使用者指南

開發人員工具主控台

開發人員工具主控台: 使用者指南

Table of Contents

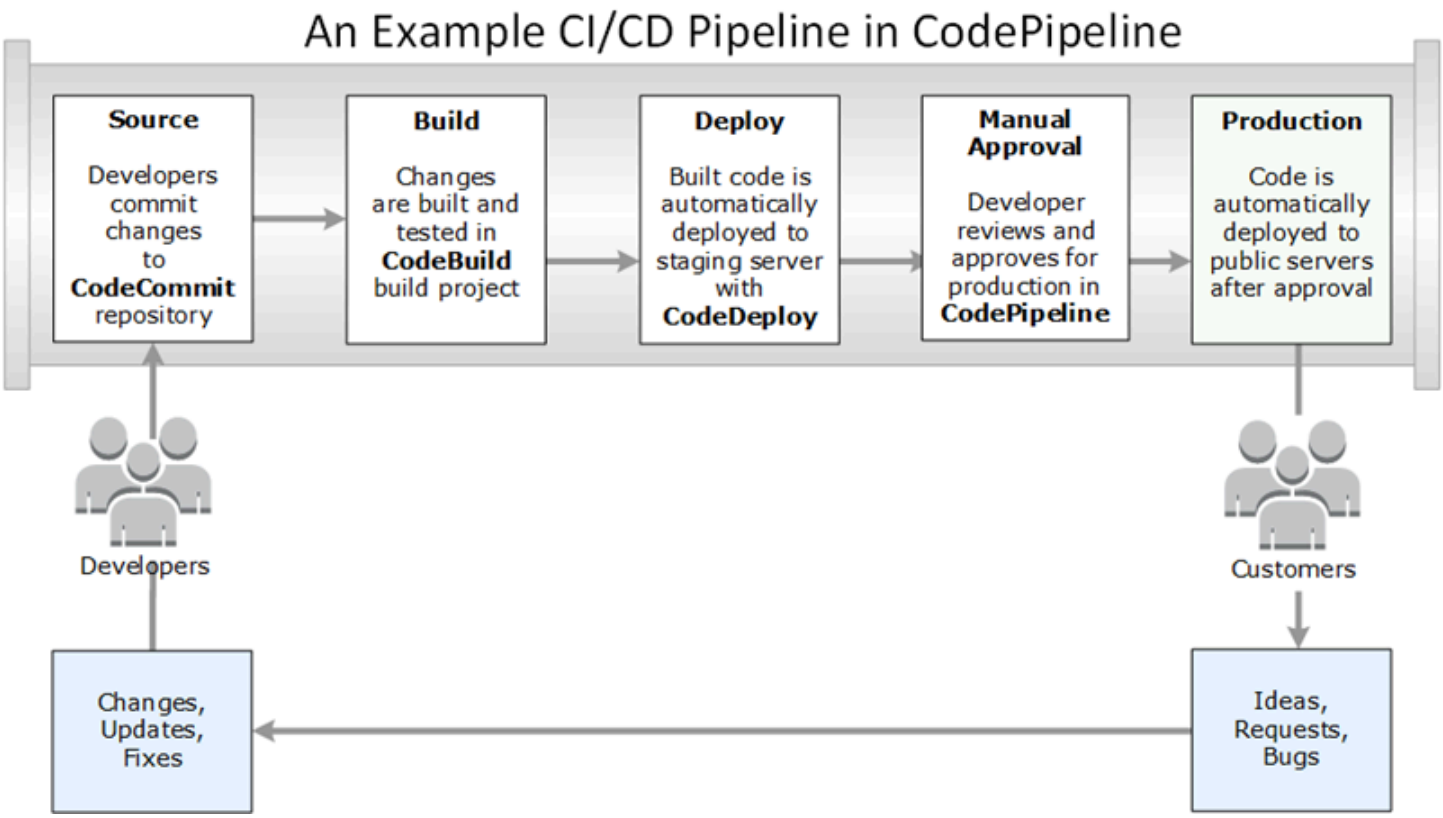
什麼是開發人員工具主控台？

開發人員工具主控台是一組服務和功能的大本營，可供您個人或集體使用，以協助您個人或團隊開發軟體。開發人員工具可協助您安全地儲存、建置、測試和部署軟體。這些工具供個人或集體使用，支援 DevOps、持續整合和持續交付 (CI/CD)。

開發人員工具主控台包含下列服務：

- AWS CodeCommit 是全受管原始碼控制服務，託管私有 Git 存儲庫。您可以使用儲存庫，以私下在 AWS 雲端 存放和管理資產 (例如，文件、原始程式碼和二進位檔案)。儲存庫存放您的專案歷史記錄，包括從第一個遞交到最新的變更。您可以協同地處理儲存庫中的程式碼，對程式碼做註解和建立提取請求，以協助確保程式碼品質。
- AWS CodeBuild 是全受管組建服務，可編譯原始碼、執行單元測試，並產生可立即部署的成品。它提供預先封裝的組建環境，適用於常見的程式設計語言和組建工具，例如 Apache Maven、Gradle 等等。您也可以自訂 CodeBuild 中的組建環境，以使用您自己的組建工具。
- AWS CodeDeploy 是全受管部署服務，可自動將軟體部署到運算服務，例如 Amazon EC2、AWS Lambda 和內部部署伺服器。它可協助您快速發行新功能、避免應用程式部署期間停機，以及處理應用程式更新時的複雜性。
- AWS CodePipeline 是持續整合和持續交付服務，可將發行軟體所需的步驟模型化、視覺化和自動化。您可以使用快速模型化和設定軟體發行程序的不同階段。根據您定義的發行程序模型，您可以在每次程式碼變更時建置、測試和部署程式碼。

以下範例描述您如何一起使用開發人員工具主控台中的服務，以協助您開發軟體。



在此範例中，開發人員在 CodeCommit 中建立儲存庫，並用來開發和協作其程式碼。他們在 CodeBuild 中建立建置專案來建置和測試程式碼，並使用 CodeDeploy 將程式碼部署到測試和生產環境。他們想要快速反覆運算，因此在 CodePipeline 中建立管道，以偵測 CodeCommit 儲存庫中的變更。建置這些變更、執行測試，然後將成功建置和測試的程式碼部署到測試伺服器。團隊將測試階段新增至管道，以便在預備伺服器上執行更多測試，例如整合或負載測試。成功完成這些測試後，團隊成員檢閱結果，如果滿意，就手動核准變更進入生產階段。CodePipeline 將已測試和核准的程式碼部署到生產執行個體。

這只是一個簡單的範例，說明如何使用開發人員工具主控台中提供的一或多個服務來協助您開發軟體。每個服務都可以自訂以符合您的需求。它們提供與其他產品和服務 (都在 AWS 中) 的許多整合，以及與其他第三方工具的整合。如需詳細資訊，請參閱下列主題：

- CodeCommit：產品和服務整合
- CodeBuild：搭配 Jenkins 使用 CodeBuild
- CodeDeploy：產品和服務整合
- CodePipeline：產品和服務整合

您是第一次使用 的新手嗎？

如果您是第一次使用開發人員工具主控台中的一個或多個服務，我們建議您先閱讀下列主題：

- 開始使用 CodeCommit
- CodeBuild 入門、概念
- CodeDeploy 入門、主要元件
- CodePipeline 入門、概念

開發人員工具主控台的功能

開發人員工具主控台包含下列功能：

- 開發人員工具主控台包含通知管理工具功能，可用來訂閱 AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy 和 AWS CodePipeline 中的事件。此功能有自己的 API：AWS CodeStar Notifications。針對儲存庫、組建專案、部署應用程式和管道中對使用者的工作來說最重要的事件，您可以使用通知功能來快速通知使用者。通知管理員可協助使用者留意儲存庫、組建、部署或管道上發生的事件，以便他們能夠快速採取動作，例如核准變更或更正錯誤。如需詳細資訊，請參閱 什麼是通知？
- 開發人員工具主控台包含連線功能，可讓您將 AWS 資源與第三方原始碼供應商建立關聯。此功能有自己的 API：AWS CodeStar Connections。您可以使用連線功能與第三方供應商建立授權連線，並將連線資源與其他 AWS 服務搭配使用。如需詳細資訊，請參閱 什麼是連線？

什麼是通知？

開發人員工具主控台中的通知功能是用於訂閱 AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy 和 AWS CodePipeline 中事件的通知管理工具。此功能有自己的 API：AWS CodeStar Notifications。針對儲存庫、組建專案、部署應用程式和管道中對使用者的工作來說最重要的事件，您可以使用通知功能來快速通知使用者。通知管理員可協助使用者留意儲存庫、組建、部署或管道上發生的事件，以便他們能夠快速採取動作，例如核准變更或更正錯誤。

我可以用通知來做什麼？

您可以使用通知功能來建立和管理通知規則，以通知使用者其資源有重大變更，包括：

- CodeBuild 建置專案中的建置成功和失敗。

- CodeDeploy 應用程式中的部署成功和失敗。
- 提取請求中的建立和更新，包括 CodeCommit 儲存庫中對程式碼的註解。
- 在 CodePipeline 中執行的手動核准狀態和管道。

您可以將通知設定為傳送至訂閱 Amazon SNS 主題的使用者電子郵件。您也可以將此功能與 AWS Chatbot 整合，並將通知傳遞至 Slack 頻道、Microsoft Teams 頻道或 Amazon Chime 聊天室。

通知如何運作？

當您為支援的資源 (例如儲存庫、建置專案、應用程式或管道) 設定通知規則時，通知功能會建立 Amazon EventBridge 規則來監控您指定的事件。當該類型的事件發生時，通知規則會傳送通知到指定為該規則目標的 Amazon SNS 主題。這些目標的訂閱者會收到關於這些事件的通知。

如何開始使用通知？

若要開始使用，以下是一些有用的主題可供檢閱：

- 了解通知的概念。
- 設定開始使用通知所需的資源。
- 開始使用您的第一個通知規則並接收您的第一個通知。

通知概念

如果您瞭解概念和術語，設定和使用通知會比較容易。以下是使用通知時需要瞭解的一些概念。

主題

- 通知
- 通知規則
- 事件
- 詳細資訊類型
- 目標
- 通知功能與 AWS CodeStar Notifications
- 儲存庫上通知規則的事件
- 建置專案上通知規則的事件

- 部署應用程式上通知規則的事件
- 管道上通知規則的事件

通知

通知是一種訊息，其中包含您和開發人員使用的資源中所發生事件的相關資訊。您可以設定通知，以便資源 (例如組建專案、儲存庫、部署應用程式或管道) 的使用者根據您建立的通知規則，接收您指定之事件類型相關的電子郵件。

AWS CodeCommit 的通知可以透過使用工作階段標籤來包含使用者身分資訊，例如顯示名稱或電子郵件地址。CodeCommit 支援使用工作階段標籤，這類標籤是您在擔任 IAM 角色、使用臨時憑證，或在 AWS Security Token Service (AWS STS) 中聯合使用者時所傳遞且為索引鍵值組的屬性。您也可以將這類標籤與 IAM 使用者建立關聯。CodeCommit 在通知內容中包含 `displayName` 和 `emailAddress` (如果這些標籤存在)。如需詳細資訊，請參閱在 CodeCommit 中使用標籤提供其他身分資訊。

> **Important**
>
> 通知包含專案特定資訊，例如組建狀態、部署狀態、具有註解的程式碼行，以及管道核准。通知內容可能隨著新功能加入而變更。定期檢閱通知規則的目標和 Amazon SNS 主題訂閱者，才是安全性最佳實務。如需更多詳細資訊，請參閱 了解通知內容和安全性。

通知規則

通知規則是您建立的 AWS 資源，用來指定傳送通知的時間與位置。其可定義：

- 建立通知的條件。這些條件是以您選擇的事件 (資源類型特有的事件) 為基礎。支援的資源類型包括 AWS CodeBuild 中的組建專案、AWS CodeDeploy 中的部署應用程式、AWS CodePipeline 中的管道，以及 AWS CodeCommit 中的儲存庫。
- 通知的傳送目標。您可以針對一個通知規則最多指定 10 個目標。

通知規則的範圍侷限於個別組建專案、部署應用程式、管道及儲存庫。通知規則同時具有使用者定義的易用名稱和 Amazon Resource Name (ARN)。通知規則必須在資源所在的相同 AWS 區域中建立。例如，如果您的建置專案位於美國東部 (俄亥俄)區域，則通知規則也必須在美國東部 (俄亥俄)區域中建立。

您可以針對一個資源最多定義 10 個通知規則。

事件

事件是您想要監控之資源的狀態變更。每個資源都有您可從中選擇的事件類型清單。當您在資源上設定通知規則時，您可以指定導致傳送通知的事件。例如，如果您在 CodeCommit 中設定儲存庫的通知，並針對 Pull request (提取請求) 和 Branches and tags (分支與標籤) 選取 Created (已建立)，則每當該儲存庫中的使用者建立提取請求、分支或 Git 標籤時，都會傳送通知。

詳細資訊類型

當您建立通知規則時，可以選擇通知中包含的細節層次和 detail type (詳細資訊類型) (Full (完整) 或 Basic (基本))。Full (完整) 設定 (預設) 包含通知中可用於事件的所有資訊，包括服務特定事件提供的所有增強資訊。Basic (基本) 設定僅包含可用資訊的子集合。

下表列出特定事件類型可用的增強資訊，並說明詳細資訊類型之間的差異。

服務	事件	完整包含	基本不包括
CodeCommit：	遞交時的註解 提取請求的註解	所有事件詳細資訊和註解的內容，包括任何回覆或註解對話。它也包含行號和對其進行註釋的程式碼行。	註解的內容、行號、程式碼行，或任何註解對話。
CodeCommit：	已建立提取請求	所有事件詳細資訊，以及與目的地分支相關之提取請求中新增、修改或刪除的檔案數目。	沒有檔案清單或關於提取請求來源分支是否已新增、修改或刪除檔案的詳細資訊。
CodePipeline	需要手動核准	所有事件詳細資訊和自訂資料 (如果已設定)。通知也包含管道中必要核准的連結。	沒有自訂資料或連結。
CodePipeline	動作執行失敗 管道執行失敗	所有事件詳細資訊和失敗的錯誤訊息內容。	沒有錯誤訊息內容。

服務	事件	完整包含	基本不包括
	階段執行失敗		

目標

目標是接收來自通知規則之通知的位置。允許的目標類型是為 Slack 頻道或 Microsoft Team 頻道設定的 Amazon SNS 主題和 AWS Chatbot 用戶端。任何訂閱目標的使用者都會收到您在通知規則中指定之事件的相關通知。

如果您想要擴展通知的範圍，可以手動設定通知與 AWS Chatbot 之間的整合，以便將通知傳送到 Amazon Chime 聊天室。然後，您可以選擇為該 AWS Chatbot 用戶端設定的 Amazon SNS 主題，做為通知規則的目標。如需更多詳細資訊，請參閱 將通知與 AWS Chatbot 和 Amazon Chime 整合。

如果您選擇使用 AWS Chatbot 用戶端做為目標，則必須先在 AWS Chatbot 中建立該用戶端。當您選擇 AWS Chatbot 用戶端做為通知規則的目標時，系統會為該 AWS Chatbot 用戶端設定 Amazon SNS 主題，其中包含將通知傳送至 Slack 或 Microsoft Team 頻道所需的所有政策。您不需要為 AWS Chatbot 用戶端設定任何現有的 Amazon SNS 主題。

您可以選擇將 Amazon SNS 主題建立為目標，做為建立通知規則的一部分 (建議)。您也可以選擇與通知規則相同 AWS 區域中的現有 Amazon SNS 主題，但必須使用必要的政策進行設定。您用於目標的 Amazon SNS 主題必須位於您的 AWS 帳戶中。該主題也必須位於與通知規則以及建立規則的 AWS 資源相同的 AWS 區域中。

例如，如果您為美國東部 (俄亥俄)區域中的儲存庫建立通知規則，Amazon SNS 主題也必須存在該區域中。如果您在建立通知規則的過程中建立了 Amazon SNS 主題，該主題會設定所需的政策，允許將事件發佈至主題。這是處理目標和通知規則的最佳方法。如果您選擇使用已存在的主題或手動建立主題，則必須設有必要的許可，使用者才會收到通知。如需更多詳細資訊，請參閱 設定通知的 Amazon SNS 主題。

> **Note**
>
> 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之

外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

通知功能與 AWS CodeStar Notifications

雖然通知是開發人員工具主控台的一項功能，但有自己的 API，即 AWS CodeStar Notifications。也有自己的 AWS 資源類型 (通知規則)、許可和事件。通知規則的事件會記錄在 AWS CloudTrail 中。可透過 IAM 政策允許或拒絕 API 動作。

儲存庫上通知規則的事件

類別	事件	事件 ID
說明	遞交時 提取請求時	codecommit-repository-comments-on-commits codecommit-repository-comments-on-pull-requests
核准	狀態已變更 規則覆寫	codecommit-repository-approvals-status-changed codecommit-repository-approvals-rule-override
提取請求	已建立 來源已更新 狀態已變更 已合併	codecommit-repository-pull-request-created codecommit-repository-pull-request-source-updated codecommit-repository-pull-request-status-changed

類別	事件	事件 ID
		codecommit-repository-pull-request-merged
分支和標籤	已建立 Deleted (已刪除) Updated	codecommit-repository-branches-and-tags-created codecommit-repository-branches-and-tags-deleted codecommit-repository-branches-and-tags-updated

建置專案上通知規則的事件

類別	事件	事件 ID
組建狀態	失敗 Succeeded 進行中 已停止	codebuild-project-build-state-failed codebuild-project-build-state-succeeded codebuild-project-build-state-in-progress codebuild-project-build-state-stopped
組建階段	失敗 Success (成功)	codebuild-project-build-phase-failure codebuild-project-build-phase-success

部署應用程式上通知規則的事件

類別	事件	事件 ID
部署	失敗 Succeeded 已開始	codedeploy-application-deployment-failed codedeploy-application-deployment-succeeded codedeploy-application-deployment-started

管道上通知規則的事件

類別	事件	事件 ID
動作執行	Succeeded 失敗 已取消 已開始	codepipeline-pipeline-action-execution-succeeded codepipeline-pipeline-action-execution-failed codepipeline-pipeline-action-execution-canceled codepipeline-pipeline-action-execution-started
階段執行	已開始 Succeeded 繼續 已取消 失敗	codepipeline-pipeline-stage-execution-started codepipeline-pipeline-stage-execution-succeeded codepipeline-pipeline-stage-execution-resumed

類別	事件	事件 ID
		codepipeline-pipeline-stage-execution-canceled codepipeline-pipeline-stage-execution-failed
管道執行	失敗 已取消 已開始 繼續 Succeeded 已取代	codepipeline-pipeline-pipeline-execution-failed codepipeline-pipeline-pipeline-execution-canceled codepipeline-pipeline-pipeline-execution-started codepipeline-pipeline-pipeline-execution-resumed codepipeline-pipeline-pipeline-execution-succeeded codepipeline-pipeline-pipeline-execution-superseded
手動核准	失敗 需要 Succeeded	codepipeline-pipeline-manual-approval-failed codepipeline-pipeline-manual-approval-needed codepipeline-pipeline-manual-approval-succeeded

設定

如果您將 AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy 或 AWS CodePipeline 的受管政策套用至 IAM 使用者或角色，則在政策所提供的角色和許可限度內，您具有使用通知所需的許可。例如，已套用

AWSCodeBuildAdminAccess、AWSCodeCommitFullAccess、AWSCodeDeployFullAccess 或 AWSCodePipeline_FullAccess 受管政策的使用者，具有通知的完整管理存取權。

如需包含範例政策的詳細資訊，請參閱身分型政策。

如果您已將其中一個政策套用至 IAM 使用者或角色，以及 CodeBuild 中的建置專案、CodeCommit 中的儲存庫、CodeDeploy 中的部署應用程式或 CodePipeline 中的管道，便可以開始建立第一個通知規則。繼續進行開始使用通知。否則請參閱下列主題：

- CodeBuild：開始使用 CodeBuild
- CodeCommit：開始使用 CodeCommit
- CodeDeploy：教學
- CodePipeline：開始使用 CodePipeline

如果您要自行管理 IAM 使用者、群組或角色的通知管理許可，請遵循本主題中的程序，設定使用服務所需的許可和資源。

如果您想要使用先前為通知建立的 Amazon SNS 主題，而不特別為通知建立主題，則必須套用允許事件發佈至 Amazon SNS 主題的政策，以設定該主題作為通知規則的目標。

> **Note**
>
> 若要執行下列程序，您必須使用具有管理許可的帳戶登入。如需詳細資訊，請參閱建立您的第一個 IAM 管理員使用者和群組。

主題

- 建立和套用通知的管理存取政策
- 設定通知的 Amazon SNS 主題
- 讓使用者訂閱做為目標的 Amazon SNS 主題

建立和套用通知的管理存取政策

您可以透過 IAM 使用者登入，或使用角色而此角色具有許可存取您要建立通知的服務 (AWS CodeBuild、AWS CodeCommit、AWS CodeDeploy 或 AWS CodePipeline)，以管理通知。您也可以建立自己的政策，並套用至使用者或群組。

下列程序顯示如何設定具有許可管理通知和新增 IAM 使用者的 IAM 群組。如果您不想設定群組，可以直接將此政策套用到 IAM 使用者，或使用者可擔任的 IAM 角色。您也可以對 CodeBuild、CodeCommit、CodeDeploy 或 CodePipeline 使用受管政策，其中包含通知功能的策略適用存取權 (視政策的範圍而定)。

針對下方的政策，輸入名稱 (例如 `AWSCodeStarNotificationsFullAccess`) 和此政策的選用說明。說明可協助您記住政策的目的 (例如，**`This policy provides full access to AWS CodeStar Notifications.`**

若要使用 JSON 政策編輯器來建立政策

1. 登入 AWS Management Console，並開啟位於 https://console.aws.amazon.com/iam/ 的 IAM 主控台。
2. 在左側的導覽窗格中，選擇 Policies (政策)。

   如果這是您第一次選擇 Policies (政策)，將會顯示 Welcome to Managed Policies (歡迎使用受管政策) 頁面。選擇 Get Started (開始使用)。
3. 在頁面頂端，選擇 Create policy (建立政策)。
4. 在政策編輯器中，選擇 JSON 選項。
5. 輸入下列 JSON 政策文件：

```
{
   "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. 選擇 Next (下一步)。

> **Note**
>
> 您可以隨時切換視覺化與 JSON 編輯器選項。不過，如果您進行變更或在視覺化編輯器中選擇下一步，IAM 就可能會調整您的政策結構，以便針對視覺化編輯器進行最佳化。如需詳細資訊，請參閱 IAM 使用者指南中的調整政策結構。

7. 在檢視與建立頁面上，為您正在建立的政策輸入政策名稱與描述 (選用)。檢視此政策中定義的許可，來查看您的政策所授予的許可。
8. 選擇 Create policy (建立政策) 儲存您的新政策。

設定通知的 Amazon SNS 主題

設定通知最簡單方法是，在建立通知規則時建立 Amazon SNS 主題。如果現有 Amazon SNS 主題符合下列要求，您可以使用該主題：

- 它是在與您要建立通知規則的資源 (組建專案、部署應用程式、儲存庫或管道) 所在的相同 AWS 區域 中建立的。
- 該主題在 2019 年 11 月 5 日之前未用於傳送 CodeCommit 的通知。如果已使用，則其將包含啟用該功能時的政策陳述式。您可以選擇使用此主題，但是您將必須依照前面程序規定來加入其他的政策。如果在 2019 年 11 月 5 日之前，仍有一或多個儲存庫設定用於通知，則現有的政策陳述式即不應移除。
- 該主題的政策允許 AWS CodeStar 將通知發佈到主題。

設定 Amazon SNS 主題做為 AWS CodeStar Notifications 通知規則的目標

1. 登入 AWS Management Console，並在 https://console.aws.amazon.com/sns/v3/home 開啟 Amazon SNS 主控台。
2. 在導覽列中，選擇 Topics (主題)、選擇您要設定的主題，然後選擇 Edit (編輯)。
3. 展開 Access policy (存取政策)，然後選擇 Advanced (進階)。

4. 在 JSON 編輯器中，為政策新增下列陳述式。包含主題 ARN、AWS 區域、AWS 帳戶 ID 和主題名稱。

```
 {
      "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"
    }
```

該政策陳述式應如以下範例所示。

```
{
  "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. 選擇 Save changes (儲存變更)。
6. 如果您想要使用 AWS KMS 加密的 Amazon SNS 主題來傳送通知，您還必須透過將以下陳述式新增至 AWS KMS key 的政策，來啟用事件來源 (AWS CodeStar Notifications) 和加密主題之間的相容性。將 AWS 區域 (在此範例中為 us-east-2) 取代為建立金鑰的 AWS 區域。

```
{
    "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"
                }
            }
```

```
        }
    ]
}
```

如需詳細資訊，請參閱「AWS Key Management Service 開發人員指南中的靜態加密和搭配 AWS KMS 使用政策條件。

讓使用者訂閱做為目標的 Amazon SNS 主題

使用者必須訂閱做為通知規則目標的 Amazon SNS 主題，才能接收通知。如果使用者透過電子郵件地址訂閱，則必須先確認訂閱，才能接收通知。若要將通知傳送給 Slack 頻道、Microsoft Teams 頻道或 Amazon Chime 聊天室中的使用者，請參閱「設定通知和 AWS Chatbot 之間的整合」。

讓使用者訂閱用於通知的 Amazon SNS 主題

1. 登入 AWS Management Console，並在 https://console.aws.amazon.com/sns/v3/home 開啟 Amazon SNS 主控台。
2. 在導覽列中，選擇 Topics (主題)，然後選擇您要讓使用者訂閱的主題。
3. 在 Subscriptions (訂閱) 中，選擇 Create subscription (建立訂閱)。
4. 在 Protocol (通訊協定) 中，選擇 Email (電子郵件)。在 Endpoint (端點) 中，輸入電子郵件地址，然後選擇 Create subscription (建立訂閱)。

開始使用通知

開始使用通知的最簡單方法是在其中一個建置專案、部署應用程式、管道或儲存庫上設定通知規則。

> **Note**
>
> 第一次建立通知規則時，系統會在您的帳戶中建立服務連結角色。如需更多詳細資訊，請參閱針對 AWS CodeStar Notifications 使用服務連結角色。

主題

- 先決條件
- 建立儲存庫的通知規則
- 建立建置專案的通知規則

- 建立部署應用程式的通知規則
- 建立管道的通知規則

先決條件

完成「設定」中的步驟。您還需要一個您建立通知規則所針對的資源。

- 在 CodeBuild 中建立建置專案或使用現有的建置專案。
- 建立應用程式或使用現有的部署應用程式。
- 在 CodePipeline 中建立管道或使用現有的管道。
- 建立 AWS CodeCommit 儲存庫或使用現有的儲存庫。

建立儲存庫的通知規則

您可以建立通知規則，以傳送對您而言很重要的儲存庫事件的相關通知。下列步驟說明如何在單一儲存庫事件上設定通知規則。這些步驟假設您的 AWS 帳戶中已設定儲存庫。

> **Important**
>
> 如果在 2019 年 11 月 5 日之前在 CodeCommit 中設定通知，則用於這些通知的 Amazon SNS 主題將包含可讓 CodeCommit 發佈至該主題的政策，其包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

1. 前往 https://console.aws.amazon.com/codecommit/ 開啟 CodeCommit 主控台。
2. 從清單中選擇儲存庫並開啟它。
3. 選擇 Notify (通知)，然後選擇 Create notification rule (建立通知規則)。您也可以選擇 Settings (設定)、選擇 Notifications (通知)，然後選擇 Create notification rule (建立通知規則)。
4. 在 Notification name (通知名稱) 中，輸入規則的名稱。
5. 如果您只希望提供給 Amazon EventBridge 的資訊包含在通知中，請在 Detail type (詳細資訊類型) 中，選擇 Basic (基本)。如果您想包含提供給 Amazon EventBridge 的資訊，以及可能由資源服務或通知管理工具提供的資訊，請選擇 Full (完整)。

   如需更多詳細資訊，請參閱 了解通知內容和安全性。

6. 在 Events that trigger notifications (觸發通知的事件) 的 Branches and tags (分支和標籤) 下，選取 Created (已建立)。
7. 在 Targets (目標) 中，選擇 Create SNS topic (建立 SNS 主題)。

> **Note**
>
> 您在建立通知規則的過程中建立主題時，系統會為您套用允許 CodeCommit 將事件發佈至主題的政策。使用針對通知規則建立的主題，有助於確保您只訂閱需要接收此儲存庫相關通知的使用者。

在 codestar-notifications- 字首之後，輸入主題的名稱，然後選擇 Submit (提交)。

> **Note**
>
> 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

8. 選擇 Submit (提交)，然後檢閱通知規則。
9. 以您的電子郵件地址訂閱您剛建立的 Amazon SNS 主題。如需更多詳細資訊，請參閱 讓使用者訂閱用於通知的 Amazon SNS 主題。
10. 前往您的儲存庫，並從預設分支建立測試分支。
11. 建立分支後，通知規則會傳送通知給所有主題訂閱者，其中包含該事件的相關資訊。

建立建置專案的通知規則

您可以建立通知規則，以傳送組建專案上對您而言很重要的事件的相關通知。下列步驟說明如何在單一組建專案事件上設定通知規則。這些步驟假設您已在 AWS 帳戶中設定組建專案。

1. 前往 https://console.aws.amazon.com/codebuild/ 開啟 CodeBuild 主控台。
2. 從清單中選擇組建專案並開啟它。

3. 選擇 Notify (通知)，然後選擇 Create notification rule (建立通知規則)。您也可以選擇 Settings (設定)，然後選擇 Create notification rule (建立通知規則)。
4. 在 Notification name (通知名稱) 中，輸入規則的名稱。
5. 如果您只希望提供給 Amazon EventBridge 的資訊包含在通知中，請在 Detail type (詳細資訊類型) 中，選擇 Basic (基本)。如果您想包含提供給 Amazon EventBridge 的資訊，以及可能由資源服務或通知管理工具提供的資訊，請選擇 Full (完整)。

   如需更多詳細資訊，請參閱 了解通知內容和安全性。
6. 在 Events that trigger notifications (觸發通知的事件) 的 Build phase (組建階段) 下，選取 Success (成功)。
7. 在 Targets (目標) 中，選擇 Create SNS topic (建立 SNS 主題)。

   > **Note**
   >
   > 您在建立通知規則的過程中建立主題時，系統會為您套用允許 CodeBuild 將事件發佈至主題的政策。使用針對通知規則建立的主題，有助於確保您只訂閱需要接收此組建專案相關通知的使用者。

   在 codestar-notifications- 字首之後，輸入主題的名稱，然後選擇 Submit (提交)。

   > **Note**
   >
   > 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。
8. 選擇 Submit (提交)，然後檢閱通知規則。
9. 以您的電子郵件地址訂閱您剛建立的 Amazon SNS 主題。如需更多詳細資訊，請參閱 讓使用者訂閱用於通知的 Amazon SNS 主題。
10. 導覽至您的組建專案並啟動組建。
11. 成功完成組建階段之後，通知規則會傳送通知給所有主題訂閱者，其中包含該事件的相關資訊。

建立部署應用程式的通知規則

您可以建立通知規則，以傳送部署應用程式上對您而言很重要的事件的相關通知。下列步驟說明如何在單一組建專案事件上設定通知規則。這些步驟假設您已在 AWS 帳戶中設定部署應用程式。

1. 前往 https://console.aws.amazon.com/codedeploy/ 開啟 CodeDeploy 主控台。
2. 從清單中選擇應用程式並開啟它。
3. 選擇 Notify (通知)，然後選擇 Create notification rule (建立通知規則)。您也可以選擇 Settings (設定)，然後選擇 Create notification rule (建立通知規則)。
4. 在 Notification name (通知名稱) 中，輸入規則的名稱。
5. 如果您只希望提供給 Amazon EventBridge 的資訊包含在通知中，請在 Detail type (詳細資訊類型) 中，選擇 Basic (基本)。如果您想包含提供給 Amazon EventBridge 的資訊，以及可能由資源服務或通知管理工具提供的資訊，請選擇 Full (完整)。

   如需更多詳細資訊，請參閱 了解通知內容和安全性。

6. 在 Events that trigger notifications (觸發通知的事件) 的 Deployment (部署) 下，選取 Succeeded (成功)。
7. 在 Targets (目標) 中，選擇 Create SNS topic (建立 SNS 主題)。

   > **Note**
   >
   > 您在建立通知規則的過程中建立主題時，系統會為您套用允許 CodeDeploy 將事件發佈至主題的政策。使用針對通知規則建立的主題，有助於確保您只訂閱需要接收此部署應用程式相關通知的使用者。

   在 codestar-notifications- 字首之後，輸入主題的名稱，然後選擇 Submit (提交)。

   > **Note**
   >
   > 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，

除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

8. 選擇 Submit (提交)，然後檢閱通知規則。
9. 以您的電子郵件地址訂閱您剛建立的 Amazon SNS 主題。如需更多詳細資訊，請參閱 讓使用者訂閱用於通知的 Amazon SNS 主題。
10. 前往您的部署應用程式並開始部署。
11. 部署成功之後，通知規則會傳送通知給所有主題訂閱者，其中包含該事件的相關資訊。

建立管道的通知規則

您可以建立通知規則，以傳送您管道上對您至關重要的事件通知。下列步驟說明如何在單一管道事件上設定通知規則。這些步驟假設您的 AWS 帳戶中已設定管道。

1. 前往 https://console.aws.amazon.com/codepipeline/ 開啟 CodePipeline 主控台。
2. 從清單中選擇管道並開啟它。
3. 選擇 Notify (通知)，然後選擇 Create notification rule (建立通知規則)。您也可以選擇 Settings (設定)，然後選擇 Create notification rule (建立通知規則)。
4. 在 Notification name (通知名稱) 中，輸入規則的名稱。
5. 如果您只希望提供給 Amazon EventBridge 的資訊包含在通知中，請在 Detail type (詳細資訊類型) 中，選擇 Basic (基本)。如果您想包含提供給 Amazon EventBridge 的資訊，以及可能由資源服務或通知管理工具提供的資訊，請選擇 Full (完整)。

   如需更多詳細資訊，請參閱 了解通知內容和安全性。
6. 在 Events that trigger notifications (觸發通知的事件) 的 Action execution (動作執行) 下，選取 Started (已開始)。
7. 在 Targets (目標) 中，選擇 Create SNS topic (建立 SNS 主題)。

   > **Note**
   >
   > 您在建立通知規則的過程中建立主題時，系統會為您套用允許 CodePipeline 將事件發佈至主題的政策。使用針對通知規則建立的主題，有助於確保您只訂閱需要接收此管道相關通知的使用者。

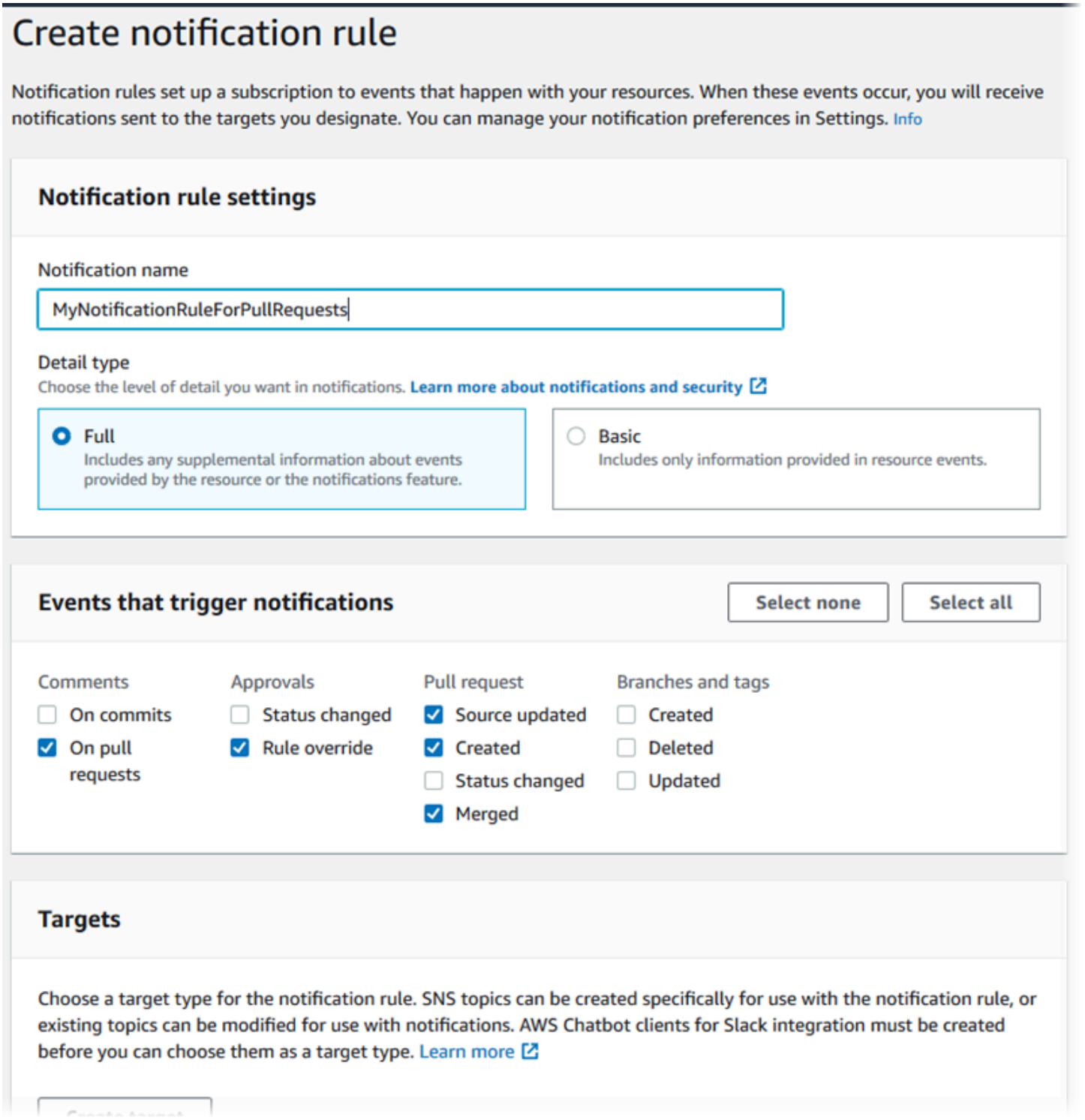
   在 codestar-notifications- 字首之後，輸入主題的名稱，然後選擇 Submit (提交)。

> **Note**
>
> 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

8. 選擇 Submit (提交)，然後檢閱通知規則。
9. 以您的電子郵件地址訂閱您剛建立的 Amazon SNS 主題。如需更多詳細資訊，請參閱 讓使用者訂閱用於通知的 Amazon SNS 主題。
10. 導覽至您的管道，然後選擇 Release change (版本變更)。
11. 動作開始時，通知規則會傳送通知給所有主題訂閱者，其中包含該事件的相關資訊。

使用通知規則

在通知規則中，您設定要使用者接收通知的事件，並指定接收這些通知的目標。您可以透過 Amazon SNS，或透過為 Slack 或 Microsoft Team 頻道設定的 AWS Chatbot 用戶端，直接傳送通知給使用者。如果您想要擴展通知的範圍，可以手動設定通知與 AWS Chatbot 之間的整合，以便將通知傳送到 Amazon Chime 聊天室。如需詳細資訊，請參閱目標及將通知與 AWS Chatbot 和 Amazon Chime 整合。



您可以使用開發人員工具主控台或 AWS CLI 來建立和管理通知規則。

主題

- 建立通知規則

- 檢視通知規則
- 編輯通知規則
- 啟用或停用通知規則的通知
- 刪除通知規則

建立通知規則

您可以使用開發人員工具主控台或 AWS CLI 來建立通知規則。您可以將 Amazon SNS 主題建立為通知規則的目標，做為建立規則的一部分。如果您要使用 AWS Chatbot 用戶端做為目標，必須先建立該用戶端，然後才能建立規則。如需更多詳細資訊，請參閱 為 Slack 頻道設定 AWS Chatbot 用戶端。

建立通知規則 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 使用導覽列瀏覽至資源。
   - 針對 CodeBuild，請選擇 Build (建置)，選擇 Build projects (建置專案)，然後選擇一個建置專案。
   - 針對 CodeCommit，選擇 Source (來源)，選擇 Repositories (儲存庫)，然後選擇一個儲存庫。
   - 針對 CodeDeploy，選擇 Applications (應用程式)，然後選擇一個應用程式。
   - 針對 CodePipeline，選擇 Pipeline (管道)，選擇 Pipelines (管道)，然後選擇一個管道。
3. 在資源頁面上，選擇 Notify (通知)，然後選擇 Create notification rule (建立通知規則)。您也可以前往資源的 Settings (設定) 頁面，前往 Notifications (通知) 或 Notification rules (通知規則)，然後選擇 Create notification rule (建立通知規則)。
4. 在 Notification name (通知名稱) 中，輸入規則的名稱。
5. 如果您只希望提供給 Amazon EventBridge 的資訊包含在通知中，請在 Detail type (詳細資訊類型) 中，選擇 Basic (基本)。如果您想包含提供給 Amazon EventBridge 的資訊，以及可能由資源服務或通知管理工具提供的資訊，請選擇 Full (完整)。

   如需更多詳細資訊，請參閱 了解通知內容和安全性。
6. 在 Events that trigger notifications (觸發通知的事件) 中，選取您要傳送通知的事件。如需資源的事件類型，請參閱下列各項：
   - CodeBuild：建置專案上通知規則的事件
   - CodeCommit：儲存庫上通知規則的事件

- CodeDeploy：部署應用程式上通知規則的事件
- CodePipeline：管道上通知規則的事件

7. 在 Targets (目標) 中，執行下列其中一個動作：

- 如果您已設定要與通知搭配使用的資源，請在選擇目標類型中，AWS Chatbot (Slack)、AWS Chatbot (Microsoft Teams) 或 SNS topic。在選擇目標中，選擇用戶端的名稱 (適用於在 AWS Chatbot 中設定的 Slack 或 Microsoft Team 用戶端)，或 Amazon SNS 主題的 Amazon Resource Name (ARN) (適用於已設定通知所需政策的 Amazon SNS 主題)。
- 如果您尚未設定要與通知搭配使用的資源，請選擇 Create target (建立目標)，然後選擇 SNS topic (SNS 主題)。在 codestar-notifications- 之後，提供主題名稱，然後選擇 Create (建立)。

> **Note**
>
> - 如果您在建立通知規則的過程中建立 Amazon SNS 主題，將會為您套用允許通知功能將事件發佈至主題的政策。使用針對通知規則建立的主題，有助於確保您只訂閱需要接收此資源相關通知的使用者。
> - 您無法在建立通知規則時建立 AWS Chatbot 用戶端。如果您選擇 AWS Chatbot (Slack) 或 AWS Chatbot (Microsoft Teams)，您會看到一個按鈕，指示您在 AWS Chatbot 中設定用戶端。選擇該選項會開啟 AWS Chatbot 主控台。如需更多詳細資訊，請參閱 為 Slack 頻道設定 AWS Chatbot 用戶端。
> - 如果您想要使用現有的 Amazon SNS 主題做為目標，除了該主題可能存在的任何其他政策以外，您必須為 AWS CodeStar Notifications 新增必要政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

8. 選擇 Submit (提交)，然後檢閱通知規則。

> **Note**
>
> 使用者必須訂閱並確認您指定為規則目標的 Amazon SNS 主題訂閱，才能收到通知。如需更多詳細資訊，請參閱 讓使用者訂閱用於通知的 Amazon SNS 主題。

建立通知規則 (AWS CLI)

1. 在終端機或命令提示字元中，執行 create-notification rule 命令，以產生 JSON 基本結構。

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

您可以將檔案命名為任何您想要的名稱。在此範例中，檔案命名為 *rule.json*。

2. 在純文字編輯器中開啟 JSON 檔案，並編輯成包含您想要用於規則的資源、事件類型和 Amazon SNS 目標。

以下範例顯示名為 **MyNotificationRule** 的通知規則，用於 AWS 帳戶 (ID 為 *123456789012*) 中的 *MyDemoRepo* 儲存庫。建立分支和標籤時，含有完整詳細資訊類型的通知會傳送到名為 *MyNotificationTopic* 的 Amazon SNS 主題。

```
{
    "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"
}
```

儲存檔案。

3. 在終端機或命令列中，再次執行 create-notification-rule 命令，使用您剛編輯的檔案建立通知規則。

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

4. 如果成功，此命令會傳回通知規則的 ARN，如下所示。

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

列出通知規則的事件類型 (AWS CLI)

1. 在終端機或命令提示字元中，執行 list-event-types 命令。您可以使用 `--filters` 選項，將回應限制為特定資源類型或其他屬性。例如，以下命令會傳回 CodeDeploy 應用程式的事件類型清單。

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

2. 此命令會產生類似下列的輸出。

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

新增標籤到通知規則 (AWS CLI)

1. 在終端機或命令提示字元中，執行 tag-resource 命令。例如，使用下列命令來新增名稱為 *Team* 且值為 *Li_Juan* 的標籤金鑰/值對。

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

2. 此命令會產生類似下列的輸出。

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

檢視通知規則

您可以使用開發人員工具主控台或 AWS CLI 來檢視 AWS 區域中所有資源的所有通知規則。您也可以檢視每個通知規則的詳細資訊。與建立通知規則的程序不同，您不必前往資源的資源頁面。

檢視通知規則 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。
3. 在 Notification rules (通知規則)中，檢閱您目前登入之 AWS 區域 區域中 AWS 帳戶 帳戶中為您的資源所設定的規則清單。使用選擇器變更 AWS 區域。
4. 若要檢視通知規則的詳細資訊，請從清單中選擇它，然後選擇 View details (檢視詳細資訊)。您也可以簡單地在清單中選擇其名稱。

檢視通知規則清單 (AWS CLI)

1. 在終端機或命令提示字元中，執行 list-notification-rules 命令來檢視所指定 AWS 區域的所有通知規則。

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

2. 如果成功，此命令會傳回 AWS 區域中每個通知規則的 ID 和 ARN，如下所示。

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

檢視通知規則的詳細資訊 (AWS CLI)

1. 在終端機或命令提示字元中，執行 describe-notification-rule 命令，並指定通知規則的 ARN。

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

2. 如果成功，此命令傳回的輸出會類似如下。

```
{
    "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"
}
```

檢視通知規則的標籤清單 (AWS CLI)

1. 在終端機或命令提示字元中，執行 list-tags-for-resource 命令來檢視所指定通知規則 ARN 的所有標籤。

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

2. 如果成功，此命令傳回的輸出會類似如下。

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

編輯通知規則

您可以編輯通知規則以變更其名稱、傳送通知所針對的事件、詳細資訊類型，或通知傳送到的一或多個目標。您可以使用開發人員工具主控台或 AWS CLI 來編輯通知規則。

編輯通知規則 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。

3. 在 Notification rules (通知規則) 中，檢閱您目前登入的 AWS 區域 中為 AWS 帳戶中的資源所設定的規則。使用選擇器變更 AWS 區域。
4. 從清單中選擇規則，然後選擇 Edit (編輯)。進行變更，然後選擇 Submit (提交)。

編輯通知規則 (AWS CLI)

1. 在終端機或命令提示字元中，執行 describe-notification-rule 命令，以檢視通知規則的結構。
2. 執行 update-notification rule 命令來產生 JSON 基本結構，然後儲存到檔案。

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

您可以將檔案命名為任何您想要的名稱。在此範例中，檔案為 *update.json*。

3. 在純文字編輯器中開啟 JSON 檔案，並對規則進行變更。

以下範例顯示名為 **MyNotificationRule** 的通知規則，用於 AWS 帳戶 (ID 為 *123456789012*) 中的 *MyDemoRepo* 儲存庫。建立分支和標籤時，通知會傳送到名為 *MyNotificationTopic* 的 Amazon SNS 主題。規則名稱變更為 *MyNewNotificationRule*。

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

儲存檔案。

4. 在終端機或命令列中，再次執行 update-notification-rule 命令，使用您剛編輯的檔案更新通知規則。

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

5. 如果成功，此命令會傳回通知規則的 Amazon Resource Name (ARN)，如下所示。

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

從通知規則移除標籤 (AWS CLI)

1. 在終端機或命令提示字元中，執行 untag-resource 命令。例如，以下命令會移除包含 *Team* 名稱的標籤。

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

2. 如果成功，此命令不會傳回任何內容。

另請參閱

- 新增或移除通知規則的目標
- 啟用或停用通知規則的通知
- 事件

啟用或停用通知規則的通知

當您建立通知規則時，預設會啟用通知。您不需要為了防止傳送通知而刪除規則。只要變更其通知狀態即可。

變更通知規則的通知狀態 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。

3. 在 Notification rules (通知規則) 中，檢閱您目前登入的 AWS 區域 中為 AWS 帳戶中的資源所設定的規則。使用選擇器變更 AWS 區域。
4. 尋找您要啟用或停用的通知規則，然後選擇它以顯示其詳細資訊。
5. 在 Notification status (通知狀態) 中，選擇滑桿以變更規則狀態：
   - Sending notifications (傳送通知)：這是預設值。
   - Notifications paused (通知暫停)：不會將通知傳送至指定的目標。

變更通知規則的通知狀態 (AWS CLI)

1. 按照編輯通知規則 (AWS CLI) 中的步驟，取得通知規則的 JSON。
2. 將 `Status` 欄位編輯為 `ENABLED` (預設值) 或 `DISABLED` (無通知)，然後執行 update-notification-rule 命令以變更狀態。

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

刪除通知規則

一個資源只能設定 10 個通知規則，因此請考慮刪除不再需要的規則。您可以使用開發人員工具主控台或 AWS CLI 來刪除通知規則。

> **Note**
> 刪除通知規則之後就無法復原，但您可以重新建立通知規則。刪除通知規則並不會刪除目標。

刪除通知規則 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。
3. 在 Notification rules (通知規則) 中，檢閱您目前登入的 AWS 區域 中為 AWS 帳戶中的資源所設定的規則。使用選擇器變更 AWS 區域。
4. 選擇通知規則，然後選擇 Delete (刪除)。
5. 輸入 **`delete`**，然後選擇 Delete (刪除)。

刪除通知規則 (AWS CLI)

1. 在終端機或命令提示字元中，執行 delete-notification-rule 命令，並指定通知規則的 ARN。

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

2. 如果成功，此命令會傳回所刪除通知規則的 ARN，如下所示。

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

使用通知規則目標

通知規則目標是一個目的地，可定義當通知規則的事件條件符合時，您要傳送通知的位置。您可以在為 Slack 或 Microsoft Team 頻道設定的 Amazon SNS 主題和 AWS Chatbot 用戶端之間進行選擇。您可以將 Amazon SNS 主題建立為目標，做為建立通知規則的一部分 (建議)。您也可以選擇與通知規則相同 AWS 區域中的現有 Amazon SNS 主題，但必須使用必要的政策進行設定。如果您選擇使用 AWS Chatbot 用戶端做為目標，則必須先在 AWS Chatbot 中建立該用戶端。

如果您想要擴展通知的範圍，可以手動設定通知與 AWS Chatbot 之間的整合，以便將通知傳送到 Amazon Chime 聊天室。然後，您可以選擇為 AWS Chatbot 用戶端設定的 Amazon SNS 主題，做為通知規則的目標。如需更多詳細資訊，請參閱 將通知與 AWS Chatbot 和 Amazon Chime 整合。

您可以使用開發人員工具主控台或 AWS CLI 來管理通知目標。您可以使用主控台或 AWS CLI 建立 Amazon SNS 主題和 AWS Chatbot 用戶端，並將其設為目標。您也可以在設為目標的 Amazon SNS 主題和 AWS Chatbot 之間設定整合。如此一來，您就可以將通知傳送至 Amazon Chime 聊天室。如需更多詳細資訊，請參閱 設定通知和 AWS Chatbot 之間的整合。

主題

- 建立或設定通知規則目標
- 檢視通知規則目標
- 新增或移除通知規則的目標
- 刪除通知規則目標

建立或設定通知規則目標

通知規則目標是為 Slack 或 Microsoft Team 頻道設定的 Amazon SNS 主題或 AWS Chatbot 用戶端。

您必須先建立 AWS Chatbot 用戶端，才能選取用戶端做為目標。當您選擇 AWS Chatbot 用戶端做為通知規則的目標時，系統會為該 AWS Chatbot 用戶端設定 Amazon SNS 主題，其中包含將通知傳送至 Slack 或 Microsoft Team 頻道所需的所有政策。您不需要為 AWS Chatbot 用戶端設定任何現有的 Amazon SNS 主題。

建立通知規則時，您可以在開發人員工具主控台中建立 Amazon SNS 通知規則目標。將會為您套用允許通知傳送至該主題的政策。這是為通知規則建立目標的最簡單方法。如需更多詳細資訊，請參閱 建立通知規則。

如果您使用現有 Amazon SNS 主題，必須為它設定存取政策，以允許資源將通知傳送至該主題。如需範例，請參閱 設定通知的 Amazon SNS 主題。

> **Note**
>
> 如果您要使用現有 Amazon SNS 主題而非建立新主題，請在 Targets (目標) 中選擇其 ARN。請確定主題具有適當的存取政策，而且訂閱者清單只包含允許查看資源相關資訊的使用者。如果 Amazon SNS 主題在 2019 年 11 月 5 日之前用於 CodeCommit 通知，該主題將包含允許 CodeCommit 發佈至該主題的政策，其中包含 AWS CodeStar Notifications 所需許可以外的不同許可。不建議使用這些主題。如果您希望針對該體驗建立一個政策，除了已存在的政策之外，您還必須為 AWS CodeStar Notifications 新增必要的政策。如需詳細資訊，請參閱設定通知的 Amazon SNS 主題及了解通知內容和安全性。

如果您想要擴展通知的範圍，可以手動設定通知與 AWS Chatbot 之間的整合，以便將通知傳送到 Amazon Chime 聊天室。如需詳細資訊，請參閱目標及將通知與 AWS Chatbot 和 Amazon Chime 整合。

設定現有 Amazon SNS 主題做為通知規則目標 (主控台)

1. 登入 AWS Management Console，並在 https://console.aws.amazon.com/sns/v3/home 開啟 Amazon SNS 主控台。
2. 於導覽列中，選擇 Topics (主題)。選擇主題，然後選擇 Edit (編輯)。
3. 展開 Access policy (存取政策)，然後選擇 Advanced (進階)。
4. 在 JSON 編輯器中，為政策新增下列陳述式。包含主題 ARN、AWS 區域、AWS 帳戶 ID 和主題名稱。

```
 {
      "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"
    }
```

該政策陳述式應如以下範例所示。

```
{
  "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. 選擇 Save changes (儲存變更)。
6. 在 Subscriptions (訂閱) 中，檢閱主題訂閱者的清單。適當地新增、編輯或刪除此通知規則目標的訂閱者。請確定訂閱者清單只包含允許查看資源相關資訊的使用者。如需更多詳細資訊，請參閱了解通知內容和安全性。

使用 Slack 建立 AWS Chatbot 用戶端做為目標

1. 請依照 AWS Chatbot 管理員指南中的在 Slack 中設定 AWS Chatbot 的指示操作。當您執行這項操作時，請考慮下列選項，以便與通知進行最佳整合：
   - 建立 IAM 角色時，請考慮選擇一個容易識別該角色的角色名稱 (例如 **AWSCodeStarNotifications-Chatbot-Slack-Role**)。這有助於識別未來角色的用途。
   - 在 SNS topics (SNS 主題) 中，您不需要選擇主題或 AWS 區域。當您選擇 AWS Chatbot 用戶端做為目標時，系統會為 AWS Chatbot 用戶端建立具有所有必要許可的 Amazon SNS 主題，並將其設為通知規則建立程序的一部分。
2. 完成用戶端建立程序。然後，您可以在建立通知規則時選擇此用戶端做為目標。如需更多詳細資訊，請參閱 建立通知規則。

> **Note**
>
> 設定 Amazon SNS 主題之後，請勿將該主題從 AWS Chatbot 用戶端移除。這樣做會讓通知無法傳送至 Slack。

使用 Microsoft Teams 頻道建立 AWS Chatbot 用戶端做為目標

1. 請依照《AWS Chatbot 管理員指南》中的在 Microsoft Teams 中設定 AWS Chatbot 的指示操作。當您執行這項操作時，請考慮下列選項，以便與通知進行最佳整合：

   - 建立 IAM 角色時，請考慮選擇一個容易識別該角色的角色名稱 (例如 **AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role**)。這有助於識別未來角色的用途。
   - 在 SNS topics (SNS 主題) 中，您不需要選擇主題或 AWS 區域。當您選擇 AWS Chatbot 用戶端做為目標時，系統會為 AWS Chatbot 用戶端建立具有所有必要許可的 Amazon SNS 主題，並將其設為通知規則建立程序的一部分。
2. 完成用戶端建立程序。然後，您可以在建立通知規則時選擇此用戶端做為目標。如需更多詳細資訊，請參閱 建立通知規則。

> **Note**
>
> 設定 Amazon SNS 主題之後，請勿將該主題從 AWS Chatbot 用戶端移除。這樣做會讓通知無法傳送至 Microsoft Teams。

檢視通知規則目標

您可以使用開發人員工具主控台 (而非 Amazon SNS 主控台) 來檢視 AWS 區域中所有資源的所有通知規則目標。您也可以檢視通知規則目標的詳細資訊。

檢視通知規則目標 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。
3. 在 Notification rule targets (通知規則目標)中，檢閱您目前登入之 AWS 區域 中 AWS 帳戶 中的通知規則所使用的目標清單。使用選擇器變更 AWS 區域。如果目標狀態顯示為 Unreachable (無法連線)，您可能需要進行調查。如需更多詳細資訊，請參閱 疑難排解 。

檢視通知規則目標清單 (AWS CLI)

1. 在終端機或命令提示字元中，執行 list-targets 命令來檢視所指定 AWS 區域的所有通知規則目標的清單：

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

2. 如果成功，此命令會傳回 AWS 區域中每個通知規則的 ID 和 ARN，如下所示：

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

新增或移除通知規則的目標

您可以編輯通知規則，以變更傳送通知的一或多個目標。您可以使用開發人員工具主控台或 AWS CLI 來變更通知規則的目標。

變更通知規則的目標 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。
3. 在 Notification rules (通知規則)中，檢閱您目前登入之 AWS 區域 中 AWS 帳戶中為您的資源所設定的規則清單。使用選擇器變更 AWS 區域。

4. 選擇規則，然後選擇 Edit (編輯)。
5. 在 Targets (目標) 中，執行下列其中一個動作：
   - 若要新增另一個目標，請選擇新增目標，然後從清單中選擇您要新增的 Amazon SNS 主題或 AWS Chatbot (Slack) 或是 AWS Chatbot (Microsoft Teams) 用戶端。您也可以選擇 Create SNS topic (建立 SNS 主題) 來建立主題並新增為目標。一個通知規則最多可有 10 個目標。
   - 若要移除目標，請選擇您要移除的目標旁邊的 Remove target (移除目標)。
6. 選擇 Submit (提交)。

新增目標到通知規則 (AWS CLI)

1. 在終端機或命令提示字元中，執行 subscribe 命令以新增目標。例如，以下命令會新增 Amazon SNS 主題做為通知規則的目標。

```
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. 如果成功，此命令會傳回所更新通知規則的 ARN，如下所示。

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

從通知規則移除目標 (AWS CLI)

1. 在終端機或命令提示字元中，執行 unsubscribe 命令以移除目標。例如，以下命令會移除做為通知規則目標的 Amazon SNS 主題。

```
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. 如果成功，此命令會傳回所更新通知規則的 ARN 和所移除目標的相關資訊，如下所示。

```
{
```

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

另請參閱

- 編輯通知規則
- 啟用或停用通知規則的通知

刪除通知規則目標

如果不再需要目標，可以將其刪除。一個資源只能設定 10 個通知規則目標，因此刪除不需要的目標有助於為可能要新增到該通知規則的其他目標騰出空間。

> **Note**
>
> 刪除通知規則目標會將目標從設定將其做為目標的所有通知規則中移除，但不會刪除目標本身。

刪除通知規則目標 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 在導覽列中，展開 Settings (設定)，然後選擇 Notification rules (通知規則)。
3. 在 Notification rule targets (通知規則目標)中，檢閱您目前登入之 AWS 區域 中 AWS 帳戶中為您的資源所設定的目標清單。使用選擇器變更 AWS 區域。
4. 選擇通知規則目標，然後選擇 Delete (刪除)。
5. 輸入 **`delete`**，然後選擇 Delete (刪除)。

刪除通知規則目標 (AWS CLI)

1. 在終端機或命令提示字元中，執行 delete-target 命令，並指定目標的 ARN。例如，下列命令會刪除使用 Amazon SNS 主題的目標。

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

2. 如果成功，此命令不會傳回任何內容。如果失敗，此命令會傳回錯誤。最常見的錯誤是主題為一或多個通知規則的目標。

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

您可以使用 `--force-unsubscribe-all` 參數，將目標從設定以其為目標的所有通知規則中移除，然後刪除該目標。

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

設定通知和 AWS Chatbot 之間的整合

AWS Chatbot 是一項 AWS 服務，可讓 DevOps 和軟體開發團隊使用 Amazon Chime 聊天室、Slack 頻道和 Microsoft Team 頻道，並監控及回應 AWS 雲端 中的操作事件。您可以設定通知規則目標與 AWS Chatbot 之間的整合，讓事件相關通知顯示在您選擇的 Amazon Chime 聊天室、Slack 頻道和 Microsoft Team 頻道中。如需詳細資訊，請參閱 AWS Chatbot 文件。

在設定與 AWS Chatbot 的整合之前，您必須先設定通知規則和規則目標。如需詳細資訊，請參閱設定及建立通知規則。您也必須在 AWS Chatbot 中設定 Slack 頻道、Microsoft Teams 頻道或 Amazon Chime 聊天室。如需詳細資訊，請參閱這些服務的文件。

主題

- 為 Slack 頻道設定 AWS Chatbot 用戶端
- 為 Microsoft Teams 頻道設定 AWS Chatbot 用戶端
- 為 Slack 或 Amazon Chime 手動設定用戶端

為 Slack 頻道設定 AWS Chatbot 用戶端

您可以建立使用 AWS Chatbot 用戶端做為目標的通知規則。如果您為 Slack 頻道建立用戶端，則可以直接使用此用戶端，做為建立通知規則之工作流程中的目標。這是設定出現在 Slack 頻道中的通知最簡單的方法。

使用 Slack 建立 AWS Chatbot 用戶端做為目標

1. 請依照 AWS Chatbot 管理員指南中的在 Slack 中設定 AWS Chatbot 的指示操作。當您執行這項操作時，請考慮下列選項，以便與通知進行最佳整合：

   - 建立 IAM 角色時，請考慮選擇一個容易識別該角色的角色名稱 (例如 **AWSCodeStarNotifications-Chatbot-Slack-Role**)。這有助於識別未來角色的用途。
   - 在 SNS topics (SNS 主題) 中，您不需要選擇主題或 AWS 區域。當您選擇 AWS Chatbot 用戶端做為目標時，系統會為 AWS Chatbot 用戶端建立具有所有必要許可的 Amazon SNS 主題，並將其設為通知規則建立程序的一部分。

2. 完成用戶端建立程序。然後，您可以在建立通知規則時選擇此用戶端做為目標。如需更多詳細資訊，請參閱 建立通知規則。

> **Note**
>
> 設定 Amazon SNS 主題之後，請勿將該主題從 AWS Chatbot 用戶端移除。這樣做會讓通知無法傳送至 Slack。

為 Microsoft Teams 頻道設定 AWS Chatbot 用戶端

您可以建立使用 AWS Chatbot 用戶端做為目標的通知規則。如果您為 或 Microsoft Team 頻道建立用戶端，則可以直接使用此用戶端，做為建立通知規則之工作流程中的目標。這是設定出現在 Microsoft Teams 頻道中的通知最簡單的方法。

使用 Microsoft Teams 頻道建立 AWS Chatbot 用戶端做為目標

1. 請依照《AWS Chatbot 管理員指南》中的在 Microsoft Teams 中設定 AWS Chatbot 的指示操作。當您執行這項操作時，請考慮下列選項，以便與通知進行最佳整合：

   - 建立 IAM 角色時，請考慮選擇一個容易識別該角色的角色名稱 (例如 **AWSCodeStarNotifications-Chatbot-Microsoft-Teams-Role**)。這有助於識別未來角色的用途。
   - 在 SNS topics (SNS 主題) 中，您不需要選擇主題或 AWS 區域。當您選擇 AWS Chatbot 用戶端做為目標時，系統會為 AWS Chatbot 用戶端建立具有所有必要許可的 Amazon SNS 主題，並將其設為通知規則建立程序的一部分。

2. 完成用戶端建立程序。然後，您可以在建立通知規則時選擇此用戶端做為目標。如需更多詳細資訊，請參閱 建立通知規則。

> **Note**
>
> 設定 Amazon SNS 主題之後，請勿將該主題從 AWS Chatbot 用戶端移除。這樣做會讓通知無法傳送至 Microsoft Teams。

為 Slack 或 Amazon Chime 手動設定用戶端

您可以選擇在通知和 Slack 或 Amazon Chime 之間直接建立整合。這是為 Amazon Chime 聊天室設定通知的唯一可用方法。手動設定此整合時，您會建立一個 AWS Chatbot 用戶端，該用戶端使用您先前設為通知規則目標的 Amazon SNS 主題。

手動將通知與 AWS Chatbot 和 Slack 整合

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 選擇 Settings (設定)，然後選擇 Notification settings (通知設定)。
3. 在 Notification rule targets (通知規則目標) 中，尋找並複製目標。

> **Note**
>
> 您可以設定多個通知規則使用與其目標相同的 Amazon SNS 主題。這可協助您整合簡訊，但如果訂閱清單專屬於一個通知規則或資源，則會產生意外後果。

4. 前往 https://console.aws.amazon.com/chatbot/ 開啟 AWS Chatbot 主控台。
5. 選擇 Configure new client (設定新用戶端)，然後選擇 Slack。
6. 選擇 Configure (設定)。
7. 登入您的 Slack 工作空間。
8. 如果系統提示您確認選擇項目，請選擇 Allow (允許)。
9. 選擇 Configure new channel (設定新頻道)。
10. 在 Configuration details (組態詳細資訊) 中的 Configuration name (組態名稱) 中，輸入您的用戶端名稱。此名稱會在您建立通知規則時，顯示在 AWS Chatbot (Slack) 目標類型的可用目標清單中。
11. 在 Configure Slack Channel (設定 Slack 頻道) 的 Channel type (頻道類型) 中，根據您要整合的頻道類型來選擇 Public (公有) 或 Private (私有)。

    - 在 Public channel (公有頻道) 中，從清單中選擇 Slack 頻道的名稱。

- 在 Private channel ID (私有頻道 ID) 中，輸入頻道代碼或 URL。

12. 在 IAM permissions (IAM 許可) 中的 Role (角色) 裡，選擇 Create an IAM role using a (使用範本建立 IAM 角色)。在 Policy templates (政策範本) 中，選擇 Notification permissions (通知許可)。在 Role name (角色名稱) 中，輸入此角色的名稱 (例如 **`AWSCodeStarNotifications-Chatbot-Slack-Role`**)。在 Policy templates (政策範本) 中，選擇 Notification permissions (通知許可)。
13. 在 SNS topics (SNS 主題) 的 SNS Region (SNS 區域) 中，選擇您建立通知規則目標的 AWS 區域。在 SNS topics (SNS 主題) 中，選擇您設定為通知規則目標的 Amazon SNS 主題名稱。

> **Note**
>
> 如果您要建立使用此用戶端做為目標的通知規則，則不需要執行此步驟。

14. 選擇 Configure (設定)。

> **Note**
>
> 如果您已搭配私有頻道設定了整合，則您必須先邀請 AWS Chatbot 加入該頻道後，才能在該頻道中看到通知。如需詳細資訊，請參閱 AWS Chatbot 文件。

15. (選用) 若要測試整合，請在資源中進行變更，以符合設定為使用 Amazon SNS 主題做為其目標之通知規則的事件類型。例如，如果您有一個通知規則設定為在對提取請求進行註解時傳送通知，請對提取請求進行註解，然後在瀏覽器中監看 Slack 頻道，以查看通知何時出現。

將通知與 AWS Chatbot 和 Amazon Chime 整合

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。
2. 選擇 Settings (設定)，然後選擇 Notification settings (通知設定)。
3. 在 Notification rule targets (通知規則目標) 中，尋找並複製目標。

> **Note**
>
> 您可以設定多個通知規則使用與其目標相同的 Amazon SNS 主題。這可協助您整合簡訊，但如果訂閱清單是針對一個通知規則或資源，也會產生意外後果。

4. 在 Amazon Chime 中，開啟您要設定進行整合的聊天室。

5. 選擇右上角的齒輪圖示，然後選擇 Manage webhooks (管理 Webhook)。
6. 在 Manage webhooks (管理 Webhook) 對話方塊中，選擇 New (新增)，輸入 Webhook 的名稱，然後選擇 Create (建立)。
7. 確認 Webhook 出現，然後選擇 Copy webhook URL (複製 Webhook URL)。
8. 前往 https://console.aws.amazon.com/chatbot/ 開啟 AWS Chatbot 主控台。
9. 選擇 Configure new client (設定新用戶端)，然後選擇 Amazon Chime。
10. 在 Configuration details (組態詳細資訊) 中的 Configuration name (組態名稱) 中，輸入您的用戶端名稱。
11. 在 Webhook URL 中，貼上 URL。在 Webhook description (Webhook 描述) 中，提供選用描述。
12. 在 IAM permissions (IAM 許可) 中的 Role (角色) 裡，選擇 Create an IAM role using a (使用範本建立 IAM 角色)。在 Policy templates (政策範本) 中，選擇 Notification permissions (通知許可)。在 Role name (角色名稱) 中，輸入此角色的名稱 (例如 **AWSCodeStarNotifications-Chatbot-Chime-Role**)。
13. 在 SNS topics (SNS 主題) 的 SNS Region (SNS 區域) 中，選擇您建立通知規則目標的 AWS 區域。在 SNS topics (SNS 主題) 中，選擇您設定為通知規則目標的 Amazon SNS 主題名稱。
14. 選擇 Configure (設定)。
15. (選用) 若要測試整合，請在資源中進行變更，以符合設定為使用 Amazon SNS 主題做為其目標之通知規則的事件類型。例如，如果您有一個通知規則設定為在對提取請求進行註解時傳送通知，請對提取請求加上註解，然後在瀏覽器中監看 Amazon Chime 聊天室，查看通知何時出現。

使用 AWS CloudTrail 記錄 AWS CodeStar Notifications 通知 API 呼叫

AWS CodeStar Notifications 與 AWS CloudTrail 整合，該服務提供由使用者、角色或 AWS 服務所採取行動的記錄。CloudTrail 會擷取通知的 API 呼叫當作事件。擷取的呼叫包括從開發人員工具主控台進行的呼叫，以及對AWS CodeStar Notifications API 作業的程式碼呼叫。如果您建立追蹤記錄，就可以將 CloudTrail 事件持續交付到 Amazon S3 儲存貯體，包括通知的事件。即使您未設定追蹤，依然可以透過 CloudTrail 主控台中的 Event history (事件歷史記錄) 檢視最新事件。您可利用 CloudTrail 所收集的資訊來判斷向 AWS CodeStar Notifications 發出的請求，以及發出請求的 IP 地址、人員、時間和其他詳細資訊。

如需詳細資訊，請參閱《AWS CloudTrail 使用者指南》。

CloudTrail 中的 AWS CodeStar Notifications 資訊

當您建立帳戶時，系統即會在 AWS 帳戶 中啟用 CloudTrail。當 AWS CodeStar Notifications 中發生活動時，該活動會記錄在 CloudTrail 事件中，其他 AWS 服務事件則記錄於 Event history (事件歷程記錄) 中。您可以檢視、搜尋和下載 AWS 帳戶 的最新事件。如需詳細資訊，請參閱使用 CloudTrail 事件歷史記錄檢視事件。

若要持續記錄 AWS 帳戶 的事件 (包括 AWS CodeStar Notifications 事件)，請建立追蹤記錄。追蹤能讓 CloudTrail 將日誌檔交付至 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中所收集的事件資料。如需詳細資訊，請參閱下列內容：

- 建立追蹤的概觀
- CloudTrail 支援的服務和整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 接收多個區域的 CloudTrail 日誌檔案和接收多個帳戶的 CloudTrail 日誌檔案

CloudTrail 會記錄所有 AWS CodeStar Notifications 動作，並記載於 *AWS CodeStar Notifications API ##*中。例如，對 `CreateNotificationRule`、`Subscribe` 和 `ListEventTypes` 動作發出的呼叫會在 CloudTrail 記錄檔案中產生項目。

每一筆事件或日誌項目都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是否透過根或 AWS Identity and Access Management (IAM) 使用者憑證來提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 該請求是否由另一項 AWS 服務提出。

如需詳細資訊，請參閱 CloudTrail userIdentity 元素。

了解 日誌檔案項目

追蹤是一種組態，可讓事件以日誌檔案的形式交付至您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌項目。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。CloudTrail 日誌檔並非依公有 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

以下範例顯示的 CloudTrail 日誌項目示範如何建立通知規則，包括 `CreateNotificationRule` 和 `Subscribe` 動作。

> **Note**
>
> 通知日誌檔案項目中的某些事件可能來自服務連結角色 AWSServiceRoleForCodeStarNotifications。


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

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type":"IAMUser",
        "principalId":"AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/Mary_Major",
        "accountId":"123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName":"Mary_Major"
    },
    "eventTime": "2019-10-07T21:34:41Z",
    "eventSource": "events.amazonaws.com",
    "eventName": "Subscribe",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "codestar-notifications.amazonaws.com",
    "userAgent": "codestar-notifications.amazonaws.com",
    "requestParameters": {
        "targets": [
            {
                "arn": "arn:aws:codestar-notifications:us-east-1:::",
                "id": "codestar-notifications-events-target"
            }
        ],
        "rule": "awscodestarnotifications-rule"
    },
    "responseElements": {
        "failedEntryCount": 0,
        "failedEntries": []
    },
    "requestID": "9466cbda-EXAMPLE",
    "eventID": "2f79fdad-EXAMPLE",
    "eventType": "AwsApiCall",
    "apiVersion": "2015-10-07",
    "recipientAccountId": "123456789012"
}
```


疑難排解

以下資訊可能有助於對通知的常見問題進行疑難排解。

主題

- 我嘗試在資源上建立通知規則時收到許可錯誤

- 我無法檢視通知規則
- 我無法建立通知規則
- 我收到無法存取資源的通知
- 我沒有收到 Amazon SNS 通知
- 我收到重複的事件通知
- 我想了解為什麼通知目標狀態顯示為無法連線
- 我想要提高通知和資源的限額

我嘗試在資源上建立通知規則時收到許可錯誤

確定您有足夠的許可。如需更多詳細資訊，請參閱 身分型政策範例。

我無法檢視通知規則

問題：您進入開發人員工具主控台後，在 Settings (設定) 底下選擇 Notifications (通知) 時，出現許可錯誤。

可能的修正方式：您可能沒有檢視通知所需的許可。雖然 AWS 開發人員工具服務 (例如 CodeCommit 和 CodePipeline) 的大部分受管政策都包含通知的許可，但目前不支援通知的服務並不包含檢視通知的許可。或者，您可能已將自訂政策套用至 IAM 使用者或角色，而該政策不允許您檢視通知。如需更多詳細資訊，請參閱 身分型政策範例。

我無法建立通知規則

您可能沒有建立通知規則所需的許可。如需更多詳細資訊，請參閱 身分型政策範例。

我收到無法存取資源的通知

當您建立通知規則並新增目標時，通知功能不會驗證收件者是否具有資源的存取權。您可能會收到有關無法存取資源的通知。如果您無法移除自己，請要求從目標的訂閱清單中移除。

我沒有收到 Amazon SNS 通知

若要疑難排解 Amazon SNS 主題的問題，請檢查下列各項：

- 請確定 Amazon SNS 主題是在與通知規則相同的 AWS 區域中建立的。
- 請確定您的電子郵件別名已訂閱正確的主題，而且您已確認訂閱。如需詳細資訊，請參閱讓端點訂閱 Amazon SNS 主題。

- 確認主題政策已編輯為允許 AWS CodeStar Notifications 將通知推送至該主題。主題政策應該包含類似以下的陳述式：

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

如需更多詳細資訊，請參閱 設定通知的 Amazon SNS 主題。

我收到重複的事件通知

以下是接收多個通知的最常見原因：

- 已為資源設定多個包含相同事件類型的通知規則，而且您已訂閱的某些 Amazon SNS 主題是這些規則的目標。若要解決此問題，請取消訂閱其中一個主題，或編輯通知規則以移除重複項目。
- 一或多個通知規則目標已與 AWS Chatbot 整合，而且您在電子郵件收件匣和 Slack 頻道、Microsoft Teams 頻道或 Amazon Chime 聊天室中接收通知。若要解決此問題，請考慮讓您的電子郵件地址取消訂閱做為規則目標的 Amazon SNS 主題，然後使用 Slack 頻道和 Microsoft Team 頻道來檢視通知。

我想了解為什麼通知目標狀態顯示為無法連線

目標有兩種可能的狀態：Active (作用中) 和 Unreachable (無法連線)。Unreachable (無法連線) 表示通知已傳送至目標，且傳遞未成功。通知會繼續傳送至該目標，如果成功，狀態會重設為 Active (作用中)。

通知規則的目標可能會因為以下其中一個原因而無法使用：

- 資源 (Amazon SNS 主題或 AWS Chatbot 用戶端) 已遭刪除。為通知規則選擇另一個目標。
- 已加密 Amazon SNS 主題，但缺少加密主題所需的政策，或已刪除 AWS KMS 金鑰。如需更多詳細資訊，請參閱 設定通知的 Amazon SNS 主題。
- Amazon SNS 主題缺少通知所需的政策。除非有政策，否則無法將通知傳送至 Amazon SNS 主題。如需更多詳細資訊，請參閱 設定通知的 Amazon SNS 主題。
- 目標 (Amazon SNS 或 AWS Chatbot) 的支援服務可能發生問題。

我想要提高通知和資源的限額

目前您無法變更任何限額。請參閱通知的配額。

通知的配額

下表列出開發人員工具主控台中通知的配額 (也稱為限額)。如需可變更限額的資訊，請參閱 AWS Service Quotas。

資源	預設限制
AWS 帳戶中的通知規則數目上限	1000
通知規則的目標數目上限	10
資源的通知規則數目上限	10

什麼是連線？

您可以使用「開發人員工具」主控台中的連線功能，將 AWS 資源 (例如 AWS CodePipeline 外部程式碼儲存庫) 連線。此功能有自己的 API，即AWS CodeStar 連接 API 參考。每個連接都是您可以提供給 AWS 服務以連接到第三方存儲庫的資源，例如 BitBucket。例如，您可以在中新增連線，以 CodePipeline 便在對協力廠商程式碼儲存庫進行程式碼變更時觸發管道。每個連線都有名稱，且與用來參考連線的唯一 Amazon Resource Name (ARN) 相關聯。

我能用連線做什麼？

您可以在開發人員工具中使用連線功能，將第三方供應商資源與您的 AWS 資源整合，包括：

- Connect 線至第三方供應商 (例如 Bitbucket)，並使用第三方連線做為與您的資源整合的來 AWS 源，例如 CodePipeline。
- 在第三方 CodePipeline 供應商的 CodeBuild 建置專案、 CodeDeploy 應用程式和管道中，統一管理跨資源連線的存取。
- 在堆疊範本中使用連線 ARN 來 CodeBuild 建置專案、 CodeDeploy應用程式和管道 CodePipeline，而不需要參考已儲存的密碼或參數。

我可以為哪些第三方供應商建立連線？

連線可以將您的 AWS 資源與下列第三方存放庫建立關聯：

- Bitbucket Cloud
- GitHub
- GitHub 企業雲
- GitHub 企業伺服器
- GitLab
- GitLab 自我管理安裝 (適用於企業版或社群版)

如需連線工作流程的概觀，請參閱「建立或更新連線的工作流程」。

針對雲端提供者類型 (例如) 建立連線的步驟 GitHub，與已安裝的提供者類型 (例如 GitHub 企業伺服器) 的步驟不同。如需為各種供應商類型建立連線的高階步驟，請參閱「使用連線」。

> **Note**
>
> 要在歐洲（米蘭）使用連接 AWS 區域，您必須：
>
> 1. 安裝區域特定的應用程式
> 2. 啟用區域
>
> 此區域特定的應用程式支援歐洲 (米蘭) 區域中的連線。它會在第三方供應商網站上發佈，並且它會與支援其他區域連線的現有應用程式分開。透過安裝此應用程式，您授權第三方提供商僅使用該區域服務來共用您的資料，並且您可以透過解除安裝該應用程式來隨時撤銷許可。除非您啟用區域，否則服務不會處理或儲存您的資料。啟用此區域，即表示您授予我們服務許可來處理和儲存您的資料。

即使未啟用該區域，如果仍已安裝區域特定的應用程式，第三方供應商仍然可以使用我們的服務來共用您的資料，因此請確保在停用該區域後解除安裝該應用程式。如需詳細資訊，請參閱啟用區域。

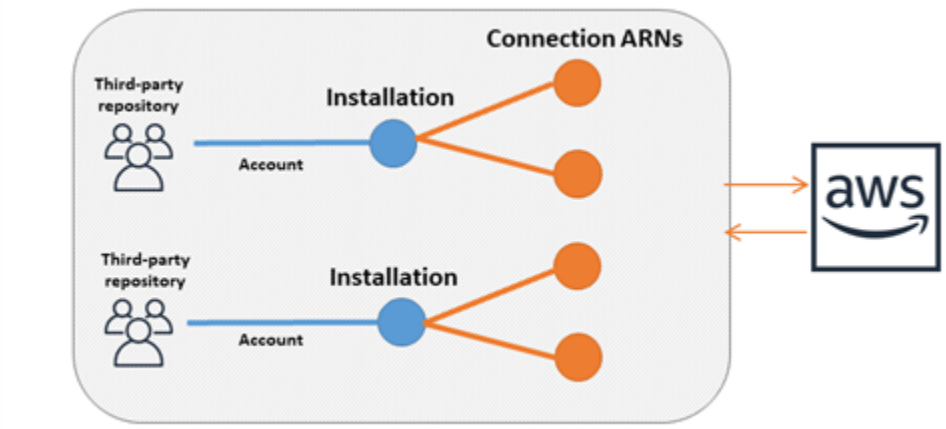
什麼 AWS 服務 與連接集成？

您可以使用連線整合第三方儲存庫與其他 AWS 服務。若要檢視連線的服務整合，請參閱 與 AWS CodeStar Connections 整合的產品和服務。

連線如何運作？

您必須先在第三方帳戶上安裝 AWS 身分驗證應用程式或提供存取權，才能建立連線。安裝連線之後，更新即可使用此安裝。建立連線時，您可以提供第三方帳戶中 AWS 資源的存取權。這可讓連線代表您的 AWS 資源存取協力廠商帳戶中的內容，例如來源儲存庫。然後，您可以與其他連接共享該連接，AWS 服務 以在資源之間提供安全的 OAuth 連接。

如果您想要建立與已安裝提供者類型 (例如 GitHub Enterprise Server) 的連線，請先使用 AWS Management Console.



連線是由建立連線 AWS 帳戶 的擁有者。連線的識別方式為包含連線 ID 的 ARN。連線 ID 是無法變更或重新映射的 UUID。刪除並重新建立連線會產生新的連線 ID，因此會有新的連線 ARN。這表示連線 ARN 絕不會重複使用。

新建立的連線處於 Pending 狀態。需要第三方交握 (OAuth 流程) 程序才能完成連線的設定，也就是從 Pending 變為 Available 狀態。完成此操作之後，連接就可以Available與 AWS 服務一起使用，例如 CodePipeline。

新建立的主機處於 Pending 狀態。需要第三方註冊程序才能完成主機的設定，也就是從 Pending 變為 Available 狀態。完成此操作之後，主機會變為 Available，且可用於與安裝式供應商類型的連線。

如需連線工作流程的概觀，請參閱「建立或更新連線的工作流程」。如需為已安裝供應商建立主機工作流程的概觀，請參閱 建立或更新主機的工作流程。如需為各種供應商類型建立連線的高階步驟，請參閱「使用連線」。

AWS CodeStar 連線中的全球資源

連線是全球資源，這表示會跨所有 AWS 區域複寫資源。

雖然連線 ARN 格式會反映建立時的區域名稱，不過資源不受限於任何區域。建立連線資源的區域是控制連線資源資料更新的區域。控制連線資源資料更新的 API 作業範例包括：建立連線、更新安裝、刪除連線或標記連線。

連線的主機資源不是全球可用的資源。您只能在建立主機資源的區域中使用資源。

- 您只需建立一次連線，然後就可以在任何 AWS 區域中使用。
- 如果建立連線的區域發生問題，會影響控制連線資源資料的 API，但您仍可成功在其他各個區域中使用該連線。
- 當您在主控台或 CLI 中列出連線資源時，清單會顯示所有區域中與您帳戶相關聯的所有連線資源。
- 當您在主控台或 CLI 中列出主機資源時，清單只會顯示所選區域中與您帳戶相關聯的主機資源。
- 使用 CLI 列出或檢視與相關主機資源的連線時，無論設定的 CLI 區域為何，輸出都會傳回主機 ARN。

建立或更新主機的工作流程

建立已安裝供應商的連線之前，需先建立主機。

主機可能有以下狀態：

- Pending - pending 主機是已建立的主機，必須先設定 (移至 available) 才能使用。
- Available - 您可以使用或將 available 主機傳遞給連接。

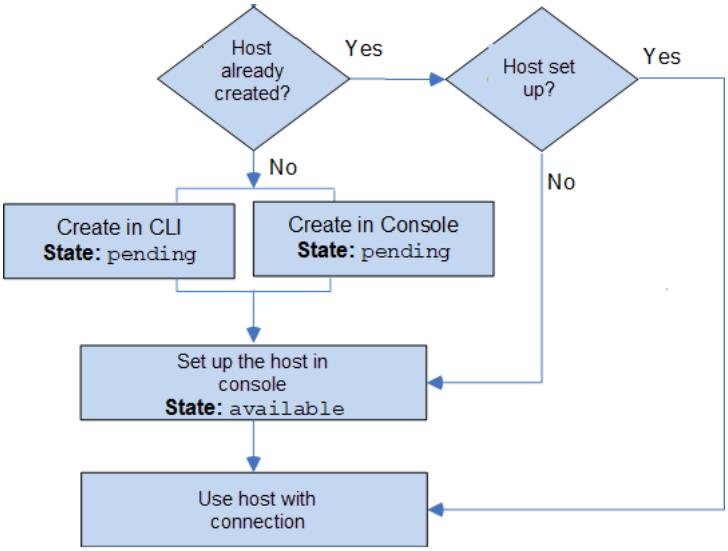
工作流程：透過 CLI、SDK 或 AWS CloudFormation建立或更新主機：

您可以使用 CreateHostAPI 來建立使用 AWS Command Line Interface (AWS CLI)、SDK 或的主機 AWS CloudFormation。建立後，主機處於 pending 狀態。您可以使用主控台的 設定 選項來完成程序。

工作流程：用主控台建立或更新主機

如果您要建立與已安裝提供者類型 (例如 GitHub Enterprise Server 或 GitLab 自我管理) 的連線，您必須先建立主機。如果您要連線到雲端供應商類型 (例如 Bitbucket)，請略過建立主機的步驟並繼續建立連線。

使用主控台設定主機並將其狀態從 pending 變更為 available。



建立或更新連線的工作流程

建立連線時，您也可以建立或使用現有的安裝，以與第三方供應商進行驗證交握。

連線可能為下列狀態：

- Pending - pending 連線是必須完成 (已變為 available) 才能使用的連線。
- Available - 您可以使用 available 連線或傳遞給帳戶中的其他資源和使用者。
- Error - 處於 error 狀態的連線會自動重試。直到變為 available 前都無法使用。

工作流程：透過 CLI、軟體開發套件或以下項目建立或更新： AWS CloudFormation

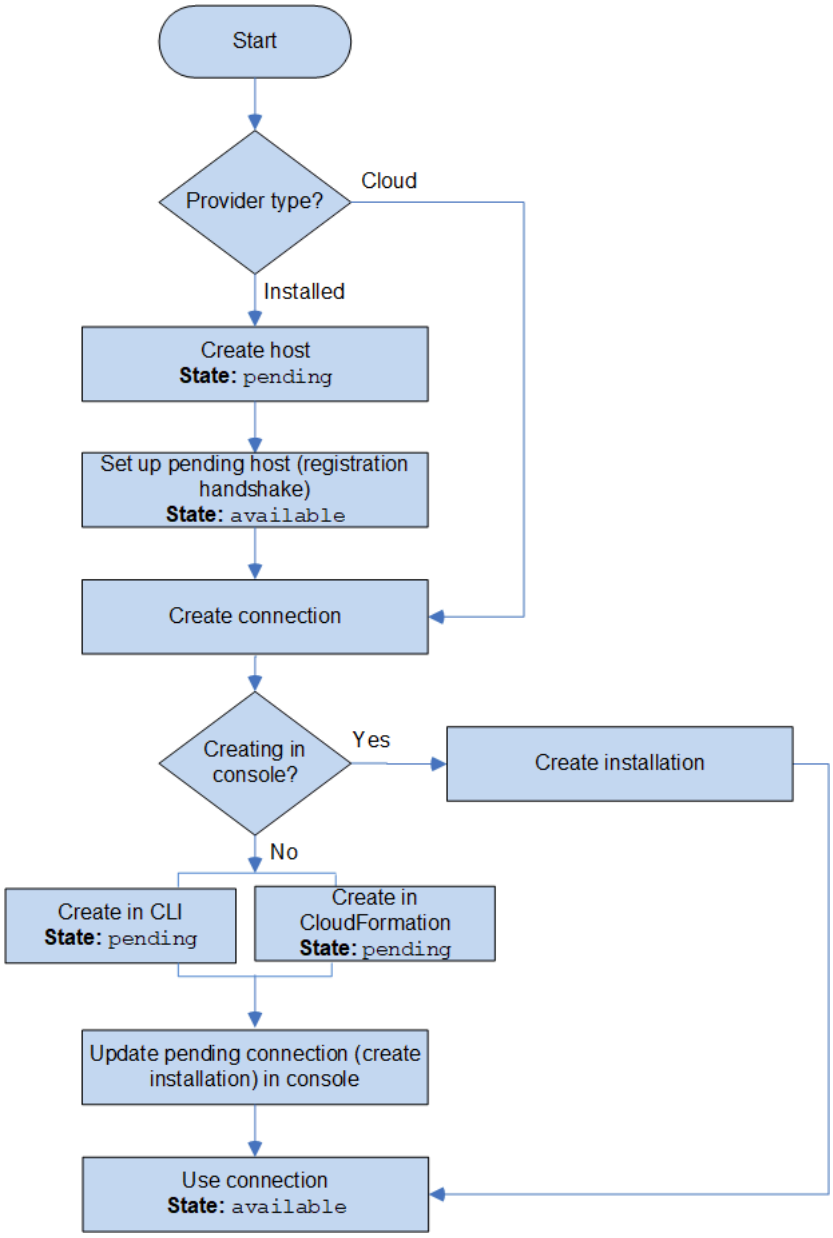
您可以使用 CreateConnectionAPI 來建立使用 AWS Command Line Interface (AWS CLI)、SDK 或的連線 AWS CloudFormation。連線建立後處於 pending 狀態。您可以使用主控台的 Set up pending connection (設定待定連線) 選項來完成程序。主控台會提示您建立安裝或使用現有的安裝來進行連

線。然後您可以使用主控台完成交握，並在主控台中選擇 Complete connection (完成連線)，將連線移至 `available` 狀態。

工作流程：建立或更新與主控台的連線

如果您要建立與已安裝提供者類型 (例如 GitHub企業伺服器) 的連線，請先建立主機。如果您要連線到雲端供應商類型 (例如 Bitbucket)，請略過建立主機的步驟並繼續建立連線。

若要使用主控台建立或更新連線，請使用主控台上的 CodePipeline 編輯動作頁面來選擇您的第三方提供者。主控台會提示您為連線建立安裝或使用現有的安裝，然後使用主控台來建立連線。主控台完成交握，並自動將連線從 `pending` 變為 `available` 狀態。



如何開始使用連線？

若要開始使用，以下是一些有用的主題可供檢閱：

- 了解關於連線的概念。
- 設定開始使用連線所需的資源。
- 開始使用使用您的第一個連線並將它們連線到資源。

連線概念

如果您了解概念和術語，設定和使用連線功能會比較容易。以下是使用開發人員工具主控台中的連線時需要了解的一些概念：

安裝

第三方帳戶上的 AWS 應用程式執行個體。安裝 AWS CodeStar Connector 應用程式可讓 AWS 存取第三方帳戶內的資源。只能在第三方供應商的網站上編輯安裝。

連線

AWS 資源用於將第三方來源儲存庫連線到其他 AWS 服務。

第三方儲存庫

由服務或公司提供且不屬於 AWS 的儲存庫。例如，BitBucket 儲存庫是第三方儲存庫。

供應商類型

提供您所連線第三方來源儲存庫的服務或公司。您可以將 AWS 資源連線至外部供應商類型。在網路和基礎設施上安裝來源儲存庫的供應商類型是安裝式供應商類型。例如，GitHub Enterprise Server 是安裝式供應商類型。

託管

代表安裝第三方供應商之基礎設施的資源。連線會使用主機來代表安裝第三方供應商的伺服器，例如 GitHub Enterprise Server。您可以為該供應商類型的所有連線建立一個主機。

> **Note**
>
> 使用主控台建立連至 GitHub Enterprise Server 的連線時，主控台會為您建立主機資源作為程序的一部分。

AWS CodeStar 連線支援的提供者和版本

本章提供有關 AWS CodeStar 連線支援的提供者和版本的資訊。

主題

Bitbucket 支援的供應商類型

您可以將應用 AWS CodeStar 程序與阿特拉西亞 Bitbucket 雲一起使用。

不支援安裝式 Bitbucket 供應商類型，例如 Bitbucket 伺服器。

GitHub 企業雲支援的 GitHub 提供者類型

您可以將 AWS 連接器用於 GitHub 應用程序 GitHub 和 GitHub 企業雲。

GitHub 企業伺服器支援的提供者類型和版本

您可以將應用 AWS CodeStar 程式與支援的 GitHub 企業伺服器版本搭配使用。如需支援的版本的清單，請參閱https://enterprise.github.com/releases/。

> **Important**
>
> AWS CodeStar 連線不支援已淘汰的 GitHub 企業伺服器版本。例如， AWS CodeStar 連線不支援 GitHub 企業伺服器 2.22.0 版，因為發行版本中的已知問題。若要連線，請升級至 2.22.1 版或最新的可用版本。

支援的提供者類型 GitLab

您可以使用連線 GitLab。如需詳細資訊，請參閱 建立連線 GitLab。

GitLab 自我管理的支援提供者類型

您可以使用 GitLab 自我管理安裝的連線 (適用於企業版或社群版)。如需更多詳細資訊，請參閱 建立與 GitLab自我管理的連線。

與 AWS CodeStar Connections 整合的產品和服務

AWS CodeStar Connections 與多種 AWS 服務與合作夥伴產品和服務整合。使用下列各節中的資訊，協助您設定連線以整合您要使用的產品和服務。

以下相關資源可協助您使用此服務。

主題

- 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 是用於監控儲存庫程式碼的服務。您可以使用連線與含有您想要審查之程式碼的第三方儲存庫建立關聯。如需瞭解如何設定 CodeGuru Reviewer 以監控 GitHub 儲存庫中原始碼的教學課程，以便建立改善程式碼的建議，請參閱 Amazon CodeGuru Reviewer 使用者指南中的教學課程：監控 GitHub 儲存庫中的原始碼。

Amazon CodeWhisperer

Amazon CodeWhisperer 是用於檢閱儲存庫程式碼的服務。CodeWhisperer 會檢閱您的程式碼，並即時為您提供程式碼建議。如需在您使用連線存取資料來源的 CodeWhisperer 中設定自訂步驟，請參閱《Amazon CodeWhisperer 使用者指南》中的建立自訂。

Amazon SageMaker

Amazon SageMaker 是一項用於建置、訓練和部署機器學習語言模型的服務。如需設定 GitHub 儲存庫連線的教學課程，請參閱《Amazon SageMaker 開發人員指南》中的使用第三方 Git 儲存庫的 SageMaker MLOps 專案逐步解說。

AWS App Runner

AWS App Runner 這項服務提供快速、簡單且符合成本效益的方式，可在 AWS 雲端 中直接將原始碼或容器映像部署到可擴展的安全 Web 應用程式上。您可以使用 App Runner 自動整合和交付管道，從儲存庫部署應用程式的程式碼。您可以使用連線從私有 GitHub 儲存庫將原始碼部署至 App Runner 服務。如需詳細資訊，請參閱在 AWS App Runner 開發人員指南中的原始碼儲存庫提供者。

AWS CloudFormation

AWS CloudFormation 是一個能幫助您模型化與設定 AWS 資源的服務，讓您能花較少的時間管理這些資源，並且有更多時間專注在 AWS 中執行的應用程式上。您建立一個描述所有所需之 AWS 資源的範本 (如 Amazon EC2 執行個體或 Amazon RDS 資料庫執行個體)，而 CloudFormation 負責為您佈建與設定這些資源。如需詳細資訊，請參閱《CloudFormation Command Line Interface 使用者指南》中的註冊您的帳戶以發佈 CloudFormation 擴充功能。

AWS CodePipeline

CodePipeline 是一種持續交付的服務，讓您能夠將發行軟體所需的步驟模型化、視覺化和自動化。您可以使用連線為 CodePipeline 來源動作設定第三方儲存庫。

進一步了解：

- `CodeStarSourceConnection` 動作請參閱 CodePipeline 動作組態參考頁面。若要檢視設定參數和範例 JSON/YAML 程式碼片段，請參閱 AWS CodePipeline 使用者指南的 CodeStarSourceConnection。
- 若要檢視使用第三方來源儲存庫建立管道的入門教學，請參閱 開始使用連線。

AWS CodeStar

AWS CodeStar 是一種雲端服務，用於建立、管理和處理 AWS 上的軟體開發專案。您可以使用 AWS CodeStar 專案在 AWS 上迅速開發、建置及部署應用程式。您可以使用連線為 AWS CodeStar 專案中的管道設定第三方儲存庫。如需建立 AWS CodeStar 專案並連線至 GitHub 儲存庫的教學課程，請參閱《AWS CodeStar 使用者指南》中的建立儲存庫的連結。

Service Catalog

Service Catalog 可讓組織建立和管理已核准在 AWS 上使用的產品目錄。

當您授權 AWS 帳戶 與外部儲存庫提供者 (例如 GitHub、GitHub Enterprise 或 BitBucket) 之間的連線時，連線可讓您將 Service Catalog 產品同步至透過第三方供應商儲存庫管理的範本檔案。

如需詳細資訊，請參閱《Service Catalog 使用者指南》中將 Service Catalog 產品同步至 GitHub、GitHub Enterprise 或 Bitbucket 中的範本檔案。

AWS Proton

AWS Proton 是用於部署到雲端基礎架構的雲端服務。您可以使用連線為 AWS Proton 範本中的資源建立第三方儲存庫的連結。如需詳細資訊，請參閱 AWS Proton 使用者指南中的建立儲存庫的連結。

設定連線

完成本節中的任務，進行設定以建立和使用開發人員工具主控台中的連線功能。

主題

- 註冊 AWS
- 建立並套用具有建立連線許可的策略

註冊 AWS

註冊 AWS 帳戶

如果您還沒有 AWS 帳戶，請完成以下步驟建立新帳戶。

註冊 AWS 帳戶

1. 開啟 https://portal.aws.amazon.com/billing/signup。
2. 請遵循線上指示進行。

   部分註冊程序需接收來電，並在電話鍵盤輸入驗證碼。

   註冊 AWS 帳戶時，會建立 AWS 帳戶根使用者。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為最佳安全實務，將管理存取權指派給管理使用者，並且僅使用根使用者來執行需要根使用者存取權的任務。

註冊程序完成後，AWS 會傳送一封確認電子郵件給您。您可以隨時登錄 https://aws.amazon.com/ 並選擇 我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

建立管理使用者

在您註冊 AWS 帳戶 之後，請保護您的 AWS 帳戶根使用者、啟用 AWS IAM Identity Center，以及建立管理使用者，讓您可以不使用根使用者處理日常作業。

保護您的 AWS 帳戶根使用者

1. 選擇 根使用者 並輸入您的 AWS 帳戶電子郵件地址，以帳戶擁有者身分登入 AWS Management Console。在下一頁中，輸入您的密碼。

   如需使用根使用者登入的說明，請參閱 AWS 登入使用者指南中的以根使用者身分登入。
2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

   如需指示，請參閱《IAM 使用者指南》中的為 AWS 帳戶根使用者啟用虛擬 MFA 裝置 (主控台)。

建立管理使用者

1. 啟用 IAM Identity Center。

   如需指示，請參閱AWS IAM Identity Center使用者指南中的啟用 AWS IAM Identity Center。
2. 在 IAM Identity Center 中，將管理存取權授予管理使用者。

   有關如何使用 IAM Identity Center 目錄 作為身分來源的教學課程，請參閱《AWS IAM Identity Center 使用者指南》中的以預設 IAM Identity Center 目錄 設定使用者存取權。

以管理員的身分登入

- 若要使用您的 IAM 身分中心使用者登入，請使用建立 IAM 身分中心使用者時傳送至您電子郵件地址的登入 URL。

  如需有關如何使用 IAM Identity Center 使用者登入的說明，請參閱《AWS 登入 使用者指南》中的登入 AWS存取入口網站。

建立並套用具有建立連線許可的策略

若要使用 JSON 政策編輯器來建立政策

1. 登入 AWS Management Console，並開啟位於 https://console.aws.amazon.com/iam/ 的 IAM 主控台。
2. 在左側的導覽窗格中，選擇 Policies (政策)。

   如果這是您第一次選擇 Policies (政策)，將會顯示 Welcome to Managed Policies (歡迎使用受管政策) 頁面。選擇 Get Started (開始使用)。
3. 在頁面頂端，選擇 Create policy (建立政策)。
4. 在政策編輯器中，選擇 JSON 選項。
5. 輸入下列 JSON 政策文件：

```
{
   "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. 選擇 Next (下一步)。

> **Note**
>
> 您可以隨時切換視覺化與 JSON 編輯器選項。不過，如果您進行變更或在視覺化編輯器中選擇下一步，IAM 就可能會調整您的政策結構，以便針對視覺化編輯器進行最佳化。如需詳細資訊，請參閱 IAM 使用者指南中的調整政策結構。

7. 在檢視與建立頁面上，為您正在建立的政策輸入政策名稱與描述 (選用)。檢視此政策中定義的許可，來查看您的政策所授予的許可。
8. 選擇 Create policy (建立政策) 儲存您的新政策。

開始使用連線

開始使用連線的最簡單方法是設定第三方來源儲存庫與 AWS 資源相關聯的連線。如果您想要將管道連線到 AWS 來源 (例如 CodeCommit)，您可以將其作為來源動作進行連線。但是如果您有外部儲存庫，則必須建立連線來為儲存庫與管道建立關聯。在本教學課程中，您將設定與您的 Bitbucket 儲存庫和管道間的連線。

在本節中，您會透過下列項目使用連線：

- AWS CodePipeline：在這些步驟中，您可以使用 Bitbucket 儲存庫做為管道來源建立管道。
- Amazon CodeGuru Reviewer：接下來，您可以將 Bitbucket 儲存庫與 CodeGuru Reviewer 中的意見回饋和分析工具建立關聯。

主題

先決條件

開始之前，請完成 設定 中的步驟。您還需要一個要連線到 AWS 服務的第三方來源儲存庫，並允許連線為您管理身分驗證。例如，您可能想要將 Bitbucket 儲存庫連線到與來源儲存庫整合的 AWS 服務。

- 使用您的 Bitbucket 帳戶建立一個 Bitbucket 儲存庫。

- 準備好您的 Bitbucket 憑證。使用 AWS Management Console 設定連線時，系統會要求您使用 Bitbucket 憑證登入。

步驟 1：編輯來源檔案

建立 Bitbucket 儲存庫時會內含一個預設的 `README.md` 檔案，您需編輯這個檔案。

1. 登入您的 Bitbucket 儲存庫並選擇 Source (來源)。
2. 選擇 `README.md` 檔案並選擇頁面頂端的 Edit (編輯)。刪除現有的文字，並新增下列文字。

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

3. 選擇 Commit (遞交)。

   確定 `README.md` 檔案位於儲存庫的根層級。

步驟 2：建立管道

在本節中，您可以採取下列動作建立管道：

- 使用連至 Bitbucket 儲存庫和動作的連線之來源階段。
- 使用 AWS CodeBuild 建置動作的建置階段。

使用精靈建立管道

1. 前往 https://console.aws.amazon.com/codepipeline/ 登入 CodePipeline 主控台。
2. 在 Welcome (歡迎) 頁面、Getting started (入門) 頁面、或者 Pipelines (管道) 頁面上，選擇 Create pipeline (建立管道)。
3. 在 Step 1: Choose pipeline settings (步驟 1：選擇管道設定) 的 Pipeline name (管道名稱) 中，輸入 **`MyBitbucketPipeline`**。
4. 在 Service role (服務角色) 中，選擇 New service role (新服務角色)。

> **Note**
>
> 如果您選擇改用現有的 CodePipeline 服務角色，請確定您已將 `codestar-connections:UseConnection` IAM 許可新增至您的服務角色政策。如需 CodePipeline 服務角色的說明，請參閱為 CodePipeline 服務角色新增許可。

5. 在進階設定底下，請保留預設值。在Artifact store (成品存放區) 中，針對您為管道所選取區域中的管道，選擇 Default location (預設位置)，即可使用預設成品存放區 (例如指定為預設值的 Amazon S3 成品儲存貯體)。

> **Note**
>
> 這不是原始碼的來源儲存貯體。這是管道的成品存放區。每個管道都需要有個別成品存放區，例如 S3 儲存貯體。

選擇 Next (下一步)。

6. 在 Step 2: Add source stage (步驟 2：新增來源階段) 頁面上，新增來源階段：

   a. 在 Source provider (來源供應商) 中，選擇 Bitbucket。

   b. 在 Connection (連線) 底下，選擇 Connect to Bitbucket (連線至 Bitbucket)。

   c. 在 Connect to Bitbucket (連線至 Bitbucket) 頁面的 Connection name (連線名稱) 中，輸入您要建立的連線名稱。此名稱可協助您稍後識別此連線。

      在 Bitbucket apps (Bitbucket 應用程式) 底下中，選擇 Install a new app (安裝新應用程式)。

   d. 在應用程式安裝頁面上，有訊息顯示 AWS CodeStar 應用程式正在嘗試連線到您的 Bitbucket 帳戶。選擇 Grant access (授與存取權)。授權連線之後，系統會偵測 Bitbucket 上的儲存庫，您可以選擇為其中一個儲存庫與 AWS 資源建立關聯。

   e. 隨即顯示新安裝的連線 ID。選擇 Complete connection (完成連線)。您將返回 CodePipeline 主控台。

   f. 在 Repository name (儲存庫名稱) 中，選擇 Bitbucket 儲存庫的名稱。

   g. 在 Branch name (分支名稱) 中，選擇儲存庫的分支。

   h. 請確認已選取在原始程式碼變更時啟動管道選項。

   i. 在輸出成品格式底下，選擇下列其中一項：CodePipeline 預設。

      - 選擇 CodePipeline 預設，以針對管道中的成品使用預設 zip 格式。
      - 選擇完整複製，以包含與管道中成品的儲存庫相關的 Git 中繼資料。只有 CodeBuild 動作支援此項。

   選擇 Next (下一步)。

7. 在 Add build stage (新增建置階段) 中，新增建置階段：

a. 在 Build provider (建置供應商) 中，選擇 AWS CodeBuild。允許 Region (區域) 預設為管道區域。

b. 選擇 Create project (建立專案)。

c. 在 Project name (專案名稱) 中，輸入此建置專案的名稱。

d. 在 Environment image (環境映像) 中，選擇 Managed image (受管映像)。針對 Operating system (作業系統)，選擇 Ubuntu。

e. 針對 Runtime (執行時間)，選擇 Standard (標準)。針對映像，選擇 aws/codebuild/standard:5.0。

f. 對於 Service role (服務角色)，選擇 New service role (新服務角色)。

g. 在 BuildSpec 底下，針對 Build specifications (建置規格) 選擇 Insert build commands (插入建置命令)。選擇 Switch to editor (切換到編輯器)，並將下方內容貼到 Build commands (建置命令) 底下：

```
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
```

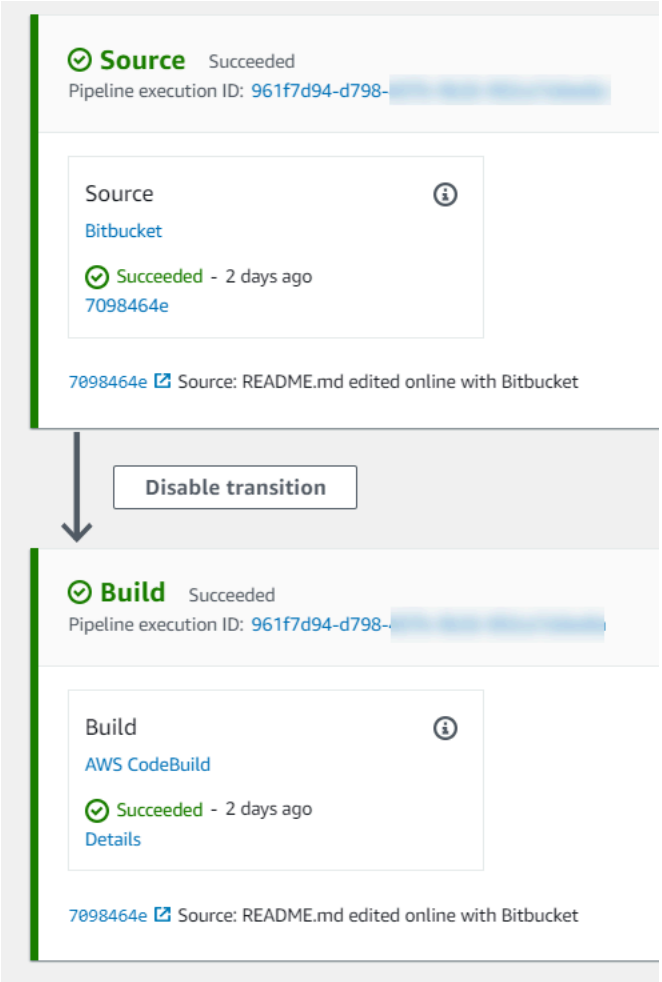
h. 選擇 Continue to CodePipeline (繼續 CodePipeline)。這會傳回 CodePipeline 主控台，並建立使用您的建置命令進行設定的 CodeBuild 專案。建置專案會使用服務角色管理 AWS 服務許可。此步驟可能需要數分鐘。

i. 選擇 Next (下一步)。

8. 在 Step 4: Add deploy stage (步驟 4：新增部署階段) 頁面上，選擇 Skip deploy stage (跳過部署階段)，然後再次選擇 Skip (跳過) 來接受警告訊息。選擇 Next (下一步)。

9. 在 Step 5: Review (步驟 5：檢閱) 上，選擇 Create pipeline (建立管道)。

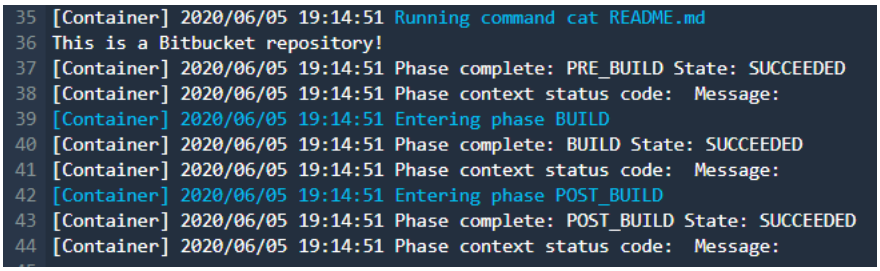
10. 成功建立管道後，管道就會開始執行。



11. 在成功的建置階段上，選擇 Details (詳細資訊)。

    在 Execution details (執行詳細資訊) 底下，檢視 CodeBuild 建置輸出。這些命令會輸出 `README.md` 檔案內容，如下所示：

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



步驟 3：為儲存庫與 CodeGuru Reviewer 建立關聯

建立連線後，您可以將該連線用於同一個帳戶中的所有 AWS 資源。例如，您可以針對管道中的 CodePipeline 來源動作使用相同的 Bitbucket 連線，並在 CodeGuru Reviewer 中執行儲存庫遞交分析。

1. 登入 CodeGuru Reviewer 主控台。
2. 在 CodeGuru Reviewer 底下，選擇Associate repository (為儲存庫建立關聯)。

   隨即會開啟單頁式精靈。
3. 在 Select source provider (選取來源提供程式) 底下，選擇 Bitbucket。
4. 在連線至 Bitbucket (透過 AWS CodeStar Connections) 底下，選擇您為管道建立的連線。
5. 在 Repository location (儲存庫位置) 底下，選擇 Bitbucket 儲存庫的名稱，然後選擇 Associate (建立關聯)。

   您可以繼續設定程式碼檢閱。如需詳細資訊，請參閱 Amazon CodeGuru Reviewer 使用者指南中的連線至 Bitbucket 並為儲存庫與 CodeGuru Reviewer 建立關聯。

使用連線

連線是用於將 AWS 資源連線到外部程式碼儲存庫的組態。每個連接都是可以提供給服務的資源，例如連接 AWS CodePipeline 到第三方存儲庫，例如 Bitbucket。例如，您可以在中新增連線，以 CodePipeline 便在對協力廠商程式碼儲存庫進行程式碼變更時觸發管道。您也可以將資 AWS 源連線到已安裝的提供者類型，例如 GitHub 企業伺服器。

如果您想要建立與已安裝的提供者類型 (例如 GitHub Enterprise Server) 的連線，則主控台會為您建立主機。主機是您建立的資源，用來代表安裝供應商的伺服器。如需詳細資訊，請參閱 使用主機。

建立連線時，您可以使用主控台中的精靈將應用程式安裝至第三方供 AWS CodeStar 應商，並將其與新連線建立關聯。如果您已經安裝了該 AWS CodeStar 應用程序，則可以使用它。

> **Note**
>
> 要在歐洲（米蘭）使用連接 AWS 區域，您必須：
>
> 1. 安裝區域特定的應用程式
> 2. 啟用區域

此區域特定的應用程式支援歐洲 (米蘭) 區域中的連線。它會在第三方供應商網站上發佈，並且它會與支援其他區域連線的現有應用程式分開。透過安裝此應用程式，您授權第三方提供商僅使用該區域服務來共用您的資料，並且您可以透過解除安裝該應用程式來隨時撤銷許可。
除非您啟用區域，否則服務不會處理或儲存您的資料。啟用此區域，即表示您授予我們服務許可來處理和儲存您的資料。
即使未啟用該區域，如果仍已安裝區域特定的應用程式，第三方供應商仍然可以使用我們的服務來共用您的資料，因此請確保在停用該區域後解除安裝該應用程式。如需詳細資訊，請參閱啟用區域。

如需有關連線的詳細資訊，請參閱AWS CodeStar 連線 API 參考資料。如需 Bitbucket 之 CodePipeline 來源動作的詳細資訊，請參閱AWS CodePipeline 使用者指南CodestarConnectionSource中的。

若要建立政策或將政策附加到您的 AWS Identity and Access Management (IAM) 使用者或角色，並具有使用 AWS CodeStar 連線所需的權限，請參閱AWS CodeConnections 權限參考。視建立 CodePipeline 服務角色的時間而定，您可能需要更新其權限以支援 AWS CodeStar 連線。如需說明，請參閱《AWS CodePipeline 使用者指南》中的更新服務角色。

主題

建立連線

您可以建立連至下列第三方供應商類型的連線：

- 若要建立連至 Bitbucket 的連線，請參閱「建立連至 Bitbucket 的連線」。
- 若要建立與 GitHub 或 GitHub 企業雲端的連線，請參閱建立連線 GitHub。
- 若要建立與 GitHub 企業伺服器的連線，包括建立主機資源，請參閱建立與 GitHub 企業伺服器的連線。
- 若要建立連線 GitLab，請參閱建立連線 GitLab。

建立連至 Bitbucket 的連線

您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 來建立一個與位於 bitbucket.org 上託管的儲存庫的連線。

開始之前：

- 您必須已建立 Bitbucket 帳戶。
- 您必須已在 bitbucket.org 上建立程式碼儲存庫。

> **Note**
>
> 您可以建立連至 Bitbucket Cloud 儲存庫的連線。不支援安裝式 Bitbucket 供應商類型，例如 Bitbucket 伺服器。請參閱AWS CodeStar 連線支援的提供者和版本。

> **Note**
>
> 連線只能存取用於建立連線之帳戶擁有的儲存庫。
> 若應用程式正安裝於 Bitbucket 工作區中，則需要 Administer workspace (管理工作區) 許可。否則，安裝該應用程式的選項將不會顯示。

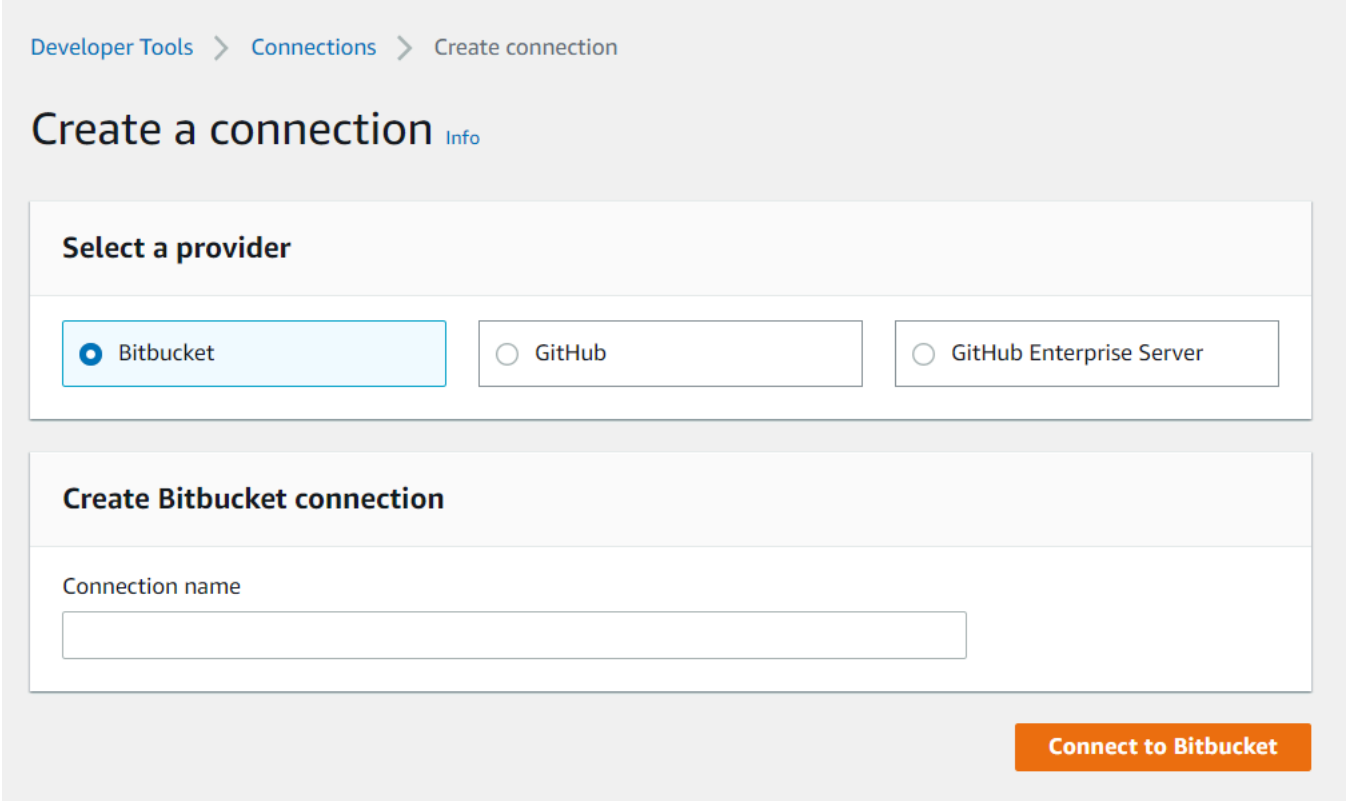
主題

- 建立連至 Bitbucket 的連線 (主控台)
- 建立連至 Bitbucket 的連線 (CLI)

建立連至 Bitbucket 的連線 (主控台)

步驟 1：建立連線

1. 登入 AWS Management Console，然後開啟開發 AWS 人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 選擇 Settings > Connections (設定 > 連線)，然後選擇 Create connection (建立連線)。
3. 若要建立連至 Bitbucket 儲存庫的連線，請在 Select a provider (選取供應商) 底下選擇 Bitbucket。在 Connection name (連線名稱) 底下，輸入您要建立的連線名稱。選擇 Connect to Bitbucket (連線至 Bitbucket)，然後繼續進行步驟 2。

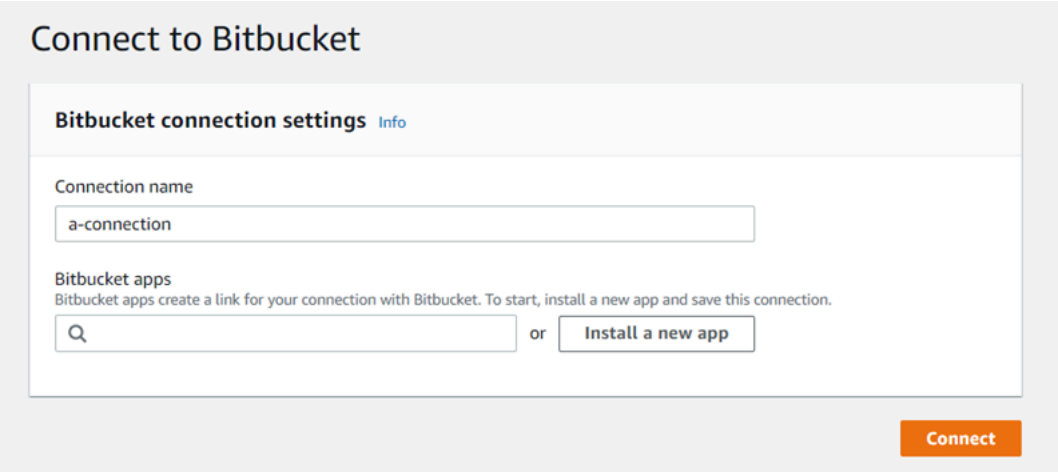


步驟 2：連線至 Bitbucket

1. Connect to Bitbucket (連線至 Bitbucket) 設定頁面上會顯示您的連線名稱。

   在 Bitbucket apps (Bitbucket 應用程式) 底下，選擇應用程式安裝，或選擇 Install a new app (安裝新應用程式) 以建立安裝。

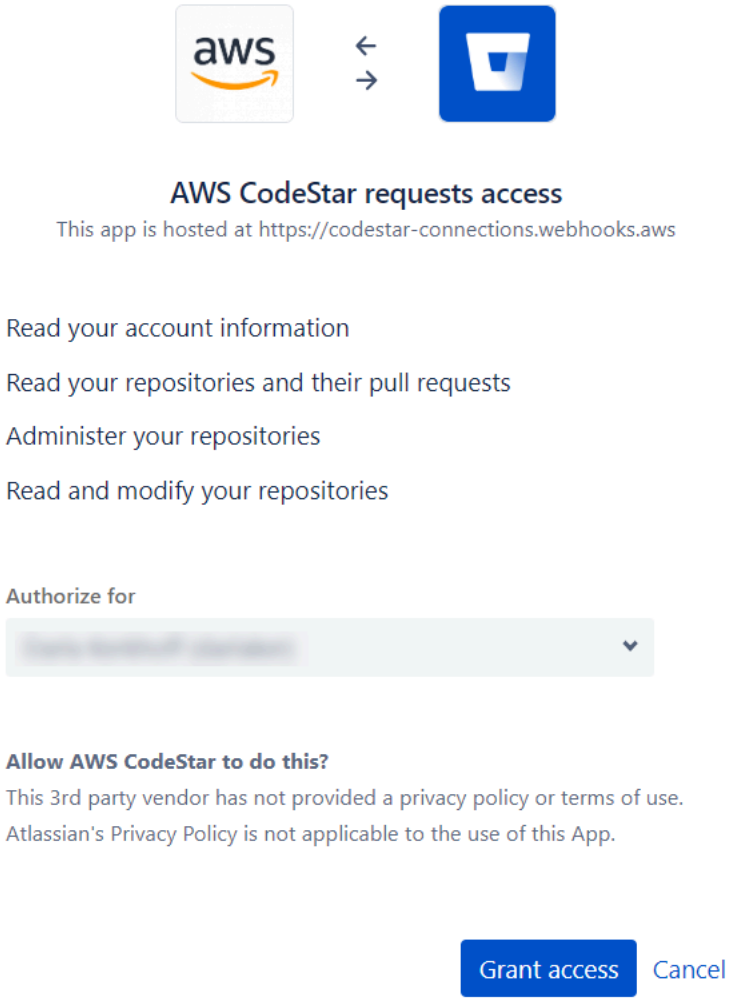
> **Note**
>
> 您只能為每個 Bitbucket 工作區或帳戶安裝一次應用程式。如果您已安裝 Bitbucket 應用程式，請選擇該應用程式並移至本節的最後一個步驟。



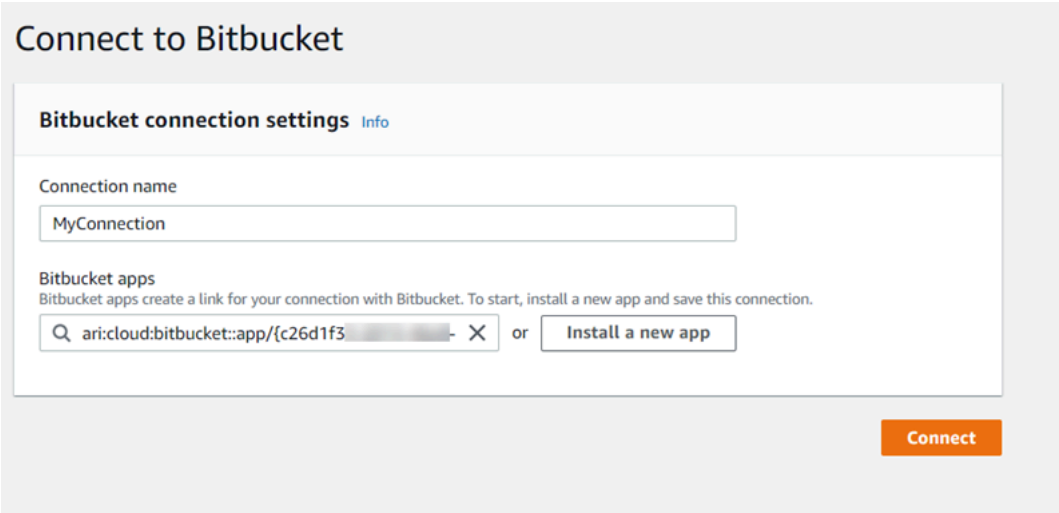
2. 如果顯示 Bitbucket 的登入頁面，請使用您的憑證登入，然後選擇繼續進行。
3. 在應用程式安裝頁面上，會有訊息顯示 AWS CodeStar 應用程式正在嘗試連線至您的 Bitbucket 帳戶。

   若您使用的是 Bitbucket 工作區，請將 Authorize for (授權) 選項變更為工作區。僅會顯示您具有管理員存取權的工作區。

   選擇 Grant access (授與存取權)。



4. 在 Bitbucket apps (Bitbucket 應用程式) 中，會顯示新安裝的連線 ID。選擇 Connect (連線)。建立的連線會顯示在連線清單中。



建立連至 Bitbucket 的連線 (CLI)

您可以使用 AWS Command Line Interface (AWS CLI) 來建立連線。

若要這麼做，請使用 create-connection 命令。

> **Important**
>
> 依預設，透過 AWS CLI 或建立 AWS CloudFormation 的連線處於PENDING狀態。建立與 CLI 的連線之後 AWS CloudFormation，或使用主控台編輯連線以顯示其狀態AVAILABLE。

建立連至 Bitbucket 的連線

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-connection令，--connection-name為您的連線指定--provider-type和。在此範例中，第三方供應商名稱為 Bitbucket，而指定的連線名稱為 MyConnection。

   ```
   aws codestar-connections create-connection --provider-type Bitbucket --connection-
   name MyConnection
   ```

   如果成功，此命令會傳回類似下列內容的連線 ARN 資訊。

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

2. 使用主控台完成連線。如需詳細資訊，請參閱 更新待定連線。

建立連線 GitHub

您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 來建立與的連接 GitHub。

開始之前：

- 您必須已經在建立帳戶 GitHub。
- 您必須已建立第三方程式碼儲存庫。

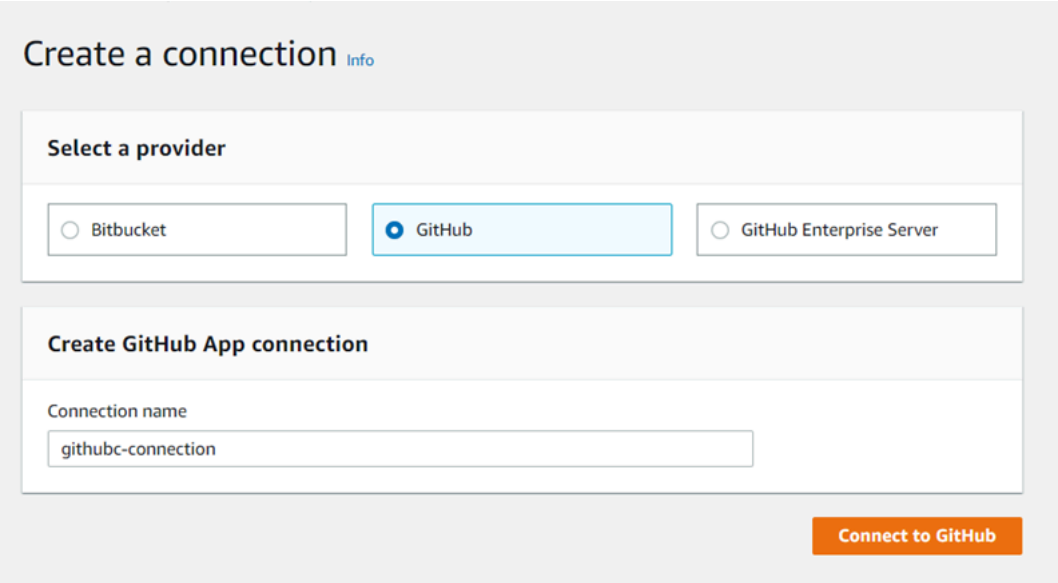
> **Note**
>
> 若要建立連線，您必須是 GitHub 組織擁有者。對於不在組織下的儲存庫，您必須是儲存庫擁有者。

主題

- 建立連線至 GitHub (主控台)
- 建立與 GitHub (CLI) 的連線

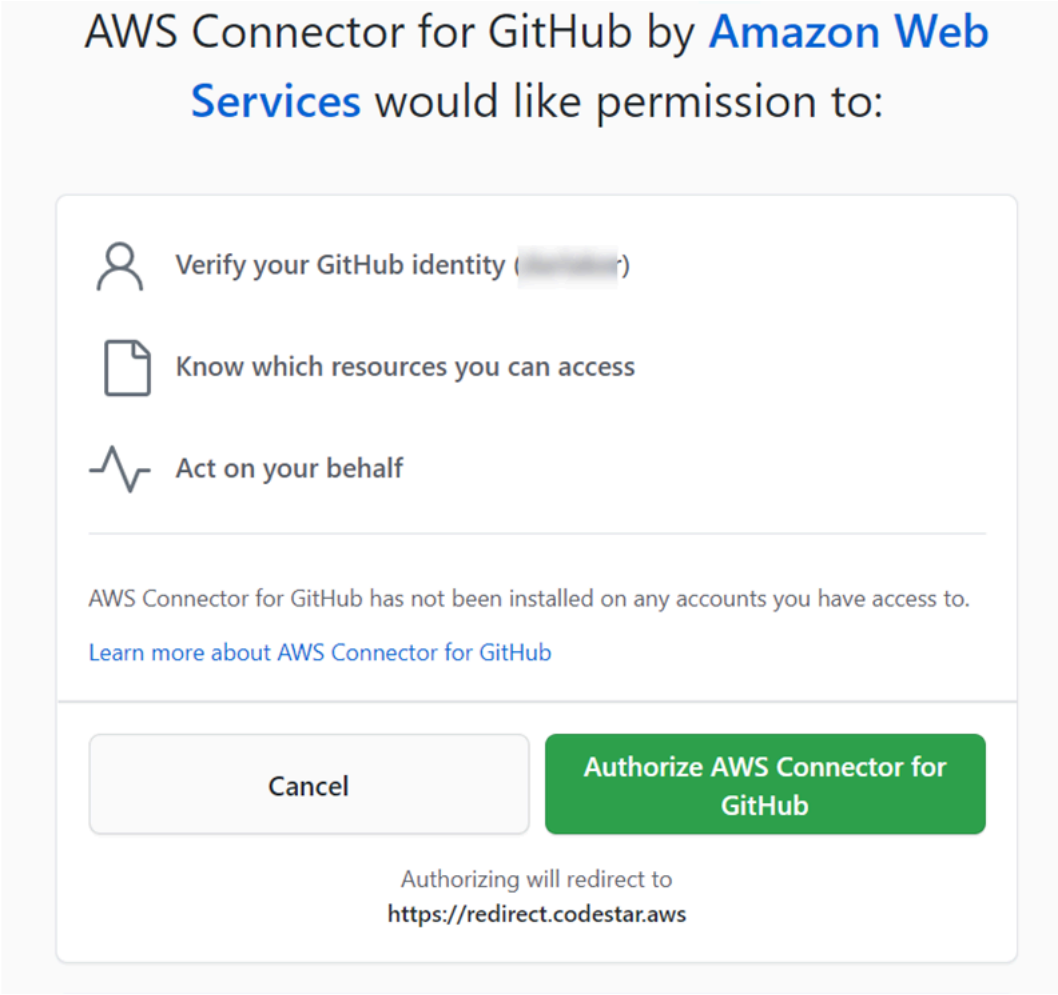
建立連線至 GitHub (主控台)

1. 登入 AWS Management Console，然後開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 選擇 Settings > Connections (設定 > 連線)，然後選擇 Create connection (建立連線)。
3. 若要建立與 GitHub 或 GitHub 企業雲端儲存庫的連線，請在 [選取提供者] 下方選擇GitHub。在 Connection name (連線名稱) 底下，輸入您要建立的連線名稱。選擇「Connect 至」 GitHub，然後繼續執行步驟 2。

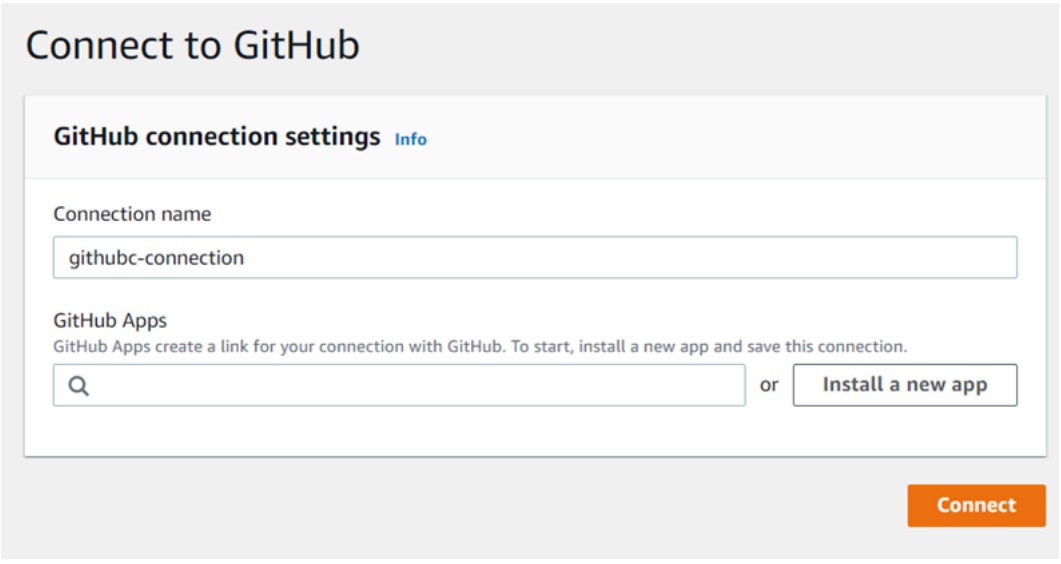


若要建立連線 GitHub

1. 在GitHub 連線設定下，您的連線名稱會出現在連線名稱中。選擇「Connect 至」 GitHub。隨即會顯示存取請求頁面。



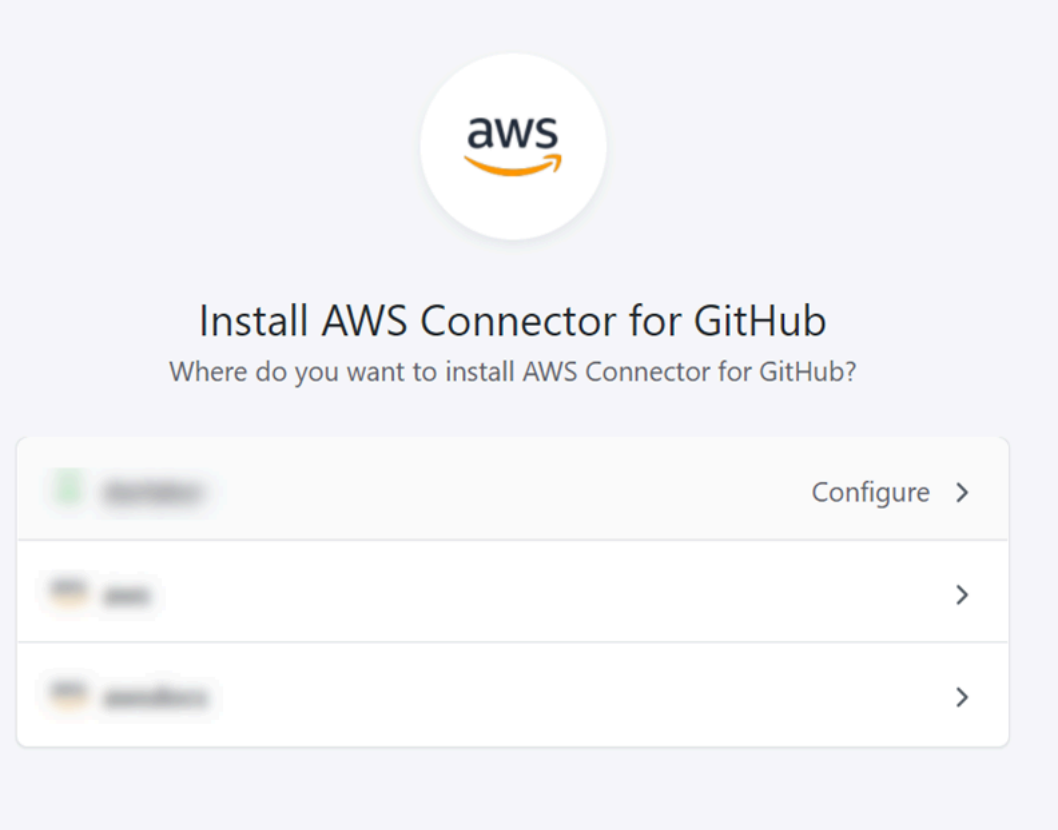
2. 選擇 [授權 AWS 連接器] GitHub。連線頁面隨即顯示並顯示 [GitHub 應用程式] 欄位。



3. 在 [GitHub 應用程式] 下方，選擇應用程式安裝，或選擇 [安裝新的應用程式] 來建立

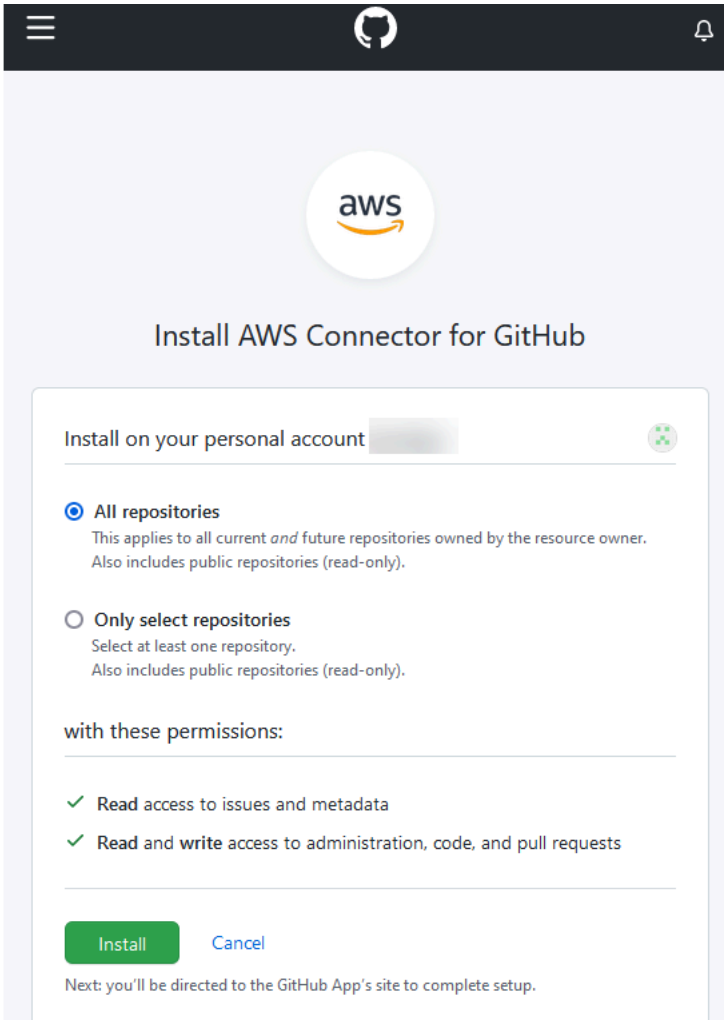
> **Note**
>
> 您可以為您連至特定供應商的所有連線安裝一個應用程式。如果您已經安裝 GitHub 應用程式的 AWS 連接器，請選擇該連接器並略過此步驟。

4. 在 [安裝AWS 連接器 GitHub] 頁面上，選擇您要安裝應用程式的帳戶。



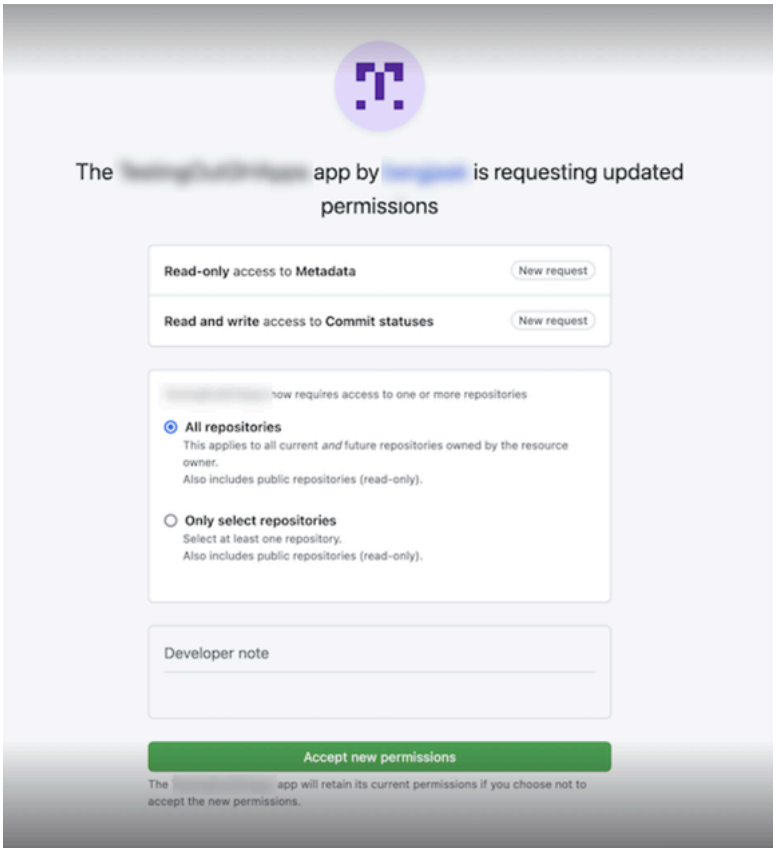
> **Note**
>
> 您只能為每個 GitHub 帳戶安裝一次該應用程序。如果您先前已安裝應用程式，可以選擇 Configure (設定)，繼續前往應用程式安裝的修改頁面，或者您可以使用上一步按鈕返回主控台。

5. 在 [安裝 AWS 連接器 GitHub] 頁面上，保留預設值，然後選擇 [安裝]。



完成此步驟之後，更新的權限頁面可能會顯示在中 GitHub。

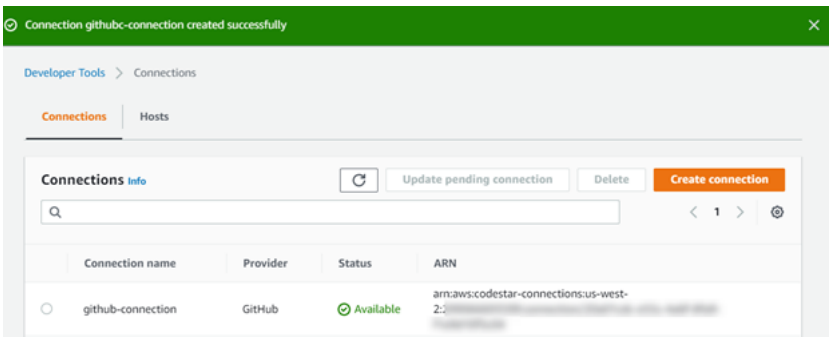
6. 如果顯示的頁面顯示 GitHub 應用程式 AWS 連接器已更新權限，請選擇 [接受新的權限]。



7. 您將會返回 [Connect 至 GitHub] 頁面。新安裝的連線 ID 會出現在GitHub應用程式中。選擇連線。

檢視您建立的連線

- 建立的連線會顯示在連線清單中。



建立與 GitHub (CLI) 的連線

您可以使用 AWS Command Line Interface (AWS CLI) 建立與的連接 GitHub。

若要這麼做，請使用 create-connection 命令。

> **Important**
>
> 依預設，透過 AWS CLI 或建立 AWS CloudFormation 的連線處於PENDING狀態。建立與 CLI 的連線之後 AWS CloudFormation，或使用主控台編輯連線以顯示其狀態AVAILABLE。

若要建立連線 GitHub

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-connection令，`--connection-name`為您的連線指定`--provider-type`和。在此範例中，第三方供應商名稱為 `GitHub`，而指定的連線名稱為 `MyConnection`。

   ```
   aws codestar-connections create-connection --provider-type GitHub --connection-name
    MyConnection
   ```

   如果成功，此命令會傳回類似下列內容的連線 ARN 資訊。

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

2. 使用主控台完成連線。如需詳細資訊，請參閱 更新待定連線。

建立與 GitHub 企業伺服器的連線

您可以使用連線將資 AWS 源與第三方存放庫建立關聯。您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 來建立與 GitHub 企業伺服器的連線。

連線僅提供對 GitHub 企業伺服器帳戶擁有的存放庫的存取權，該存放庫在建立連線期間用於授權 GitHub應用程式的安裝。

開始之前：

- 您必須已經擁有 GitHub 企業伺服器執行個體和儲存庫。
- 您必須是 GitHub Enterprise Server 執行個體的系統管理員，才能建立 GitHub 應用程式並建立主機資源，如本節所示。

> **Important**
>
> 當您為 GitHub 企業伺服器設定主機時，會為您建立 Webhook 事件資料的 VPC 端點。如果您在 2020 年 11 月 24 日之前建立了主機，並且想要使用 VPC PrivateLink Webhook 端點，則必須先刪除主機，然後再建立新主機。

主題

- 建立與 GitHub企業伺服器 (主控台) 的連線
- 建立與 GitHub企業伺服器 (CLI) 的連線

建立與 GitHub企業伺服器 (主控台) 的連線

若要建立 GitHub 企業伺服器連線，請提供 GitHub 企業伺服器安裝位置的資訊，並使用您的 GitHub 企業認證授權建立連線。

主題

- 建立您的 GitHub企業伺服器連線 (主控台)

建立您的 GitHub企業伺服器連線 (主控台)

若要建立與 GitHub 企業伺服器的連線，請準備好伺服器 URL 和 GitHub 企業認證。

建立主機

1. 登入 AWS Management Console，然後開啟開發 AWS 人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 在 Hosts (主機) 索引標籤中，選擇 Create host (建立主機)。
3. 在 Host name (主機名稱) 中，輸入您想要使用的主機名稱。
4. 在選取提供者中，選擇下列其中一項：
   - GitHub 企業伺服器
   - GitLab 自我管理
5. 在 URL 中，輸入安裝供應商之基礎設施的端點。
6. 如果您的伺服器是在 Amazon VPC 內設定的，而您想要與 VPC 連線，請選擇 Use a VPC (使用 VPC)。否則，請選擇 No VPC (無 VPC)。

7. 如果您已將執行個體啟動到 Amazon VPC 中，且想與 VPC 連線，請選擇 Use a VPC (使用 VPC)，然後完成以下操作。

   a. 在 VPC ID 底下，選擇您的 VPC ID。請務必選擇安裝執行個體的基礎設施之 VPC，或是可透過 VPN 或 Direct Connect 存取執行個體的 VPC。

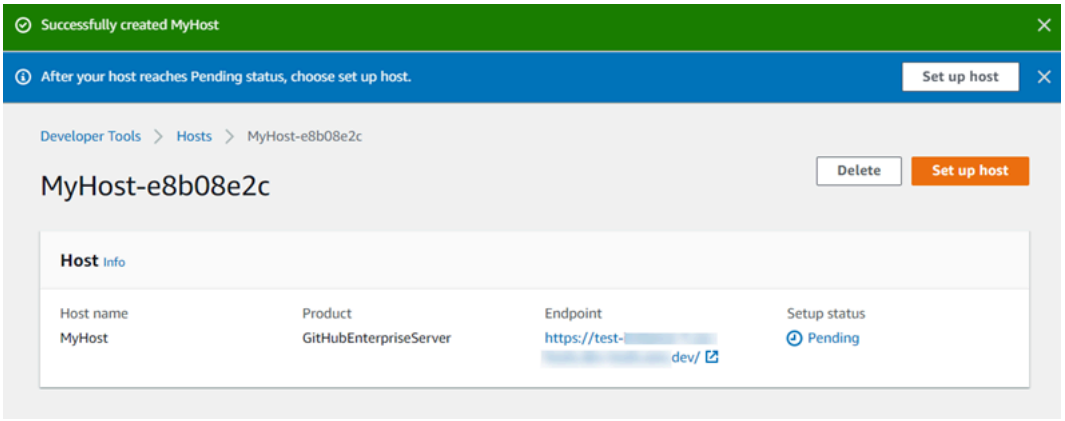
   b. 如果您已設定私有 VPC，且已將執行個體設定為使用非公有憑證授權機構執行 TLS 驗證，請在 TLS certificate (TLS 憑證) 中輸入您的憑證 ID。TLS 憑證值應該是憑證的公有金鑰。

8. 選擇 Create host (建立主機)。

9. 主機詳細資訊頁面顯示後，主機狀態會隨主機建立而變更。

> **Note**
>
> 如果您的主機設定包含 VPC 組態，需花幾分鐘的時間來佈建主機網路元件。

等待您的主機到達 Pending (待定) 狀態，然後完成設定。如需詳細資訊，請參閱 設定待定主機。



步驟 2：建立與 GitHub 企業伺服器的連線 (主控台)

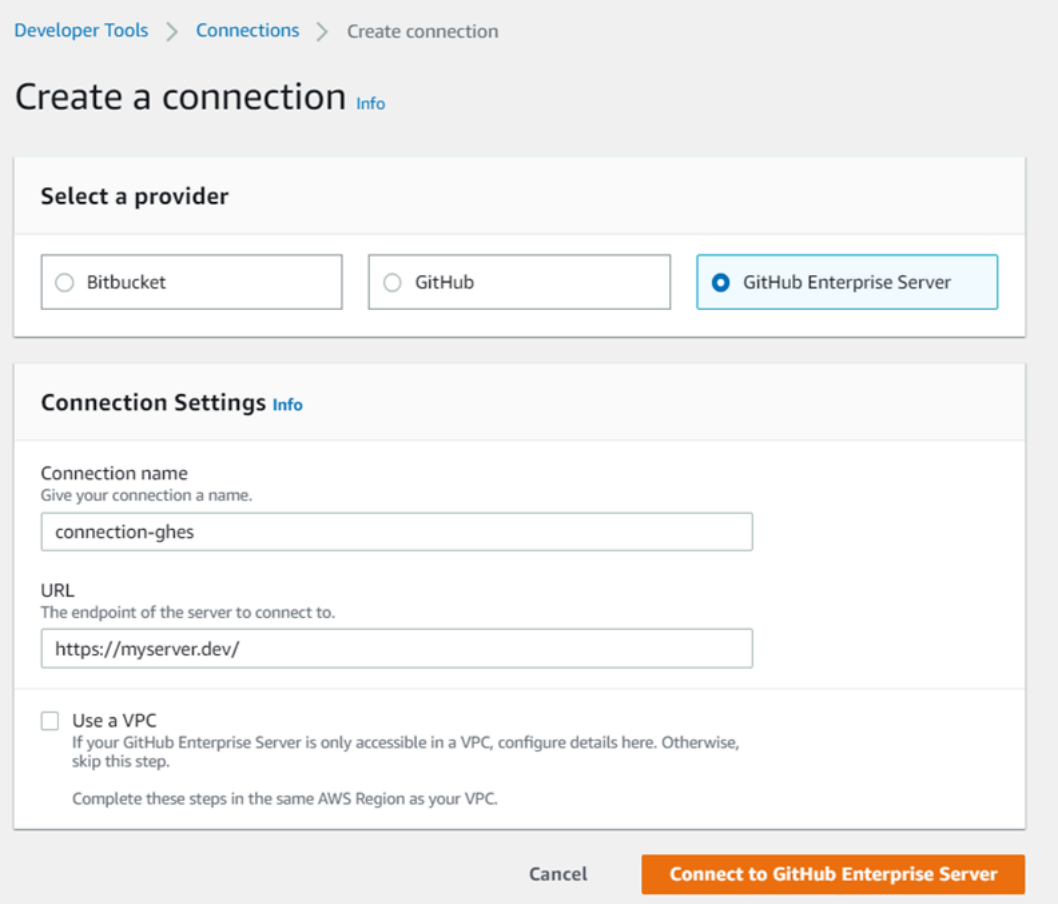
1. 登入 AWS Management Console 並開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。

2. 選擇 Settings > Connections (設定 > 連線)，然後選擇 Create connection (建立連線)。

3. 若要建立與已安裝之 GitHub 企業伺服器儲存區域的連線，請選擇「GitHub 企業伺服器

Connect 到 GitHub 企業伺服器

1. 針對 Connection name (連線名稱)，請輸入連線的名稱。



2. 在 URL 中，輸入伺服器的端點。

> **Note**
>
> 如果提供的 URL 已用於為連線設定 GitHub企業伺服器，系統會提示您選擇先前為該端點建立的主機資源 ARN。

3. （選用）如果您已將伺服器啟動到 Amazon VPC 中，且想與 VPC 連線，請選擇 Use a VPC (使用 VPC)，然後完成以下操作。

   a. 在 VPC ID 底下，選擇您的 VPC ID。請務必針對安裝 GitHub 企業伺服器執行個體的基礎結構選擇 VPC，或是可透過 VPN 或直 Connect 存取 GitHub企業伺服器執行個體的 VPC。

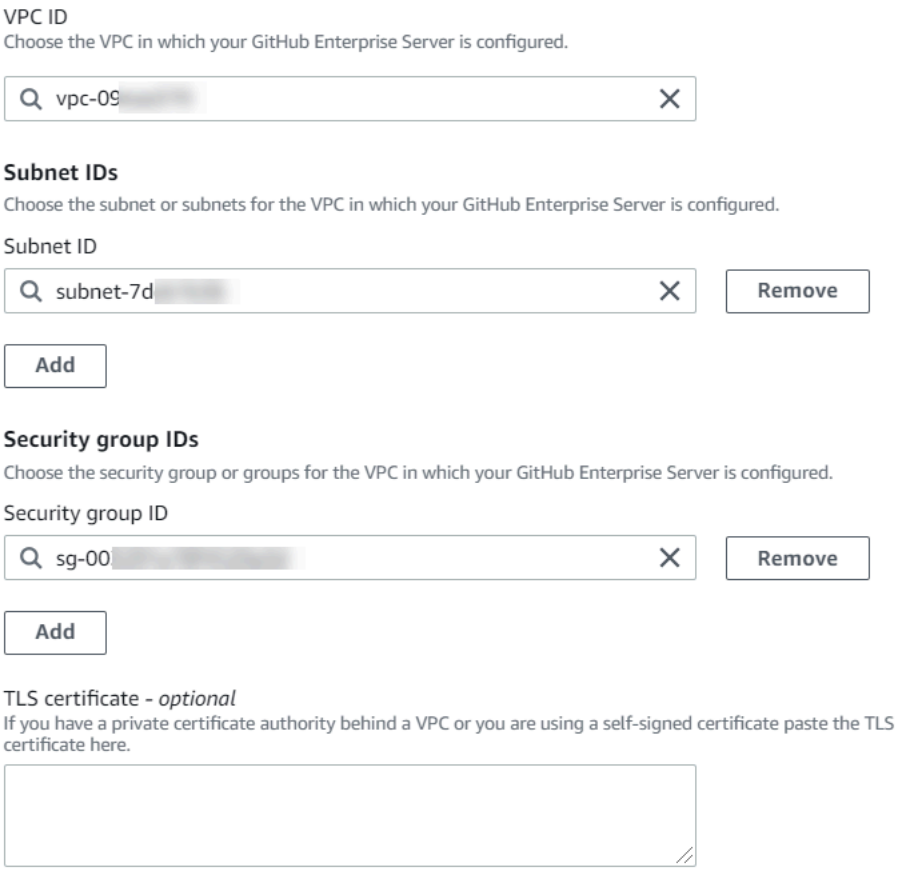
   b. 在 Subnet ID (子網路 ID) 底下，選擇 Add (新增)。在欄位中，選擇您要用於主機的子網路 ID。您最多可選擇 10 個子網路。

      請務必針對安裝 GitHub Enterprise Server 執行個體的基礎結構選擇子網路，或是可透過 VPN 或直 Connect 存取已安裝 GitHub 企業伺服器執行個體的子網路。

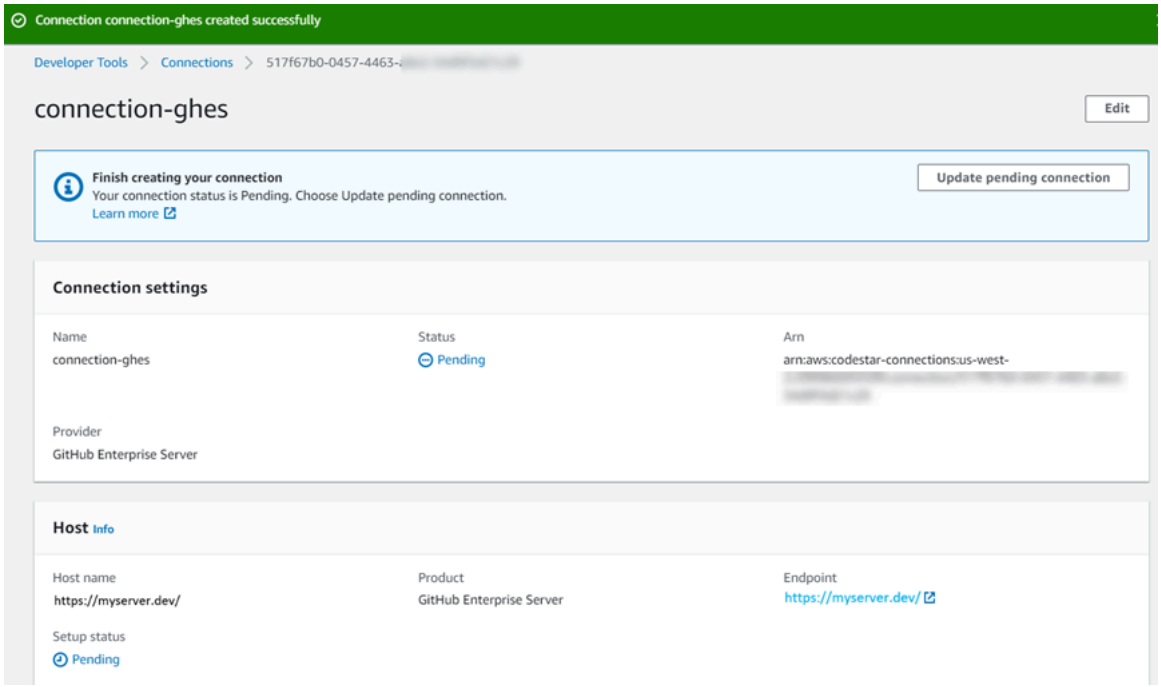
c. 在 Security group IDs (安全群組 ID) 底下，選擇 Add (新增)。在欄位中，選擇您要用於主機的安全群組。您最多可以選擇 10 個安全群組。

   請務必針對安裝 GitHub Enterprise Server 執行個體的基礎結構選擇安全性群組，或可透過 VPN 或直 Connect 存取已安裝 GitHub 企業伺服器執行個體的安全性群組。

d. 如果您已設定私有 VPC，並且已將 GitHub 企業伺服器執行個體設定為使用非公用憑證授權單位執行 TLS 驗證，請在 TLS 憑證中輸入您的憑證 ID。TLS 憑證值應該是憑證的公有金鑰。

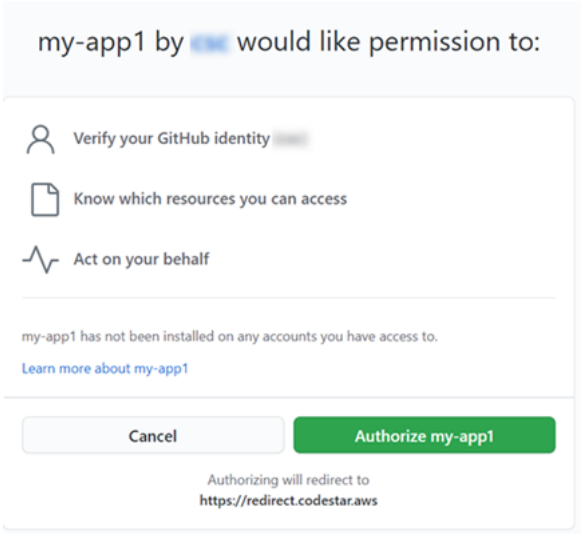


4. 選擇 [Connect 到 GitHub 企業伺服器]。建立的連線會顯示 Pending (待定) 狀態。系統會利用您提供的伺服器資訊為連線建立主機資源。主機名稱會使用 URL。
5. 選擇 Update pending connection (更新待定連線)。



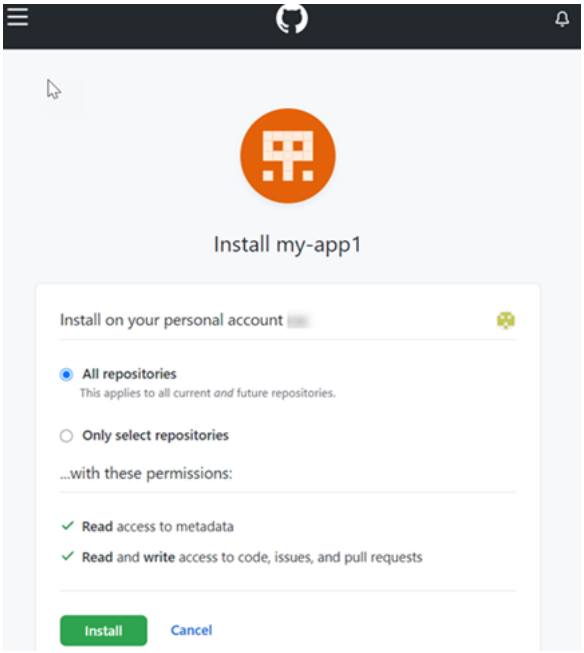
6. 如果出現提示，請在 GitHub 企業登入頁面上，使用您的 GitHub企業認證登入。
7. 在「建立 GitHub 應用程式」頁面上，選擇應用程式的名稱。

8. 在 GitHub 授權頁面上，選擇授權<app-name>。



9. 在應用程式安裝頁面上，會出現一則訊息，顯示 AWS CodeStar連接器應用程式已準備好安裝。如果您有多個組織，系統可能會提示您選擇要安裝應用程式的組織。

   選擇您要安裝應用程式的儲存庫設定。選擇 Install (安裝)。



10. 連線頁面會顯示建立的連線處於 Available (可用) 狀態。

建立與 GitHub企業伺服器 (CLI) 的連線

您可以使用 AWS Command Line Interface (AWS CLI) 來建立連線。

若要這麼做，請使用 create-host 與 create-connection 命令。

> **Important**
>
> 依預設，透過 AWS CLI 或建立 AWS CloudFormation 的連線處於PENDING狀態。建立與 CLI 的連線之後 AWS CloudFormation，或使用主控台編輯連線以顯示其狀態AVAILABLE。

步驟 1：若要建立 GitHub 企業伺服器 (CLI) 的主機

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-host令，--provider-endpoint為您的連線指定--provider-type、和。--name在此範例中，第三方供應商名稱為 GitHubEnterpriseServer，而端點為 my-instance.dev。

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

   如果成功，此命令會傳回類似下列內容的主機 Amazon Resource Name (ARN) 資訊。

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

   完成此步驟後，主機會處於 PENDING 狀態。
2. 使用主控台完成主機設定，並將主機變為 Available 狀態。如需詳細資訊，請參閱 設定待定主機。

步驟 2：在主控台中設定待處理主機

1. 登入 AWS Management Console 並開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 使用主控台完成主機設定，並將主機變為 Available 狀態。請參閱設定待定主機。

步驟 3：若要建立 GitHub 企業伺服器 (CLI) 的連線

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-connection令，--connection-name為您的連線指定--host-arn和。

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

如果成功，此命令會傳回類似下列內容的連線 ARN 資訊。

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

2. 使用主控台來設定待定連線。如需詳細資訊，請參閱 更新待定連線。

步驟 4：若要在主控台中完成 GitHub企業伺服器的連線

1. 登入 AWS Management Console 並開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 使用主控台來設定待處理連線，並將連線移動到 `Available` 狀態。如需詳細資訊，請參閱 更新待定連線。

建立連線 GitLab

您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 來建立與 gitlab.com 上託管的儲存庫的連線。

> **Note**
>
> 透過在中授權此連線安裝 GitLab，即表示您授予我們的服務處理您資料的權限，並且您可以隨時透過解除安裝應用程式來撤銷權限。

開始之前：

- 您必須已經在建立帳戶 GitLab。

> **Note**
>
> 連線只能存取用於建立和授權連線之帳戶。

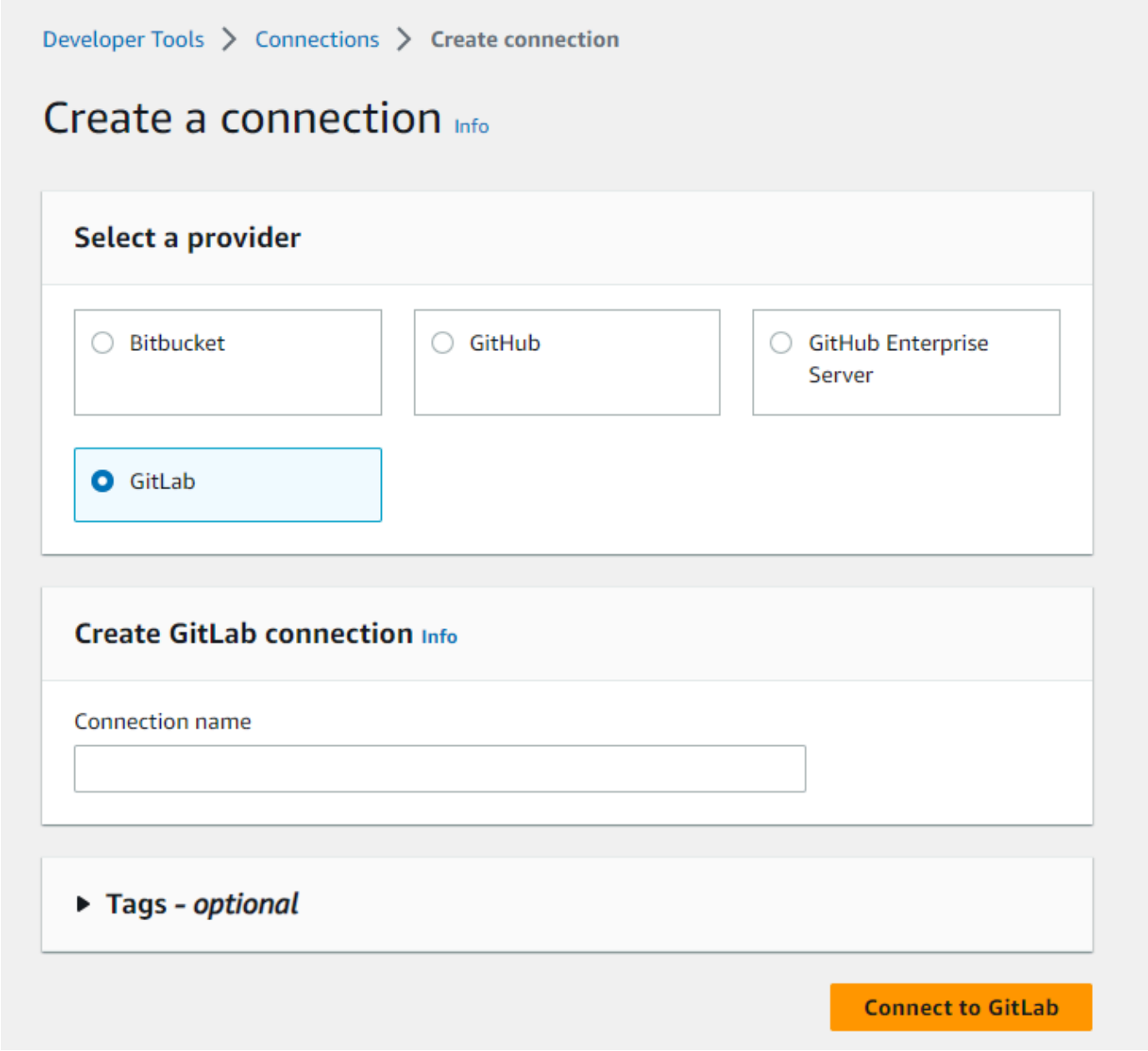
> **Note**
>
> 您可以在中建立具有 Owner 角色的連線 GitLab，然後連線可與具有諸如資源的存放庫一起使用 CodePipeline。如果是群組中的儲存庫，您不需要為群組擁有者。

主題

建立連線至 GitLab (主控台)

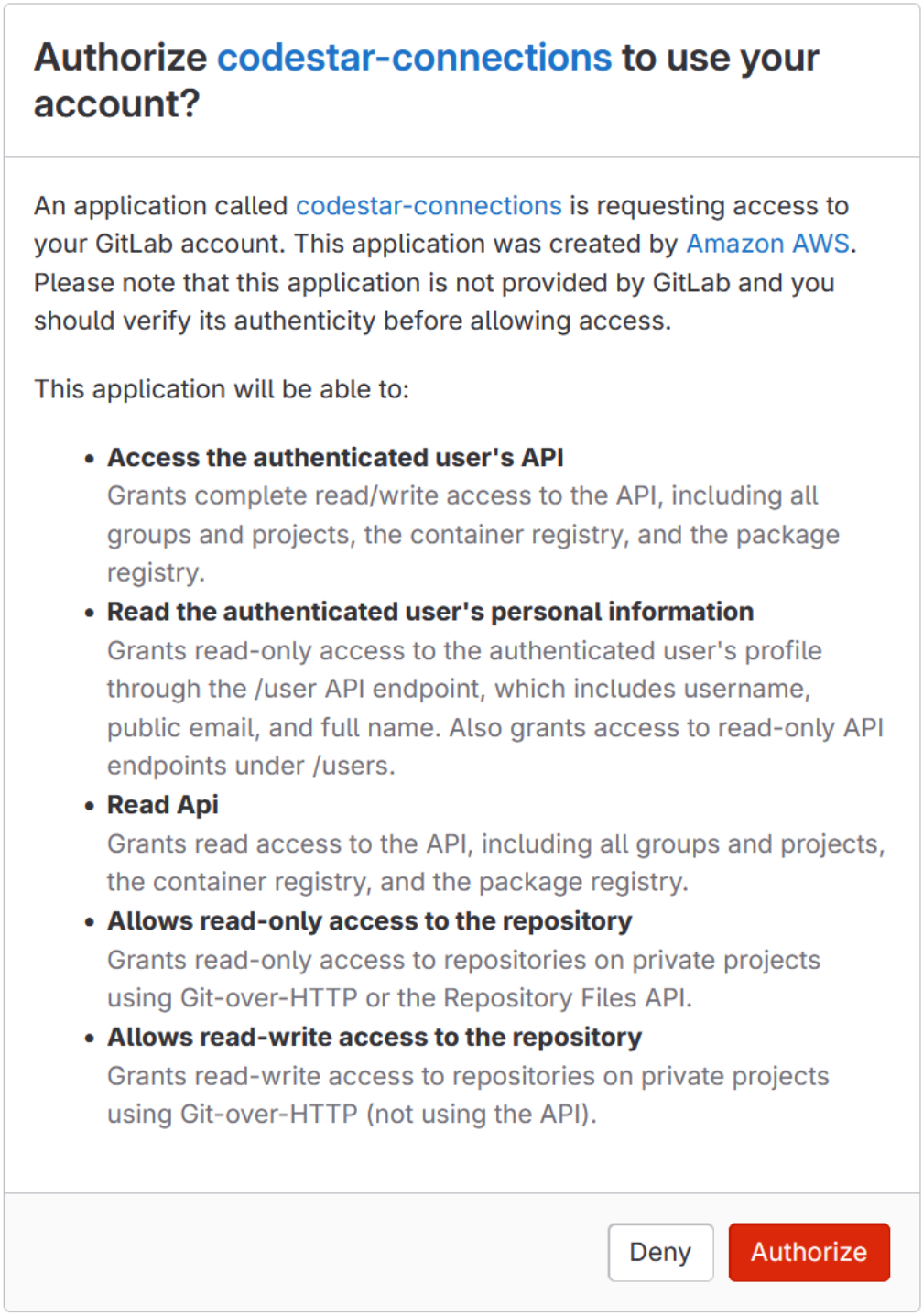
步驟 1：建立連線

1. 登入 AWS Management Console，然後在開啟 [開 AWS 發人員工具] 主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 選擇設定，然後選擇連線。選擇建立連線。
3. 若要建立 GitLab 存放庫的連線，請在 [選取提供者] 下選擇GitLab。在 Connection name (連線名稱) 底下，輸入您要建立的連線名稱。選擇「Connect 至」 GitLab。



4. GitLab 顯示的登入頁面時，使用您的認證登入，然後選擇 [登入]。
5. 隨即顯示授權頁面，其中會顯示一則訊息，要求授權存取您的 GitLab 帳戶的連線。

   選擇 Authorize (授權)。



6. 瀏覽器會返回連線主控台頁面。在「建立 GitLab 連線」下，新的連線會顯示在「連線名稱」中。
7. 選擇「Connect 至」 GitLab。

成功建立連線後，會顯示成功橫幅。連線詳細資訊會在連線設定頁面上顯示。

建立與 GitLab (CLI) 的連線

您可以使用 AWS Command Line Interface (AWS CLI) 來建立連線。

若要這麼做，請使用 create-connection 命令。

> **Important**
>
> 依預設，透過 AWS CLI 或建立 AWS CloudFormation 的連線處於PENDING狀態。建立與 CLI 的連線之後 AWS CloudFormation，或使用主控台編輯連線以顯示其狀態AVAILABLE。

若要建立連線 GitLab

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-connection令，`--connection-name`為您的連線指定`--provider-type`和。在此範例中，第三方供應商名稱為 `GitLab`，而指定的連線名稱為 `MyConnection`。

   ```
   aws codestar-connections create-connection --provider-type GitLab --connection-name
    MyConnection
   ```

   如果成功，此命令會傳回類似下列內容的連線 ARN 資訊。

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

2. 使用主控台完成連線。如需詳細資訊，請參閱 更新待定連線。

建立與 GitLab自我管理的連線

您可以使用自我管理的安裝建立 GitLab 企業版或 GitLab 社群版的連線。

您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 建立 GitLab 自我管理的連線和主機。

> **Note**
>
> 通過在 GitLab 自我管理中授權此連接應用程序，您授予我們的服務處理您的數據的權限，並且您可以通過卸載該應用程序隨時撤消許可。

在您建立與 GitLab 自我管理的連線之前，您必須先建立一個用於連線的主機，如下列步驟所述。如需為已安裝供應商建立主機工作流程的概觀，請參閱 建立或更新主機的工作流程。

您可以選擇性地使用 VPC 來設定主機。如需主機資源網路與 VPC 組態的詳細資訊，請參閱 (選用) 先決條件：連線的網路或 Amazon VPC 組態 和 針對主機的 VPC 組態進行疑難排解 中的 VPC 先決條件。

開始之前：

- 您必須已經建立了具 GitLab 有自我管理安裝的 GitLab 企業版或 GitLab 社群版的帳戶。如需詳細資訊，請參閱 https://docs.gitlab.com/ee/subscriptions/self_managed/。

  > **Note**
  >
  > 連線只能存取用於建立和授權連線之帳戶。

  > **Note**
  >
  > 您可以建立與具有 Owner 角色的存放庫的連線 GitLab，然後連線可與資源 (例如) 搭配使用 CodePipeline。如果是群組中的儲存庫，您不需要為群組擁有者。

- 您必須已經創建了具有以下範圍縮小權限的 GitLab 個人訪問令牌（PAT）：api。如需詳細資訊，請參閱 https://docs.gitlab.com/ee/user/profile/personal_access_tokens.html。只能使用管理員使用的 PAT。

  > **Note**
  >
  > 您的 PAT 會用於授權主機，不會以其他方式儲存或由連線使用。若要設置主體，您可以建立臨時 PAT，然後在設置主體後刪除 PAT。

主題

- 建立與 GitLab 自我管理 (主控台) 的連線
- 建立與 GitLab自我管理 (CLI) 的連線

建立與 GitLab 自我管理 (主控台) 的連線

使用下列步驟建立主機，並在主控台中建立與 GitLab 自我管理的連線。針對在 VPC 中設定主機的考量事項，請參閱 (選用) 先決條件：連線的網路或 Amazon VPC 組態。

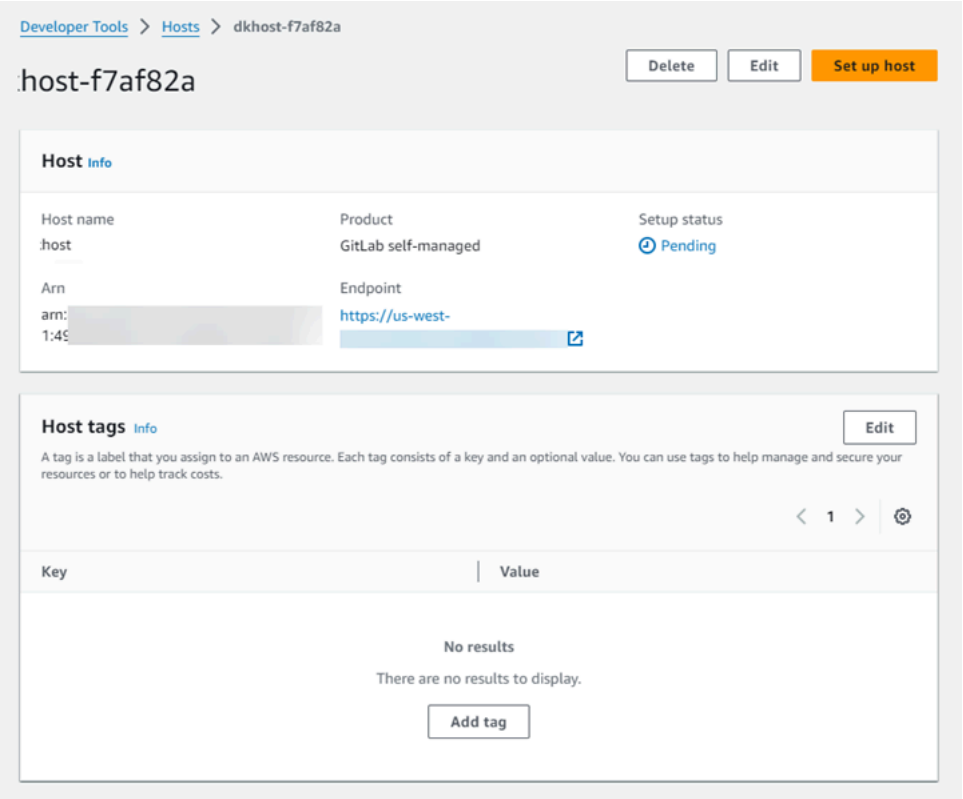
> **Note**
>
> 您可以為單一 GitLab 自我管理的安裝建立主機，然後管理與該主機的一或多個 GitLab 自我管理連線。

步驟 1：建立您的主機

1. 登入 AWS Management Console，然後在開啟 [開 AWS 發人員工具] 主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 在 Hosts (主機) 索引標籤中，選擇 Create host (建立主機)。
3. 在 Host name (主機名稱) 中，輸入您想要使用的主機名稱。
4. 在 [選取提供者] 中，選擇 [GitLab自我管理]。
5. 在 URL 中，輸入安裝供應商之基礎設施的端點。
6. 如果您的伺服器是在 Amazon VPC 內設定的，而您想要與 VPC 連線，請選擇 Use a VPC (使用 VPC)。否則，請選擇 No VPC (無 VPC)。
7. （選用）如果您已將主機啟動到 Amazon VPC 中，且想與 VPC 連線，請選擇 使用 VPC，然後完成以下操作。
   a. 在 VPC ID 底下，選擇您的 VPC ID。請務必選擇安裝主機的基礎設施之 VPC，或是可透過 VPN 或 Direct Connect 存取執行個體的 VPC。
   b. 如果您已設定私有 VPC，且已將主機設定為使用非公有憑證授權機構執行 TLS 驗證，請在 TLS 憑證) 中輸入您的憑證 ID。TLS 憑證值應該是憑證的公有金鑰。
8. 選擇 Create host (建立主機)。
9. 主機詳細資訊頁面顯示後，主機狀態會隨主機建立而變更。

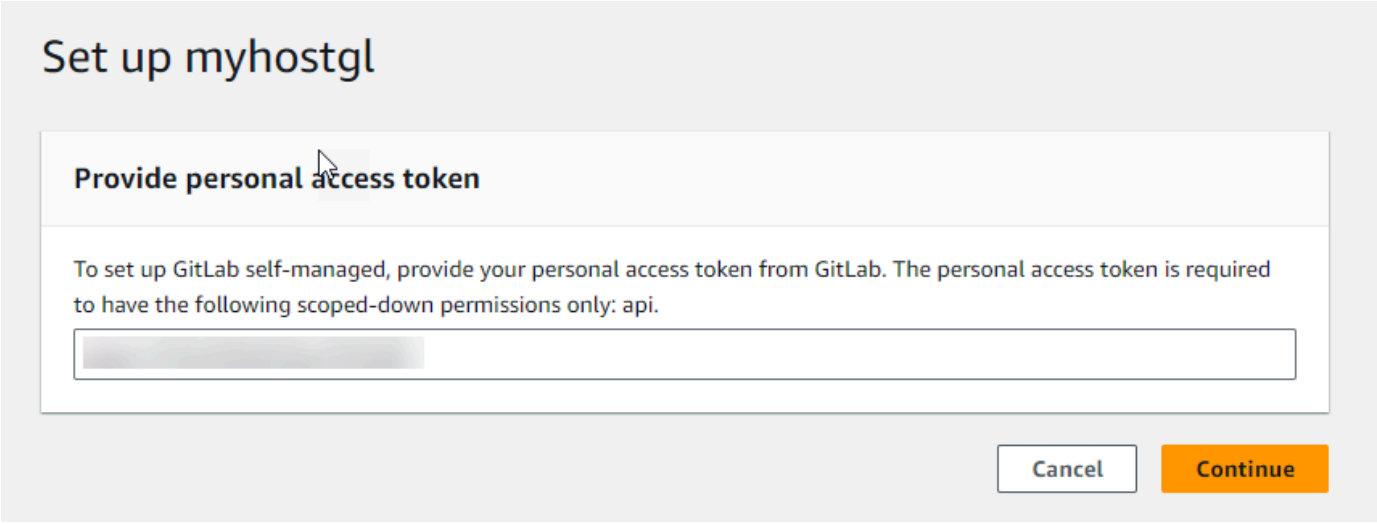
> **Note**
>
> 如果您的主機設定包含 VPC 組態，需花幾分鐘的時間來佈建主機網路元件。

等待您的主機到達 Pending (待定) 狀態，然後完成設定。如需詳細資訊，請參閱 設定待定主機。

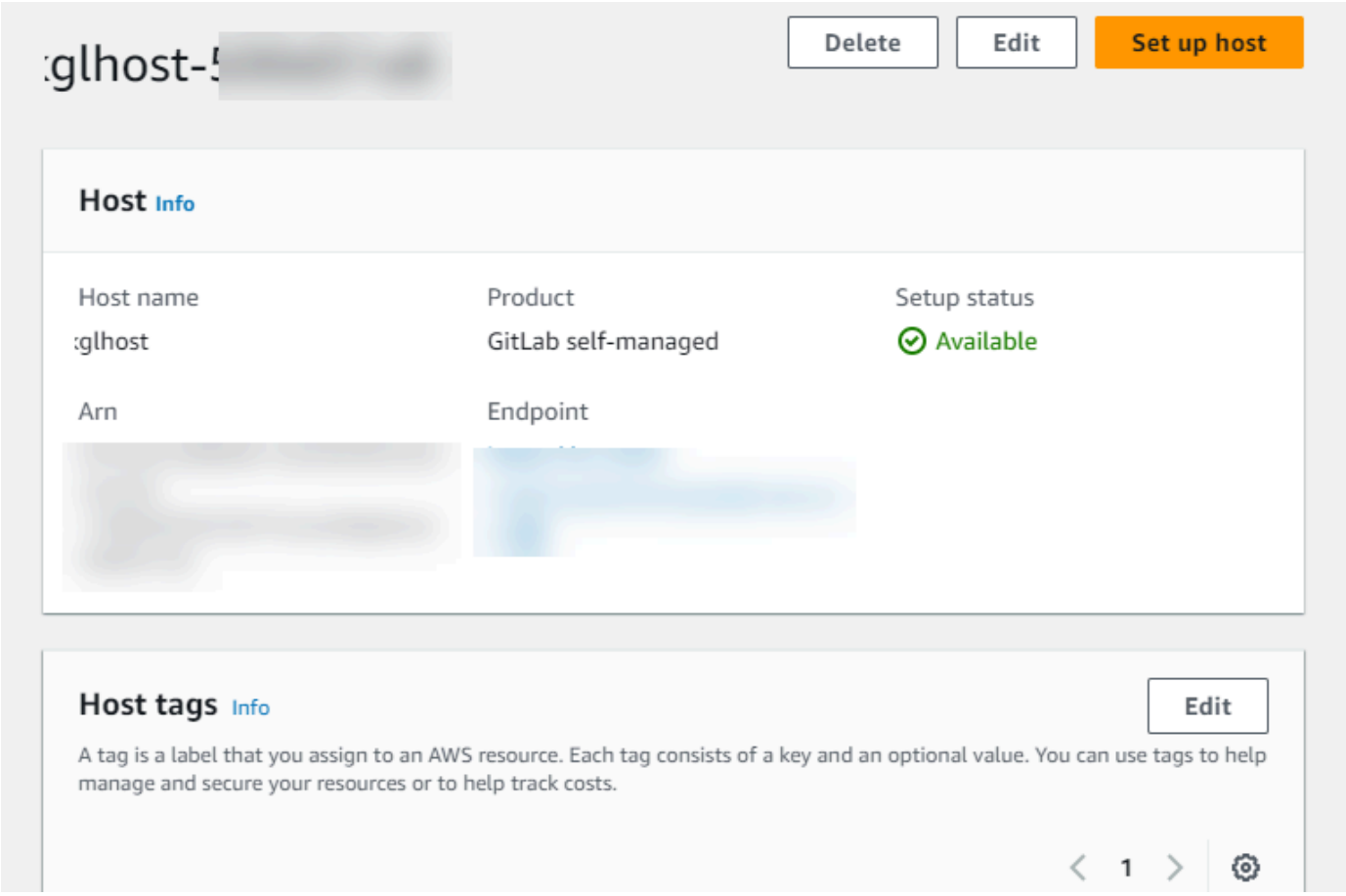


步驟 2：設定待處理主機

1. 選擇設定主機。
2. 將顯示設定 ***host_name*** 頁面。在「提供個人存取權杖」中，僅向您的 GitLab PAT 提供下列範圍權限：api。

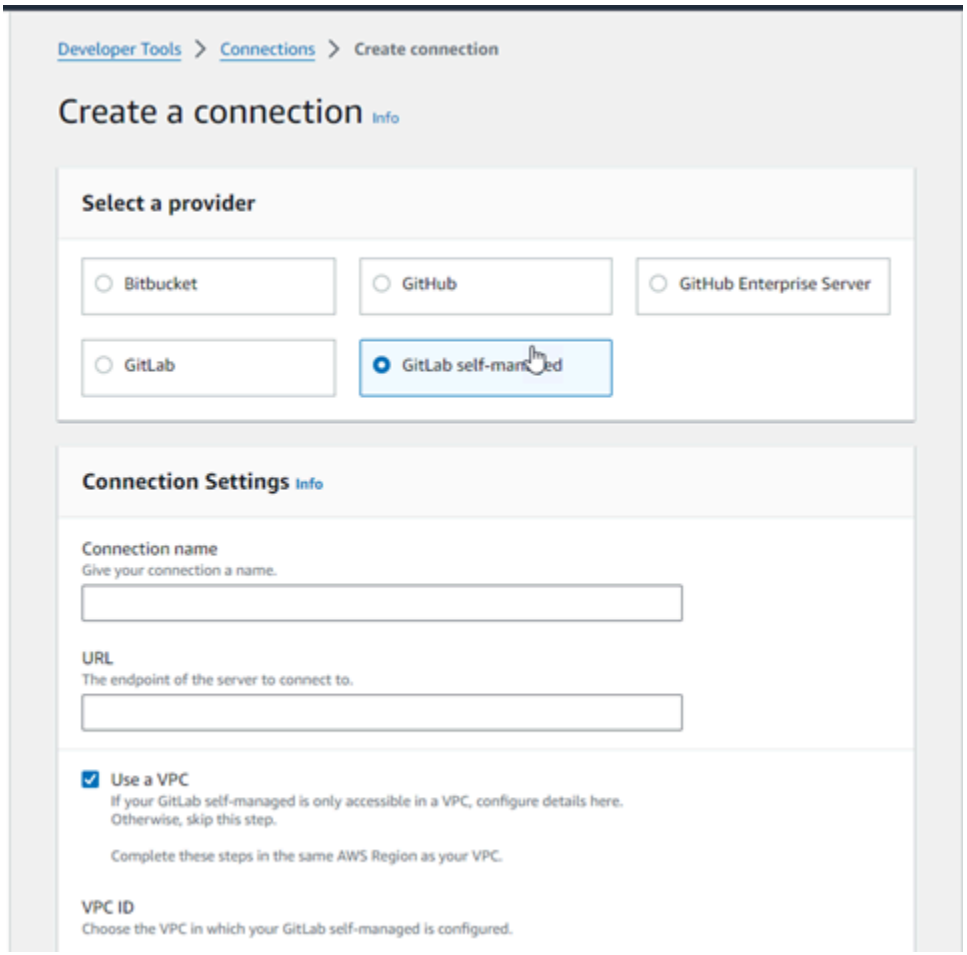


3. 主機成功註冊後，會出現主機詳細資訊頁面，並顯示主機狀態為 Available (可用)。



步驟 3：建立連線

1. 登入 AWS Management Console，然後在開啟 [開 AWS 發人員工具] 主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 選擇設定，然後選擇連線。選擇建立連線。
3. 若要建立 GitLab 存放庫的連線，請在 [選取提供者] 底下，選擇 [GitLab 自我管理]。在 Connection name (連線名稱) 底下，輸入您要建立的連線名稱。



4. 在 URL 中，輸入伺服器的端點。
5. 如果您已將伺服器啟動到 Amazon VPC 中，且想與 VPC 連線，請選擇 Use a VPC (使用 VPC)，然後完成以下操作。

   a. 在 VPC ID 底下，選擇您的 VPC ID。請務必選擇安裝主機的基礎設施之 VPC，或是可透過 VPN 或 Direct Connect 存取主機的 VPC。

   b. 在 Subnet ID (子網路 ID) 底下，選擇 Add (新增)。在欄位中，選擇您要用於主機的子網路 ID。您最多可選擇 10 個子網路。

請務必選擇安裝主機的基礎設施之子網路，或是可透過 VPN 或 Direct Connect 存取已安裝主機的子網路。

c. 在 Security group IDs (安全群組 ID) 底下，選擇 Add (新增)。在欄位中，選擇您要用於主機的安全群組。您最多可以選擇 10 個安全群組。

請務必選擇安裝主機的基礎設施之安全群組，或是可透過 VPN 或 Direct Connect 存取已安裝主機的安全群組。

d. 如果您已設定私有 VPC，且已將主機設定為使用非公有憑證授權機構執行 TLS 驗證，請在 TLS 憑證) 中輸入您的憑證 ID。TLS 憑證值應該是憑證的公有金鑰。

6. 選擇「Connect 至 GitLab 自我管理」。建立的連線會顯示 Pending (待定) 狀態。系統會利用您提供的伺服器資訊為連線建立主機資源。主機名稱會使用 URL。
7. 選擇 Update pending connection (更新待定連線)。
8. GitLab 顯示的登入頁面時，使用您的認證登入，然後選擇 [登入]。
9. 隨即顯示授權頁面，其中會顯示一則訊息，要求授權存取您的 GitLab 帳戶的連線。

選擇 Authorize (授權)。

10. 瀏覽器會返回連線主控台頁面。在「建立 GitLab 連線」下，新的連線會顯示在「連線名稱」中。
11. 選擇「Connect 至 GitLab 自我管理」。

成功建立連線後，會顯示成功橫幅。連線詳細資訊會在連線設定頁面上顯示。

建立與 GitLab自我管理 (CLI) 的連線

您可以使用 AWS Command Line Interface (AWS CLI) 來建立 GitLab自我管理的主機和連線。

若要這麼做，請使用 create-host 與 create-connection 命令。

> **Important**
>
> 依預設，透過 AWS CLI 或建立 AWS CloudFormation 的連線處於`PENDING`狀態。建立與 CLI 的連線之後 AWS CloudFormation，或使用主控台編輯連線以顯示其狀態`AVAILABLE`。

步驟 1：建立 GitLab 自我管理 (CLI) 的主機

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-host令，--provider-endpoint為您的連線指定--provider-type、和。--name在此範例中，第三方供應商名稱為 GitLabSelfManaged，而端點為 my-instance.dev。

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

如果成功，此命令會傳回類似下列內容的主機 Amazon Resource Name (ARN) 資訊。

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

完成此步驟後，主機會處於 PENDING 狀態。

2. 使用主控台完成主機設定，並依下列步驟將主機變為 Available 狀態。

步驟 2：在主控台中設定待處理主機

1. 登入 AWS Management Console 並開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 使用主控台完成主機設定，並將主機變為 Available 狀態。請參閱設定待定主機。

步驟 3：建立 GitLab 自我管理 (CLI) 的連線

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指create-connection令，--connection-name為您的連線指定--host-arn和。

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

如果成功，此命令會傳回類似下列內容的連線 ARN 資訊。

```
{
```

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

2. 在下列步驟使用主控台來設定待處理連線。

步驟 4：在主控台中完成 GitLab 自我管理的連線

1. 登入 AWS Management Console 並開啟開發人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 使用主控台來設定待處理連線，並將連線移動到 `Available` 狀態。如需詳細資訊，請參閱 更新待定連線。

更新待定連線

依預設，透過 AWS Command Line Interface (AWS CLI) 建立或處 AWS CloudFormation 於`PENDING`狀態的連線。建立與 AWS CLI 或的連線之後 AWS CloudFormation，請使用主控台更新連線以顯示其狀態`AVAILABLE`。

> **Note**
>
> 您必須使用主控台更新待定連線。您無法使用 AWS CLI更新待定連線。

第一次使用主控台將新連線新增至第三方供應商時，您必須使用與連線相關聯的安裝來完成與第三方供應商的 OAuth 交握。

您可以使用開發人員工具主控台來完成待定連線。

完成連線

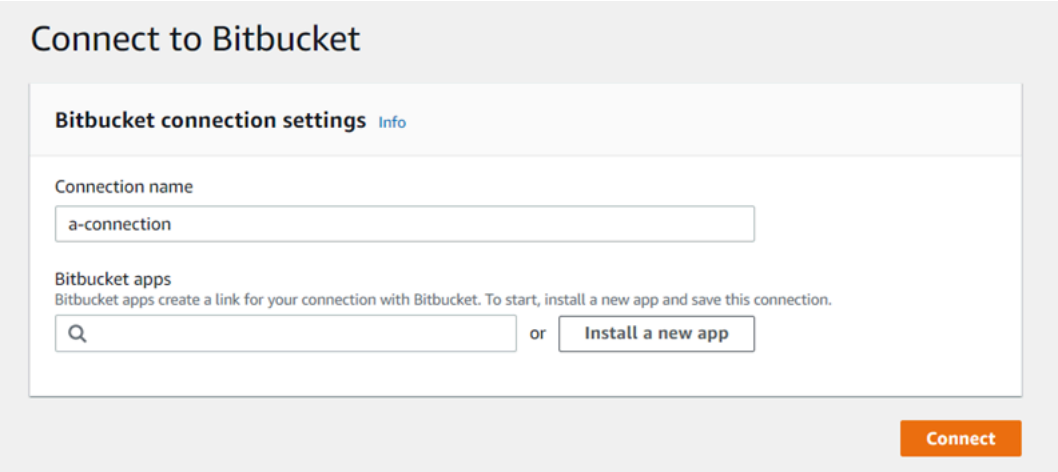
1. 開啟開發 AWS 人員工具主控台，位於https://console.aws.amazon.com/codesuite/settings/connections。
2. 選擇 Settings > Connections (設定 > 連線)。

   會顯示與您 AWS 帳戶相關聯的所有連線名稱。
3. 在 Name (名稱) 中，選擇您要更新的待定連線名稱。

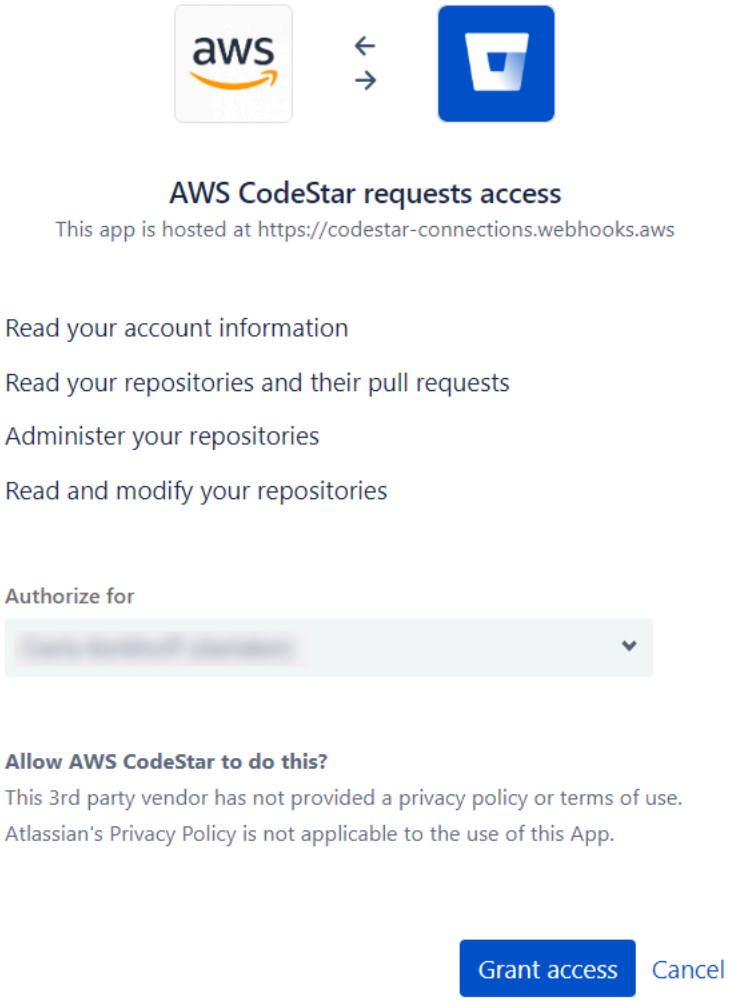
   選擇處於 Pending (待定) 狀態的連線時，會啟用 Update a pending connection (更新待定連線)。

4. 選擇 Update pending connection (更新待定連線)。
5. 在 Connect to Bitbucket (連線至 Bitbucket) 頁面的 Connection name (連線名稱) 中，確認連線的名稱。

   在 Bitbucket apps (Bitbucket 應用程式) 底下，選擇應用程式安裝，或選擇 Install a new app (安裝新應用程式) 以建立安裝。



6. 在應用程式安裝頁面上，會有訊息顯示 AWS CodeStar 應用程式正在嘗試連線至您的 Bitbucket 帳戶。選擇 Grant access (授與存取權)。



7. 隨即顯示新安裝的連線 ID。選擇 Complete connection (完成連線)。

列出連線

您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 list-connections 命令來檢視帳戶中的連線清單。

列出連線 (主控台)

列出連線

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Settings > Connections (設定 > 連線)。
3. 檢視連線的名稱、狀態和 ARN。

列出連線 (CLI)

您可以使用列出您與第三方程式碼儲存庫的連線。 AWS CLI 對於與主機資源相關聯的連接，例如連接到 GitHub Enterprise 服務器，輸出額外返回主機 ARN。

若要這麼做，請使用 list-connections 命令。

列出連線

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示字元 (視窗)，然後使用 AWS CLI 來執行list-connections指令。

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

此命令會傳回下列輸出。

```
{
     "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"
}
```

刪除一個連線

您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 delete-connection 命令來刪除連線。

主題

刪除連線 (主控台)

刪除連線

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Settings > Connections (設定 > 連線)。
3. 在 Connection name (連線名稱) 中，選擇您要刪除的連線名稱。
4. 選擇刪除。
5. 在欄位中輸入 **`delete`** 以確認，然後選擇 Delete (刪除)。

> **Important**
>
> 這個操作無法復原。

刪除連線 (CLI)

您可以使用 AWS Command Line Interface (AWS CLI) 刪除連線。

若要這麼做，請使用 delete-connection 命令。

> **Important**
>
> 執行命令之後，就會刪除連線。不會顯示確認對話方塊。您可以建立新連線，但是 Amazon Resource Name (ARN) 永遠不會重複使用。

刪除連線

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行delete-connection命令，指定要刪除之連線的 ARN。

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

此命令不會傳回任何結果。

標記連線資源

標籤是您或 AWS 指派給 AWS 資源的自訂屬性標籤。每個 AWS 標籤都有兩個部分：

- 標籤鍵 (例如，CostCenter、Environment 或 Project)。標籤鍵會區分大小寫。
- 一個名為標籤值 (例如，111122223333、Production 或團隊名稱) 的選用欄位。忽略標籤值基本上等同於使用空字串。與標籤鍵相同，標籤值會區分大小寫。

這些合稱為鍵值組。

您可以使用主控台或 CLI 以標記資源。

您可以在 CodeConnections中標記下列資源類型：

- 連線
- 主機

這些步驟假設您已安裝最新版本， AWS CLI 或已更新至目前版本。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的安裝 AWS CLI。

除了使用標籤識別、組織和追蹤資源之外，您還可以在 AWS Identity and Access Management (IAM) 政策中使用標籤來協助控制哪些人可以檢視您的資源並與其互動。如需以標籤為基礎的存取政策範例，請參閱使用標籤來控制對 AWS CodeStar 連線資源的存取。

主題

- 標記資源 (主控台)
- 標記資源 (CLI)

標記資源 (主控台)

您可以使用主控台來新增、更新或移除連線資源的標籤。

主題

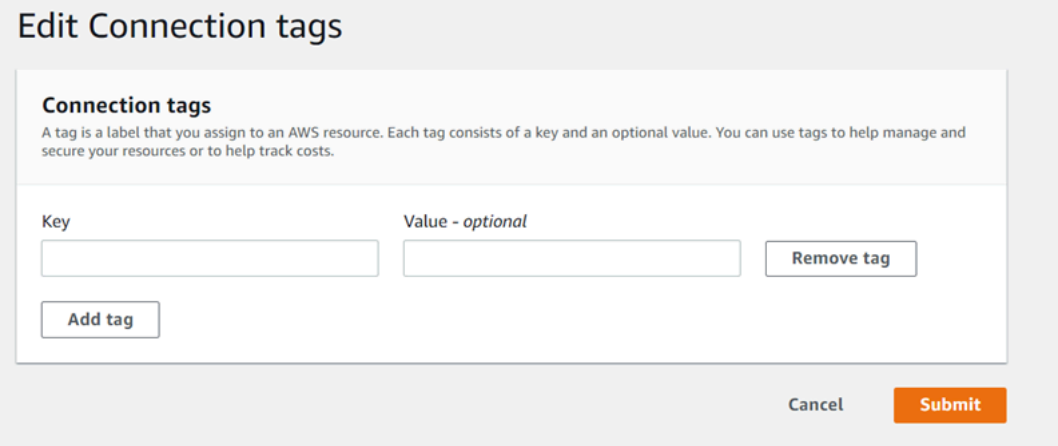
將標籤新增至連線資源 (主控台)

您可以使用主控台將標籤新增到現有連線或主機。

> **Note**
>
> 當您為已安裝的提供者 (例如 GitHub Enterprise Server) 建立連線時，也會為您建立主機資源時，建立期間的標籤只會新增至連線。如果您想要將標籤重複用於新連線，這可讓您分別標記主機。如果您想要將標籤新增到主機，請使用這裡說明的步驟。

為連線新增標籤

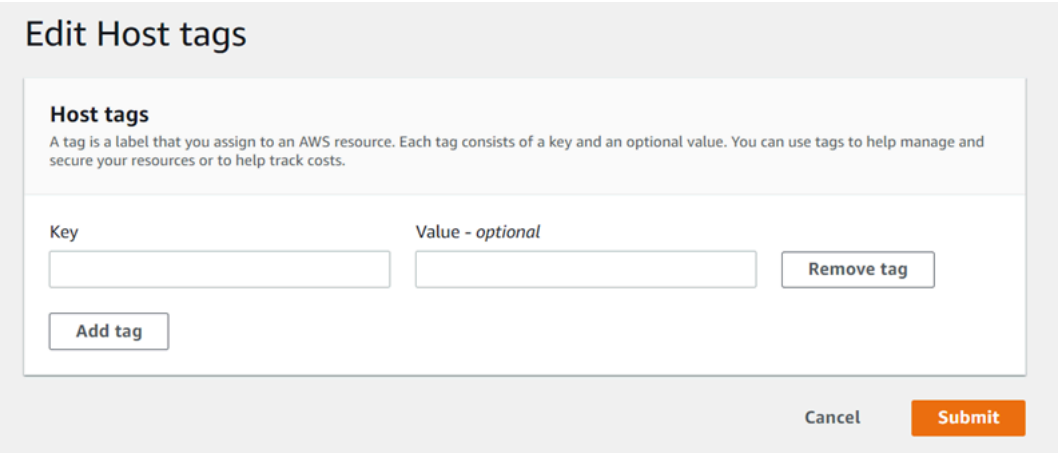
1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Connections (連線) 索引標籤。
3. 選擇您要編輯的連線。隨即會顯示連線設定頁面。
4. 在 Connection tags (連線標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Edit Connection tags (編輯連線標籤) 頁面。
5. 在 Key (索引鍵) 和ValueValue (值) 欄中，在你想新增的各組標籤中輸入金鑰對。(Value (值) 欄為選用。) 例如，在 Key (索引鍵) 中輸入 **`Project`**。在 Value (值) 中輸入 **`ProjectA`**。



6. (選用) 選擇 Add tag (新增標籤)，新增更多列，然後輸入更多標籤。
7. 選擇提交。標籤會列在連線設定之下。

為主機新增標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Hosts (主機) 索引標籤。
3. 選擇您要編輯的主機。隨即會顯示主機設定頁面。
4. 在 Host tags(主機標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Host tags (主機標籤) 頁面。
5. 在 Key (索引鍵) 和ValueValue (值) 欄中，在你想新增的各組標籤中輸入金鑰對。(Value (值) 欄為選用。) 例如，在 Key (索引鍵) 中輸入 **`Project`**。在 Value (值) 中輸入 **`ProjectA`**。



6. (選用) 選擇 Add tag (新增標籤)，新增更多列，然後為主機輸入更多標籤。
7. 選擇提交。標籤列在主機設定之下。

檢視連線資源的標籤 (主控台)

您可以使用主控台檢視現有資源的標籤。

檢視連線的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Connections (連線) 索引標籤。
3. 選擇您要檢視的連線。隨即會顯示連線設定頁面。
4. 在 Connection tags (連線標籤) 底下，檢視 Key (索引鍵) 和 Value (值) 欄下方的連線標籤。

檢視主機的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Hosts (主機) 索引標籤。
3. 選擇您要檢視的主機。
4. 在 Host tags (主機標籤) 底下，檢視 Key (索引鍵) 和 Value (值) 欄下方的主機標籤。

編輯連線資源的標籤 (主控台)

您可以使用主控台來編輯已新增到連線資源的標籤。

編輯連線的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Connections (連線) 索引標籤。
3. 選擇您要編輯的連線。隨即會顯示連線設定頁面。
4. 在 Connection tags (連線標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Connection tags (連線標籤) 頁面。
5. 在 Key (金鑰) 和 Value (加值) 欄，視需要更新每個欄位的值。例如，針對 **`Project`** 索引鍵，在 Value (值) 中將 **`ProjectA`** 變為 **`ProjectB`**。
6. 選擇提交。

編輯主機的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。

2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Hosts (主機) 索引標籤。
3. 選擇您要編輯的主機。隨即會顯示主機設定頁面。
4. 在 Host tags(主機標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Host tags (主機標籤) 頁面。
5. 在 Key (金鑰) 和 Value (加值) 欄，視需要更新每個欄位的值。例如，針對 **`Project`** 索引鍵，在 Value (值) 中將 **`ProjectA`** 變為 **`ProjectB`**。
6. 選擇提交。

移除連線資源的標籤 (主控台)

您可以使用主控台或移除連線資源的標籤。當您從關聯的資源移除標籤時，將會刪除這些標籤。

移除連線的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Connections (連線) 索引標籤。
3. 選擇您要編輯的連線。隨即會顯示連線設定頁面。
4. 在 Connection tags (連線標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Connection tags (連線標籤) 頁面。
5. 在您要刪除的金鑰和值的每個標籤旁邊，選擇 移除標籤 (Remove tag)。
6. 選擇提交。

移除主機的標籤

1. 登入主控台。從導覽窗格中，選擇 Settings (設定)。
2. 在 Settings (設定) 底下，選擇 Connections (連線)。選擇 Hosts (主機) 索引標籤。
3. 選擇您要編輯的主機。隨即會顯示主機設定頁面。
4. 在 Host tags(主機標籤) 底下，選擇 Edit (編輯)。隨即會顯示 Host tags (主機標籤) 頁面。
5. 在您要刪除的金鑰和值的每個標籤旁邊，選擇 移除標籤 (Remove tag)。
6. 選擇提交。

標記資源 (CLI)

您可以使用 CLI 來檢視、新增、更新或移除連線資源的標籤。

主題

- 將標籤新增至連線資源 (CLI)
- 檢視連線資源的標籤 (CLI)
- 編輯連線資源的標籤 (CLI)
- 移除連線資源的標籤 (CLI)

將標籤新增至連線資源 (CLI)

您可以使用 AWS CLI 來標記連線中的資源。

在終端機或命令列，執行 tag-resource 命令，為您要新增標籤的資源指定 Amazon Resource Name (ARN)，以及您想新增之標籤的索引鍵和數值。您可以新增一個以上的標籤。

為連線新增標籤

1. 為您的資源取得 ARN。使用 列出連線 中顯示的 list-connections 命令來取得連線 ARN。
2. 在終端機或命令列上執行 tag-resource 命令。

   例如，使用下列指令來標記具有兩個標籤的連接，一個名為 Pro *ject* 的標籤鍵 (具有 *Project* A 的標籤值)，以及以 *true* 標籤值命名*ReadOnly*的標籤鍵。

   ```
   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
   ```

   若成功，此命令不會傳回任何內容。

為主機新增標籤

1. 為您的資源取得 ARN。使用 列出主機 中顯示的 list-hosts 命令來取得主機 ARN。
2. 在終端機或命令列上執行 tag-resource 命令。

   例如，使用下列指令來為主機加上兩個標籤、一個名為 Pro *ject* 的標籤關鍵字 (其標籤值為 *Project* A ject)，以及以 *true* 標籤值命名*IscontainerBased*的標籤鍵。

   ```
   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
   ```

若成功，此命令不會傳回任何內容。

檢視連線資源的標籤 (CLI)

您可以使 AWS CLI 用檢視連線資源的 AWS 標籤。若未新增標籤，傳回的清單空白。使用 list-tags-for-resource 命令檢視已新增至連線或主機的標籤。

檢視連線的標籤

1. 為您的資源取得 ARN。使用 列出連線 中顯示的 list-connections 命令來取得連線 ARN。
2. 在終端機或命令列上執行 list-tags-for-resource 命令。例如，使用下列命令來檢視連線的標籤索引鍵和標籤值清單。

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

這個命令會傳回與資源相關聯的標籤。這個範例會顯示針對連線傳回的兩個索引鍵值組。

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

檢視主機的標籤

1. 為您的資源取得 ARN。使用 列出主機 中顯示的 list-hosts 命令來取得主機 ARN。
2. 在終端機或命令列上執行 list-tags-for-resource 命令。例如，使用下列命令來檢視主機的標籤索引鍵和標籤值清單。

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

這個命令會傳回與資源相關聯的標籤。此範例顯示針對主機傳回的兩個索引鍵值組。

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

編輯連線資源的標籤 (CLI)

您可以使 AWS CLI 用編輯資源的標籤。您可以變更現有索引鍵的值或新增其他索引鍵。

在終端機或命令列，執行 tag-resource 命令，指定您要更新標籤的資源 ARN，並指定要更新的標籤索引鍵和標籤值。

編輯標籤時，任何未指定的標籤索引鍵都會保留，任何項目的索引鍵都相同，但會以新值更新。使用 edit 命令新增的索引鍵會新增為一對新的鍵值組。

編輯連線的標籤

1. 為您的資源取得 ARN。使用 列出連線 中顯示的 list-connections 命令來取得連線 ARN。
2. 在終端機或命令列上執行 tag-resource 命令。

   在這個範例中，索引鍵 `Project` 的數值會變更為 `ProjectB`。

```
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
```

若成功，此命令不會傳回任何內容。若要驗證與連線相關聯的標籤，請執行 list-tags-for-resource 命令。

編輯主機的標籤

1. 為您的資源取得 ARN。使用 列出主機 中顯示的 list-hosts 命令來取得主機 ARN。
2. 在終端機或命令列上執行 tag-resource 命令。

   在這個範例中，索引鍵 `Project` 的數值會變更為 `ProjectB`。

   ```
   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
   ```

   若成功，此命令不會傳回任何內容。若要驗證與主機相關聯的標籤，請執行 list-tags-for-resource 命令。

移除連線資源的標籤 (CLI)

請按照下列步驟使 AWS CLI 用從資源中移除標籤。當您從關聯的資源移除標籤時，將會刪除這些標籤。

> **Note**
>
> 如果您刪除連線資源，則會從刪除的資源中移除所有標籤關聯。刪除連線資源之前，您不需要移除標籤。

在終端機或命令列，執行 untag-resource 命令，指定您想移除標籤的資源 ARN，和您想移除的標籤的標籤金鑰。例如，要刪除與標籤鍵 *`Project`* 連接上的多個標籤 *`ReadOnly`*，並使用以下命令。

```
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
```

若成功，此命令不會傳回任何內容。若要驗證與管道相關的標籤，請執行 list-tags-for-resource 命令。輸出顯示所有標籤皆已移除。

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

檢視連線詳細資訊

您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 get-connection 命令來檢視連線的詳細資訊。若要使用 AWS CLI，您必須已安裝的最新版本， AWS CLI 或已更新至目前版本。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的安裝 AWS CLI。

檢視連線 (主控台)

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Settings > Connections (設定 > 連線)。
3. 選擇您要檢視的連線旁邊的按鈕，然後選擇 View details (檢視詳細資訊)。
4. 隨即會顯示連線的下列資訊：
   - 連線名稱。
   - 連線的供應商類型。
   - VPN 連線狀態。
   - 連線 ARN。
   - 如果已安裝的提供者 (例如 GitHub企業伺服器) 建立連線，則為與連線相關聯的主機資訊。
   - 如果連線是針對已安裝的提供者 (例如 GitHub Enterprise Server) 建立的，則會顯示與連線之主機相關聯的端點資訊。
5. 如果連線處於 Pending (待定) 狀態，若要完成連線，請選擇 Update pending connection (更新待定連線)。如需詳細資訊，請參閱更新待定連線。

檢視連線 (CLI)

- 在終端機或命令列上執行 get-connection 命令。例如，使用下列命令來檢視具有 `arn:aws:codestar-connections:us-west-2:account_id:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f` ARN 值之連線的詳細資訊。

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

如果成功，此命令會傳回連線詳細資訊。

Bitbucket 連線的輸出範例：

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

GitHub 連線的範例輸出：

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

GitHub 企業伺服器連線的輸出範例：

```
{
    "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"
    }
```

```
}
```

使用主機

若要建立連至安裝式供應商類型 (例如 GitHub Enterprise Server) 的連線，請先使用 AWS Management Console 建立一個主機。主機是您建立的資源，用來表示安裝供應商的基礎設施。然後使用該主機建立連線。如需更多詳細資訊，請參閱 使用連線。

例如，您可以為連線建立主機，讓供應商的第三方應用程式能註冊並代表您的基礎設施。為某個供應商類型建立一個主機，然後使用這個主機建立連至該供應商類型的所有連線。

使用主控台來建立連至安裝式供應商類型 (例如 GitHub Enterprise Server) 的連線時，主控台會為您建立主機資源。

主題

- 建立主機
- 設定待定主機
- 列出主機
- 編輯主機
- 刪除主機
- 檢視主機詳細資訊

建立主機

您可以使用 AWS Management Console 或 AWS Command Line Interface (AWS CLI) 建立連至第三方程式碼儲存庫 (已安裝在您的基礎設施上) 的連線。例如，您可能在 Amazon EC2 執行個體上將 GitHub Enterprise Server 做為虛擬機器執行。建立連至 GitHub Enterprise Server 的連線之前，需先建立用於該連線的主機。

如需為已安裝供應商建立主機工作流程的概觀，請參閱 建立或更新主機的工作流程。

開始之前：

- (選用) 如果要使用 VPC 建立主機，您必須先建立好網路或虛擬私有雲端 (VPC)。
- 您必須先建立好執行個體，且如果打算與 VPC 連線，須先將主機啟動到 VPC 中。

> **Note**
>
> 每個 VPC 一次只能與一個主機相關聯。

您可以選擇性地使用 VPC 來設定主機。如需主機資源網路與 VPC 組態的詳細資訊，請參閱 (選用) 先決條件：連線的網路或 Amazon VPC 組態 和 針對主機的 VPC 組態進行疑難排解 中的 VPC 先決條件。

若要使用主控台建立主機與連至 GitHub Enterprise Server 的連線，請參閱 建立您的 GitHub企業伺服器連線 (主控台)。主控台會為您建立一個主機。

若要使用主控台建立主機與連至 GitHub 自我管理的連線，請參閱 建立與 GitLab自我管理的連線。主控台會為您建立一個主機。

(選用) 先決條件：連線的網路或 Amazon VPC 組態

如果您的基礎設施已設定網路連線，您可以略過本節。

如果您的主機只能在 VPC 中存取，請遵循這些 VPC 要求再繼續進行。

VPC 要求

您可以選擇性地使用 VPC 建立主機。以下是一般 VPC 要求，具體取決於您為安裝設定的 VPC。

- 您可以透過公有和私有子網路設定公有 VPC。如果您沒有慣用的 CIDR 區塊或子網路，您可以使用 AWS 帳戶 的預設 VPC。
- 如果您已設定私有 VPC，且已將 GitHub Enterprise Server 執行個體設定為使用非公有憑證授權機構執行 TLS 驗證，則需要為您的主機資源提供 TLS 憑證。
- AWS CodeStar Connections 建立您的主機時，系統會為您建立 Webhook 的 VPC 端點 (PrivateLink)。如需更多詳細資訊，請參閱 AWS CodeStar Connections 和界面 VPC 端點 (AWS PrivateLink)。
- 安全群組組態：
  - 主機建立期間使用的安全群組需要傳入和傳出規則，以允許網路介面連線到您的 GitHub Enterprise Server 執行個體
  - 連接到 GitHub Enterprise Server 執行個體 (不是主機設定的一部分) 的安全群組需要從連線所建立的網路介面進行傳入和傳出的權限。

- VPC 子網路必須位於您區域中的不同可用區域。可用區域是不同的位置，與其他可用區域的故障隔離。各個子網必須完全位於某一可用區域內，不得跨越多個區域。

如需使用 VPC 和子網路的詳細資訊，請參閱《Amazon VPC 使用者指南》中的 IPv4 的 VPC 和子網路規模調整。

您為主機設定提供的 VPC 資訊

在下一個步驟中為連線建立主機資源時，您需要提供下列項目：

- VPC ID：安裝 GitHub Enterprise Server 執行個體的伺服器之 VPC ID，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的 VPC。
- 子網路 ID 或 ID：安裝 GitHub Enterprise Server 執行個體的伺服器之子網路 ID，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的子網路。
- 安全群組或群組：安裝 GitHub Enterprise Server 執行個體的伺服器之安全群組，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的安全群組。
- 端點：準備好您的伺服器端點，繼續執行下一個步驟。

如需包含 VPC 或主機連線疑難排解在內的詳細資訊，請參閱「針對主機的 VPC 組態進行疑難排解」。

許可要求

在建立主機的過程中，AWS CodeStar Connections 會代表您建立網路資源，以簡化 VPC 連線。這包括用於向主機查詢資料的 AWS CodeStar Connections 網路介面，以及主機用於透過 Webhook 將事件資料傳送至 AWS CodeStar Connections 的 VPC 端點或 PrivateLink。若要建立這些網路資源，請確定建立主機所使用的角色具有下列許可：

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

如需針對 VPC 中許可或主機連線進行疑難排解的詳細資訊，請參閱「針對主機的 VPC 組態進行疑難排解」。

如需 Webhook VPC 端點的詳細資訊，請參閱「AWS CodeStar Connections 和界面 VPC 端點 (AWS PrivateLink)」。

主題

- 為連線建立主機 (主控台)
- 為連線建立主機 (CLI)

為連線建立主機 (主控台)

針對安裝連線 (如 GitHub Enterprise Server 或 GitLab 自我管理)，您可以使用主機來代表安裝第三方供應商的基礎設施之端點。

若要了解在 VPC 中設定主機的考量事項，請參閱「建立與 GitLab自我管理的連線」。

若要使用主控台建立主機與連至 GitHub Enterprise Server 的連線，請參閱 建立您的 GitHub企業伺服器連線 (主控台)。主控台會為您建立一個主機。

若要使用主控台建立主機與連至 GitHub 自我管理的連線，請參閱 建立與 GitLab自我管理的連線。主控台會為您建立一個主機。

> **Note**
>
> 一個 GitHub Enterprise Server 或 GitLab 自我管理帳戶一次只能建立一個主機。所有連至特定 GitHub Enterprise Server 或 GitLab 自我管理帳戶的連線都會使用相同的主機。

為連線建立主機 (CLI)

您可以使用 AWS Command Line Interface (AWS CLI) 為已安裝的連線建立主機。

> **Note**
>
> 一個 GitHub Enterprise Server 帳戶只能建立一個主機。所有連至特定 GitHub Enterprise Server 帳戶的連線都會使用相同的主機。

您可以使用主機來代表安裝第三方供應商的基礎設施之端點。若要使用 CLI 建立主機，請使用 create-host 命令。主機建立完成後，該主機會處於 Pending (待定) 狀態。接著需設定主機，以將其變為 Available (可用) 狀態。主機變為可用後，便可完成建立連線的步驟。

> **Important**
>
> 根據預設，透過 AWS CLI 建立的主機會處於 `Pending` 狀態。使用 CLI 建立主機後，需使用主控台將主機狀態設為 `Available`。

若要使用主控台建立主機與連至 GitHub Enterprise Server 的連線，請參閱 建立您的 GitHub企業伺服器連線 (主控台)。主控台會為您建立一個主機。

若要使用主控台建立主機與連至 GitHub 自我管理的連線，請參閱 建立與 GitLab自我管理的連線。主控台會為您建立一個主機。

設定待定主機

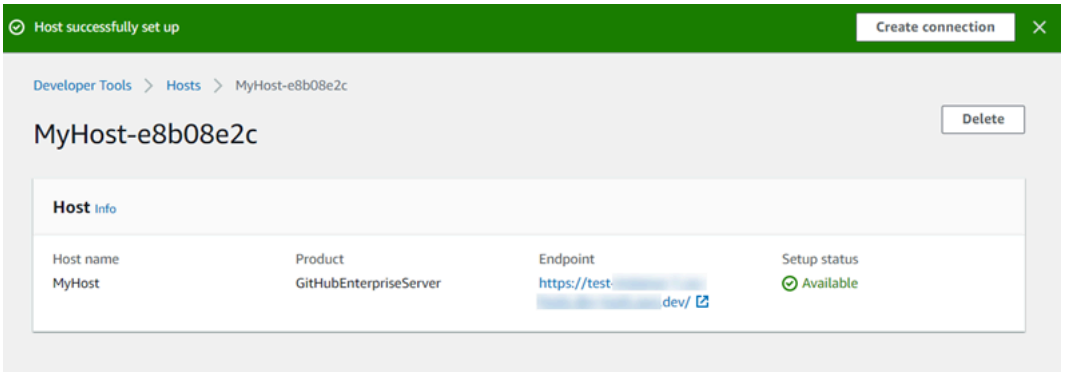
根據預設，透過 AWS Command Line Interface (AWS CLI) 或軟體開發套件建立的主機會處於 `Pending` 狀態。透過主控台、AWS CLI 或軟體開發套件建立連線後，請使用主控台將主機狀態設為 `Available`。

您必須已建立主機。如需詳細資訊，請參閱建立主機。

設定待定主機

主機建立後會處於 Pending (待定) 狀態。若要讓主機從 Pending (待定) 變為 Available (可用)，請完成以下步驟。此程序會與第三方供應商執行交握，在主機上註冊 AWS 連線應用程式。

1. 在 AWS 開發人員工具主控台上，當您的主機到達 Pending (待定) 狀態後，請選擇 Set up host。
2. 如果要為 GitLab 自我管理建立主機，則會顯示 設置 頁面。在提供個人存取字符中，僅向您的 GitLab PAT 提供下列範圍縮小許可：api。
3. 在第三方安裝式供應商登入頁面上 (例如 GitHub Enterprise Server 登入頁面)，如果出現提示，請使用您的帳戶憑證登入。
4. 在應用程式安裝頁面上，於 GitHub App name (GitHub 應用程式名稱) 中，輸入您要為主機安裝的應用程式名稱。選擇 Create GitHub App (建立 GitHub 應用程式)。
5. 主機成功註冊後，會出現主機詳細資訊頁面，並顯示主機狀態為 Available (可用)。



6. 主機變為可用後，您可以繼續建立連線。在成功橫幅上，選擇 Create connection (建立連線)。完成 Create connection (建立連線) 中的步驟。

列出主機

您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 list-connections 命令來檢視帳戶中的連線清單。

列出主機 (主控台)

列出主機

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Hosts (主機) 索引標籤。檢視主機的名稱、狀態和 ARN。

列出主機 (CLI)

您可以使用 AWS CLI 列出安裝式第三方供應商連線的主機。

若要這麼做，請使用 list-hosts 命令。

列出主機

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)，然後使用 AWS CLI 執行 list-hosts 命令。

```
aws codestar-connections list-hosts
```

此命令會傳回下列輸出。

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

編輯主機

您可以為處於 Pending 狀態的主機編輯主機設定。您可以編輯主機名稱、URL 或 VPC 組態。

您不能為多個主機使用相同的 URL。

> **Note**
>
> 若要了解在 VPC 中設定主機的考量事項，請參閱「(選用) 先決條件：連線的網路或 Amazon VPC 組態」。

編輯主機

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Settings > Connections (設定 > 連線)。
3. 選擇 Hosts (主機) 索引標籤。

   隨即會顯示與您的 AWS 帳戶相關聯且是在所選 AWS 區域建立的主機。
4. 若要編輯主機名稱，請在 Name (名稱) 中輸入新值。
5. 若要編輯主機端點，請在 URL 中輸入新值。
6. 若要編輯主機 VPC 組態，請在 VPC ID 中輸入新值。
7. 選擇 Edit host (編輯主機)。

8. 隨即會顯示更新的設定。選擇 Set up Pending host (設定待定主機)。

刪除主機

您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 delete-host 命令來刪除主機。

主題

刪除主機 (主控台)

刪除主機

1. 開啟位於 https://console.aws.amazon.com/codesuite/settings/connections 的開發人員工具主控台。
2. 選擇 Hosts (主機) 索引標籤。在 Name (名稱) 中，選擇您想刪除的主機名稱。
3. 選擇 Delete (刪除)。
4. 在欄位中輸入 **`delete`** 以確認，然後選擇 Delete (刪除)。

> **Important**
> 這個操作無法復原。

刪除主機 (CLI)

您可以使用 AWS Command Line Interface (AWS CLI) 來刪除主機。

若要這麼做，請使用 delete-host 命令。

> **Important**
> 您必須先刪除與主機相關聯的所有連線，才能刪除該主機。
> 執行命令之後，就會刪除主機。不會顯示確認對話方塊。

刪除主機

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 執行 delete-host 命令，指定要刪除的主機之 Amazon Resource Name (ARN)。

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

此命令不會傳回任何結果。

檢視主機詳細資訊

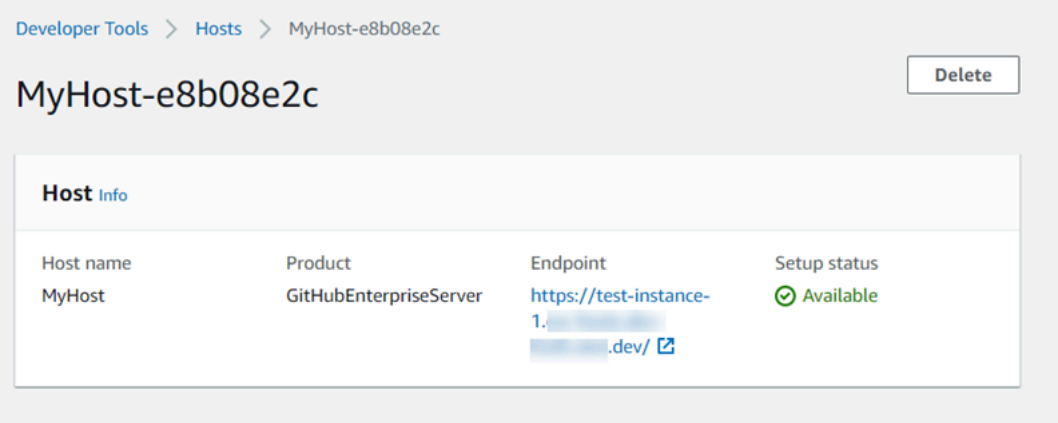
您可以使用開發人員工具主控台或 AWS Command Line Interface (AWS CLI) 中的 get-host 命令來檢視主機的詳細資訊。

檢視主機詳細資訊 (主控台)

1. 前往 https://console.aws.amazon.com/codesuite/settings/connections，登入 AWS Management Console 並開啟開發人員工具主控台。
2. 選擇 Settings > Connections (設定 > 連線)，然後選擇 Hosts (主機) 索引標籤。
3. 選擇您要檢視的主機旁邊的按鈕，然後選擇 View details (檢視詳細資訊)。
4. 隨即會顯示主機的下列資訊：
   - 主機名稱。
   - 連線的供應商類型。
   - 安裝供應商的基礎設施之端點。
   - 主機的設定狀態。準備用於連線的主機處於 Available (可用) 狀態。如果您的主機已建立但未完成安裝，該主機可能處於不同的狀態。

     可使用的狀態如下：
     - PENDING - 主機已完成建立，並準備好在主機上註冊提供商應用程式以啟動設定。
     - AVAILABLE - 主機已完成建立和設定，可與連線搭配使用。
     - ERROR - 主機建立或註冊期間發生錯誤。
     - VPC_CONFIG_VPC_INITIALIZING - 正在建立主機的 VPC 組態。
     - VPC_CONFIG_VPC_FAILED_INITIALIZATION - 主機的 VPC 組態發生錯誤並失敗。

- VPC_CONFIG_VPC_AVAILABLE - 主機的 VPC 組態已完成設定且可使用。
- VPC_CONFIG_VPC_DELETING - 正在刪除主機的 VPC 組態。



5. 選擇 Delete (刪除) 即可刪除主機。
6. 如果主機處於 Pending (待定) 狀態，若要完成設定，請選擇 Set up host (設定主機)。如需詳細資訊，請參閱設定待定主機。

檢視主機詳細資訊 (CLI)

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)，然後使用 AWS CLI 執行 get-host 命令，指定要檢視詳細資訊的主機之 Amazon Resource Name (ARN)。

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

此命令會傳回下列輸出。

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

使用連結儲存庫的同步組態

在 [AWS CodeStar 連線] 中，您可以使用連線將 AWS 資源與第三方儲存庫 (例如 GitHub Bitbucket 雲端、 GitHub 企業伺服器和 GitLab. 使用CFN_STACK_SYNC同步類型，您可以創建一個同步配置，該配置 AWS 允許從 Git 存儲庫同步內容以更新指定的 AWS 資源。 AWS CloudFormation 與連接集成，以便您可以使用 Git sync 來管理您與之同步的鏈接存儲庫中的模板和參數文件。

建立連線後，您可以使用連線 CLI 或 AWS CloudFormation 主控台來建立存放庫連結和同步組態。

- 儲存庫連結：儲存庫連結會在連線和外部 Git 儲存庫之間建立關聯。儲存庫連結可讓 Git 同步監控並同步指定 Git 儲存庫中檔案的變更。
- 同步配置：使用同步配置同步 Git 存儲庫中的內容以更新指定的 AWS 資源。

如需詳細資訊，請參閱AWS CodeStar 連線 API 參考。

如需逐步引導您使用 AWS CloudFormation 主控台建立 AWS CloudFormation 堆疊同步設定的教學課程，請參閱使用CloudFormation 者指南中的使用 AWS CloudFormation Git sync。

主題

- 使用儲存庫連結
- 使用同步組態

使用儲存庫連結

儲存庫連結會在連線和外部 Git 儲存庫之間建立關聯。儲存庫連結可讓 Git 同步監控指定 Git 儲存庫中檔案的變更，並將其同步至 AWS CloudFormation 堆疊。

如需有關儲存庫連結的詳細資訊，請參閱AWS CodeStar 連線 API 參考資料。

主題

- 建立儲存庫連結
- 更新儲存庫連結
- 列出儲存庫連結
- 刪除儲存庫連結
- 檢視儲存庫連結詳細資訊

建立儲存庫連結

您可以使用 AWS Command Line Interface (AWS CLI) 中的create-repository-link指令，在連線與要同步的外部儲存庫之間建立連結。

在建立存放庫連結之前，您必須先與第三方提供者建立外部存放庫，例如 GitHub。

建立儲存庫連結

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行命create-repository-link令。指定關聯連線的 ARN、擁有者 ID 和儲存庫名稱。

```
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. 此命令會傳回下列輸出。

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

更新儲存庫連結

您可以使用 AWS Command Line Interface (AWS CLI) 中的指update-repository-link令來更新指定的存放庫連結。

您可以更新儲存庫連結的下列資訊：

- `--connection-arn`
- `--owner-id`

- `--repository-name`

當您想要變更與儲存庫相關聯的連線時，可以更新儲存庫連結。若要使用不同的連線，您需要指定連線 ARN。如需檢視連線 ARN 的步驟，請參閱檢視連線詳細資料。

更新儲存庫連結

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用執 AWS CLI 行指update-repository-link令，為存放庫連結指定要更新的值。例如，下列命令會更新與儲存庫連結 ID 相關聯的連線。這會使用 `--connection` 參數指定新連線 ARN。

```
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. 此命令會傳回下列輸出。

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

列出儲存庫連結

您可以使用 AWS Command Line Interface (AWS CLI) 中的list-repository-links指令列出您帳戶的儲存庫連結。

列出儲存庫連結

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行命list-repository-links 令。

```
aws codestar-connections list-repository-links
```

2. 此命令會傳回下列輸出。

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

刪除儲存庫連結

您可以使用 AWS Command Line Interface (AWS CLI) 中的delete-repository-link指令來刪除存放庫連結。

刪除儲存庫連結之前，您必須先刪除與儲存庫連結相關聯的所有同步組態。

> **Important**
>
> 執行命令之後，就會刪除儲存庫連結。不會顯示確認對話方塊。您可以建立新儲存庫連結，但是 Amazon Resource Name (ARN) 不會重複使用。

刪除儲存庫連結

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指delete-repository-link令，指定要刪除之存放庫連結的 ID。

```
aws codestar-connections delete-repository-link --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173
```

此命令不會傳回任何結果。

檢視儲存庫連結詳細資訊

您可以使用 AWS Command Line Interface (AWS CLI) 中的get-repository-link指令來檢視有關存放庫連結的詳細資訊。

檢視儲存庫連結詳細資訊

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指get-repository-link令，指定存放庫連結 ID。

```
aws codestar-connections get-repository-link --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173
```

2. 此命令會傳回下列輸出。

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

使用同步組態

同步組態會在指定的儲存庫和連線之間建立關聯。使用同步組態來同步 Git 儲存庫中的內容，以更新指定的 AWS 資源。

如需有關連線的詳細資訊，請參閱AWS CodeStar 連線 API 參考資料。

主題

- 建立同步組態
- 更新同步組態
- 列出同步組態
- 刪除同步組態
- 檢視同步組態詳細資訊

建立同步組態

您可以使用 AWS Command Line Interface (AWS CLI) 中的create-repository-link指令，在連線與要同步的外部儲存庫之間建立連結。

建立同步組態之前，您必須先在連線和第三方儲存庫之間建立儲存庫連結。

建立同步組態

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行命create-repository-link令。指定關聯連線的 ARN、擁有者 ID 和儲存庫名稱。下列命令會為 AWS CloudFormation中的資源建立具有同步類型的同步組態。這也會指定儲存庫中的儲存庫分支和組態檔案。在此範例中，資源是命名為 **mystack** 的堆疊。

```
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. 此命令會傳回下列輸出。

```
{
    "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"
```

```
    }
```

更新同步組態

您可以使用 AWS Command Line Interface (AWS CLI) 中的 update-sync-configuration 命令來更新指定的同步組態。

您可以更新同步組態的下列資訊：

- `--branch`
- `--config-file`
- `--repository-link-id`
- `--resource-name`
- `--role-arn`

更新同步組態

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行命update-sync-configuration令，指定您要更新的值，以及資源名稱和同步類型。例如，下列命令會使用 `--branch` 參數更新與同步組態相關聯的分支名稱。

```
aws codestar-connections update-sync-configuration --sync-type CFN_STACK_SYNC --
resource-name mystack --branch feature-branch
```

2. 此命令會傳回下列輸出。

```
{
    "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"
    }
```

列出同步組態

您可以使用 AWS Command Line Interface (AWS CLI) 中的 list-sync-configurations 命令列出您帳戶的儲存庫連結。

列出儲存庫連結

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指list-sync-configurations令，指定同步類型和存放庫連結 ID。

```
aws codestar-connections list-sync-configurations --repository-link-id
 6053346f-8a33-4edb-9397-10394b695173  --sync-type CFN_STACK_SYNC
```

2. 此命令會傳回下列輸出。

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

刪除同步組態

您可以使用 AWS Command Line Interface (AWS CLI) 中的 delete-sync-configuration 命令來刪除同步組態。

> **Important**
>
> 執行命令之後，就會刪除同步組態。不會顯示確認對話方塊。您可以建立新同步組態，但是 Amazon Resource Name (ARN) 不會重複使用。

刪除同步組態

- 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行命delete-sync-configuration令，為您要刪除的同步配置指定同步類型和資源名稱。

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

此命令不會傳回任何結果。

檢視同步組態詳細資訊

您可以使用 AWS Command Line Interface (AWS CLI) 中的get-sync-configuration命令來檢視同步配置的詳細資料。

檢視同步組態的詳細資訊

1. 開啟終端機 (Linux、macOS 或 Unix) 或命令提示 (Windows)。使用 AWS CLI 來執行指get-sync-configuration令，指定存放庫連結 ID。

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

2. 此命令會傳回下列輸出。

```
{
    "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"
    }
}
```

使用 AWS CloudTrail 記錄 AWS CodeConnections API 呼叫

AWS CodeConnections 整合了 AWS CloudTrail，這是一種提供記錄使用者、角色或 AWS 服務所採取之動作的服務。CloudTrail 會擷取通知的 API 呼叫當作事件。擷取的呼叫包括從開發人員工具主控台進行的呼叫，以及對 AWS CodeConnections API 操作的程式碼呼叫。

如果您建立追蹤記錄，就可以將 CloudTrail 事件持續交付到 Amazon Simple Storage Service (Amazon S3) 儲存貯體，包括通知的事件。即使您未設定追蹤，仍可透過 CloudTrail 主控台中的 Event history (事件歷史記錄) 檢視最新事件。您可利用 CloudTrail 所收集的資訊來判斷向 AWS CodeConnections 發出的請求，以及發出請求的 IP 地址、人員、時間和其他詳細資訊。

如需詳細資訊，請參閱《AWS CloudTrail 使用者指南》。

CloudTrail 中的 AWS CodeConnections 資訊

當您建立帳戶時，系統即會在 AWS 帳戶中啟用 CloudTrail。當 AWS CodeConnections 中發生活動時，該活動會記錄在 CloudTrail 事件中，其他 AWS 服務事件則記錄於 Event history (事件歷史記錄) 中。您可以檢視、搜尋和下載 AWS 帳戶的最新事件。如需詳細資訊，請參閱 AWS CloudTrail 使用者指南中的 使用 CloudTrail 事件歷史記錄檢視事件。

如需您 AWS 帳戶中正在進行事件的記錄 (包含 AWS CodeConnections 的事件)，請建立線索。追蹤能讓 CloudTrail 將日誌檔交付至 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中所收集的事件資料。

如需詳細資訊，請參閱 AWS CloudTrail 使用者指南中的以下主題：

- 建立追蹤的概觀
- CloudTrail 支援的服務和整合
- 設定 CloudTrail 的 Amazon SNS 通知
- 從多個區域接收 CloudTrail 日誌檔案
- 從多個帳戶接收 CloudTrail 日誌檔案

CloudTrail 會記錄所有 AWS CodeConnections 動作，並記載於 AWS CodeConnections API 參考中。例如，對 `CreateConnection`、`DeleteConnection` 和 `GetConnection` 動作發出的呼叫會在 CloudTrail 記錄檔案中產生項目。

每一筆事件或日誌項目都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是否使用根或其他 IAM 憑證提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 該請求是否由另一項 AWS 服務提出。

如需詳細資訊，請參閱 CloudTrail userIdentity 元素。

了解 日誌檔案項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付到您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌項目。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。CloudTrail 日誌檔並非依公有 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

以下範例顯示的是展示 `CreateConnection` 動作的 CloudTrail 日誌項目。

```
{
    "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 和界面 VPC 端點 (AWS PrivateLink)

您可以建立介面 VPC 端點，以在您的 VPC 與 AWS CodeStar Connections 間建立私有連線。介面端點是由 AWS PrivateLink 提供技術支援，這項技術可讓您在沒有網際網路閘道、NAT 裝置、VPN 連接或 AWS Direct Connect 連線的情況下私密地存取 AWS CodeStar Connections API。VPC 中的執行個體不需要公有 IP 地址，就能與 AWS CodeStar Connections API 通訊，因為 VPC 與 AWS CodeStar Connections 之間的流量不會離開 Amazon 網路。

每個介面端點都是由您子網路中的一或多個彈性網絡介面表示。

如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 介面 VPC 端點 (AWS PrivateLink)。

AWS CodeStar Connections VPC 端點的考量事項

設定 AWS CodeStar Connections 的介面 VPC 端點前，請務必參閱 Amazon VPC 使用者指南中的介面端點。

AWS CodeStar Connections 支援從您的 VPC 呼叫其所有 API 動作。

所有 AWS CodeStar Connections 區域都支援 VPC 端點。

VPC 端點概念

以下是 VPC 端點的重要概念：

VPC 端點

VPC 中的進入點，可讓您以私密方式連線至服務。以下是不同類型的 VPC 端點。您可以建立受支援服務所需的 VPC 端點類型。

- AWS CodeStar Connections 動作的 VPC 端點
- AWS CodeStar Connections Webhook 的 VPC 端點

AWS PrivateLink

提供 VPC 和服務之間私有連線的技術。

AWS CodeStar Connections 動作的 VPC 端點

您可以管理 AWS CodeStar Connections 服務的 VPC 端點。

為 AWS CodeStar Connections 建立界面 VPC 端點

您可使用 Amazon VPC 主控台或 AWS Command Line Interface (AWS CLI) 來為 AWS CodeStar Connections 服務建立 VPC 端點。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的建立介面端點。

若要透過 VPC 開始使用連線，請為 AWS CodeStar Connections 建立一個界面 VPC 端點。為 AWS CodeStar Connections 建立 VPC 端點時，請選擇 AWS Services，並在 Service Name 中選擇以下項目：

- com.amazonaws.***region***.codestar-connections.api：此選項會為 AWS CodeStar Connections API 作業建立一個 VPC 端點。例如，如果您的使用者是使用 AWS CLI、AWS CodeStar Connections API 或 AWS 軟體開發套件來與 AWS CodeStar Connections 互動，以執行 `CreateConnection`、`ListConnections` 及 `CreateHost` 等作業，便需選擇此選項。

若選擇 Enable DNS name (啟用 DNS 名稱) 選項，當您為端點選取了私有 DNS，便可使用該區域的預設 DNS 名稱 (例如 `codestar-connections.us-east-1.amazonaws.com`)，向 AWS CodeStar Connections 發出 API 請求。

> **Important**
>
> 針對為 AWS 服務和 AWS Marketplace 合作夥伴服務建立的端點，私有 DNS 根據預設為啟用。

如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的透過介面端點存取服務。

為 AWS CodeStar Connections 動作建立 VPC 端點政策

您可以將端點政策連接至控制 AWS CodeStar Connections 存取權限的 VPC 端點。此政策會指定下列資訊：

- 可執行動作的主體。
- 可執行的動作。
- 可供執行動作的資源。

如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的使用 VPC 端點控制對服務的存取。

> **Note**
>
> com.amazonaws.*region*.codestar-connections.webhooks 端點不支援政策。

範例：AWS CodeStar Connections 動作的 VPC 端點政策

以下是 AWS CodeStar Connections 端點政策的範例。連接至端點後，此政策會為所有資源的所有委託人授予列出的 AWS CodeStar Connections 動作的存取權限。

```
{
  "Statement": [
    {
      "Sid": "GetConnectionOnly",
      "Principal": "*",
      "Action": [
        "codestar-connections:GetConnection"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

AWS CodeStar Connections Webhook 的 VPC 端點

建立或刪除具有 VPC 組態的主機時，AWS CodeStar Connections 會為您建立 Webhook 端點。端點名稱為 com.amazonaws.***region***.codestar-connections.webhooks。

使用 GitHub Webhook 的 VPC 端點，主機可以透過 Webhook 將事件資料經由 Amazon 網路傳送到您的整合式 AWS 服務。

> **Important**
>
> 您為 GitHub Enterprise Server, 設定主機時，AWS CodeStar Connections 會為您建立 Webhook 事件資料的 VPC 端點。如果您是在 2020 年 11 月 24 日之前建立主機，且想要使用 VPC PrivateLink Webhook 端點，您必須先刪除主機，然後建立新主機。

AWS CodeStar Connections 會管理這些端點的生命週期。若要刪除端點，您必須刪除對應的主機資源。

AWS CodeStar Connections 主機的 Webhook 端點使用方式

Webhook 端點是從第三方儲存庫傳送 Webhook 以進行 AWS CodeStar Connections 處理的位置。一個 Webhook 描述一個客戶動作。執行 `git push` 時，Webhook 端點會收到來自供應商的 Webhook 詳細解析該推送。例如，AWS CodeStar Connections 可以通知 CodePipeline 啟動您的管道。

對於不使用 VPC 的雲端供應商 (例如 Bitbucket 或 GitHub Enterprise Server 主機)，因為供應商會將 Webhook 傳送到不使用 Amazon 網路的 AWS CodeStar Connections，所以 Webhook VPC 端點不適用。

對連線進行疑難排解

以下資訊可能有助於排解 AWS CodeBuild、AWS CodeDeploy 及 AWS CodePipeline 中有關資源連線的常見疑難問題。

主題

我無法建立連線

您可能沒有建立連線的許可。如需更多詳細資訊，請參閱 權限和範例 AWS CodeConnections。

嘗試建立或完成連線時收到許可錯誤

嘗試在 CodePipeline 主控台中建立或檢視連線時，可能會傳回下列錯誤訊息。

使用者：*#####*未獲授權在資源：*## ARN*上執行*##*

如果顯示這則訊息，請確定您有足夠的許可。

在 AWS Command Line Interface (AWS CLI) 或 AWS Management Console 中建立和檢視連線的許可，只是在主控台上建立和完成連線所需許可的一部分。單純檢視、編輯或建立連線，然後完成待定連線所需的許可，範圍應該限制為只需要執行特定任務的使用者。如需更多詳細資訊，請參閱 權限和範例 AWS CodeConnections。

嘗試使用連線時收到許可錯誤

如果您嘗試在 CodePipeline 主控台中使用連線，即使您具有列出、取得和建立許可，可能仍會傳回下列其中一個或兩個錯誤訊息。

無法驗證您的帳戶。

使用者：*#####*未獲授權在資源：*## ARN ###*：codestar-connections:UseConnection

如果出現此訊息，請確定您有足夠的許可。

請確定您擁有使用連線的許可，包括列出位於供應商位置的可用儲存庫。如需更多詳細資訊，請參閱 權限和範例 AWS CodeConnections。

連線並非處於可用狀態或已脫離待定狀態

如果主控台顯示連線並非處於可用狀態的訊息，請選擇 Complete connection (完成連線)。

如果您選擇完成連線，且出現連線不處於待定狀態的訊息，您可以取消請求，因為連線已處於可用狀態。

新增 GitClone 的連線許可

當您在來源動作和 CodeBuild 動作中使用 AWS CodeStar 連線時，有兩種方式可以將輸入成品傳遞至組建：

- 預設值：來源動作會產生 zip 檔，其中包含 CodeBuild 下載項目的程式碼。

- Git 複製：來源程式碼可以直接下載到建置環境。

  Git 複製模式可讓您將原始程式碼當成工作中 Git 儲存庫來互動。若要使用此模式，您必須准許 CodeBuild 環境使用連線。

若要將許可新增至 CodeBuild 服務角色政策，請建立客戶受管政策以連接至 CodeBuild 服務角色。下列步驟建立政策，其中，`action` 欄位中指定 `UseConnection` 許可，而 `Resource` 欄位中指定連線的 Amazon Resource Name (ARN)。

使用主控台新增 UseConnection 許可

1. 若要尋找管道的連線 ARN，請開啟管道，然後選擇來源動作上的 (i) 圖示。「Configuration (組態)」窗格隨即會開啟，而連線 ARN 會顯示在 ConnectionArn 旁邊。您可以將連線 ARN 新增至 CodeBuild 服務角色政策。
2. 若要尋找 CodeBuild 服務角色，請開啟管道中使用的建置專案，然後瀏覽至 Build details (建置詳細資訊) 索引標籤。
3. 在「Environment (環境)」區段中，選擇 Service role (服務角色) 連結。這會開啟 AWS Identity and Access Management (IAM) 主控台，讓您新增政策以授予您連線的存取權。
4. 在 IAM 主控台，選擇 Attach policies (連接政策)，然後選擇 Create policy (建立政策)。

   使用下列政策範本範例。在 `Resource` 欄位中新增連線 ARN，如下列範例所示。

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Effect": "Allow",
               "Action": "codestar-connections:UseConnection",
               "Resource": "insert connection ARN here"
           }
       ]
   }
   ```

   在 JSON 索引標籤上，貼上您的政策。
5. 選擇 Review policy (檢閱政策)。輸入政策的名稱 (例如 **`connection-permissions`**)，然後選擇 Create policy (建立政策)。
6. 返回服務角色的 Attach Permissions (連接許可) 頁面，重新整理政策清單，然後選取您剛建立的政策。選擇 Attach policies (連接政策)。

主機並非處於可用狀態

如果主控台顯示訊息，指出主機並非處於 `Available` 狀態，請選擇 Set up host (設定主機)。

建立主機的第一個步驟會導致建立的主機現在處於 `Pending` 狀態。若要讓主機變為 `Available` 狀態，您必須選擇在主控台中設定主機。如需更多詳細資訊，請參閱 設定待定主機。

> **Note**
>
> 您無法使用 AWS CLI 設定 `Pending` 主機。

針對發生連線錯誤的主機進行疑難排解

如果刪除或修改基礎 GitHub 應用程式，連線和主機可能會變為錯誤狀態。處於錯誤狀態的主機和連線無法復原，且必須重新建立主機。

- 更改應用程式 pem 金鑰、更改應用程式名稱 (初始建立後) 等動作，會導致主機和所有相關聯的連線進入錯誤狀態。

如果主控台或 CLI 傳回處於 `Error` 狀態的主機或與主機相關的連線，您可能需要執行下列步驟：

- 刪除並重新建立主機資源，然後重新安裝主機註冊應用程式。如需更多詳細資訊，請參閱 建立主機。

我無法為主機建立連線

若要建立連線或主機，必須符合以下條件。

- 您的主機必須處於 AVAILABLE (可用) 狀態。如需詳細資訊，請參閱
- 必須在與主機相同的區域建立連線。

針對主機的 VPC 組態進行疑難排解

建立主機資源時，您必須提供安裝 GitHub Enterprise Server 執行個體的基礎設施之網路連線或 VPC 資訊。若要對主機的 VPC 或子網路組態進行疑難排解，請使用此處顯示的範例 VPC 資訊做為參考。

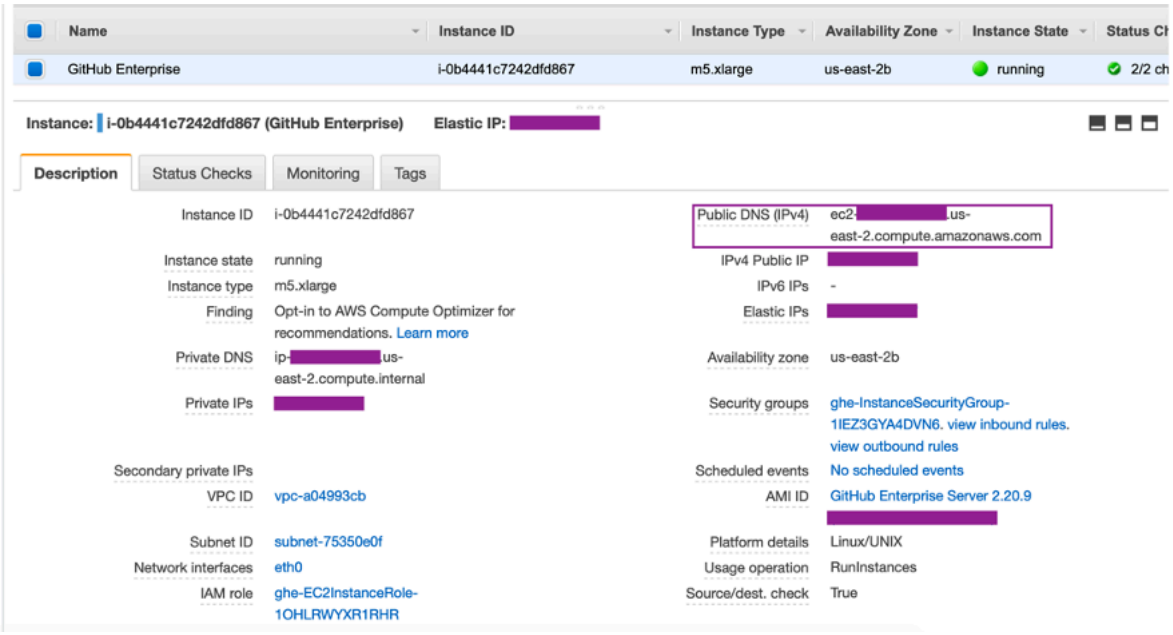
> **Note**
>
> 使用本節內容，對 Amazon VPC 內 GitHub Enterprise Server 主機組態相關問題進行疑難排解。如需針對設為使用 VPC 的 Webhook 端點 (PrivateLink) 之連線相關問題進行疑難排解，請參閱「針對 GitHub Enterprise Server 連線的 Webhook VPC 端點 (PrivateLink) 進行疑難排解」。

在此範例中，您可以使用下列程序來設定 VPC 和伺服器 (GitHub Enterprise Server 執行個體的安裝位置)：

1. 建立 VPC。如需更多詳細資訊，請參閱 https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#Create-VPC。
2. 在 VPC 中建立子網路。如需更多詳細資訊，請參閱 https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#AddaSubnet。
3. 將執行個體啟動至 VPC 中。如需更多詳細資訊，請參閱 https://docs.aws.amazon.com/vpc/latest/userguide/working-with-vpcs.html#VPC_Launch_Instance。

> **Note**
>
> 每個 VPC 一次只能與一個主機相關聯 (GitHub 企業伺服器執行個體) 相關聯。

下圖顯示使用 GitHub Enterprise AMI 啟動的 EC2 執行個體。



使用 VPC 進行 GitHub Enterprise Server 連線時，您必須在設定主機時為基礎設施提供下列資訊：

- VPC ID：安裝 GitHub Enterprise Server 執行個體的伺服器之 VPC，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的 VPC。
- 子網路 ID 或 ID：安裝 GitHub Enterprise Server 執行個體的伺服器之子網路，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的子網路。
- 安全群組或群組：安裝 GitHub Enterprise Server 執行個體的伺服器之安全群組，或可透過 VPN 或 Direct Connect 存取已安裝 GitHub Enterprise Server 執行個體的安全群組。
- 端點：準備好您的伺服器端點，繼續執行下一個步驟。

如需使用 VPC 和子網路的詳細資訊，請參閱 Amazon VPC 使用者指南中的 IPv4 的 VPC 和子網路規模調整。

主題

我無法讓主機進入待定狀態

如果您的主機進入了 VPC_CONFIG_FAILED_INITIALIZATION 狀態，可能是因為您為主機選取的 VPC、子網路或安全群組出現問題。

- VPC、子網路和安全群組都必須屬於建立主機的帳戶。
- 子網路和安全群組必須屬於選取的 VPC。
- 提供的每個子網路都必須位於不同的可用區域中。
- 建立主機的使用者必須具有下列 IAM 權限：

```
ec2:CreateNetworkInterface
ec2:CreateTags
ec2:DescribeDhcpOptionsec2:DescribeNetworkInterfaces
ec2:DescribeSubnets
ec2:DeleteNetworkInterface
ec2:DescribeVpcs
ec2:CreateVpcEndpoint
ec2:DeleteVpcEndpoints
ec2:DescribeVpcEndpoints
```

我無法讓主機進入可用狀態

如果您無法完成主機的 AWS CodeStar Connections 應用程式設定，可能是因為您的 VPC 組態或 GitHub Enterprise Server 執行個體出現問題。

- 如果您不是使用公有憑證授權機構，則需要為 GitHub Enterprise 執行個體所使用的主機提供 TLS 憑證。TLS 憑證值應該是憑證的公有金鑰。
- 您需為 GitHub Enterprise Server 執行個體的管理員才能建立 GitHub 應用程式。

我的連線/主機本來正常運作，現在卻停止運作

如果連線/主機之前正常工作，現在卻無法正常運作，可能是因為 VPC 中的組態變更或 GitHub 應用程式經過修改。請檢查以下內容：

- 連接至您為連線所建立主機資源的安全群組現在有所變更，或無法再存取 GitHub Enterprise Server。AWSCodeStar Connections 需具有可連線至 GitHub Enterprise Server 執行個體的安全群組。

- DNS 伺服器 IP 最近有所變更。檢查連接至您為連線所建立主機資源中指定的 VPC 的 DHCP 選項，即可確認是否有變更。請注意，如果您最近從 AmazonProviedDNS 移轉至自訂 DNS 伺服器，或是開始使用新的自訂 DNS 伺服器，則主機/連線會停止運作。若要解決這個問題，請刪除您現有的主機並重新建立，這麼做會將最新的 DNS 設定存放在我的資料庫中。
- 網路 ACL 設定已變更，不再允許 HTTP 連線到您的 GitHub Enterprise Server 基礎設施所在的子網路。
- GitHub Enterprise Server 上 AWS CodeStar Connections 應用程式的任何設定有所變更。修改任何組態 (例如 URL 或應用程式密碼) 可能會中斷已安裝的 GitHub Enterprise Server 執行個體與 AWS CodeStar Connections 之間的連線。

我無法刪除網路介面

如果您無法偵測到您的網路介面，請檢查下列項目：

- AWS CodeStar Connections 建立的網路介面只能藉由刪除主機來刪除。使用者無法手動刪除這些網路介面。
- 您必須具備下列許可：

```
ec2:DescribeNetworkInterfaces
ec2:DeleteNetworkInterface
```

針對 GitHub Enterprise Server 連線的 Webhook VPC 端點 (PrivateLink) 進行疑難排解

使用 VPC 組態建立主機時，系統會為您建立 Webhook VPC 端點。

> **Note**
>
> 使用本節內容，針對設為使用 VPC 的 Webhook 端點 (PrivateLink) 之連線相關問題進行疑難排解。如需對 Amazon VPC 內 GitHub Enterprise Server 主機組態相關問題進行疑難排解，請參閱「針對主機的 VPC 組態進行疑難排解」。

若您建立連至安裝式供應商類型的連線，並指定您的伺服器是在 VPC 內設定，AWS CodeStar Connections 會建立您的主機，並為您建立 Webhook 的 VPC 端點 (PrivateLink)。這可讓主機透過 Webhook 將事件資料經由 Amazon 網路傳送到您的整合式 AWS 服務。如需更多詳細資訊，請參閱 AWS CodeStar Connections 和界面 VPC 端點 (AWS PrivateLink)。

主題

- 我無法刪除 Webhook VPC 端點

我無法刪除 Webhook VPC 端點

AWS Codestar Connections 會管理主機的 Webhook VPC 端點生命週期。若要刪除端點，您必須刪除對應的主機資源。

- AWS CodeStar Connections 建立的 Webhook VPC 端點 (PrivateLink) 只能藉由刪除主機來刪除。這些端點無法手動刪除。
- 您必須具備下列許可：

```
ec2:DescribeNetworkInterfaces
ec2:DeleteNetworkInterface
```

針對 2020 年 11 月 24 日之前建立的主機進行疑難排解

自 2020 年 11 月 24 日起，AWS CodeStar Connections 設定您的主機時，同時會為您設定額外的 VPC 端點 (PrivateLink) 支援。對於在此更新之前建立的主機，請使用本節疑難排解內容。

如需更多詳細資訊，請參閱 AWS CodeStar Connections 和界面 VPC 端點 (AWS PrivateLink)。

主題

- 我有一個 2020 年 11 月 24 日之前建立的主機，而且想將 VPC 端點 (PrivateLink) 用於 Webhook
- 我無法讓主機進入可用狀態 (VPC 錯誤)

我有一個 2020 年 11 月 24 日之前建立的主機，而且想將 VPC 端點 (PrivateLink) 用於 Webhook

您為 GitHub Enterprise Server 設定主機時，系統會為您建立 Webhook 端點。連線現在使用 VPC PrivateLink Webhook 端點。如果您是在 2020 年 11 月 24 日之前建立主機，且想要使用 VPC PrivateLink Webhook 端點，您必須先刪除主機，然後建立新主機。

我無法讓主機進入可用狀態 (VPC 錯誤)

如果您的主機是在 2020 年 11 月 24 日之前建立的，而且您無法完成主機的 AWS CodeStar Connections 應用程式設定，可能是因為您的 VPC 組態或 GitHub Enterprise Server 執行個體出現問題。

您的 VPC 需要一個 NAT 閘道 (或傳出網際網路存取權)，才能讓您的 GitHub Enterprise Server 執行個體傳送 GitHub Webhook 的輸出網路流量。

無法為 GitHub 儲存庫建立連線

問題：

由於連至 GitHub 儲存庫的連線使用適用於 GitHub 的 AWS Connector，因此您需有組織擁有者許可或儲存庫的管理員許可才能建立連線。

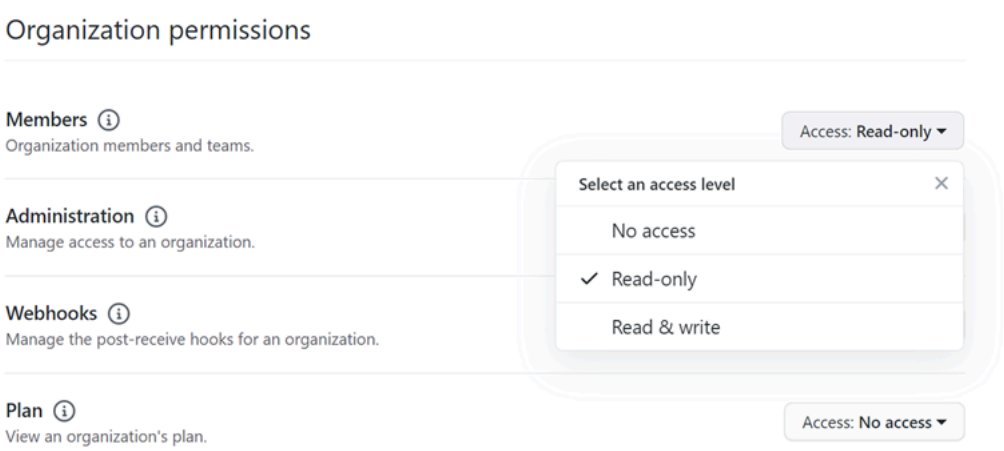
可能的修正方式：如需 GitHub 儲存庫許可層級的詳細資訊，請參閱 https://docs.github.com/en/free-pro-team@latest/github/setting-up-and-managing-organizations-and-teams/permission-levels-for-an-organization。

編輯 GitHub Enterprise Server 連線應用程式許可

如果您在 2020 年 12 月 23 日或之前安裝 GitHub Enterprise Server 的應用程式，您可能需要將應用程式的唯讀存取權提供給組織成員。如果您是 GitHub 應用程式擁有者，請依照下列步驟編輯建立主機時所安裝應用程式的許可。

> **Note**
>
> 您必須在 GitHub Enterprise Server 執行個體上完成這些步驟，而且您必須是 GitHub 應用程式擁有者。

1. 在 GitHub Enterprise Server 中，從您個人檔案照片上的下拉式選項中，選擇 Settings (設定)。
2. 選擇 Developer settings (開發人員設定)，然後選擇 GitHub Apps (GitHub 應用程式)。
3. 在應用程式清單中，為連線選擇應用程式名稱，然後在設定畫面中選擇 Permissions and events (許可與事件)。
4. 在 Organization permissions (組織許可) 下，對於 Members (成員)，從 Access (存取權) 下拉式清單中選擇 Read-only (唯讀)。



5. 在 Add a note to users (為使用者新增附註) 底下，新增更新原因說明。選擇 Save changes (儲存變更)。

連線至 GitHub 時發生連線錯誤：「發生問題，請確定您的瀏覽器已啟用 Cookie」或「組織擁有者必須安裝 GitHub 應用程式」

問題：

如要建立 GitHub 儲存庫的連線，您必須是 GitHub 組織擁有者。對於不在組織下的儲存庫，您必須是儲存庫擁有者。由組織擁有者以外的其他人員建立連線時，會針對組織擁有者建立請求，並顯示下列其中一個錯誤：

發生問題，請確保您的瀏覽器已啟用 cookie

或

組織擁有者必須安裝 GitHub 應用程式

可能修正：對於 GitHub 組織中的儲存庫，組織擁有者必須建立與 GitHub 儲存庫的連線。對於不在組織下的儲存庫，您必須是儲存庫擁有者。

我想提高連線的限額

您可以請求提高 AWS CodeStar Connections 的某些限額。如需更多詳細資訊，請參閱 連線的配額。

連線的配額

下表列出開發人員工具主控台中連線的配額 (也稱為限額)。

此表格中的配額適用於各個 AWS 區域 且可以提高。若要請求提高配額，請使用支援中心主控台。若要取得 AWS 區域 資訊和可變更的配額，請參閱 AWS 服務配額。

> **Note**
>
> 您必須先啟用歐洲 (米蘭) AWS 區域，然後才能加以使用。如需詳細資訊，請參閱啟用區域。

資源	預設限制
每個 AWS 帳戶 的連線數上限	250

此表中的配額是固定的，而且無法變更。

資源	預設限制
連線名稱字元上限	32 個字元
每個 AWS 帳戶 的主機數上限	50
儲存庫連結的數目上限	100
AWS CloudFormation 堆疊同步組態的數目上限	100
每個儲存庫連結的同步組態數目上限	100
每個分支的同步組態數目上限	50

要新增至允許清單的 IP 地址

如果您在 Amazon EC2 執行個體上實作 IP 篩選或允許特定 IP 地址，請將以下 IP 地址新增到允許清單中。這樣做可以連接到提供程序，例如 GitHub 和 Bitbucket。

下表按 AWS 區域 列出在開發人員工具主控台中連線的 IP 地址。

> **Note**
>
> 對於歐洲 (米蘭) 區域，您必須先啟用此區域，才能加以使用。如需詳細資訊，請參閱啟用區域。

區域	IP 地址
美國西部 (奧勒岡) (us-west-2)	35.160.210.199, 54.71.206.108, 54.71.36.205
美國東部 (維吉尼亞北部) (us-east-1)	3.216.216.90, 3.216.243.220, 3.217.241.85
歐洲 (愛爾蘭) (eu-west-1)	34.242.64.82, 52.18.37.201, 54.77.75.62
美國東部 (俄亥俄) (us-east-2)	18.217.188.190, 18.218.158.91, 18.220.4.80
亞太區域 (新加坡) (ap-southeast-1)	18.138.171.151, 18.139.22.70, 3.1.157.176
亞太區域 (雪梨) (ap-southeast-2)	13.236.59.253, 52.64.166.86, 54.206.1.112
亞太區域 (東京) (ap-northeast-1)	52.196.132.231, 54.95.133.227, 18.181.13.91
歐洲 (法蘭克福) (eu-central-1)	18.196.145.164, 3.121.252.59, 52.59.104.195
亞太區域 (首爾) (ap-northeast-2)	13.125.8.239, 13.209.223.177, 3.37.200.23
亞太區域 (孟買) (ap-south-1)	13.234.199.152, 13.235.29.220, 35.154.23 0.124
南美洲 (聖保羅) (sa-east-1)	18.229.77.26, 54.233.226.52, 54.233.207.69
加拿大 (中部) (ca-central-1)	15.222.219.210, 35.182.166.138, 99.79.111 .198
歐洲 (倫敦) (eu-west-2)	3.9.97.205, 35.177.150.185, 35.177.200.225
美國西部 (加利佛尼亞北部) (us-west-1)	52.52.16.175, 52.8.63.87
歐洲 (巴黎) (eu-west-3)	35.181.127.138, 35.181.145.22, 35.181.20.200
歐洲 (斯德哥爾摩) (eu-north-1)	13.48.66.148, 13.48.8.79, 13.53.78.182

區域	IP 地址
歐洲 (米蘭) (eu-south-1)	18.102.28.105, 18.102.35.130, 18.102.8.116

開發人員工具主控台功能的安全性

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，該架構專為滿足對安全性最敏感的組織的需求而打造。

安全是 AWS 與您之間共同承擔的責任。共同責任模型 將此描述為雲端的安全和雲端內的安全：

- 雲端的安全性 — AWS 負責保護在 AWS 雲端中執行 AWS 服務的基礎架構。 AWS 還為您提供可以安全使用的服務。在 AWS 合規計畫中，第三方稽核員會定期測試並驗證我們的安全功效。若要瞭解適用於「 AWS CodeStar 通知與 AWS CodeStar 連線」的規範遵循方案，請參閱規範遵循方案的 AWS 服務。
- 雲端中的安全性 — 您的責任取決於您使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的要求和適用法律和法規。

本文件可協助您瞭解如何在使用「 AWS CodeStar 通知與 AWS CodeStar 連線」時套用共用職責模型。下列主題說明如何設定「 AWS CodeStar 通知與 AWS CodeStar 連線」，以符合您的安全性與合規性目標。您還將學習如何使用其他 AWS 服務來幫助您監控和保護您的 AWS CodeStar 通知和 AWS CodeStar 連接資源。

如需深入了解開發人員工具主控台中服務的安全性，請參閱下列內容：

- CodeBuild 安全
- CodeCommit 安全
- CodeDeploy 安全
- CodePipeline 安全

了解通知內容和安全性

通知會將資源相關資訊提供給訂閱您設定之通知規則目標的使用者。這些資訊可能包含開發人員工具資源的詳細資訊，包括儲存庫內容、建置狀態、部署狀態和管道執行。

例如，您可以為中的存放庫配置通知規則， CodeCommit 以包含對提交或提取請求的註解。若是如此，為了回應該規則而傳送的通知可能包含該註解中參考的一行或多行程式碼。同樣地，您也可以在中設定建置專案的通知規則， CodeBuild 以納入組建狀態和階段的成功或失敗。為了回應該規則而傳送的通知將包含該資訊。

您可以在中設定管道的通知規則， CodePipeline 以包含手動核准的相關資訊，而針對該規則傳送的通知可能包含提供該核准的人員姓名。您可以在中設定應用程式的通知規則， CodeDeploy 以指出部署成功，而針對該規則傳送的通知可能包含部署目標的相關資訊。

通知可包含專案專屬資訊，例如建置狀態、具有註解的程式碼行、部署狀態，以及管道核准。為了協助確保專案的安全性，請務必定期檢閱通知規則的目標，以及指定為目標之 Amazon SNS 主題的訂閱者清單。此外，為了回應事件而傳送的通知內容可能會隨著其他功能新增到基礎服務而變更。此變更可能會在不通知現有通知規則的情況下發生。請考慮定期檢閱通知訊息的內容，以協助確保您了解傳送的內容以及傳送的對象。

如需通知規則可用的事件類型的詳細資訊，請參閱通知概念。

您可以選擇將通知中包含的詳細資訊限制為僅包含在事件中的資訊。這就是所謂的基本詳細資訊類型。這些事件包含與傳送到 Amazon EventBridge 和 Amazon CloudWatch 活動完全相同的資訊。

開發人員工具主控台服務 (例如) 可能會選擇在通知訊息中新增有關其部分或所有事件類型的資訊 CodeCommit，而不是事件中可用的資訊。您可以隨時新增此補充資訊，以增強目前的事件類型或補充未來的事件類型。您可以選擇 Full (完整) 詳細資訊類型，以決定將事件相關的補充資訊 (若有的話) 納入通知中。如需更多詳細資訊，請參閱 詳細資訊類型。

AWS CodeStar Notifications 和 AWS CodeStar Connections 中的資料保護功能

AWS 共同的責任模型適用於 AWS CodeStar Notifications 和 AWS CodeStar Connections 中的資料保護功能。如此模型所述，AWS 負責保護執行所有 AWS 雲端 的全球基礎設施。您必須負責維護在此基礎設施上託管之內容的控制權。您也必須負責您所使用的 AWS 服務 安全性設定和管理工作。如需有關資料隱私權的詳細資訊，請參閱資料隱私權常見問答集。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 AWS 共同的責任模型和 GDPR 部落格文章。

基於資料保護目的，建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 憑證，並設定個人使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用 AWS CloudTrail 設定 API 和使用者活動日誌記錄。
- 使用 AWS 加密解決方案，以及 AWS 服務 內的所有預設安全控制項。
- 使用進階的受管安全服務(例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。

- 如果您在透過命令列介面或 API 存取 AWS 時，需要 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的詳細資訊，請參閱聯邦資訊處理標準 (FIPS) 140-2 概觀。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您使用 AWS CodeStar Notifications 和 AWS CodeStar Connections，或採用主控台、API、AWS CLI 或 AWS 軟體開發套件的其他 AWS 服務 時。您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

AWS CodeStar 通知與連線的身分識別 AWS CodeStar 與存取管理

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM 管理員控制哪些人可以驗證 (登入) 和授權 (具有權限) 以使用 AWS CodeStar 通知和 AWS CodeStar 連線資源。IAM 是您可以使用的 AWS 服務 ，無需額外付費。

主題

- 物件
- 使用身分驗證
- 使用政策管理存取權
- 開發人員工具主控台中的功能如何與 IAM 搭配使用
- AWS CodeConnections 權限參考
- 身分型政策範例
- 使用標籤來控制對 AWS CodeStar 連線資源的存取
- 在主控台中使用通知和連線
- 允許使用者檢視他們自己的許可
- 疑難排解 AWS CodeStar 通知與 AWS CodeStar 連線身分與存取
- 針對 AWS CodeStar Notifications 使用服務連結角色
- 使用 AWS CodeConnections 的服務連結角色
- AWS CodeConnections 的 AWS 受管政策

物件

您使用 AWS Identity and Access Management (IAM) 的方式會根據您在「 AWS CodeStar 通知和 AWS CodeStar 連線」中執行的工作而有所不同。

服務使用者 — 如果您使用「 AWS CodeStar 通知和 AWS CodeStar 連線」服務來執行工作，則管理員會為您提供所需的認證和權限。當您使用更多「 AWS CodeStar 通知」和「 AWS CodeStar 連線」功能來完成工作時，您可能需要其他權限。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果您無法存取「 AWS CodeStar 通知與 AWS CodeStar 連線」中的功能，請參閱疑難排解 AWS CodeStar 通知與 AWS CodeStar 連線身分與存取。

服務管理員 — 如果您負責公司的「 AWS CodeStar 通知」和「 AWS CodeStar 連線」資源，您可能擁有「 AWS CodeStar 通知和 AWS CodeStar 連線」的完整存取權。決定您的服務使用者應存取哪些「 AWS CodeStar 通知」和「 AWS CodeStar 連線」功能和資源是您的工作。接著，您必須將請求提交給您的 IAM 管理員，來變更您服務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要進一步了解貴公司如何透過 AWS CodeStar 通知和 AWS CodeStar 連線使用 IAM，請參閱開發人員工具主控台中的功能如何與 IAM 搭配使用。

IAM 管理員 — 如果您是 IAM 管理員，您可能想要瞭解如何撰寫政策以管理 AWS CodeStar 通知和 AWS CodeStar 連線存取權限的詳細資訊。若要檢視您可以在 IAM 中使用的「 AWS CodeStar 通知和 AWS CodeStar 連線」以身分識別為基礎的政策範例，請參閱。 身分型政策範例

使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以 IAM 使用者身分或假設 IAM 角色進行驗證 (登入 AWS)。 AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。 AWS IAM Identity Center (IAM 身分中心) 使用者、貴公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料都是聯合身分識別的範例。您以聯合身分登入時，您的管理員先前已設定使用 IAM 角色的聯合身分。當您使 AWS 用同盟存取時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需有關登入的詳細資訊 AWS，請參閱《AWS 登入 使用指南》 AWS 帳戶中的如何登入您的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以便使用您的認證以加密方式簽署要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱 IAM 使用者指南中的簽署 AWS API 請求。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如， AWS 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。如需更多資訊，請參閱 AWS IAM Identity Center 使用者指南中的多重要素驗證和 IAM 使用者指南中的在 AWS中使用多重要素驗證 (MFA)。

AWS 帳戶根使用者

當您建立時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需這些任務的完整清單，了解需以根使用者登入的任務，請參閱《IAM 使用者指南》中的需要根使用者憑證的任務。

IAM 使用者和群組

IAM 使用者是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定許可。建議您盡可能依賴暫時憑證，而不是擁有建立長期憑證 (例如密碼和存取金鑰) 的 IAM 使用者。但是如果特定使用案例需要擁有長期憑證的 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱 IAM 使用者指南中的為需要長期憑證的使用案例定期輪換存取金鑰。

IAM 群組是一種指定 IAM 使用者集合的身分。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為 IAMAdmins 的群組，並給予該群組管理 IAM 資源的許可。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。如需進一步了解，請參閱 IAM 使用者指南中的建立 IAM 使用者 (而非角色) 的時機。

IAM 角色

IAM 角色是您 AWS 帳戶 內部具有特定許可的身分。它類似 IAM 使用者，但不與特定的人員相關聯。您可以切換角色，在中暫時擔任 IAM 角色。 AWS Management Console 您可以透過呼叫 AWS CLI 或 AWS API 作業或使用自訂 URL 來擔任角色。如需使用角色的方法更多相關資訊，請參閱 IAM 使用者指南中的使用 IAM 角色。

使用暫時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 – 若要向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱 IAM 使用者指南中的為第三方身分提供者建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的許可集。
- 暫時 IAM 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。
- 跨帳戶存取權 – 您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的委託人) 存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加

到資源（而不是使用角色作為代理）。若要了解跨帳戶存取權角色和資源型政策間的差異，請參閱 IAM 使用者指南中的 IAM 角色與資源類型政策的差異。

- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (Amazon S3) 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 轉寄存取工作階段 (FAS) — 當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱《轉發存取工作階段》。
  - 服務角色 – 服務角色是服務擔任的 IAM 角色，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需更多資訊，請參閱 IAM 使用者指南中的建立角色以委派許可給 AWS 服務。
  - 服務連結角色 — 服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 — 您可以使用 IAM 角色來管理在 EC2 執行個體上執行的應用程式以及發出 AWS CLI 或 AWS API 請求的臨時登入資料。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並提供給其所有應用程式，請建立連接至執行個體的執行個體設定檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得暫時憑證。如需更多資訊，請參閱 IAM 使用者指南中的利用 IAM 角色來授予許可給 Amazon EC2 執行個體上執行的應用程式。

若要了解是否要使用 IAM 角色或 IAM 使用者，請參閱 IAM 使用者指南中的建立 IAM 角色 (而非使用者) 的時機。

使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS ，當與身分識別或資源相關聯時，會定義其權限。 AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以 JSON 文件的形式儲存在中。如需 JSON 政策文件結構和內容的更多相關資訊，請參閱 IAM 使用者指南中的 JSON 政策概觀。

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

IAM 政策定義該動作的許可，無論您使用何種方法來執行操作。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該原則的使用者可以從 AWS Management Console AWS CLI、或 AWS API 取得角色資訊。

身分型政策

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分類型政策，請參閱 IAM 使用者指南中的建立 IAM 政策。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶. 受管政策包括 AWS 受管政策和客戶管理的策略。如需了解如何在受管政策及內嵌政策間選擇，請參閱《IAM 使用者指南》中的在受管政策和內嵌政策間選擇。

開發人員工具主控台中的功能如何與 IAM 搭配使用

使用 IAM 管理開發人員工具主控台功能的存取權前，應先了解可與此主控台搭配使用的 IAM 功能有哪些。若要取得通知和其他 AWS 服務如何與 IAM 搭配運作的高階檢視，請參閱 IAM 使用者指南中的與 IAM 搭配使用的AWS 服務。

主題

- 開發人員工具主控台中的身分型政策
- AWS CodeStar 通知和 AWS CodeStar 連接資源為基礎的策略
- 以標籤為基礎的授權
- IAM 角色

開發人員工具主控台中的身分型政策

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。AWS CodeStar 通知和 AWS CodeStar 連線支援特定動作、資源和條件索引鍵。若要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素參考。

動作

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼条件下可以對什麼資源執行哪些動作。

JSON 政策的 Action 元素描述您可以用來允許或拒絕政策中存取的動作。原則動作通常與關聯的 AWS API 作業具有相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

開發人員工具主控台中通知的政策動作在動作前面使用下列字首：codestar-notifications and codestar-connections。例如，若要授予某人檢視其帳戶中的所有通知規則，請在其政策中包含 codestar-notifications:ListNotificationRules 動作。原則陳述式必須包含Action或NotAction元素。 AWS CodeStar 「通知和 AWS CodeStar 連線」會定義它自己的一組動作，用來描述您可以使用此服務執行的工作。

若要在單一陳述式中指定多個「 AWS CodeStar 通知」動作，請以逗號分隔，如下所示。

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

若要在單一陳述式中指定多個 AWS CodeConnections 動作，請以逗號分隔，如下所示。

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

您也可以使用萬用字元 (*) 來指定多個動作。例如，如需指定開頭是 List 文字的所有動作，請包含以下動作：

```
"Action": "codestar-notifications:List*"
```

AWS CodeStar 通知 API 動作包括：

- CreateNotificationRule
- DeleteNotificationRule
- DeleteTarget

- DescribeNotificationRule
- ListEventTypes
- ListNotificationRules
- ListTagsForResource
- ListTargets
- Subscribe
- TagResource
- Unsubscribe
- UntagResource
- UpdateNotificationRule

AWS CodeConnections API 動作包括下列項目：

- CreateConnection
- DeleteConnection
- GetConnection
- ListConnections
- ListTagsForResource
- TagResource
- UntagResource

在中需要以下僅權限操作才 AWS CodeConnections 能完成身份驗證握手：

- GetIndividualAccessToken
- GetInstallationUrl
- ListInstallationTargets
- StartOAuthHandshake
- UpdateConnectionInstallation

在使用連 AWS CodeConnections 接時，需要執行以下僅限權限的操作：

- UseConnection

若要將連線傳遞 AWS CodeConnections 至服務，需要執行下列僅限權限的動作：

- `PassConnection`

若要查看「 AWS CodeStar 通知和 AWS CodeStar 連線」動作清單，請參閱 IAM 使用指南中的「通 AWS CodeStar 知」和「 AWS CodeStar 連線定義的動作」。

資源

AWS CodeStar 通知和 AWS CodeStar 連線不支援在策略中指定資源 ARN。

條件索引鍵

AWS CodeStar 通知和 AWS CodeStar 連接定義自己的條件鍵集，並支持使用一些全局條件鍵。若要查看所有 AWS 全域條件金鑰，請參閱《IAM 使用者指南》中的AWS 全域條件內容金鑰。

所有 AWS CodeStar 通知動作都支援`codestar-notifications:NotificationsForResource`條件索引鍵。如需詳細資訊，請參閱 身分型政策範例。

AWS CodeConnections 定義下列可用於 IAM 政策`Condition`元素的條件金鑰。您可以使用這些索引鍵來縮小套用政策陳述式的條件。如需詳細資訊，請參閱 AWS CodeConnections 權限參考。

條件索引鍵	描述
`codestar-connections:BranchName`	根據第三方儲存庫的分支名稱來篩選存取權
`codestar-connections:FullRepositoryId`	透過請求中傳遞的儲存庫來篩選存取。僅適用於用來存取特定儲存庫的 `UseConnection` 請求
`codestar-connections:InstallationId`	根據用來更新連線的第三方 ID (例如 Bitbucket 應用程式安裝 ID) 來篩選存取權。可讓您限制哪些第三方應用程式安裝可以用來建立連線
`codestar-connections:OwnerId`	根據第三方供應商的擁有者或帳戶 ID 來篩選存取權
`codestar-connections:PassedToService`	根據委託人允許傳遞連線的服務來篩選存取權

條件索引鍵	描述
codestar-connections:ProviderAction	根據 UseConnection 請求中的供應商動作 (例如 ListRepositories) 來篩選存取權。
codestar-connections:ProviderPermissionsRequired	根據第三方供應商許可類型來篩選存取權
codestar-connections:ProviderType	根據請求中傳遞的第三方供應商類型來篩選存取
codestar-connections:ProviderTypeFilter	根據用來篩選結果的第三方供應商類型來篩選存取
codestar-connections:RepositoryName	根據第三方儲存庫名稱來篩選存取權

範例

若要檢視以身分識別為基礎的 AWS CodeStar 原則範例，請參閱。 AWS CodeStar 身分型政策範例

AWS CodeStar 通知和 AWS CodeStar 連接資源為基礎的策略

AWS CodeStar 通知和 AWS CodeStar 連線不支援以資源為基礎的政策。

以標籤為基礎的授權

您可以將標籤附加至「 AWS CodeStar 通知」和「 AWS CodeStar 連線」資源，或在要求中傳遞標籤。若要根據標籤控制存取，請使用 codestar-notifications and codestar-connections:ResourceTag/*key-name*、aws:RequestTag/*key-name* 或 aws:TagKeys 條件金鑰，在政策的條件元素中，提供標籤資訊。有關標記策略的詳細資訊，請參閱標記 AWS 資源。如需有關標記 AWS CodeStar 通知和 AWS CodeStar 連線資源的詳細資訊，請參閱標記連線資源。

若要檢視身分型政策範例，用於根據該資源的標籤來限制資源的存取權，請參閱「使用標籤來控制對 AWS CodeStar 連線資源的存取」。

IAM 角色

IAM 角色是您 AWS 帳戶中具有特定許可的實體。

使用暫時登入資料

您可以搭配聯合功能使用暫時憑證來登入、擔任 IAM 角色或跨帳戶角色。您可以透過呼叫AssumeRole或等 AWS STS API 作業來取得臨時安全登入資料GetFederationToken。

AWS CodeStar 通知和 AWS CodeStar 連線支援使用臨時登入資料。

服務連結角色

服務連結角色可讓 AWS 服務存取其他服務中的資源，以代表您完成動作。服務連結角色會顯示在您的 IAM 帳戶中，並由該服務所擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

AWS CodeStar 通知支援服務連結角色。如需有關建立或管理 AWS CodeStar 通知和 AWS CodeStar 連線服務連結角色的詳細資訊，請參閱針對 AWS CodeStar Notifications 使用服務連結角色。

AWS CodeStar 連線不支援服務連結角色。

AWS CodeConnections 權限參考

下表列出每個 AWS CodeConnections API 作業、您可以授與權限的對應動作，以及用於授與權限的資源 ARN 格式。 AWS CodeConnections API 會根據該 API 允許的動作範圍分組到表格中。撰寫可連接至 IAM 身分的許可政策 (身分型政策) 時，請參考這些表格。

當您建立許可政策時，您需要在政策的 `Action` 欄位中指定動作。您需要在政策的 `Resource` 欄位中指定資源值做為 ARN，可包含或不含萬用字元 (*)。

若要在連線政策中表達條件，請使用此處所述和 條件索引鍵 中列出的條件索引鍵。您也可以使用 AWS寬度條件鍵。如需完整的 AWS全金鑰清單，請參閱 IAM 使用者指南中的可用金鑰。

若要指定動作，請使用 `codestar-connections:` 字首，後面接著 API 操作名稱 (例如 `codestar-connections:ListConnections` 或 `codestar-connections:CreateConnection`)。

使用萬用字元

若要指定多個動作或資源，請在 ARN 中使用萬用字元 (*)。例如，`codestar-connections:*`指定所有 AWS CodeConnections 動作，並`codestar-connections:Get*`指定以該字開頭的所有 AWS CodeConnections 動作`Get`。以下範例授予對所有以 `MyConnection` 為名稱開頭之資源的存取權。

```
arn:aws:codestar-connections:us-west-2:account-ID:connection/*
```

您只能針對下表列出的*##*資源使用萬用字元。您不能對 *region* 或 *account-id* 資源使用萬用字元。如需萬用字元的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 識別符。

主題

管理連線的許可

指定用於使用或 SDK 來檢視、建立 AWS CLI 或刪除連線的角色或使用者應具有下列權限的權限。

> **Note**
>
> 只具有下列許可，並無法在主控台中完成或使用連線。您需要在 完成連線的許可 中新增許可。

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

AWS CodeStar 通知和 AWS CodeStar 連線需要權限才能管理連線的動作

CreateConnection

動作：`codestar-connections:CreateConnection`

使用 CLI 或主控台來建立連線時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

DeleteConnection

動作：codestar-connections:DeleteConnection

使用 CLI 或主控台來刪除連線時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetConnection

動作：codestar-connections:GetConnection

需要使用 CLI 或主控台來檢視有關連線的詳細資訊。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListConnections

動作：codestar-connections:ListConnections

使用 CLI 或主控台列出帳戶中的所有連線時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

這些操作支援下列條件金鑰：

動作	條件索引鍵
codestar-connections:CreateConnection	codestar-connections:ProviderType
codestar-connections:DeleteConnection	N/A

動作	條件索引鍵
codestar-connections:GetConnection	N/A
codestar-connections:ListConnections	codestar-connections:ProviderTypeFilter

管理主機的許可

指定用於使用或 SDK 來檢視、建立 AWS CLI 或刪除主機的角色或使用者應具有下列權限的權限。

> **Note**
>
> 只具有下列許可，並無法在主機中完成或使用主機。您需要在 設定主機的許可 中新增許可。

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

AWS CodeStar 通知和 AWS CodeStar 連線管理主機所需動作的權限

CreateHost

動作：codestar-connections:CreateHost

使用 CLI 或主控台來建立主機時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

DeleteHost

動作：codestar-connections:DeleteHost

使用 CLI 或主控台來刪除主機時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

GetHost

動作：codestar-connections:GetHost

使用 CLI 或主控台來檢視有關主機的詳細資訊時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

ListHosts

動作：codestar-connections:ListHosts

使用 CLI 或主控台列出帳戶中的所有主機時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

這些操作支援下列條件金鑰：

動作	條件索引鍵
codestar-connections:CreateHost	codestar-connections:ProviderType
codestar-connections:DeleteHost	N/A
codestar-connections:GetHost	N/A
codestar-connections:ListHosts	codestar-connections:ProviderTypeFilter

完成連線的許可

指定要管理在主控台中連線的角色或使用者，應具備完成主控台中的連線及建立安裝所需的許可，包括授權供應商進行交握，以及建立供連線使用的安裝。除了上述許可之外，也請使用下列許可。

執行以瀏覽器為基礎的交握時，主控台會使用下列 IAM 操作。ListInstallationTargets、GetInstallationUrl、StartOAuthHandshake、UpdateConnec及 GetIndividualAccessToken 是 IAM 政策許可。不是 API 動作。

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

以此為基礎，需要下列許可才能在主控台中使用、建立、更新或刪除連線。

```
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 完成連線動作所需的權限

GetIndividualAccessToken

動作：codestar-connections:GetIndividualAccessToken

使用主控台完成連線時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetInstallationUrl

動作：codestar-connections:GetInstallationUrl

使用主控台完成連線時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListInstallationTargets

動作：codestar-connections:ListInstallationTargets

使用主控台完成連線時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

斯塔圖 AuthHandshake

動作：codestar-connections:Start0AuthHandshake

使用主控台完成連線時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

UpdateConnectionInstallation

動作：codestar-connections:UpdateConnectionInstallation

使用主控台完成連線時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

這些作業支援下列條件索引鍵。

動作	條件索引鍵
codestar-connections:GetIndividualAccessToken	codestar-connections:ProviderType
codestar-connections:GetInstallationUrl	codestar-connections:ProviderType
codestar-connections:ListInstallationTargets	N/A
codestar-connections:Start0AuthHandshake	codestar-connections:ProviderType
codestar-connections:UpdateConnectionInstallation	codestar-connections:InstallationId

設定主機的許可

指定管理在主控台中連線的角色或使用者，應具備在主控台中設定主機所需的許可，包括授權供應商進行交握，以及安裝主機應用程式。除了上述主機許可之外，也請使用下列許可。

執行瀏覽器式主機註冊時，主控台會使用下列 IAM 作業。RegisterAppCode 和 StartAppRegistrationHandshake 是 IAM 政策許可。不是 API 動作。

```
codestar-connections:RegisterAppCode
codestar-connections:StartAppRegistrationHandshake
```

以此為基礎，需要下列許可才能在主控台中使用、建立、更新或刪除需要主機的連線 (例如已安裝的供應商類型)。

```
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 完成主機設定所需動作的權限

RegisterAppCode

動作：codestar-connections:RegisterAppCode

使用主控台來完成主機設定時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

StartAppRegistrationHandshake

動作：codestar-connections:StartAppRegistrationHandshake

使用主控台來完成主機設定時需要。這只是 IAM 政策許可，不是 API 動作。

資源：arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

這些作業支援下列條件索引鍵。

將連線傳遞至服務

將連線傳遞至服務時 (例如，在管道定義中提供連線 ARN 以建立或更新管線)，使用者必須具有 codestar-connections:PassConnection 許可。

AWS CodeConnections 傳遞連接所需的權限

PassConnection

動作：codestar-connections:PassConnection

將連線傳遞到服務時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

此操作也支援下列條件金鑰：

- codestar-connections:PassedToService

支援的條件金鑰值

金鑰	有效動作供應商
codestar-connections:PassedToService	• codeguru-reviewer • codepipeline.amazonaws.com • proton.amazonaws.com

使用連線

當類似的服務 CodePipeline 使用連線時，服務角色必須具有指定連線的codestar-connections:UseConnection權限。

若要在主控台中管理連線，使用者政策必須具有 codestar-connections:UseConnection 許可。

AWS CodeConnections 使用連線所需的動作

UseConnection

動作：codestar-connections:UseConnection

使用連線時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

此操作也支援下列條件金鑰：

- codestar-connections:BranchName
- codestar-connections:FullRepositoryId
- codestar-connections:OwnerId
- codestar-connections:ProviderAction
- codestar-connections:ProviderPermissionsRequired
- codestar-connections:RepositoryName

支援的條件金鑰值

金鑰	有效動作供應商
codestar-connections:FullRepositoryId	儲存庫的使用者名稱和儲存庫名稱，例如 my-owner/my-repository 。僅當使用連線來存取特定儲存庫時才支援。
codestar-connections:ProviderPermissionsRequired	read_only 或 read_write
codestar-connections:ProviderAction	GetBranch ，ListRepositories ，ListOwners ，ListBranches ，StartUploadArchiveToS3 ，GitPush, GitPull, GetUploadArchiveToS3Status ，CreatePullRequestDiffComment ，GetPullRequest ，ListBranchCommits ，ListCommitFiles ，

金鑰	有效動作供應商
	ListPullRequestComments ，ListPullRequestCommits 。 如需相關資訊，請參閱下一節。

某些功能的必要條件金鑰可能隨著時間而變更。除非您的存取控制需求需要不同的許可，否則建議您使用 codestar-connections:UseConnection 來控制對連線的存取。

支援的 **ProviderAction** 存取權類型

當 AWS 服務使用連接時，會導致對源代碼提供者進行 API 調用。例如，服務可能呼叫 https://api.bitbucket.org/2.0/repositories/*username* API 來列出 Bitbucket 連線的儲存庫。

ProviderAction 條件金鑰可讓您限制供應商上可呼叫的 API。因為 API 路徑可能是動態產生，而且路徑隨不同供應商而不同，因此 ProviderAction 值對應至抽象動作名稱，而不是 API 的 URL。不論連線的供應商類型為何，這可讓您撰寫具有相同效果的政策。

以下是針對每個支援的 ProviderAction 值授予的存取類型。以下說明 IAM 政策許可。不是 API 動作。

AWS CodeConnections 支援的存取類型 **ProviderAction**

GetBranch

動作：codestar-connections:GetBranch

存取分支的相關資訊 (例如該分支的最新遞交) 時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListRepositories

動作：codestar-connections:ListRepositories

存取屬於擁有者的公有和私有儲存庫清單 (包括這些儲存庫的詳細資訊) 時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListOwners

動作：codestar-connections:ListOwners

存取連線可存取的擁有者清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListBranches

動作：codestar-connections:ListBranches

存取特定儲存庫上存在的分支清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

StartUploadArchiveToS3

動作：codestar-connections:StartUploadArchiveToS3

讀取原始碼並上傳至 Amazon S3 時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GitPush

動作：codestar-connections:GitPush

使用 Git 寫入儲存庫時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GitPull

動作：codestar-connections:GitPull

使用 Git 讀取儲存庫時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetUploadArchiveTo中三身份

動作：codestar-connections:GetUploadArchiveToS3Status

存取由 StartUploadArchiveToS3 起始的上傳狀態 (包括任何錯誤訊息) 時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

CreatePullRequestDiffComment

動作：codestar-connections:CreatePullRequestDiffComment

存取提取請求上的註解時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

GetPullRequest

動作：codestar-connections:GetPullRequest

檢視儲存庫的提取請求時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListBranchCommits

動作：codestar-connections:ListBranchCommits

儲存庫分支的遞交清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListCommitFiles

動作：codestar-connections:ListCommitFiles

檢視遞交的檔案清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListPullRequestComments

動作：codestar-connections:ListPullRequestComments

檢視提取請求的註解清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

ListPullRequestCommits

動作：codestar-connections:ListPullRequestCommits

檢視提取請求的遞交清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*

標記連線資源的支援許可

標記連線資源時，會使用下列 IAM 操作。

```
codestar-connections:ListTagsForResource
codestar-connections:TagResource
codestar-connections:UntagResource
```

AWS CodeConnections 標記連線資源的必要動作

ListTagsForResource

動作：codestar-connections:ListTagsForResource

檢視與連線資源關聯之標籤清單時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

TagResource

動作：codestar-connections:TagResource

標記連線資源時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

UntagResource

動作：codestar-connections:UntagResource

從連線資源中移除標籤時需要。

資源：arn:aws:codestar-connections:*region*:*account-id*:connection/*connection-id*、 arn:aws:codestar-connections:*region*:*account-id*:host/*host-id*

將連線傳遞到儲存庫連結

在同步組態中提供儲存庫連結時，使用者必須擁有存放庫連結 ARN/資源的 codestar-connections:PassRepository 許可。

AWS CodeConnections 傳遞連接所需的權限

PassRepository

動作：codestar-connections:PassRepository

需要將儲存庫連結傳遞至同步組態。

資源：arn:aws:codestar-connections:*region*:*account-id*:repository-link/*repository-link-id*

此操作也支援下列條件金鑰：

- codestar-connections:PassedToService

支援的條件金鑰值

金鑰	有效動作供應商
codestar-connections:PassedToService	• cloudformation.sync.codeconnections.amazonaws.com

儲存庫連結的可支援條件金鑰

下列條件金鑰支援儲存庫連結和同步組態資源的操作：

- `codestar-connections:Branch`

  透過請求中傳遞的分支名稱來篩選存取。

條件金鑰的可支援動作

金鑰	有效值
`codestar-connections:Branch`	此條件金鑰支援下列動作： • CreateSyncConfiguration • UpdateSyncConfiguration • GetRepositorySyncStatus

身分型政策範例

根據預設，具有、或 AWS CodePipeline 套用其中一個受管政策的 IAM 使用者和角色 AWS CodeCommit AWS CodeBuild AWS CodeDeploy，都具有與這些政策意圖一致的連線、通知和通知規則的許可。例如，已套用其中一個完整存取政策 (AWSCodeCommitFullAccess、、或 AWSCodePipeline_FullAccess) 的 IAM 使用者或角色 AWSCodeBuildAdminAccessAWSCodeDeployFullAccess，也可以完整存取針對這些服務的資源建立的通知和通知規則。

其他 IAM 使用者和角色沒有建立或修改 AWS CodeStar 通知和 AWS CodeStar 連線資源的權限。他們也無法使用 AWS Management Console AWS CLI、或 AWS API 執行工作。IAM 管理員必須建立 IAM 政策，授予使用者和角色在指定資源上執行 API 作業的所需許可。管理員接著必須將這些政策連接至需要這些許可的 IAM 使用者或群組。

AWS CodeStar 通知的權限和範例

下列政策陳述式和範例可協助您管理「 AWS CodeStar 通知」。

完整存取受管政策中的通知相關許可

AWSCodeCommitFullAccess、AWSCodeBuildAdminAccessAWSCodeDeployFullAccess、和AWSCodePipeline_FullAccess受管理的政策包括下列陳述式，可讓您完整存取開發人員工具主控台中的通知。已套用上述其中一個受管政策的使用者，也可以建立和管理通知的 Amazon SNS 主題、讓使用者訂閱和取消訂閱主題，以及列出可選擇作為通知規則目標的主題。

> **Note**
>
> 在受管理政策中，條件金鑰 `codestar-notifications:NotificationsForResource` 具有服務的資源類型所特有的值。例如，在的完整存取原則中 CodeCommit，值為`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": "*"
    }
```

唯讀受管政策中的通知相關許可

AWSCodeCommitReadOnlyAccess、AWSCodeBuildReadOnlyAccessAWSCodeDeployReadOnlyAccess、和AWSCodePipeline_ReadOnlyAccess受管理的策略包括下列陳述式，以允許唯讀存取通知。例如，他們可以在開發人員主控台中檢視資源的通知，但無法建立、管理或訂閱通知。

> **Note**
>
> 在受管理政策中，條件金鑰 `codestar-notifications:NotificationsForResource` 具有服務的資源類型所特有的值。例如，在的完整存取原則中 CodeCommit，值為`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": "*"
    }
```

其他受管政策中的通知相關許可

AWSCodeCommitPowerUserAWSCodeBuildDeveloperAccess、
和AWSCodeBuildDeveloperAccess受管理的政策包括下列陳述式，可讓開發人員套用這些受管理原則之一來建立、編輯和訂閱通知。他們無法刪除通知規則或管理資源的標籤。

> **Note**
>
> 在受管理政策中，條件金鑰 `codestar-notifications:NotificationsForResource` 具有服務的資源類型所特有的值。例如，在的完整存取原則中 CodeCommit，值為`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": "*"
    }
```

範例：管理通知的系統管理員層級原則 AWS CodeStar

在此範例中，您想要授與 AWS 帳戶中的 IAM 使用者完整存取「 AWS CodeStar 通知」權限，以便使用者可以檢閱通知規則的詳細資料，並列出通知規則、目標和事件類型。您也希望允許使用者新增、更新和刪除通知規則。這是完整存取原則，相當於包含在AWSCodeBuildAdminAccess、AWSCodeCommitFullAccess、AWSCodeDeployFullAccess和AWSCodePi管理策略中的通知權限。就像這些受管政策一樣，您只應將這種政策聲明附加到需要對整個 AWS 帳戶通知和通知規則進行完整管理存取權的 IAM 使用者、群組或角色。

> **Note**
>
> 此政策包含 `CreateNotificationRule`。任何將此政策套用至其 IAM 使用者或角色的使用者都可以針對 AWS 帳戶中 Notification 支援的任何和所有資源類型建立通 AWS CodeStar 知規則，即使該使用者本身無法存取這些資源也是如此。例如，具有此原則的使用者可以在沒有存取 CodeCommit自身權限的情況下為 CodeCommit 存放庫建立通知規則。

```
{
    "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": "*"
      }
   ]
}
```

範例：使用通知的參與者層級原則 AWS CodeStar

在此範例中，您想要授與「 AWS CodeStar 通知」 day-to-day 使用方式的存取權，例如建立和訂閱通知，但不要授與更具破壞性的動作，例如刪除通知規則或目標。這等同於AWSCodeBuildDeveloperAccess、AWSCodeDeployDeveloperAccess和AWSCodeCommitPowerUser受管理策略中提供的存取權。

> **Note**
>
> 此政策包含 `CreateNotificationRule`。任何將此政策套用至其 IAM 使用者或角色的使用者都可以針對 AWS 帳戶中 Notification 支援的任何和所有資源類型建立通 AWS CodeStar 知規則，即使該使用者本身無法存取這些資源也是如此。例如，具有此原則的使用者可以在沒有存取 CodeCommit自身權限的情況下為 CodeCommit 存放庫建立通知規則。

```
{
    "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": "*"
        }
    ]
}
```

範例：使用 AWS CodeStar 通知的 read-only-level 政策

在此範例中，您希望授予您帳戶中 IAM 使用者對 AWS 帳戶中的通知規則、目標和事件類型的唯讀存取權。此範例會示範如何建立允許檢視這些項目的政策。這等同於包含在AWSCodeBuildReadOnlyAccess、AWSCodeCommitReadOnly和AWSCodePipeline_ReadOnlyAccess受管理策略中的權限。

```
{
    "Version": "2012-10-17",
    "Id": "CodeNotification__ReadOnly",
    "Statement": [
        {
            "Sid": "Reads_API_Access",
            "Effect": "Allow",
```

```
            "Action": [
                "CodeNotification:DescribeNotificationRule",
                "CodeNotification:ListNotificationRules",
                "CodeNotification:ListTargets",
                "CodeNotification:ListEventTypes"
            ],
            "Resource": "*"
        }
    ]
}
```

權限和範例 AWS CodeConnections

下列政策陳述式和範例可協助您管理 AWS CodeConnections。

若要了解如何使用這些範例 JSON 政策文件來建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的在 JSON 標籤上建立政策。

範例：使用 CLI 建立和 AWS CodeConnections 使用主控台檢視的原則

指定用於使用或 SDK 來檢視、建立、標記 AWS CLI 或刪除連線的角色或使用者應具有下列權限的權限。

> **Note**
>
> 只具有下列許可，並無法在主控台中完成連線。您需要新增下一節中的許可。

若要使用主控台來檢視可用連線的清單、檢視標籤和使用連線，請使用下列政策。

```
{
    "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": "*"
     }
   ]
}
```

範例：使用主控台建立 AWS CodeConnections 的政策

指定要管理在主控台中連線的角色或使用者，應具備完成主控台中的連線及建立安裝所需的許可，包括授權供應商進行交握，以及建立供連線使用的安裝。也應該新增 UseConnection 以使用主控台中的連線。請使用下列政策來在主控台中檢視、使用、建立、標記或刪除連線。

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

範例：用於管理的系統管理員層級原則 AWS CodeConnections

在此範例中，您想要授與 AWS 帳戶中的 IAM 使用者完整存取權限，以 CodeConnections 便使用者可以新增、更新和刪除連線。這是完整存取原則，相當於AWSCodePipeline_FullAccess受管理的原則。就像該受管政策一樣，您應該只將這種政策聲明附加到需要完全管理存取跨 AWS 帳戶連線的 IAM 使用者、群組或角色。

```
{
    "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": "*"
     }
   ]
}
```

範例：使用的參與者層級原則 AWS CodeConnections

在此範例中，您想要授與存取使用情 day-to-day 況 (例如建立和檢視連線的詳細資料)，但不要授與更具破壞性的動作 (例如刪除連線)。 CodeConnections

```
{
    "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": "*"
        }
    ]
}
```

範例：使用的 read-only-level 政策 AWS CodeConnections

在此範例中，您想要授與帳戶中的 IAM 使用者對帳戶中連線的唯讀存取權。 AWS 此範例會示範如何建立允許檢視這些項目的政策。

```
{
    "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": "*"
        }
    ]
}
```

範例：與指定存放庫 AWS CodeConnections 搭配使用的範圍縮短原則

在下列範例中，客戶希望 CodeBuild 服務角色存取指定的 Bitbucket 存放庫。 CodeBuild 服務角色的原則：

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

範例：使用連線的原則 CodePipeline

在下列範例中，系統管理員想要使用者使用與的連線 CodePipeline。連接至使用者的政策：

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

範例：使用 CodeBuild 服務角色進行 Bitbucket 讀取作業 AWS CodeConnections

在下列範例中，無論存放庫為何，客戶都希望 CodeBuild 服務角色在 Bitbucket 上執行讀取作業。 CodeBuild 服務角色的原則：

```
{
```

```
    "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"}}
    }
}
```

範例：限制 CodeBuild 服務角色執行作業 AWS CodeConnections

在下列範例中，客戶想要防止 CodeBuild 服務角色執行類似的作業`CreateRepository`。 CodeBuild 服務角色的原則：

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

使用標籤來控制對 AWS CodeStar 連線資源的存取

可以將標記連接到資源或在請求中將標記傳遞至支援標記的服務。在中 CodeConnections，資源可以有標籤，而某些動作可以包含標籤。建立 IAM 政策時，可使用標籤條件索引鍵來控制以下項目：

- 可在管道資源上執行動作的使用者 (根據資源已具有的標籤)。
- 可在動作請求中傳遞的標籤。
- 請求中是否可使用特定的標籤鍵。

以下範例顯示了如何指定 CodeConnections 使用者政策中的標記條件。

Example 1：根據請求中的標籤允許動作

下列政策會授予使用者在 CodeConnections 中建立連線的許可。

若要這樣做，它會在請求指定名為 Project 且值為 ProjectA 的標籤時允許 CreateConnection 和 TagResource 動作。(aws:RequestTag 條件索引鍵用來控制哪些標籤可在 IAM 請求中傳遞。) aws:TagKeys 條件可確保標籤索引鍵區分大小寫。

```
{
  "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：根據資源標籤允許動作

下列政策授予使用者對 CodeConnections 中的資源執行動作和取得相關資訊的許可。

若要這樣做，它會在管道有名為 Project 且值為 ProjectA 的標籤時允許特定動作。(aws:RequestTag 條件索引鍵用來控制哪些標籤可在 IAM 請求中傳遞。) aws:TagKeys 條件可確保標籤索引鍵區分大小寫。

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

在主控台中使用通知和連線

通知體驗內建於 CodeBuild、、和主控 CodePipeline台 CodeCommit CodeDeploy，以及 「設定」 導覽列本身的「開發人員工具」主控台中。若要存取主控台中的通知，您必須套用這些服務相關的其中一個受管政策，或者您必須擁有至少一組許可。這些權限必須允許您列出並檢視您 AWS 帳戶中「 AWS CodeStar 通知」和「 AWS CodeStar 連線」資源的詳細資料。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (IAM 使用者或角色) 而言，主控台就無法如預期運作。如需有關授與 AWS CodeBuild、、和存取權的詳細資訊 AWS CodeCommit AWS CodeDeploy AWS CodePipeline，包括這些主控台的存取權，請參閱下列主題：

- CodeBuild：使用以身分為基礎的原則 CodeBuild
- CodeCommit：使用以身分為基礎的原則 CodeCommit
- AWS CodeDeploy: 身分識別與存取管理 AWS CodeDeploy
- CodePipeline: 使用 IAM 政策進行存取控制

AWS CodeStar 通知沒有任何 AWS 受管理的策略。若要提供通知功能的存取權，您必須針對上述其中一項服務套用其中一項受管政策，或者，您必須建立具有您要授予使用者或實體之許可層級的政策，然後將這些政策連接到需要這些許可的使用者、群組或角色。如需更多資訊和範例，請參閱下列內容：

- 範例：管理通知的系統管理員層級原則 AWS CodeStar
- 範例：使用通知的參與者層級原則 AWS CodeStar
- 範例：使用 AWS CodeStar 通知的 read-only-level 政策.

AWS CodeStar 連線沒有任何 AWS 受管理的策略。您可以針對存取權使用許可和許可組合，如 完成連線的許可 中詳述的許可權限。

如需詳細資訊，請參閱下列內容：

- 範例：用於管理的系統管理員層級原則 AWS CodeConnections
- 範例：使用的參與者層級原則 AWS CodeConnections
- 範例：使用的 read-only-level 政策 AWS CodeConnections

您不需要為僅對 AWS CLI 或 AWS API 進行呼叫的使用者允許主控台權限。反之，只需允許存取符合您嘗試執行之 API 操作的動作就可以了。

允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此原則包含在主控台上或以程式設計方式使用 AWS CLI 或 AWS API 完成此動作的權限。

```
{
    "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": "*"
        }
    ]
}
```

疑難排解 AWS CodeStar 通知與 AWS CodeStar 連線身分與存取

使用下列資訊，協助您診斷和修復您使用通知和 IAM 時可能遇到的常見問題。

主題

- 我是管理員，想要允許其他人存取通知
- 我已建立 Amazon SNS 主題，並將其新增為通知規則目標，但我未收到有關事件的電子郵件
- 我想允許 AWS 帳戶以外的人員存取我的 AWS CodeStar 通知和 AWS CodeStar 連線資源

我是管理員，想要允許其他人存取通知

若要允許其他人存取 AWS CodeStar 通知和 AWS CodeStar 連線，您必須為需要存取的人員或應用程式建立 IAM 實體 (使用者或角色)。他們將使用該實體的憑證來存取 AWS。然後，您必須將原則附加至實體，以便在 AWS CodeStar 通知和 AWS CodeStar 連線中授與他們正確的權限。

若要立即開始使用，請參閱《IAM 使用者指南》中的建立您的第一個 IAM 委派使用者及群組。

如需 AWS CodeStar 通知的特定資訊，請參閱AWS CodeStar 通知的權限和範例。

我已建立 Amazon SNS 主題，並將其新增為通知規則目標，但我未收到有關事件的電子郵件

為了接收有關事件的通知，您必須將有效的 Amazon SNS 主題訂閱為通知規則的目標，而且您的電子郵件地址必須訂閱該 Amazon SNS 主題。若要疑難排解 Amazon SNS 主題的問題，請檢查下列各項：

- 確定 Amazon SNS 主題與通知規則位於相同的 AWS 區域。
- 檢查並確定您的電子郵件別名已訂閱正確的主題，而且您已確認訂閱。如需詳細資訊，請參閱讓端點訂閱 Amazon SNS 主題。
- 確認主題原則已修改，以允許「 AWS CodeStar 通知」將通知推送至該主題。主題政策應該包含類似以下的陳述式：

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

如需詳細資訊，請參閱 設定。

我想允許 AWS 帳戶以外的人員存取我的 AWS CodeStar 通知和 AWS CodeStar 連線資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要瞭解 AWS CodeStar 「通知與 AWS CodeStar 連線」是否支援這些功能，請參閱開發人員工具主控台中的功能如何與 IAM 搭配使用。
- 若要了解如何提供對您所擁有資源 AWS 帳戶 的存取權，請參閱《IAM 使用者指南》中您擁有的另一 AWS 帳戶 個 IAM 使用者提供存取權限。

- 若要了解如何將資源存取權提供給第三方 AWS 帳戶，請參閱 IAM 使用者指南中的提供第三方 AWS 帳戶 擁有的存取權。
- 若要了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的將存取權提供給在外部進行身分驗證的使用者 (聯合身分)。
- 若要了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 角色與資源型政策的差異。

針對 AWS CodeStar Notifications 使用服務連結角色

AWS CodeStar Notifications 可使用 AWS Identity and Access Management (IAM) 服務連結角色。服務連結角色是直接連結至 AWS CodeStar Notifications 的一種特殊 IAM 角色類型。服務連結角色由 AWS CodeStar Notifications 預先定義，且內含該服務代您呼叫其他 AWS 服務所需的所有許可。此角色是您第一次建立通知規則時為您建立的。您不必建立角色。

服務連結角色可讓設定 AWS CodeStar Notifications 更為簡單，因為您不必手動新增必要的許可。AWSCodeStar Notifications 會定義其服務連結角色的許可，除非另有定義，否則僅有 AWS CodeStar Notifications 可以擔任其角色。定義的許可包括信任政策和許可政策，並且該許可政策不能連接到任何其他 IAM 實體。

若要刪除服務連結角色，您必須先刪除其相關資源。如此可保護您的 AWS CodeStar Notifications 資源，避免您不小心移除資源的存取許可。

關於支援服務連結角色的其他服務，如需相關資訊，請參閱與 IAM 搭配運作的 AWS 服務。

AWS CodeStar Notifications 的服務連結角色許可

AWS CodeStar Notifications 使用 AWSServiceRoleForCodeStarNotifications 服務連結角色，以擷取工具鏈中發生的事件相關資訊，以及將通知傳送至您指定的目標。

AWSServiceRoleForCodeStarNotifications 服務連結角色信任下列服務可擔任該角色：

- `codestar-notifications.amazonaws.com`

此角色許可政策允許 AWS CodeStar Notifications 對指定資源完成下列動作：

- 動作：`CloudWatch Event rules that are named awscodestar-notifications-*` 上的 `PutRule`
- 動作：`CloudWatch Event rules that are named awscodestar-notifications-*` 上的 `DescribeRule`

- 動作：CloudWatch Event rules that are named awscodestar-notifications-* 上的 PutTargets
- 動作： CreateTopic 至 create Amazon SNS topics for use with AWS CodeStar Notifications with the prefix CodeStarNotifications-
- 動作：all comments on all pull requests in all CodeCommit repositories in the AWS account 上的 GetCommentsForPullRequests
- 動作：all comments on all commits in all CodeCommit repositories in the AWS account 上的 GetCommentsForComparedCommit
- 動作：all commits in all CodeCommit repositories in the AWS account 上的 GetDifferences
- 動作：all comments on all commits in all CodeCommit repositories in the AWS account 上的 GetCommentsForComparedCommit
- 動作：all commits in all CodeCommit repositories in the AWS account 上的 GetDifferences
- 動作：all AWS Chatbot clients in the AWS account 上的 DescribeSlackChannelConfigurations
- 動作：all AWS Chatbot clients in the AWS account 上的 UpdateSlackChannelConfiguration
- 動作：all actions in all pipelines in the AWS account 上的 ListActionExecutions
- 動作：all files in all CodeCommit repositories in the AWS account unless otherwise tagged 上的 GetFile

您可以在 AWSServiceRoleForCodeStarNotifications 服務連結角色的政策陳述式中看到這些動作。

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

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的服務連結角色許可。

為 AWS CodeStar Notifications 建立服務連結角色

您不需要手動建立一個服務連結角色。您可以使用開發人員工具主控台或 AWS CLI 或軟體開發套件中的 CreateNotificationRule API 來建立通知規則。您也可以直接呼叫 API。無論您使用哪一種方法，系統都會為您建立服務連結角色。

若您刪除此服務連結角色，之後需要再次建立，您可以在帳戶中使用相同程序重新建立角色。您可以使用開發人員工具主控台或 AWS CLI 或軟體開發套件中的 CreateNotificationRule API 來建立通知規則。您也可以直接呼叫 API。無論您使用哪一種方法，系統都會為您建立服務連結角色。

為 AWS CodeStar Notifications 編輯服務連結角色

因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。不過，您可以使用 IAM 來編輯角色描述。如需更多資訊，請參閱 IAM 使用者指南中的編輯服務連結角色。

為 AWS CodeStar Notifications 刪除服務連結角色

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。刪除服務連結角色之前，您必須先清理資源。對於 AWS CodeStar Notifications，這表示刪除您的 AWS 帳戶中使用此服務角色的所有通知規則。

> **Note**
>
> 若 AWS CodeStar Notifications 服務在您試圖刪除資源時正在使用該角色，刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

刪除 AWSServiceRoleForCodeStarNotifications 使用的 AWS CodeStar Notifications 資源

1. 前往 https://console.aws.amazon.com/codesuite/settings/notifications 開啟 AWS 開發人員工具主控台。

   > **Note**
   >
   > 通知規則套用至其建立所在的 AWS 區域。如果您在多個 AWS 區域中有通知規則，請使用區域選擇器來變更 AWS 區域。

2. 選擇清單中出現的所有通知規則，然後選擇 Delete (刪除)。
3. 在您建立通知規則的所有 AWS 區域中重複這些步驟。

使用 IAM 來刪除服務連結角色

使用 IAM 主控台、AWS CLI 或 AWS Identity and Access Management API 來刪除 AWSServiceRoleForCodeStarNotifications 服務連結角色。如需詳細資訊，請參閱 IAM 使用者指南中的刪除服務連結角色。

AWS CodeStar Notifications 服務連結角色的支援區域

AWS CodeStar Notifications 支援在所有提供服務的 AWS 區域中使用服務連結角色。如需詳細資訊，請參閱 AWS 區域與端點和 AWS CodeStar Notifications。

使用 AWS CodeConnections 的服務連結角色

AWS CodeConnections 使用 AWS Identity and Access Management (IAM) 服務連結的角色。服務連結角色是直接連結至 AWS CodeConnections 的一種特殊 IAM 角色類型。服務連結角色由 AWS CodeConnections 預先定義，且內含該服務代您呼叫其他 AWS 服務所需的所有許可。此角色是您第一次建立連線時為您建立的。您不必建立角色。

服務連結的角色可讓設定 AWS CodeConnections 更為簡單，因為您不必手動新增許可。AWS CodeConnections 會定義其服務連結角色的許可，且除非另有定義，否則僅有 AWS CodeConnections 能取得其角色。定義的許可包括信任政策和許可政策，並且該許可政策不能連接到任何其他 IAM 實體。

若要刪除服務連結角色，您必須先刪除其相關資源。如此可保護您 AWS CodeConnections 的資源，避免您不小心移除資源的存取許可。

關於支援服務連結角色的其他服務，如需相關資訊，請參閱與 IAM 搭配運作的 AWS 服務。

AWS CodeConnections 的服務連結角色許可

AWS CodeConnections 會使用 AWSServiceRoleForGitSync 服務連結角色來搭配已連線的 Git 型儲存庫使用 Git 同步。

AWSServiceRoleForGitSync 服務連結角色信任下列服務擔任該角色：

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

名為 AWSGitSyncServiceRolePolicy 的角色許可政策允許 AWS CodeConnections 在指定的資源上完成下列動作：

- 動作：授與許可，以允許使用者與外部 Git 型儲存庫建立連線，以及搭配這些儲存庫使用 Git 同步。

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的服務連結角色許可。

為 AWS CodeConnections 建立服務連結角色

您不需要手動建立一個服務連結角色。當您使用 CreateRepositoryLink API 為 Git 同步的專案建立資源時，您就會建立該角色。

若您刪除此服務連結角色，之後需要再次建立，您可以在帳戶中使用相同程序重新建立角色。

為 AWS CodeConnections 編輯服務連結角色

因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。不過，您可以使用 IAM 來編輯角色描述。如需詳細資訊，請參閱《IAM 使用者指南》中的編輯服務連結角色。

為 AWS CodeConnections 刪除服務連結角色

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。刪除服務連結角色之前，您必須先清理資源。這表示刪除您的 AWS 帳戶中使用此服務角色的所有連線。

> **Note**
>
> 若 AWS CodeConnections 服務在您試圖刪除資源時正在使用該角色，刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

刪除 AWSServiceRoleForGitSync 所使用的 AWS CodeConnections 資源

1. 開啟 [開發人員工具] 主控台，然後選擇設定。
2. 選擇清單中出現的所有連線，然後選擇刪除。
3. 在您建立連線的所有 AWS 區域中重複這些步驟。

使用 IAM 來刪除服務連結角色

使用 IAM 主控台、AWS CLI 或 AWS Identity and Access Management API 來刪除 AWSServiceRoleForGitSync 服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的刪除服務連結角色。

AWS CodeConnections 服務連結角色的支援區域

AWS CodeConnections 支援在所有提供服務的 AWS 區域中使用服務連結角色。如需詳細資訊，請參閱 AWS 區域與端點。

AWS CodeConnections 的 AWS 受管政策

AWS 管理的政策是由 AWS 建立和管理的獨立政策。AWS 管理的政策的設計在於為許多常見使用案例提供許可，如此您就可以開始將許可指派給使用者、群組和角色。

請謹記，AWS 管理的政策可能不會授予您特定使用案例的最低權限許可，因為它們可供所有 AWS 客戶使用。我們建議您定義使用案例專屬的客戶管理政策，以便進一步減少許可。

您無法更改 AWS 管理的政策中定義的許可。如果 AWS 更新 AWS 管理的政策中定義的許可，更新會影響政策連接的所有主體身分 (使用者、群組和角色)。在推出新的 AWS 服務 或有新的 API 操作可供現有服務使用時，AWS 很可能會更新 AWS 管理的政策。

如需詳細資訊，請參閱《IAM 使用者指南》中的 AWS 受管政策。

AWS 管理的政策：AWSGitSyncServiceRolePolicy

您無法將 AWSGitSyncServiceRolePolicy 附加至您的 IAM 實體。此政策會連接到服務連結角色，而此角色可讓 AWS CodeConnections 代表您執行動作。如需更多詳細資訊，請參閱 使用 AWS CodeConnections 的服務連結角色。

此政策可讓客戶存取 Git 型儲存庫，以便與連線搭配使用。客戶將在使用 CreateRepositoryLink API 後存取這些資源。

許可詳細資訊

此政策包含以下許可。

- `codestar-connections` – 授與許可以允許使用者與外部 Git 型儲存庫建立連線。

```
{
 "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 管理的政策的 AWS CodeConnections 更新項目

檢視自 AWS CodeConnections 開始追蹤 AWS 管理的政策變更以來的更新詳細資訊。如需有關此頁面變更的自動提醒，請訂閱 AWS CodeConnections 文件歷史記錄頁面上的 RSS 摘要。

變更	描述	日期
AWSGitSyncServiceRolePolicy – 新政策	AWS CodeConnections 已新增此政策。 授與許可，以允許 AWS CodeConnections 使用者搭配已連線的 Git 型儲存庫使用 Git 同步。	2023 年 11 月 26 日
AWS CodeConnections 已開始追蹤變更	AWS CodeConnections 已開始追蹤其 AWS 管理的政策的變更。	2023 年 11 月 26 日

AWS CodeStar 通知與 AWS CodeStar 連線的合規性驗證

AWS CodeStar 通知和 AWS CodeStar 連線不在任何 AWS 合規性方案的範圍內。

如需特定法規遵循計劃範圍內的 AWS 服AWS 務清單，請參閱合規計劃範圍內的服務。如需一般資訊，請參閱 AWS 合規計劃。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱在 AWS Artifact 中下載報表。

使用「 AWS CodeStar 通知與 AWS CodeStar 連線」時，您的合規責任取決於資料的敏感度、公司的合規目標，以及適用的法律與法規。 AWS 提供下列資源以協助遵循法規：

- 安全性與合規性快速入門指南 — 這些部署指南討論架構考量，並提供在上部署以安全性和法規遵循為重點的基準環境的步驟。 AWS
- AWS 合規性資源 — 此工作簿和指南集合可能適用於您的產業和所在地。
- AWS Config— 此 AWS 服務評估您的資源配置是否符合內部實踐，行業準則和法規。
- AWS Security Hub— 此 AWS 服務提供安全狀態的全面檢視，協助您檢查您 AWS 是否符合安全性產業標準和最佳做法。

AWS CodeStar Notifications 和 AWS CodeStar Connections 中的復原功能

AWS 全球基礎設施是以 AWS 區域與可用區域為中心建置的。AWS區域提供多個分開且隔離的實際可用區域，它們以低延遲、高輸送量和高度備援聯網功能相互連結。透過可用區域，您所設計與操作的應用程式和資料庫，就能夠在可用區域之間自動容錯移轉，而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域與可用區域的詳細資訊，請參閱 AWS 全球基礎設施。

- 通知規則專屬於其建立所在的 AWS 區域。如果您在多個 AWS 區域 中有通知規則，請使用區域選擇器來檢閱每個 AWS 區域 中的通知規則。
- AWS CodeStar Notifications 仰賴 Amazon Simple Notification Service (Amazon SNS) 主題做為通知規則目標。Amazon SNS 主題和通知規則目標的相關資訊，可以存放在與您設定通知規則所在 AWS 區域不同的區域。

AWS CodeStar Notifications 和 AWS CodeStar Connections 中的基礎設施安全

作為受管服務中的功能，AWS CodeStar Notifications 和 AWS CodeStar Connections 受到 AWS 全球網路安全程序所保護，如 Amazon Web Services：安全程序概觀白皮書中所述。

您使用 AWS 發佈的 API 呼叫來 透過網路存取 AWS CodeStar Notifications 和 AWS CodeStar Connections。用戶端必須支援 Transport Layer Security (TLS) 1.0 或更新版本。用戶端也必須支援具備完美轉送私密 (PFS) 的密碼套件，例如臨時 Diffie-Hellman (DHE) 或橢圓曲線臨時 Diffie-Hellman (ECDHE)。大多數現代系統都支援這些模式。

請求必須使用存取金鑰 ID 和與 IAM 委託人相關聯的私密存取金鑰來簽署。或者，您可以使用 AWS Security Token Service (AWS STS) 產生臨時安全憑證來簽署請求。

跨區域 AWS CodeConnections 資源之間的流量

如果您使用連線功能來啟用資源的連線，表示您同意並指示我們在您使用基礎服務的 AWS 區域 區域以外的 AWS 區域 區域，存放和處理與此類連線資源的相關資訊，僅用於與僅唯一目的是在建立資源所在地區以外的地區提供連至這些資源的連線。

如需更多詳細資訊，請參閱 AWS CodeStar 連線中的全球資源。

> **Note**
>
> 如果您使用連線功能為不需要先啟用的區域中的資源啟用連線，我們將會儲存並處理上述主題所述的資訊。
> 對於在必須先啟用的區域中建立的連線，例如歐洲 (米蘭) 區域，我們將會僅儲存和處理該地區中的連線資訊。

文件歷史記錄

下表說明此版本開發人員工具主控台指南的文件。

- AWS CodeStar Notifications API 版本：2019-10-15
- AWS CodeStar Connections API 版本：2019-12-01

變更	描述	日期
GitLab 自我管理的支援	新增用於設定 AWS 資源的連線，以與 GitLab 自我管理互動的支援。更多資訊，請參閱建立或更新主機的工作流程和建立與 GitLab 自我管理的連線。	2023 年 12 月 28 日
用於連線的新儲存庫連結和同步組態	新增有關設定儲存庫連結和同步組態的資訊。使用同步組態來同步 Git 儲存庫中的內容，以更新您的 AWS CloudFormation 堆疊資源。如需詳細資訊，請參閱使用儲存庫連結和使用同步組態。	2023 年 11 月 27 日
支援連線的服務連結角色	新增設定連線以搭配 Git 儲存庫使用 Git 同步的支援。如需詳細資訊，請參閱使用 AWS CodeStar Connections 的服務連結角色和受管政策。	2023 年 11 月 26 日
GitLab 群組的支援	新增用於設定 AWS 資源的連線以與 GitLab 群組互動的支援。如需詳細資訊，請參閱建立連線和建立連至 GitLab 的連線。	2023 年 9 月 15 日
新的 GitLab 供應商類型	現在您可以建立與 GitLab 的連線。如需詳細資訊，請參閱建	2023 年 8 月 10 日

	立連線和建立連至 GitLab 的連線。	
通知規則的新目標類型	您現在可以選擇為 Microsoft Teams 頻道設定的 AWS Chatbot 用戶端，做為通知規則的目標。如需詳細資訊，請參閱建立通知規則和使用通知規則目標。	2023 年 5 月 17 日
歐洲 (米蘭) 區域中提供連線	已新增在歐洲區域 (米蘭) 區域中的連線。如需詳細資訊，請參閱跨區域的 AWS CodeStar Connections 之間的流量。	2023 年 5 月 17 日
新增了儲存庫許可連線錯誤的疑難排解	於 GitHub 組織中建立儲存庫的連線時，您必須是 GitHub 組織擁有者。如需詳細資訊，請參閱連線至 GitHub 時出現連線錯誤。	2022 年 8 月 29 日
新增標記主機資源的資訊	您現在可以使用主控台和 CLI 來標記主機。如需詳細資訊，請參閱在 AWS CodeStar Connections 中標記資源。	2021 年 4 月 19 日
連線的 VPC 端點支援	您可以搭配連線使用 VPC 端點。如需詳細資訊，請參閱 AWS CodeStar Connections 和界面 VPC 端點 (AWS PrivateLink)。	2020 年 11 月 24 日
新的 GitHub 和 GitHub Enterprise Cloud 供應商類型	您現在可以建立連至 GitHub 和 GitHub Enterprise Cloud 的連線。如需詳細資訊，請參閱建立連線和建立連至 GitHub 的連線。	2020 年 9 月 30 日

新增 GitHub Enterprise Server 供應商類型和主機資源	本指南新增有關連線之主機資源的資訊。您現在可以建立連至 GitHub Enterprise Server 的連線。如需詳細資訊，請參閱建立連線和使用主機。這是開發人員工具主控台使用者指南中連線功能的一般可用性版本。	2020 年 6 月 29 日
新增使用和標記連線的資訊	本指南新增主控台中連線功能的相關資訊。您可以檢視概念、入門步驟、包含範例政策的許可參考，以及建立、檢視和標記連線的步驟。如需詳細資訊，請參閱什麼是連線、連線概念、連線入門、建立連線、在 AWS CodeStar Connections 中標記資源、安全性、連線的配額、疑難排解，以及透過 AWS CloudTrail 使用 AWS CodeStar Connections API 呼叫。若要檢視其他供應商動作 (僅限許可動作) 的清單，請參閱 ProviderType 的動作。	2020 年 6 月 28 日
通知規則的新目標類型	您現在可以選擇為 Slack 頻道設定的 AWS Chatbot 用戶端，做為通知規則的目標。如需詳細資訊，請參閱建立通知規則和使用通知規則目標。	2020 年 4 月 2 日

已新增關於 AWS CodeCommit 事件的通知	現在，您可以為與提取請求核准相關的事件設定通知。如需詳細資訊，請參閱儲存庫上通知規則的事件和在 CodeCommit 中使用提取請求。	2020 年 2 月 10 日
可在兩個新增的 AWS 區域中使用通知	開發人員工具主控台現在支援中東 (巴林) 和亞太區域 (香港) 的通知。如需詳細資訊，請參閱《AWS 一般參考》中的 AWS CodeStar 通知。	2020 年 2 月 5 日
新增對加密 Amazon SNS 主題的支援	新增使用加密 Amazon SNS 主題做為通知目標的指導方針。如需詳細資訊，請參閱為通知設定 Amazon SNS 主題。	2020 年 2 月 4 日
通知可能包含 CodeCommit 的工作階段標籤資訊	CodeCommit 的通知現在可以透過使用工作階段標籤來包含使用者身分資訊，例如顯示名稱或電子郵件地址。如需詳細資訊，請參閱概念和在 CodeCommit 中使用標籤來提供身分識別資訊。	2019 年 12 月 19 日
初始版本	這是初版的開發人員工具主控台使用者指南。	2019 年 11 月 5 日

AWS 詞彙表

如需最新的 AWS 術語，請參閱《AWS 詞彙表 參考》中的 AWS 詞彙表。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。