "ConnectionStatus": "AVAILABLE"
    }
}
```

GitHub エンタープライズサーバー接続の出力例:

```
{
    "Connection": {
        "ConnectionName": "MyConnection",
        "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:account_id:connection/2d178fb9-EXAMPLE",
        "ProviderType": "GitHubEnterpriseServer",
        "OwnerAccountId": "account_id",
        "ConnectionStatus": "PENDING",
        "HostArn": "arn:aws:codestar-connections: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、GitHub Enterprise Server を Amazon EC2 インスタンス上で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を作成し、GitHub Enterprise Server への接続を作成するには、「GitHubエンタープライズサーバー接続 (コンソール) を作成します。」を参照してください。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を作成し、GitHub セルフマネージドへの接続を作成するには、「GitLabセルフマネージド接続を作成します。」を参照してください。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 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 GitHub Enterprise Server インスタンスを設定している場合は、ホストリソースに TLS 証明書を提供する必要があります。
- AWS CodeStar Connections でホストを作成すると、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 CodeStar Connections とインターフェイス 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
  - ホストの作成時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 (ホスト設定の一部ではない) にアタッチされたセキュリティグループには、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アクセスとアウトバウンド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、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: 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、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の ID。
- 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、AWS CodeStar Connections はユーザーの代わりにネットワークリソースを作成して VPC 接続を容易にします。これには、AWS Connections でホストのデータをク

エリするためのネットワークインターフェイスと、ホストから AWS CodeStar Connections にウェブフック経由でイベントデータを送信するための VPC エンドポイントまたは PrivateLink 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 CodeStar Connections とインターフェイス 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、インストールの接続では、ホストを使用して、サードパーティーの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。

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、「 GitLabセルフマネージド接続を作成します。」を参照してください。

コンソールを使用してホストを作成し、GitHub Enterprise Server への接続を作成するには、「 GitHubエンタープライズサーバー接続 (コンソール) を作成します。」を参照してください。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を作成し、GitHub セルフマネージドへの接続を作成するには、「GitLabセルフマネージド接続を作成します。」を参照してください。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 回だけ作成します。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接続はすべて、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します。

> **Note**
>
> ホストは、GitHub Enterprise Server アカウントごとに 1 回だけ作成しまう。特定の GitHub Enterprise Server アカウントへの接続はすべて、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**
>
> AWS CLI で作成されたホストは、デフォルトで `Pending` のステータスになっています。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を作成し、GitHub Enterprise Server への接続を作成するには、「GitHubエンタープライズサーバー接続 (コンソール) を作成します。」を参照してください。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を作成し、GitHub セルフマネージドへの接続を作成するには、「GitLabセルフマネージド接続を作成します。」を参照してください。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

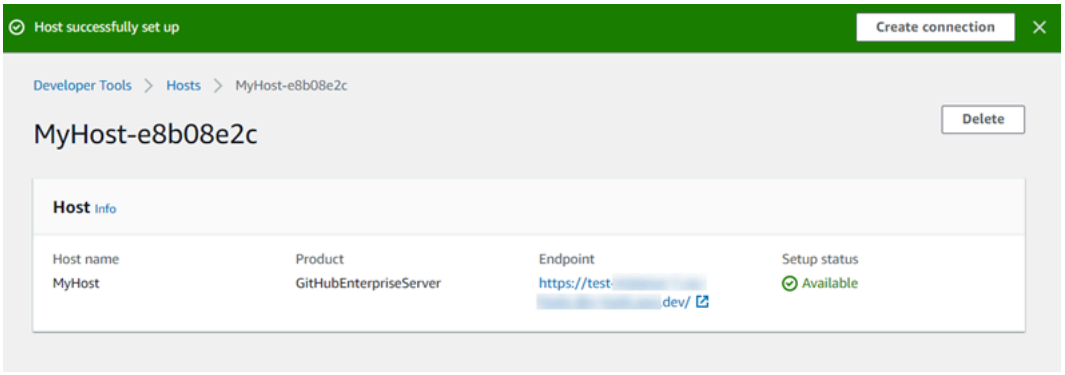
AWS Command Line Interface (AWS CLI) または SDK を使用して作成された接続のステータスは、デフォルトで `Pending` になります。AWS CLI または SDK を使用して接続を作成した後、 コンソールで接続を編集し、ステータスを `Available` に変更します。

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

デベロッパーツールコンソール 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. AWS デベロッパーツールコンソール (https://console.aws.amazon.com/codesuite/settings/connections) 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)

AWS CLI を使用して、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、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 codestar-connections list-hosts
```

このコマンドは、次の出力を返します。

```
{
    "Hosts": [
        {
            "Name": "My-Host",
            "HostArn": "arn:aws:codestar-connections: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. AWS デベロッパーツールコンソール (https://console.aws.amazon.com/codesuite/settings/connections) 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**
>
> このアクションは元に戻すことができません。

ホストの削除 (CLI)

AWS Command Line Interface (AWS CLI) を使用してホストを削除します。

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。AWS CLI を使用して delete-host コマンドを実行し、削除するホストの Amazon リソース名 (ARN) を指定します。

```
aws codestar-connections delete-host --host-arn "arn:aws:codestar-connections: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。

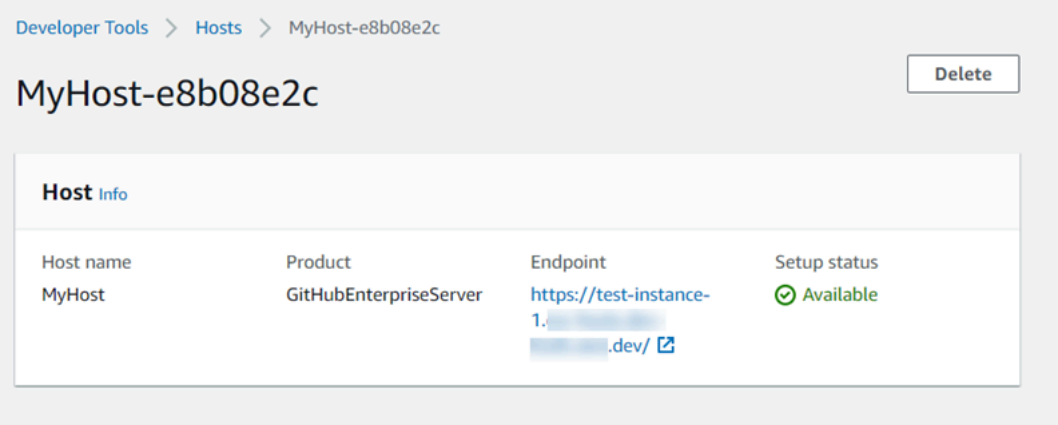
ホストの詳細を表示するには (コンソール)

1. 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。
2. [設定] > [接続] を選択して、次に [ホスト] タブを選択します。
3. 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。
4. 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、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 codestar-connections get-host --host-arn arn:aws:codestar-connections:us-
west-2:account_id:host/My-Host-28aef605
```

このコマンドは、次の出力を返します。

```
{
    "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

AWS CodeStar コネクションでは、接続を使用して Bitbucket Cloud、 GitHub Enterprise Server GitHub、 AWS などのサードパーティのリポジトリにリソースを関連付けます。GitLabCFN_STACK_SYNC同期タイプを使用すると、Git AWS AWS リポジトリのコンテンツを同期して指定されたリソースを更新できる同期設定を作成できます。 AWS CloudFormation は接続と統合されるため、Git sync を使用して、同期先のリンク先リポジトリにあるテンプレートファイル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 AWS CloudFormation またはコンソールを使用してリポジトリリンクを作成し、設定を同期できます。

- 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、 AWS 指定したリソースを更新します。

詳細については、「AWS CodeStar 接続 API リファレンス」を参照してください。

AWS CloudFormation AWS CloudFormation コンソールを使用してスタックの同期設定を作成する手順を説明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 sync の操作」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。リポジトリリンクにより、Git sync は指定された Git リポジトリ内のファイルへの変更を監視し、 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 CodeStar Connections API リファレンスをご覧ください。

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) create-repository-link のコマンドを使用して、接続と同期先の外部リポジトリとの間にリンクを作成できます。

リポジトリリンクを作成するには、外部リポジトリをサードパーティプロバイダ (など) で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 create-repository-linkを使用して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 codestar-connections create-repository-link --connection-arn  arn:aws:codestar-
connections: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:codestar-connections:us-
east-1:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "OwnerId": "account_id",
        "ProviderType": "GitHub",
        "RepositoryLinkArn": "arn:aws:codestar-connections: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) update-repository-link のコマンドを使用して、指定し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。 AWS CLI update-repository-linkを使用してコマンドを実行し、リポジトリリンクの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-adef-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-adef-4ceaefcb2167",
        "OwnerId": "owner_id",
        "ProviderType": "GitHub",
        "RepositoryLinkArn": "arn:aws:codestar-connections:us-
east-1:account_id:repository-link/6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryLinkId": "6053346f-8a33-4edb-9397-10394b695173",
        "RepositoryName": "MyRepo",
        "Tags": []
    }
}
```

リポジトリリンクを一覧表示する

AWS Command Line Interface (AWS CLI) list-repository-links 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 list-repository-links を使用してコマンドを実行します。

```
aws codestar-connections 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) delete-repository-link の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。 AWS CLI delete-repository-linkを使用してコマンドを実行し、削除するリポジトリリンクの ID を指定します。

  ```
  aws codestar-connections 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) get-repository-link 内の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。を使用してリポジトリリンク ID get-repository-link を指定してコマンドを実行します。 AWS CLI

```
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。

接続について詳しくは、AWS CodeStar Connections API リファレンスをご覧ください。

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) create-repository-link 内のコマンドを使用して、接続と同期先の外部リポジトリとの間に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 create-repository-linkを使用して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 codestar-connections 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。 AWS CLI update-sync-configurationを使用してコマンドを実行し、更新する値、リソース名、同期タイプを指定します。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 codestar-connections 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 list-sync-configurationsを使用してコマンドを実行し、同期タイプとリポジトリリンク ID を指定します。

```
aws codestar-connections 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 delete-sync-configurationを使用してコマンドを実行し、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 codestar-connections 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) get-sync-configuration 内の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

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。を使用してリポジトリリンク ID get-sync-configuration を指定してコマンドを実行します。 AWS CLI

```
aws codestar-connections get-sync-configuration --sync-type CFN_STACK_SYNC --
resource-name mystack
```

2. 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"
    }
}
```

AWS CodeConnections による AWS CloudTrail API コールのログ記録

AWS CodeConnections は AWS CloudTrail と統合されています。これは、ユーザーやロール、または AWS サービスによって実行されたアクションをレコードするサービスです。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、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。

証跡を作成すると、通知のイベントを含め、CloudTrail イベントを Amazon Simple Storage Service (Amazon S3) バケットに継続的に配信できるようになります。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 が収集した情報を使用して、AWS CodeConnections に対して行われたリクエスト、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、および追加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 CodeConnectionsCloudTrail での 情報

CloudTrail は、アカウントを作成すると AWS アカウントで有効になります。AWS CodeConnections 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)] で AWS のその他のサービスのイベントと共に 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、「AWS CloudTrail ユーザーガイド」の「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 のイベントなど、AWS CodeConnections アカウントのイベントの継続的なレコードについては、追跡を作成します。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 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 がサポートされているサービスと統合
- CloudTrail の Amazon SNS 通知の設定
- CloudTrail ログ ファイルを複数のリージョンから受け取る
- 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 CloudTrail によってログに記録され、「AWS CodeConnections API リファレンス」に記録されます。例えば、`CreateConnection`、`DeleteConnection`、`GetConnection`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。同一性情報は次の判断に役立ちます。

- 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

[トレイル] は、指定した Simple Storage Service (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、任意の出典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。

次の例は、`CreateConnection`アクションを示す CloudTrail ログエントリです。

```
{
```

```
    "eventVersion": "1.05",
    "userIdentity": {
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/Mary_Major",
        "accountId":"123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName":"Mary_Major"
    },
    "eventTime": "2020-04-21T01:09:48Z",
    "eventSource": "codestar-connections.amazonaws.com",
    "eventName": "CreateConnection",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "IP",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/80.0.3987.163 Safari/537.36",
    "requestParameters": {
        "providerType": "Bitbucket",
        "connectionName": "my-connection"
    },
    "responseElements": {
        "connectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/7EXAMPLE-5da1-4867-960c-4918175ea3ce"
    },
    "requestID": "ac1fbc15-a84f-4568-9f90-f05f1a57749c",
    "eventID": "7548f5b0-7ecf-430f-84bf-72e364644359",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


AWS CodeStar Connections とインターフェイス VPC エンドポイント (AWS PrivateLink)

VPC と AWS CodeStar Connections とのプライベート接続を確立するには、インターフェイス VPC エンドポイントを作成します。インターフェイスエンドポイントは、AWS PrivateLink を利用しており、インターネットゲートウェイ、NAT デバイス、VPN 接続、AWS Direct Connect 接続のいずれを使用せずに、AWS CodeStar Connections API にプライベートアクセスを可能にする技術です。VPC 内のインスタンスは、AWS CodeStar Connections API と通信するためにパブリック IP アドレスを必要としません。これは、VPC と AWS CodeStar Connections 間のトラフィックは Amazon ネットワークを離れることがないからです。

各インターフェー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 インターフェース VPC エンドポイント (AWS PrivateLink) をご参照ください。

AWS CodeStar Connections VPC エンドポイントに関する考慮事項

AWS CodeStar Connections のインターフェース VPC エンドポイントを設定する前に、「Amazon VPC ユーザーガイド」の「インターフェースエンドポイント」を確認してください。

AWS CodeStar Connections 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ています。

VPC エンドポイントは、すべての AWS CodeStar Connections 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 内のエントリポイント。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 CodeStar Connections アクション用の VPC エンドポイント
- AWS CodeStar Connections ウェブフック用の VPC エンドポイント

AWS PrivateLink

VPC とサービスの間にプライベート接続を提供するテクノロジー。

AWS CodeStar Connections アクション用の VPC エンドポイント

AWS CodeStar Connections サービスの VPC エンドポイントを管理できます。

AWS CodeStar Connections アクション用のインターフェイス VPC エンドポイントの作成

AWS CodeStar Connections サービス用の VPC エンドポイントは、Amazon VPC コンソールまたは AWS Command Line Interface (AWS CLI) で作成できます。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、AWS CodeStar Connections 用のインターフェイス VPC エンドポイントを作成します。AWS CodeStar Connections 用の VPC エンドポイントを作成するときは、AWS のサービス、およびサービス名で、以下を選択します。

- com.amazonaws.***region***.codestar-connections.api: このオプションで、AWS CodeStar Connections 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 CodeStar Connections API、または AWS SDK を使用して、`CreateConnection`、`ListConnections`、`CreateHost` などのオペレーションのために AWS CodeStar Connections とやり取りする場合は、このオプションを選択します。

[Enable DNS name] (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、リージョンのデフォルト DNS 名 (`codestar-connections.us-east-1.amazonaws.com` など) を使用して AWS CodeStar Connections に API リクエストを行うことができます。。

> **Important**
>
> 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は、プライベート DNS がデフォルトで有効になります。

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。

AWS CodeStar Connections アクションの VPC エンドポイントポリシーの作成

VPC エンドポイントに AWS CodeStar Connections へのアクセスをコントロール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 CodeStar Connections アクションの VPC エンドポイントポリシー

AWS CodeStar Connections のエンドポイントポリシーの例を次に示します。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、登録されている AWS CodeStar Connections 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 CodeStar Connections ウェブフック用の VPC エンドポイント

AWS CodeStar Connections 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、Amazon ネットワーク経由で統合された AWS のサービスにウェブフック経由でイベントデータを送信できます。

> **Important**
>
> GitHub Enterprise Server 用にホストをセットアップすると、AWS CodeStar Connections によりウェブフックイベントデータ用の VPC エンドポイントが作成されます。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、最初に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 CodeStar Connections が、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。

AWS CodeStar Connections ホストの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では、AWS CodeStar Connections 処理のためにサードパーティーのリポジトリからのウェブフックが送信されます。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 CodeStar Connections は、パイプラインを開始するように CodePipeline に通知できます。

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 CodeStar Connections 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

次の情報は、AWS CodeBuild、AWS CodeDeploy、および AWS CodePipeline 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つ場合があります。

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と、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と、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
- GitHub Enterprise Server 接続用の ウェブフック VPC エンドポイント (PrivateLink) 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ません
- GitHub Enterprise Server 接続アプリのアクセス許可を編集する
- GitHub への接続時の接続エラー:「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 アプリケーションをインストールする必要があります」
- 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、「の権限と例 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と、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と、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ためのアクセス許可は、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、「の権限と例 AWS CodeConnections」を参照してください。

接続を使用しようとすると、アクセス許可エラーが表示されます

CodePipeline コンソールで接続を使用しようとすると、アクセス許可の一覧表示、取得、および作成のアクセス許可がある場合でも、次のエラーメッセージのいずれか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。)

User: *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ーをリストするなど、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、「の権限と例 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 接続を使用するときに、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、 CodeBuild がダウンロードするコードを含む 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 サービスのロールポリシーにアクセス許可を追加するには、CodeBuild サービスのロールにアタッチするカスタマーマネージド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 の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。[Configuration] (設定) ペインが開き、接続 ARN が [ConnectionArn] の横に表示されます。CodeBuild サービスロールポリシーに接続 ARN を追加します。
2. CodeBuild サービスロールを確認するには、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、[Build details] (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。
5. [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。
6. 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 ホストを設定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 アプリが削除または変更された場合、接続とホストがエラー状態に移行す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。

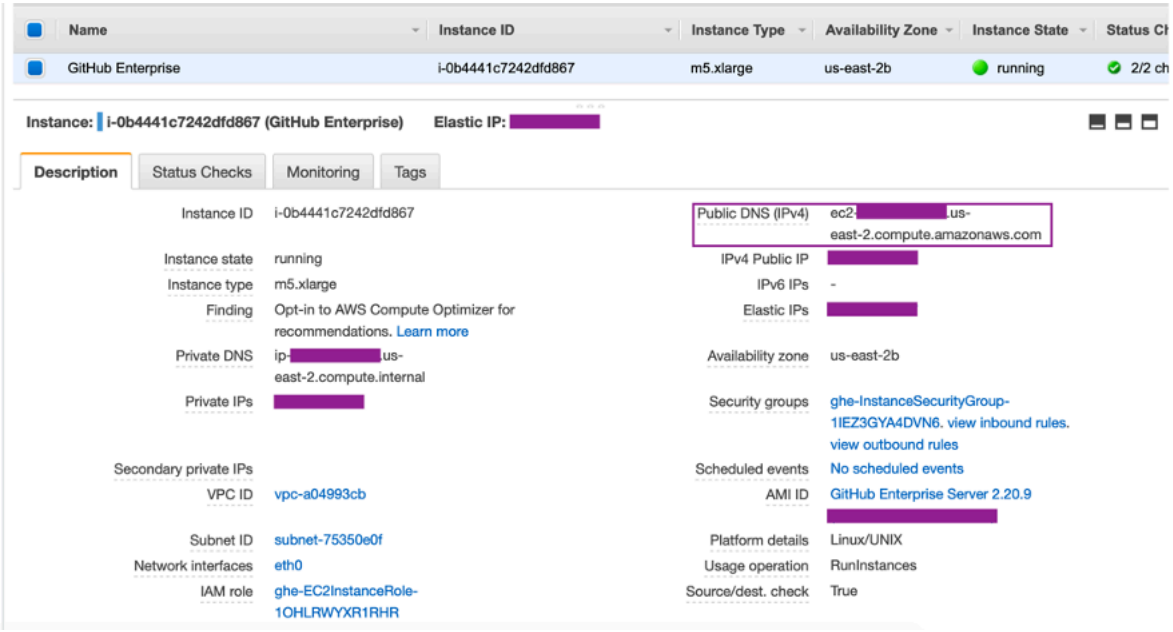
> **Note**
>
> このセクションは、Amazon VPC 内の GitHub Enterprise Server ホスト設定に関連するトラブルシューティングに使用します。VPC (PrivateLink) の ウェブフックエンドポイントを使用するように設定されている接続に関連するトラブルシューティングについては、「GitHub Enterprise Server 接続用の ウェブフック VPC エンドポイント (PrivateLink) のトラブルシューティング」を参照してください。

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#AddaSubnet」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」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。



GitHub Enterprise Server 接続に VPC を使用する場合、ホストをセットアップするときにインフラストラクチャに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。
- サブネット 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。
- 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

ホストが VPC_CONFIG_FAILEED_INTIONALITINATION 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:DescribeDhcpOptionsec2: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 AWS CodeStar Connections アプリケーションのセットアップ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。

- パブリック認証局を使用していない場合は、GitHub Enterprise 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 の設定が変更されたか、GitHub アプリが変更されたこと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たか、GitHub Enterprise Server にアクセスできなくなりました。AWSCodeStar Connections には、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が必要です。
- 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は、ホスト/接続が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 ACL 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 接続は許可されなくなりました。
- GitHub Enterprise Server 上の AWS CodeStar Connections アプリケーションの設定が変更されました。URL 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 AWS CodeStar Connections 間の接続が切断される場合があります。

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。

- AWS CodeStar Connections が作成し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
```

GitHub Enterprise Server 接続用の ウェブフック VPC エンドポイント (PrivateLink) 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 (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ている接続に関連するトラブルシューティングに使用します。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、「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。

インストールされたプロバイダーのタイプへの接続を作成し、サーバーが VPC 内に設定されるように指定すると、AWS CodeStar Connections がホストを作成し、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された AWS の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 CodeStar Connections とインターフェイス 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 Codestar Connections が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。

- AWS CodeStar Connections 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 日より、AWS CodeStar Connections がホストを設定する際、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 CodeStar Connections とインターフェイス 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 以前に作成したホストがあり、ウェブフックに VPC エンドポイント（PrivateLink）を使用したいと考えています
- 利用可能な状態（VPC エラー）のホストを取得できません

2020 年 11 月 24 日 以前に作成したホストがあり、ウェブフックに VPC エンドポイント（PrivateLink）を使用したいと考えています

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が使用され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、最初に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

ホストを 2020 年 11 月 24 日より前に作成して、ホストの AWS CodeStar Connections アプリケーションのセットアップ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。

GitHub Enterprise Server インスタンスが GitHub Webhook の出力ネットワークトラフィックを送信できるようにするために、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ません

問題:

GitHub リポジトリへの接続は AWS Connector for GitHub を使用するため、接続を作成するには、リポジトリへの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の詳細については、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。

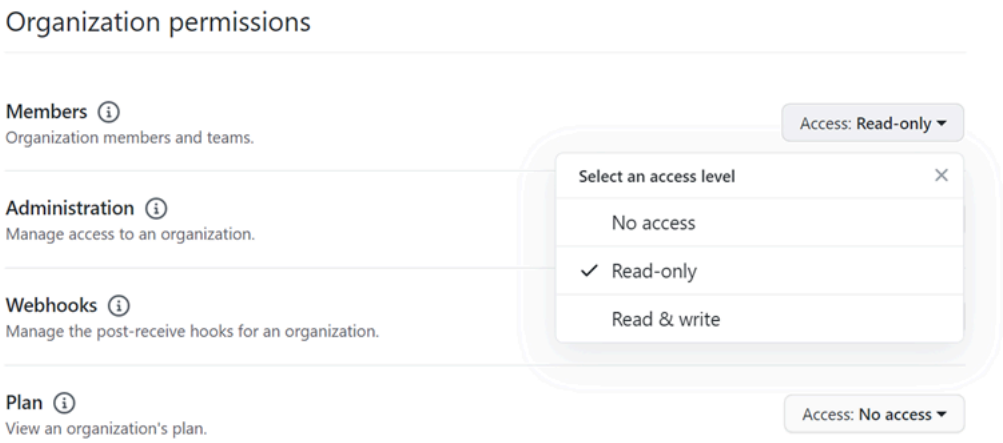
GitHub Enterprise Server 接続アプリのアクセス許可を編集する

2020 年 12 月 23 日以前に GitHub Enterprise Server 用のアプリをインストールした場合、組織のメンバーはアプリ読み取り専用のアクセス許可が必要な場合があります。GitHub アプリの所有者である場合は、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手順を完了し、GitHub アプリケーションの所有者になる必要があります。

1. GitHub Enterprise Serve で、プロフィール写真のドロップダウンオプションから、[Settings] (設定) を選択します。
2. [Developer settings] (開発者設定) を選択してから、GitHub Apps (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。



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。[Save changes] (変更の保存) をクリックします。

GitHub への接続時の接続エラー:「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 アプリケーション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 アプリケーションをインストールする必要があります

解決策: GitHub 組織のリポジトリである場合、組織の所有者が 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。

接続の制限を引き上げることはできますか

AWS CodeStar Connections では、制限の引き上げをリクエストできます。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

リソース	デフォルトの制限
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、や Bitbucket GitHub などのプロバイダ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-southeast-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-southeast-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

リージョン	IP アドレス
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
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

デベロッパーツールコンソールの機能のセキュリティ

AWS クラウドセキュリティは最優先事項です。 AWS 顧客は、最もセキュリティに敏感な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の恩恵を受けることができます。

セキュリティは、 AWS お客様とお客様と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 — AWS AWS AWS クラウド内で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があります。 AWS また、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 Notification and AWS CodeStar Connections に適用されるコンプライアンスプログラムについては、「AWS コンプライアンスプログラム別の対象サービス」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、 AWS 使用する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 AWS CodeStar 通知と接続を使用する際に責任分担モデルを適用する方法を理解するのに役立ちます。以下のトピックでは、 AWS CodeStar AWS CodeStar セキュリティとコンプライアンスの目標を満たすように通知と接続を構成する方法を示しています。また、 AWS AWS CodeStar AWS CodeStar 通知と接続リソースの監視と保護に役立つ他のサービスの使い方についても学びます。

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。

たとえば、 CodeCommit リポジトリの通知ルールを設定して、コミットやプルリクエストにコメントを追加することができます。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、 CodeBuild ビルドの状態やフェーズの成功または失敗を含むようにビルドプロジェクトの通知ルールを設定できます。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。

CodePipeline パイプラインの通知ルールを設定して、手動承認に関する情報を含めることができます。また、そのルールに応じて送信される通知には、承認を行った担当者の名前が含まれる場合があります。 CodeDeploy デプロイの成功を示すようにアプリケーションの通知ルールを設定できます。そのルールに応じ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、イベントに応答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 や Amazon EventBridge CloudWatch Events に送信されるものとまったく同じ情報が含まれています。

CodeCommit開発者ツールなどのコンソールサービスでは、イベントで利用できる情報以外に、イベントタイプの一部またはすべてに関する情報を通知メッセージに追加する場合があります。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 Notifications と AWS CodeStar Connections でのデータ保護

AWS 責任共有モデルは、AWS CodeStar Notifications と AWS CodeStar Connections のデータ保護に適用されます。このモデルで説明されているように、AWS には、AWS クラウド のすべて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。ユーザーには、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にも責任があります。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AWS 責任共有モデルおよび GDPRのブログ記事を参照してください。

データを保護するため、AWS アカウント の認証情報を保護し、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セットアップすることをお勧めします。こうすると、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、次の方法でデータを保護することをお勧めします。

- 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 が必須です。TLS 1.3 が推奨されます。
- AWS CloudTrail で API とユーザーアクティビティログをセットアップします。
- AWS のサービス内でデフォルトである、すべてのセキュリティ管理に加え、AWS の暗号化ソリューションを使用します。
- 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 を使用して AWS 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。

顧客の E メールアドレスなどの機密情報や重要情報は、タグや Name フィールドなどの自由形式のフィールドに入力しないことを強くお勧めします。これは、コンソール、API、AWS CLI、または AWS SDK を使用して、AWS CodeStar Notifications と AWS CodeStar Connections、またはその他の AWS のサービス を操作する場合も同様です。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 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 AWS CodeStar 通知と接続の ID とアクセス管理

AWS Identity and Access Management (IAM) は、 AWS のサービス AWS 管理者がリソースへのアクセスを安全に制御できるようにするものです。IAM 管理者は、 AWS CodeStar Notifications and Connections リソースを使用するユーザーを認証 (サインイン) および許可 (権限の付与) できるユーザーを制御します。 AWS CodeStar IAM AWS のサービス は追加料金なしで使用できるアプリです。

トピック

対象者

AWS Identity and Access Management (IAM) の使用方法は、 AWS CodeStar AWS CodeStar 通知と接続で行う作業によって異なります。

サービスユーザー — AWS CodeStar Notifications and AWS CodeStar Connections サービスを使用して業務を行う場合、管理者は必要な認証情報と権限を提供します。 AWS CodeStar AWS CodeStar 業務に使用する通知と接続の機能が増えるにつれて、追加の権限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と、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 AWS CodeStar 通知と接続の機能にアクセスできない場合は、を参照してくださいAWS CodeStar AWS CodeStar 通知と接続のトラブルシューティング:ID とアクセス。

サービス管理者 — AWS CodeStar AWS CodeStar 会社で通知と接続のリソースを担当している場合は、 AWS CodeStar AWS CodeStar 通知と接続にすべてアクセスできるはずです。 AWS CodeStar AWS CodeStar サービスユーザーがどの通知と接続機能やリソースにアクセスすべきかを決めるのはあなたの仕事です。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。会社が IAM AWS CodeStar AWS CodeStar を通知と接続でどのように使用できるかについての詳細は、「」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。

IAM 管理者 — IAM 管理者の方は、 AWS CodeStar 通知と接続へのアクセスを管理するポリシーを記述する方法の詳細を知りたいと思うかもしれません。 AWS CodeStar IAM AWS CodeStar AWS CodeStar で使用できる通知と接続の ID ベースのポリシーの例については、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 AWS 認証情報を使用してサインインする方法です。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 (サインイン AWS) する必要があります。

ID ソースを通じて提供された認証情報を使用して、フェデレーション ID AWS としてサインインできます。 AWS IAM Identity Center フェデレーテッド ID の例としては、(IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google や Facebook の認証情報などがあります。フェデレーティ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。 AWS フェデレーションを使用してアクセスすると、間接的にロールを引き継ぐことになります。

ユーザーのタイプによっては、 AWS Management Console AWS またはアクセスポータルにサインインできます。へのサインインについて詳しくは AWS、『AWS サインイン ユーザーガイド』の「AWS アカウントにサインインする方法」を参照してください。

AWS プログラムでアクセスする場合は、認証情報を使用してリクエストに暗号署名するためのソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供します。 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。推奨方法を使用して自分でリクエストに署名する方法の詳細については、IAM ユーザーガイドの「 AWS API リクエストへの署名」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。たとえば、アカウントのセキュリティを強化するために多要素認証 (MFA) AWS を使用することを推奨しています。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、 AWS のサービス アカウントのすべてのリソースに完全にアクセスできる 1 つのサインイン ID から始めます。この ID は AWS アカウント root ユーザーと呼ばれ、アカウントの作成に使用した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されます。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、1 人のユーザーまたはアプリケーションに対して特定の権限を持つ社内の AWS アカウント ID 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、 AWS アカウント 特定の権限を持つ社内の ID 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。 AWS Management Console ロールを切り替えることで、の IAM ロールを一時的に引き受けることができます。 AWS CLI または AWS API オペレーションを呼び出すか、カスタム URL を使用してロールを引き受けることができます。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 – フェデレーティ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。フェデレーティ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、ロールをプロキシとして使用する代わりに AWS のサービス、ポリシーをリソースに直接アタッチできるものもあります。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 — AWS のサービス AWS のサービス他の機能を使用するものもあります。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アクションを実行する場合 AWS、あなた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、 AWS のサービス を呼び出したプリンシパルの権限をリクエスト元と組み合わせて使用して AWS のサービス、ダウンストリームサービスにリクエストを行います。FASリクエストは、AWS のサービス サービスが他のユーザーとのやりとりやリソースとのやり取りを必要とするリクエストを受信したときにのみ行われます。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 のサービスに権限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、にリンクされている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。 AWS アカウント サービスにリンクされたロールはに表示され、そのサービスが所有します。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 API AWS CLI 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。EC2 AWS インスタンスにロールを割り当て、そのロールを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る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

AWS ポリシーを作成して AWS ID またはリソースにアタッチすることで、アクセスを制御します。ポリシーとは、ID またはリソースに関連付けると権限を定義するオブジェクトです。 AWS AWS 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) がリクエストを行うと、これらのポリシーを評価します。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 AWS ドキュメントとして保存されます。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、および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 管理ポリシーと顧客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。 AWS 通知やその他のサービスが IAM とどのように連携するかを大まかに把握するには、IAM ユーザーガイドの「IAM AWS と連携するサービス」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 AWS CodeStar 通知と接続-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 AWS CodeStar 通知と接続は、特定のアクション、リソース、条件キーをサポートします。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には通常、関連する AWS API オペレーションと同じ名前が付けられます。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 codestar-connections` を使用します。例えば、アカウン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` OR `NotAction` 要素を含める必要があります。 AWS CodeStar AWS CodeStar 通知と接続では、このサービスで実行できるタスクを説明する独自のアクションセットが定義されています。

1 つのステートメントで複数の AWS CodeStar Notifications アクションを指定するには、以下のようにコンマで区切ります。

```
"Action": [
      "codestar-notifications:action1",
      "codestar-notifications:action2"
```

1 AWS CodeConnections つのステートメントに複数のアクションを指定するには、次のようにコンマで区切ります。

```
"Action": [
      "codestar-connections:action1",
      "codestar-connections: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 通知 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

AWS CodeConnections 認証ハンドシェイクを完了するには、以下の権限のみのアクションが必要です。

- GetIndividualAccessToken
- GetInstallationUrl
- ListInstallationTargets
- StartOAuthHandshake
- UpdateConnectionInstallation

接続を使用するには、以下の権限のみのアクションが必要です。 AWS CodeConnections

- UseConnection

接続をサービスに渡すには、以下の権限のみのアクションが必要です。 AWS CodeConnections

- PassConnection

AWS CodeStar AWS CodeStar 通知と接続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 通知によって定義されるアクション」と「AWS CodeStar 接続によって定義されるアクション」を参照してください。

リソース

AWS CodeStar AWS CodeStar 通知と接続では、ポリシーでのリソース ARN の指定はサポートされていません。

条件キー

AWS CodeStar AWS CodeStar 通知と接続は独自の条件キーセットを定義し、一部のグローバル条件キーの使用もサポートしています。 AWS すべての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。

AWS CodeStar codestar-notifications:NotificationsForResourceすべての通知アクションは条件キーをサポートしています。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 IAM Condition ポリシーの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 権限リファレンス」を参照してください。

条件キー	説明
codestar-connections: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
codestar-connections:FullRepositoryId	リクエストで渡されたリ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
codestar-connections: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。
codestar-connections: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
codestar-connections: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
codestar-connections: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。
codestar-connections: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

条件キー	説明
codestar-connections: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。
codestar-connections: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。
codestar-connections: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 AWS CodeStar 通知と接続の ID ベースのポリシーの例については、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 AWS CodeStar 通知と接続-リソースベースのポリシー

AWS CodeStar AWS CodeStar 通知と接続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

AWS CodeStar AWS CodeStar 通知リソースと接続リソースにタグを添付したり、リクエストでタグを渡したりできます。タグに基づいてアクセスを管理するには、codestar-notifications and codestar-connections: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。タグ付け方法について詳しくは、「AWS リソースのタグ付け」を参照してください。 AWS CodeStar AWS CodeStar 通知リソースと接続リソースのタグ付けについて詳しくは、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 CodeStar Connections 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、 AWS 特定の権限を持つ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は、AWS STS AssumeRoleGetFederationTokenやなどの API オペレーションを呼び出して取得します。

AWS CodeStar AWS CodeStar 通知と接続では、一時的な認証情報の使用がサポートされます。

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 サービスは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 Notifications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 Notifications and AWS CodeStar Connections の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 Notifications のサービスにリンクされたロールの使用

AWS CodeStar Connections 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 権限リファレンス

次の表は、各 AWS CodeConnections API オペレーション、アクセス権限を付与できる対応するアクション、およびアクセス権限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 API 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ています。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-wide の条件キーも使用できます。 AWS-wide キーの全リストについては、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、`codestar-connections:` や `codestar-connections:ListConnections`) の前に `codestar-connections: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。たとえば、`codestar-connections:*` AWS CodeConnections すべてのアクションを指定し、`codestar-connections:Get*` AWS CodeConnections その単語で始まるすべてのアクションを指定します。`Get`次の例では、`MyConnection` で始まる名前のすべてのリソースへのアクセスを許可します。

```
arn:aws:codestar-connections: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

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、以下の権限に制限されている必要があります。 AWS CLI

> **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。

```
codestar-connections:CreateConnection
codestar-connections:DeleteConnection
codestar-connections:GetConnection
codestar-connections:ListConnections
```

AWS CodeStar AWS CodeStar 通知と接続:接続を管理するためのアクションに必要な権限

CreateConnection

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

DeleteConnection

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetConnection

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListConnections

アクション:codestar-connections: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:codestar-connections:*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
codestar-connections:CreateConnection	codestar-connections:ProviderType
codestar-connections:DeleteConnection	該当なし
codestar-connections:GetConnection	該当なし
codestar-connections:ListConnections	codestar-connections:ProviderTypeFilter

ホストを管理するためのアクセス許可

または SDK を使用してホストを表示、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、以下の権限に制限されている必要があります。 AWS CLI

> **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。

```
codestar-connections:CreateHost
codestar-connections:DeleteHost
codestar-connections:GetHost
codestar-connections:ListHosts
```

AWS CodeStar AWS CodeStar 通知と接続:ホストを管理するためのアクションに必要な権限

CreateHost

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:host/*host-id*

DeleteHost

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:host/*host-id*

GetHost

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:host/*host-id*

ListHosts

アクション:codestar-connections: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:codestar-connections:*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
codestar-connections:CreateHost	codestar-connections:ProviderType
codestar-connections:DeleteHost	該当なし
codestar-connections:GetHost	該当なし
codestar-connections:ListHosts	codestar-connections: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、UpdateConnec
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。

```
codestar-connections:GetIndividualAccessToken
codestar-connections:GetInstallationUrl
codestar-connections:ListInstallationTargets
codestar-connections:StartOAuthHandshake
codestar-connections: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。

```
codestar-connections:CreateConnection
codestar-connections:DeleteConnection
codestar-connections:GetConnection
codestar-connections:ListConnections
codestar-connections:UseConnection
codestar-connections:ListInstallationTargets
codestar-connections:GetInstallationUrl
codestar-connections:StartOAuthHandshake
codestar-connections:UpdateConnectionInstallation
codestar-connections:GetIndividualAccessToken
```

AWS CodeConnections 接続を完了するためのアクションに必要な権限

GetIndividualAccessToken

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetInstallationUrl

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListInstallationTargets

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

[開始] AuthHandshake

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

UpdateConnectionInstallation

アクション:codestar-connections: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:codestar-connections:*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
codestar-connections:GetIndividualAccessToken	codestar-connections:ProviderType
codestar-connections:GetInstallationUrl	codestar-connections:ProviderType
codestar-connections:ListInstallationTargets	該当なし
codestar-connections:StartOAuthHandshake	codestar-connections:ProviderType
codestar-connections:UpdateConnectionInstallation	codestar-connections: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。

```
codestar-connections:RegisterAppCode
codestar-connections: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ータイプなど) を使用、作成、更新、または削除します。

```
codestar-connections:CreateConnection
codestar-connections:DeleteConnection
codestar-connections:GetConnection
```

```
codestar-connections:ListConnections
codestar-connections:UseConnection
codestar-connections:ListInstallationTargets
codestar-connections:GetInstallationUrl
codestar-connections:StartOAuthHandshake
codestar-connections:UpdateConnectionInstallation
codestar-connections:GetIndividualAccessToken
codestar-connections:RegisterAppCode
codestar-connections:StartAppRegistrationHandshake
```

AWS CodeConnections ホスト設定を完了するためのアクションに必要な権限

RegisterAppCode

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:host/*host-id*

StartAppRegistrationHandshake

アクション:codestar-connections: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:codestar-connections:*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 codestar-connections: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権限

PassConnection

アクション:codestar-connections:PassConnection

サービスに接続を渡すために必要です。

リソース:arn:aws:codestar-connections:*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。

- codestar-connections: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
codestar-connections:PassedToService	• codeguru-reviewer • codepipeline.amazonaws.com • proton.amazonaws.com

接続の使用

CodePipeline のようなサービスが接続を使用する場合、codestar-connections:UseConnectionサービスロールには特定の接続に対する権限が必要です。

コンソールで接続を管理するには、ユーザーポリシーに codestar-connections: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:codestar-connections:UseConnection

接続を使用するために必要です。

リソース:arn:aws:codestar-connections:*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。

- codestar-connections:BranchName

- codestar-connections:FullRepositoryId
- codestar-connections:OwnerId
- codestar-connections:ProviderAction
- codestar-connections:ProviderPermissionsRequired
- codestar-connections: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
codestar-connections: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。
codestar-connections:ProviderPermissionsRequired	read_only または read_write
codestar-connections:ProviderAction	GetBranch , ListRepositories , ListOwners , ListBranches , StartUploadArchiveToS3 , GitPush, GitPull, GetUploadArchiveToS3Status , CreatePullRequestDiffComment , GetPullRequest , ListBranchCommits , ListCommitFiles , ListPullRequestComments , ListPullRequestCommits . 詳細については、次のセクションをご覧ください。

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、codestar-connections: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 接続をサービスが使用すると、ソースコードプロバイダーに対して API 呼び出しが行われます。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、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 対応しているアクセスタイプ **ProviderAction**

GetBranch

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListRepositories

アクション:codestar-connections:ListRepositories

所有者に属する公開および非公開リポジトリーのリストにアクセスするために必要です。

リソース:arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListOwners

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListBranches

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

StartUploadArchiveToS3

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GitPush

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GitPull

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetUploadArchiveToS3 ステータス

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

CreatePullRequestDiffComment

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetPullRequest

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListBranchCommits

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListCommitFiles

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListPullRequestComments

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListPullRequestCommits

アクション:codestar-connections: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:codestar-connections:*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。

```
codestar-connections:ListTagsForResource
codestar-connections:TagResource
codestar-connections: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:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

TagResource

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

UntagResource

アクション:codestar-connections: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:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*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 codestar-connections: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権限

PassRepository

アクション:codestar-connections: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:codestar-connections:*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。

- codestar-connections: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
codestar-connections:PassedToService	• cloudformation.sync.codeconnections.amazonaws.com

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。

- codestar-connections: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
codestar-connections:Branch	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。 • CreateSyncConfiguration • UpdateSyncConfiguration • GetRepositorySyncStatus

アイデンティティベースポリシーの例

デフォルトでは、、、、 AWS CodePipeline のいずれかの管理ポリシーが適用されている 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目的に合った接続、通知、通知ルールに対するアクセス権限が付与されます。たとえば、フルアクセスポリシー (、、、または AWSCodePipeline_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AWSCodeCommitFullAccessAWSCodeBuildAdminAccessAWSCodeDeployFullAccess、それらのサービスのリソース用に作成された通知や通知ルールにもフルアクセスできます。

その他の IAM ユーザーやロールには、 AWS CodeStar AWS CodeStar 通知リソースや接続リソースを作成または変更する権限がありません。また、 AWS Management Console AWS CLI、または AWS API を使用してタスクを実行することもできません。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 通知の権限と例

AWS CodeStar 以下のポリシーステートメントと例は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、AWSCodeBuildAdminAccessAWSCodeDeployFullAccess、AWSCodePipelin
よび管理ポリシーには、開発者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。たとえば、のフルアクセスポリシーでは CodeCommit、値はですarn:aws:codecommit:*。

```
    {
        "Sid": "CodeStarNotificationsReadWriteAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:CreateNotificationRule",
            "codestar-notifications:DescribeNotificationRule",
            "codestar-notifications:UpdateNotificationRule",
            "codestar-notifications:DeleteNotificationRule",
            "codestar-notifications:Subscribe",
            "codestar-notifications:Unsubscribe"
        ],
        "Resource": "*",
        "Condition" : {
            "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:<vendor-code>:*"}
        }
    },
    {
        "Sid": "CodeStarNotificationsListAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:ListTargets",
            "codestar-notifications:ListTagsforResource",
            "codestar-notifications:ListEventTypes"
        ],
        "Resource": "*"
    },
    {
        "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
        "Effect": "Allow",
```

```
        "Action": [
            "sns:CreateTopic",
            "sns:SetTopicAttributes"
        ],
        "Resource": "arn:aws:sns:*:*:codestar-notifications*"
    },
    {
        "Sid": "SNSTopicListAccess",
        "Effect": "Allow",
        "Action": [
            "sns:ListTopics"
        ],
        "Resource": "*"
    },
    {
        "Sid": "CodeStarNotificationsChatbotAccess",
        "Effect": "Allow",
        "Action": [
            "chatbot:DescribeSlackChannelConfigurations",
            "chatbot:ListMicrosoftTeamsChannelConfigurations"
          ],
       "Resource": "*"
    }
```

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

、AWSCodeCommitReadOnlyAccessAWSCodeBuildReadOnlyAccessAWSCodeDeployReadOnlyAccess、A
よび管理ポリシーには、通知への読み取り専用アクセスを許可する次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。たとえば、のフルアクセスポリシーでは CodeCommit、値はです`arn:aws:codecommit:*`。

```
  {
        "Sid": "CodeStarNotificationsPowerUserAccess",
        "Effect": "Allow",
        "Action": [
```

```
            "codestar-notifications:DescribeNotificationRule"
        ],
        "Resource": "*",
        "Condition" : {
            "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:<vendor-code>:*"}
        }
    },
    {
        "Sid": "CodeStarNotificationsListAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-notifications:ListNotificationRules",
            "codestar-notifications:ListEventTypes",
            "codestar-notifications:ListTargets"
        ],
        "Resource": "*"
    }
```

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、AWSCodeBuildDeveloperAccess管理ポリシーには以下のステートメントが含まれており、これらの管理ポリシーのいずれかを適用した開発者が通知を作成、編集、購読できるようになっています。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。たとえば、のフルアクセスポリシーでは CodeCommit、値はです`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:通知を管理するための管理者レベルのポリシー AWS CodeStar

この例では、 AWS アカウント内の IAM ユーザーに通知へのフルアクセス権を付与して、 AWS CodeStar その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したいと考えています。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、、、AWSCodePipeline_FullAccess管理ポリシーの一部として含まれている通知権限と同等で

す。AWSCodeBuildAdminAccessAWSCodeCommitFullAccessAWSCodeDeployFullAccess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と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添付してください。

> **Note**
>
> このポリシーには、許可として CreateNotificationRule が含まれています。このポリシーを自分の IAM ユーザーまたはロールに適用したユーザーは、そのユーザー自身がそれらのリソースにアクセスできない場合でも、 AWS アカウントの Notifications AWS CodeStar がサポートす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。たとえば、このポリシーを持つユーザーは、 CodeCommit CodeCommit自身にアクセスする権限がなくても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:通知を使用するためのコントリビューターレベルのポリシー AWS CodeStar

この例では、 day-to-day AWS CodeStar 通知の作成や購読などの通知の使用にはアクセス権を付与したいが、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はアクセス権を付与したくないと考えている。これは、AWSCodeBuildDeveloperAccessAWSCodeDeployDeveloperAccess、AWSCodeCommitPowerUserおよび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。このポリシーを自分の IAM ユーザーまたはロールに適用したユーザーは、そのユーザー自身がそれらのリソースにアクセスできない場合でも、 AWS アカウントの Notifications AWS CodeStar がサポートす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。たとえば、このポリシーを持つユーザーは、 CodeCommit CodeCommit自身にアクセスする権限がなくても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: read-only-level AWS CodeStar 通知を使用するためのポリシー

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、、、AWSCodePipeline_ReadOnlyAccess管理ポリシーに含まれる権限と同等です。AWSCodeBuildReadOnlyAccessAWSCodeCommitReadOnly

```
{
    "Version": "2012-10-17",
    "Id": "CodeNotification__ReadOnly",
    "Statement": [
        {
            "Sid": "Reads_API_Access",
            "Effect": "Allow",
            "Action": [
                "CodeNotification:DescribeNotificationRule",
                "CodeNotification:ListNotificationRules",
                "CodeNotification:ListTargets",
                "CodeNotification:ListEventTypes"
            ],
            "Resource": "*"
        }
    ]
}
```

の権限と例 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、以下の権限に制限されている必要があります。

> **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": [
            "codestar-connections:CreateConnection",
            "codestar-connections:DeleteConnection",
            "codestar-connections:UseConnection",
            "codestar-connections:GetConnection",
            "codestar-connections:ListConnections",
            "codestar-connections:TagResource",
            "codestar-connections:ListTagsForResource",
            "codestar-connections:UntagResource"
        ],
        "Resource": "*"
     }
   ]
}
```

例: AWS CodeConnections コンソールで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": [
                "codestar-connections:CreateConnection",
                "codestar-connections:DeleteConnection",
                "codestar-connections:GetConnection",
                "codestar-connections:ListConnections",
                "codestar-connections:GetInstallationUrl",
                "codestar-connections:GetIndividualAccessToken",
                "codestar-connections:ListInstallationTargets",
```

```
                "codestar-connections:StartOAuthHandshake",
                "codestar-connections:UpdateConnectionInstallation",
                "codestar-connections:UseConnection",
                "codestar-connections:TagResource",
                "codestar-connections:ListTagsForResource",
                "codestar-connections:UntagResource"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

例:管理者レベルの管理ポリシー AWS CodeConnections

この例では、 AWS アカウント内の IAM ユーザーにフルアクセス権を付与して、そのユーザーが接続を追加、更新、 CodeConnections 削除できるようにしたいと考えています。これはフルアクセスポリシーで、AWSCodePipeline_FullAccess管理ポリシーと同等です。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添付してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "ConnectionsFullAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-connections:CreateConnection",
            "codestar-connections:DeleteConnection",
            "codestar-connections:UseConnection",
            "codestar-connections:GetConnection",
            "codestar-connections:ListConnections",
            "codestar-connections:ListInstallationTargets",
            "codestar-connections:GetInstallationUrl",
            "codestar-connections:StartOAuthHandshake",
            "codestar-connections:UpdateConnectionInstallation",
            "codestar-connections:GetIndividualAccessToken",
            "codestar-connections:TagResource",
            "codestar-connections:ListTagsForResource",
            "codestar-connections:UntagResource"
        ],
```

```
        "Resource": "*"
     }
   ]
}
```

例:コントリビューターレベルの使用に関するポリシー AWS CodeConnections

この例では、 day-to-day 接続の詳細の作成や表示などの使用にはアクセス権を付与したいが CodeConnections、接続の削除などのより破壊的なアクションにはアクセス権を付与したくないと考えている。

```
{
    "Version": "2012-10-17",
    "Sid": "AWSCodeStarConnectionsPowerUserAccess",
        "Effect": "Allow",
        "Action": [
            "codestar-connections:CreateConnection",
            "codestar-connections:UseConnection",
            "codestar-connections:GetConnection",
            "codestar-connections:ListConnections",
            "codestar-connections:ListInstallationTargets",
            "codestar-connections:GetInstallationUrl",
            "codestar-connections:GetIndividualAccessToken",
            "codestar-connections:StartOAuthHandshake",
            "codestar-connections:UpdateConnectionInstallation",
            "codestar-connections:ListTagsForResource"
        ],
        "Resource": "*"
        }
    ]
}
```

例: read-only-level 使用に関するポリシー AWS CodeConnections

この例では、アカウントの IAM ユーザーに、 AWS アカウント内の接続への読み取り専用アクセス権を付与します。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。

```
{
    "Version": "2012-10-17",
    "Id": "Connections__ReadOnly",
    "Statement": [
        {
            "Sid": "Reads_API_Access",
```

```
            "Effect": "Allow",
            "Action": [
            "codestar-connections:GetConnection",
            "codestar-connections:ListConnections",
            "codestar-connections:ListInstallationTargets",
            "codestar-connections:GetInstallationUrl",
            "codestar-connections:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

例: AWS CodeConnections 指定したリポジトリで使用するためのスコープダウンポリシー

次の例では、 CodeBuild 顧客は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望んでいます。 CodeBuild サービスロールのポリシー:

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "codestar-connections:UseConnection"
        ],
        "Resource": "arn:aws:codestar-connections:us-
west-2:connection:3dee99b9-172f-4ebe-a257-722365a39557",
        "Condition": {"ForAllValues:StringEquals": {"codestar-
connections:FullRepositoryId": "myrepoowner/myreponame"}}
    }
}
```

例:接続を使用するポリシー CodePipeline

次の例では、管理者はユーザにとの接続を使用させたいと考えています 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": [
            "codestar-connections:PassConnection"
```

```
        ],
        "Resource": "arn:aws:codestar-connections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "Condition": {"ForAllValues:StringEquals": {"codestar-
connections:PassedToService": "codepipeline.amazonaws.com"}}
    }
}
```

例:Bitbucket CodeBuild の読み取り操作にサービスロールを使用する AWS CodeConnections

次の例では、お客様はリポジトリに関係なく Bitbucket CodeBuild の読み取り操作をサービスロールに実行させたいと考えています。 CodeBuild サービスロールに関するポリシー:

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "codestar-connections:UseConnection"
        ],
        "Resource": "arn:aws:codestar-connections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "Condition": {"ForAllValues:StringEquals": {"codestar-
connections:ProviderPermissionsRequired": "read_only"}}
    }
}
```

例: CodeBuild サービスロールによる操作の実行を制限する AWS CodeConnections

次の例では、 CodeBuild 顧客はサービスロールが次のような操作を実行しないようにしたいと考えていますCreateRepository。 CodeBuild サービスロールに関するポリシー:

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "codestar-connections:UseConnection"
        ],
        "Resource": "arn:aws:codestar-connections:us-west-2:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "Condition": {"ForAllValues:StringNotEquals": {"codestar-
connections:ProviderPermissionsRequired": "CreateRepository"}}
```

```
    }
}
```

タグを使用して AWS CodeStar Connections 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 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は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 ユーザー用のポリシー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

次のポリシーでは、 CodeConnectionsで接続を作成するアクセス許可をユーザーに付与します。

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": [
        "codestar-connections:CreateConnection",
        "codestar-connections: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、 CodeConnectionsのリソースに対してアクションを実行して情報を取得するアクセス許可をユーザーに付与します。

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": [
        "codestar-connections:CreateConnection",
        "codestar-connections:DeleteConnection",
        "codestar-connections: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

通知機能は CodeBuild、、 CodeCommit、 CodePipelineコンソールのほか CodeDeploy、開発者ツールコンソールの設定ナビゲーションバー自体にも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。これらの権限では、 AWS CodeStar AWS CodeStar AWS アカウント内の通知リソースと接続リソースの詳細を一覧表示して表示できる必要があります。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。 AWS CodeBuild、 AWS CodeCommit、 AWS CodeDeploy AWS CodePipelineおよびへのアクセス権の付与 (これらのコンソールへのアクセス権を含む) 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: ID ベースのポリシーの使用には CodeBuild
- CodeCommit: ID ベースのポリシーの使用 CodeCommit
- AWS CodeDeploy: ID およびアクセス管理 AWS CodeDeploy
- CodePipeline: IAM ポリシーによるアクセス制御

AWS CodeStar AWS 通知には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:通知を管理するための管理者レベルのポリシー AWS CodeStar
- 例:通知を使用するためのコントリビューターレベルのポリシー AWS CodeStar
- 例: read-only-level AWS CodeStar 通知を使用するためのポリシー.

AWS CodeStar Connections AWS には管理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:管理者レベルの管理ポリシー AWS CodeConnections
- 例:コントリビューターレベルの使用に関するポリシー AWS CodeConnections

- 例: read-only-level 使用に関するポリシー AWS CodeConnections

AWS CLI または AWS API のみを呼び出しているユーザーには、コンソール権限を許可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、またはまたは API を使用してこのアクションをプログラムで実行するための権限が含まれています。 AWS CLI AWS

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

```
        }
    ]
}
```

AWS CodeStar AWS CodeStar 通知と接続のトラブルシューティング:ID とアクセス

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
- AWSAWS CodeStar AWS CodeStar 自分のアカウント以外のユーザーにも通知と接続の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 AWS CodeStar 他のユーザーが通知と接続にアクセスできるようにするには、アクセスが必要なユーザーまたはアプリケーション用の 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 Notifications and AWS CodeStar Connections で適切な権限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。

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 通知の権限と例。

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 AWS トピックが通知ルールと同じリージョンにあることを確認してください。
- 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。
- Notifications AWS CodeStar 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。

AWSAWS CodeStar AWS CodeStar 自分のアカウント以外のユーザーにも通知と接続の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制御リスト (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 AWS CodeStar 通知と接続がこれらの機能をサポートしているかどうかについては、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。

- AWS アカウント 所有しているリソース全体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「AWS アカウント 所有する別の 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第三者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「AWS アカウント 第三者が所有するリソースへのアクセスの提供」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 Notifications のサービスにリンクされたロールの使用

AWS CodeStar Notificaios 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 Notifications に直接リンクされた一意のタイプの IAM ロールです。サービスリンクロールは AWS CodeStar Notifications によって事前に定義されており、サービスがユーザーに代わって AWS の他のサービスを呼び出すために必要なすべての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ことで、必要なアクセス許可を手動で追加する必要がなくなるため、AWS CodeStar Notifications の設定が簡単になります。AWS CodeStar Notifications 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ます。特に定義されている場合を除き、AWS CodeStar Notifications のみがそのロールを引き受けることができます。定義される許可には、信頼ポリシーと許可ポリシーが含まれており、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、リソースへの意図しないアクセスによるアクセス許可の削除が防止され、AWS CodeStar Notifications リソースは保護されます。

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 Notifications は、AWSServiceRoleForCodeStarNotifications の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した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 サービスリンクロールは、以下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、指定し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 Notifications 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"
        }
    ]
}
```

サービスにリンクされたロールの作成、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、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「Service-Linked Role Permissions」(サービスリンクロールのアクセス権限) 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

サービスにリンクされたロールを手動で作成する必要はありません。SDK からデベロッパーツールコンソールまたは 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。

このサービスにリンクされたロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。SDK からデベロッパーツールコンソールまたは 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「サービスにリンクされたロールの編集」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 Notifications の場合、これは AWS アカウントで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**
>
> リソースを削除する際に、AWS CodeStar Notifications サービスでロールが使用されている場合、削除は失敗することがあります。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。

AWSServiceRoleForCodeStarNotifications が使用するAWS CodeStar Notifications リソースを削除するには

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

   > **Note**
   >
   > 通知ルールは、これらのルールを作成した 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 AWS リージョン を変更します。

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、AWS CLI、または AWS Identity and Access Management API を使用して、AWSServiceRoleForCodeStarNotifications サービスリンク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「サービスにリンクされたロールの削除」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をサポートするリージョン

AWS CodeStar Notifications は、サービスを利用できるすべての 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 のリージョンとエンドポイント」および「AWS CodeStar Notifications」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、AWS CodeConnections 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 CodeConnections による事前定義済みのロールであり、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許可を備えています。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くなるため、AWS CodeConnections の設定が簡単になります。AWS CodeConnections はこの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 CodeConnections のみがそのロールを引き受けます。定義される許可には、信頼ポリシーと許可ポリシーが含まれており、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、リソースへの意図しないアクセスによる許可の削除が防止され、AWS CodeConnections リソースは保護されます。

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 CodeConnections のサービスにリンクされたロールの許可

AWS CodeConnections は、サービスにリンクされたロール AWSServiceRoleForGitSync を使用して、接続された Git ベースのリポジトリで Git 同期を使用します。

サービスにリンクされたロール AWSServiceRoleForGitSync は、次のサービスを信頼してそのロールを引き受けます。

- `repository.sync.codeconnections.amazonaws.com`

AWSGitSyncServiceRolePolicy という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おける以下のアクションの実行を AWS CodeConnections に許可します。

- 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。

サービスにリンクされたロールの作成、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、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 のサービスにリンクされたロールの作成

サービスにリンクされたロールを手動で作成する必要はありません。ロールは、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作成します。

このサービスにリンクされたロールを削除した後で再度作成する必要が生じた場合は、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。これは、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することを意味します。

> **Note**
>
> リソースを削除する際に、AWS CodeConnections のサービスでロールが使用されている場合、削除は失敗することがあります。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。

AWSServiceRoleForGitSync が使用する AWS CodeConnections リソースを削除するには

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、AWS CLI、または AWS Identity and Access Management API を使用して、サービスにリンクされたロール AWSServiceRoleForGitSync を削除します。詳細については、IAM ユーザーガイドの「サービスにリンクされたロールの削除」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をサポートするリージョン

AWS CodeConnections では、この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」を参照してください。

AWS CodeConnections の AWS マネージドポリシー

AWS マネージドポリシーは、AWS が作成および管理するスタンドアロンポリシーです。AWS マネージドポリシーは、多くの一般的なユースケースでアクセス許可を提供できるように設計されているため、ユーザー、グループ、ロールへのアクセス許可の割り当てを開始できます。

AWS マネージドポリシーは、ご利用の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ご注意ください。AWS のすべてのお客様が使用できるようになるのを避けるためです。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することで、アクセス許可を絞り込むことをお勧めします。

AWS マネージドポリシーで定義したアクセス権限は変更できません。AWS が AWS マネージドポリシーに定義されているアクセス許可を更新すると、更新はポリシーがアタッチされているすべてのプリンシパルアイデンティティ (ユーザー、グループ、ロール) に影響します。新しい AWS のサービスを起動するか、既存のサービスで新しい API オペレーションが使用可能になると、AWS が AWS マネージド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 管理ポリシーを参照してください。

AWS マネージドポリシー: AWSGitSyncServiceRolePolicy

お客様の IAM エンティティに、AWSGitSyncServiceRolePolicy をアタッチすることはできません。このポリシーは、ユーザーに代わって AWS CodeConnections がアクションを実行することを許可する、サービスにリンクされたロール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、以下の許可が含まれています。

- `codestar-connections` – 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"
   ],
   "Resource": "arn:aws:codestar-connections:*:*:connection/*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  }
 ]
}
```

AWS CodeConnections マネージドポリシーの AWS 更新

このサービスがこれらの変更の追跡を開始してからの、AWS の AWS CodeConnections 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通知については、AWS CodeConnections [Document history](ドキュメントの履歴) ページの RSS フィードをご覧ください。

変更	説明	日付
AWSGitSyncServiceRolePolicy – 新しいポリシー	AWS CodeConnections でポリシーが追加されました。 接続された Git ベースのリポジトリで AWS CodeConnections ユーザーが 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 は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 AWS CodeStar 通知と接続のコンプライアンス検証

AWS CodeStar AWS CodeStar 通知と接続は、 AWS どのコンプライアンスプログラムの対象にもなりません。

AWS 特定のコンプライアンスプログラムの対象となるサービスの一覧については、「AWS コンプライアンスプログラム別の対象サービス」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。

サードパーティの監査レポートはを使用して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 Notifications and AWS CodeStar Connections を使用する際のコンプライアンス責任は、データの機密性、会社のコンプライアンス目標、および適用される法律と規制によって決まります。 AWS 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 — これらの導入ガイドでは、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導入する手順を説明しています。 AWS
- AWS コンプライアンスリソース — この一連のワークブックとガイドは、お客様の業界や地域に当てはまる場合があります。
- AWS Config— AWS このサービスでは、リソース構成が社内の慣行、業界のガイドライン、規制にどの程度準拠しているかを評価します。
- AWS Security Hub— AWS このサービスでは、内部のセキュリティ状態を包括的に把握できるため、 AWS セキュリティ業界の標準やベストプラクティスに準拠しているかどうかを確認できます。

AWS CodeStar Notifications と AWS CodeStar Connectionsでの耐障害性

AWS の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ます。AWSリージョンには、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立し隔離されたアベイラビリティーゾーンがあります。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、および拡張性が優れています。

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、これらのルールを作成した AWS リージョン に固有です。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各 AWS リージョン で通知ルールを確認します。
- AWS CodeStar Notifications は、通知ルールターゲットとして Amazon Simple Notification Service (Amazon SNS) トピックに依存しています。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 と AWS CodeStar Connections でのインフラストラクチャセキュリティ

マネージドサービスであるAWS CodeStar Notifications と AWS CodeStar Connections 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。

AWS が公開している API コールを使用して、ネットワークを介して AWS CodeStar Notifications と AWS CodeStar Connections にアクセスします。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は、リソースが作成されたリージョン以外のリージョンでリソースへの接続を提供することに関し、唯一の目的として基盤となるサービスを使用している AWS リージョン の外側にある AWS リージョン に、接続リソースに関連する情報を保存や処理することに同意し、指示します。

詳細については、「AWS CodeStar Connections のグローバルリソース」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。

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 CodeStar Connections API バージョン: 2019-12-01

変更	説明	日付
GitLab セルフマネージドのサポート	AWS リソースが GitLab セルフマネージドとやり取りするための接続およびホスト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」と「GitLab セルフマネージドへの接続を作成する」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
接続のサービスにリンクされたロールのサポート	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、「AWS CodeStar Connections のサービスにリンクされたロール	2023 年 11 月 26 日

	の使用」と「マネージドポリシー」を参照してください。	
GitLab グループのサポート	AWS リソースが GitLab グループとやり取り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、Create a connection および Create a connection to 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	GitLab 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、Create a connection および Create a connection to 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、「リージョンをまたぐ AWS CodeStar Connections 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 組織のリポジトリへの接続を作成する場合は、GitHub 組織の所有者である必要があります。詳細については、「GitHub への接続時の接続エラー」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、「AWS CodeStar Connections でのリソースへのタグ付け」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 CodeStar Connections とインターフェイス 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 プロバイダーのタイプ	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、Create a connection および Create a connection to 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」、「接続の使用開始」、「接続を作成する」、「AWS CodeStar Connections でのリソースへのタグ付け」、「セキュリティ」、「接続のクォータ」、「トラブルシューティング」、および「AWS CloudTrail を使用した AWS CodeStar Connections API コール」を参照してください。追加のプロバイダのアクション（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、Actions for 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。	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、「リポジトリでの通知ルールのイベント」および「CodeCommit でのプルリクエストの操作 」を参照してください。	2020 年 2 月 10 日
新たに 2 つの AWS リージョンで通知が利用可能に	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) の通知をサポートするようになりました。詳細については、「AWS 全般のリファレンス」の「AWS CodeStar 通知」を参照してください。	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、CodeCommit のセッションタグ情報を含めることができる	セッションタグを使用して、CodeCommit の通知に表示名や E メールアドレスなどのユーザー ID 情報を含めることができるようになりました。詳細については、「概念」および「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。	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。