



Guía del usuario

AWS CodeStar



AWS CodeStar: Guía del usuario

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

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

.....	viii
¿Qué es AWS CodeStar?	1
¿Qué puedo hacer con AWS CodeStar?	1
¿Cómo empiezo a trabajar con AWS CodeStar?	2
Configuración	3
Paso 1: crear una cuenta de	3
Inscríbese en una Cuenta de AWS	3
Creación de un usuario con acceso administrativo	4
Paso 2: Crear el rol de AWS CodeStar servicio	5
Paso 3: configurar los permisos de IAM del usuario	5
Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar	6
Paso 5: Abra la consola AWS CodeStar	6
Sigüientes pasos	7
Introducción a AWS CodeStar	8
Paso 1: crear un proyecto de AWS CodeStar	9
Paso 2: añadir información de visualización para su perfil de usuario de AWS CodeStar	15
Paso 3: ver el proyecto	15
Paso 4: confirmar un cambio	17
Paso 5: añadir más miembros del equipo	22
Paso 6: eliminación	24
Paso 7: preparar el proyecto para un entorno de producción	25
Pasos siguientes	25
Tutorial del proyecto sin servidor	26
Información general	27
Paso 1: creación del proyecto	28
Paso 2: explorar recursos del proyecto	29
Paso 3: probar el servicio web	32
Paso 4: configurar la estación de trabajo para editar código de proyecto	33
Paso 5: añadir lógica al servicio web	34
Paso 6: probar el servicio web mejorado	36
Paso 7: añadir una prueba de unidad al servicio web	37
Paso 8: ver los resultados de pruebas de la unidad	39
Paso 9: Eliminación	40
Pasos siguientes	41

Tutorial del proyecto de AWS CLI	41
Paso 1: Descargar y revisar el código fuente de muestra	42
Paso 2: Descargar la plantilla de la cadena de herramientas de muestra	43
Paso 3: Comprobar la plantilla de la cadena de herramientas en AWS CloudFormation	44
Paso 4: Cargar el código fuente y la plantilla de la cadena de herramientas	45
Paso 5: Crear un proyecto en AWS CodeStar	46
Tutorial de proyecto de una habilidad de Alexa	49
Requisitos previos	49
Paso 1: crear el proyecto y conectar su cuenta de desarrollador de Amazon	50
Paso 2: probar la habilidad en el simulador de Alexa	51
Paso 3: explorar los recursos del proyecto	52
Paso 4: haga un cambio a la respuesta de la habilidad	52
Paso 5: configuración de la estación de trabajo local para conectarla al repositorio del proyecto	53
Pasos siguientes	54
Tutorial: creación de un proyecto con un repositorio de código fuente de GitHub	54
Paso 1: crear el proyecto y crear el repositorio de GitHub	54
Paso 2: ver el código fuente	58
Paso 3: crear una solicitud de extracción de GitHub	58
Plantillas de proyecto	60
Archivos de proyectos de AWS CodeStar y recursos	60
Introducción: elija una plantilla del proyecto	62
Elegir una plataforma de computación de plantillas	62
Elija un tipo de aplicación de plantilla	63
Elegir un lenguaje de programación de la plantilla	64
Cómo hacer cambios en su proyecto de AWS CodeStar	64
Cambiar código fuente de aplicación y enviar los cambios	65
Cambiar recursos de aplicaciones con el archivo Template.yml	65
.....	66
Prácticas recomendadas de AWS CodeStar	67
Prácticas recomendadas de seguridad para recursos de AWS CodeStar	67
Prácticas recomendadas para configurar las versiones de dependencias	67
Prácticas recomendadas de monitorización y registro para recursos de AWS CodeStar	68
Trabajar con proyectos de	69
Creación de un proyecto	71
Crear un proyecto en AWS CodeStar (consola)	71

Crear un proyecto en AWS CodeStar (AWS CLI)	77
Utilizar un IDE con AWS CodeStar	84
Usar AWS Cloud9 con AWS CodeStar	85
Utilizar Eclipse con AWS CodeStar	93
Utilizar Visual Studio con AWS CodeStar	98
Cambiar los recursos del proyecto	100
Cambios de recursos admitidos	100
Añadir una etapa a AWS CodePipeline	102
Cambiar la configuración del entorno de AWS Elastic Beanstalk	103
Cambiar una función de AWS Lambda en código fuente	103
Habilitar el seguimiento para un proyecto	103
Añadir un recurso a un proyecto	107
Añadir un rol de IAM a un proyecto	113
Añadir una etapa Prod y un punto de conexión a un proyecto	114
Utilice de forma segura los parámetros de SSM en un proyecto AWS CodeStar	123
Desviar el tráfico para un proyecto de AWS Lambda	125
Llevar su proyecto de AWS CodeStar a producción	133
Creación de un repositorio de GitHub	134
Trabajar con etiquetas de proyectos	135
Añadir una etiqueta a un proyecto	135
Eliminar una etiqueta de un proyecto	135
Obtener una lista de etiquetas para un proyecto	136
Eliminar un proyecto	136
Eliminar un proyecto en AWS CodeStar (consola)	137
Eliminar un proyecto en AWS CodeStar (AWS CLI)	138
Trabajar con equipos de	140
Añadir miembros del equipo a un proyecto	142
Añadir un miembro del equipo (consola)	144
Añadir y ver miembros del equipo (AWS CLI)	146
Administrar permisos de equipo	147
Administrar permisos de equipo (consola)	148
Administrar permisos de equipo (AWS CLI)	149
Eliminar miembros del equipo de un proyecto	150
Eliminar miembros del equipo (consola)	151
Eliminar miembros del equipo (AWS CLI)	151
Trabajar con el perfil de usuario de AWS CodeStar	152

Administrar la información de la visualización	152
Administrar el perfil de usuario (consola)	153
Administrar perfiles de usuario (AWS CLI)	154
Añadir una clave pública al perfil de usuario	157
Administrar la clave pública (consola)	157
Administrar la clave pública (AWS CLI)	158
Conectarse a la instancia de Amazon EC2 con la clave privada	159
Seguridad	161
Protección de los datos	162
Cifrado de datos en AWS CodeStar	163
Identity and Access Management	163
Público	164
Autenticación con identidades	164
Administración de acceso mediante políticas	168
¿Cómo AWS CodeStar funciona con IAM	170
AWS CodeStar Políticas y permisos a nivel de proyecto	181
Ejemplos de políticas basadas en identidades	188
Resolución de problemas	219
Registro de llamadas a la API de AWS CodeStar con AWS CloudTrail	221
Información de AWS CodeStar en CloudTrail	222
Descripción de las entradas de archivos de registro de AWS CodeStar	223
Validación de la conformidad	224
Resiliencia	224
Seguridad de infraestructuras	225
Límites	226
Solución de problemas AWS CodeStar	228
Error al crear el proyecto: el proyecto no se ha creado	228
Creación de proyectos: aparece un error cuando intento editar la configuración de Amazon EC2 al crear un proyecto	229
Eliminación de un AWS CodeStar proyecto: se ha eliminado un proyecto, pero aún existen recursos	230
Fallo en la gestión del equipo: no se ha podido añadir un usuario de IAM a un equipo de un proyecto AWS CodeStar	231
Error de acceso: un usuario federado no puede acceder a un proyecto AWS CodeStar	232
Error de acceso: un usuario federado no puede acceder ni crear un entorno AWS Cloud9	232

Error de acceso: un usuario federado puede crear un AWS CodeStar proyecto, pero no puede ver los recursos del proyecto	233
Error del rol de servicio: el rol de servicio no se ha podido crear	233
Error del rol de servicio: el rol de servicio no es válido o falta	233
Problema con el rol del proyecto: las comprobaciones del estado de AWS Elastic Beanstalk salud fallan en las instancias de un AWS CodeStar proyecto	234
Error del rol de proyecto: un rol de proyecto no es válido o falta	235
Extensiones del proyecto: no se puede conectar a JIRA	235
GitHub: No se puede acceder al historial de confirmaciones, los problemas o el código de un repositorio	235
AWS CloudFormation: Restauración de creación de pila para permisos ausentes	236
AWS CloudFormation no está autorizado a realizar la función de ejecución iam: PassRole on Lambda	236
No se ha podido crear la conexión para un repositorio GitHub	237
Notas de la versión	238
Glosario de AWS	244

El 31 de julio de 2024, Amazon Web Services (AWS) dejará de ofrecer soporte para la creación y visualización de AWS CodeStar proyectos. Después del 31 de julio de 2024, ya no podrá acceder a la AWS CodeStar consola ni crear nuevos proyectos. Sin embargo, los AWS recursos creados mediante este cambio AWS CodeStar, incluidos los repositorios de código fuente, las canalizaciones y las compilaciones, no se verán afectados por este cambio y seguirán funcionando. AWS CodeStar Esta interrupción no afectará a las conexiones ni a las AWS CodeStar notificaciones.

Si desea realizar un seguimiento del trabajo, desarrollar código y crear, probar e implementar sus aplicaciones, Amazon CodeCatalyst ofrece un proceso de inicio simplificado y funciones adicionales para administrar sus proyectos de software. Obtén más información sobre las [funciones](#) y [los precios](#) de Amazon CodeCatalyst.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la versión original de inglés, prevalecerá la versión en inglés.

¿Qué es AWS CodeStar?

AWS CodeStar es un servicio basado en la nube para crear, administrar y trabajar con proyectos de desarrollo de software en AWS. Puede desarrollar, compilar e implementar aplicaciones rápidamente en AWS con un proyecto de AWS CodeStar. Los proyectos de AWS CodeStar crean e integran los servicios de AWS para su cadena de herramientas de proyecto. En función de su selección de plantilla del proyecto de AWS CodeStar, esa cadena de herramientas puede incluir control de origen, compilación, implementación, servidores virtuales o recursos sin servidor, etc. AWS CodeStar también administra los permisos necesarios para los usuarios de proyectos (denominados miembros del equipo). Al añadir usuarios como miembros del equipo a un proyecto de AWS CodeStar, los propietarios de proyectos pueden conceder a cada miembro del equipo acceso a un proyecto y sus recursos apropiado a su rol y de forma rápida y sencilla.

Temas

- [¿Qué puedo hacer con AWS CodeStar?](#)
- [¿Cómo empiezo a trabajar con AWS CodeStar?](#)

¿Qué puedo hacer con AWS CodeStar?

Puede utilizar AWS CodeStar para ayudarle a configurar el desarrollo de aplicaciones en la nube y gestionar el desarrollo desde un único panel centralizado. En concreto, puede:

- Iniciar nuevos proyectos de software en AWS en cuestión de minutos utilizando plantillas para las aplicaciones web, los servicios web y mucho más: AWS CodeStar incluye plantillas de proyectos para varios tipos de proyectos y lenguajes de programación. Como AWS CodeStar se encarga de la configuración, todos sus recursos de proyectos están configurados para funcionar de forma conjunta.
- Administrar el acceso a proyectos de su equipo: AWS CodeStar proporciona una consola central donde puede asignar a los miembros del equipo del proyecto los roles que necesitan para obtener acceso a las herramientas y los recursos. Estos permisos se aplican automáticamente en todos los servicios de AWS que se utilizan en el proyecto, por lo que no es necesario crear ni administrar políticas de IAM complejas.
- Visualizar, operar y colaborar en los proyectos en un único lugar: AWS CodeStar incluye un panel de proyectos que proporciona una visión general del proyecto, su cadena de herramientas y eventos importantes. Puede monitorizar la actividad del proyecto más reciente, como

confirmaciones de código recientes y hacer un seguimiento del estado de los cambios de código, crear resultados e implementaciones, todo ello desde la misma página web. Puede supervisar lo que sucede en el proyecto desde un único panel y profundizar en los problemas a investigar.

- Iterar rápidamente con todas las herramientas que necesita: AWS CodeStar incluye una cadena de herramientas de desarrollo integrada para el proyecto. Los miembros del equipo insertan código y los cambios se implementan automáticamente. La integración con seguimiento de problemas permite a los miembros del equipo hacer un seguimiento de lo que hay que hacer a continuación. Puede trabajar junto con su equipo con mayor rapidez y eficacia en todas las fases de entrega del código.

¿Cómo empiezo a trabajar con AWS CodeStar?

Para empezar a trabajar con AWS CodeStar:

1. Prepárese para usar AWS CodeStar siguiendo los pasos de [Configuración AWS CodeStar](#).
2. Pruebe con AWS CodeStar siguiendo los pasos del tutorial [Introducción a AWS CodeStar](#).
3. Comparta el proyecto con otros desarrolladores siguiendo los pasos de [Añadir miembros de equipo a un proyecto de AWS CodeStar](#).
4. Integre su IDE favorito siguiendo los pasos que se indican en [Utilizar un IDE con AWS CodeStar](#).

Configuración AWS CodeStar

Antes de empezar a usarlo AWS CodeStar, debe completar los siguientes pasos.

Temas

- [Paso 1: crear una cuenta de](#)
- [Paso 2: Crear el rol de AWS CodeStar servicio](#)
- [Paso 3: configurar los permisos de IAM del usuario](#)
- [Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar](#)
- [Paso 5: Abra la consola AWS CodeStar](#)
- [Siguiendo pasos](#)

Paso 1: crear una cuenta de

Inscríbese en una Cuenta de AWS

Si no tiene una Cuenta de AWS, complete los siguientes pasos para crearlo.

Para suscribirte a una Cuenta de AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga las instrucciones que se le indiquen.

Parte del procedimiento de registro consiste en recibir una llamada telefónica e indicar un código de verificación en el teclado del teléfono.

Cuando te registras en una Cuenta de AWS, Usuario raíz de la cuenta de AWS se crea un. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar [tareas que requieren acceso de usuario raíz](#).

AWS te envía un correo electrónico de confirmación una vez finalizado el proceso de registro. Puede ver la actividad de la cuenta y administrar la cuenta en cualquier momento entrando en <https://aws.amazon.com/> y seleccionando Mi cuenta.

Creación de un usuario con acceso administrativo

Después de crear un usuario administrativo Cuenta de AWS, asegúrelo Usuario raíz de la cuenta de AWS AWS IAM Identity Center, habilite y cree un usuario administrativo para no usar el usuario root en las tareas diarias.

Proteja su Usuario raíz de la cuenta de AWS

1. Inicie sesión [AWS Management Console](#) como propietario de la cuenta seleccionando el usuario root e introduciendo su dirección de Cuenta de AWS correo electrónico. En la siguiente página, escriba su contraseña.

Para obtener ayuda para iniciar sesión con el usuario raíz, consulte [Signing in as the root user](#) en la Guía del usuario de AWS Sign-In .

2. Active la autenticación multifactor (MFA) para el usuario raíz.

Para obtener instrucciones, consulte [Habilitar un dispositivo MFA virtual para el usuario Cuenta de AWS raíz \(consola\)](#) en la Guía del usuario de IAM.

Creación de un usuario con acceso administrativo

1. Activar IAM Identity Center.

Consulte las instrucciones en [Activar AWS IAM Identity Center](#) en la Guía del usuario de AWS IAM Identity Center .

2. En IAM Identity Center, conceda acceso administrativo a un usuario.

Para ver un tutorial sobre su uso Directorio de IAM Identity Center como fuente de identidad, consulte [Configurar el acceso de los usuarios con la configuración predeterminada Directorio de IAM Identity Center en la](#) Guía del AWS IAM Identity Center usuario.

Iniciar sesión como usuario con acceso de administrador

- Para iniciar sesión con el usuario de IAM Identity Center, utilice la URL de inicio de sesión que se envió a la dirección de correo electrónico cuando creó el usuario de IAM Identity Center.

Para obtener ayuda para iniciar sesión con un usuario del Centro de identidades de IAM, consulte [Iniciar sesión en el portal de AWS acceso](#) en la Guía del AWS Sign-In usuario.

Concesión de acceso a usuarios adicionales

1. En IAM Identity Center, cree un conjunto de permisos que siga la práctica recomendada de aplicar permisos de privilegios mínimos.

Para conocer las instrucciones, consulte [Create a permission set](#) en la Guía del usuario de AWS IAM Identity Center .

2. Asigne usuarios a un grupo y, a continuación, asigne el acceso de inicio de sesión único al grupo.

Para conocer las instrucciones, consulte [Add groups](#) en la Guía del usuario de AWS IAM Identity Center .

Paso 2: Crear el rol de AWS CodeStar servicio

Cree un [rol de servicio](#) que se utilice para conceder AWS CodeStar permisos de administración de AWS recursos y de IAM en su nombre. Solo tiene que crear el rol del servicio una vez.

Important

Para crear el rol de servicio, debe haber iniciado sesión como usuario administrativo de (o cuenta raíz). Para obtener más información, consulte [Creación del primer grupo y usuario de IAM](#).

1. Abra la AWS CodeStar consola en <https://console.aws.amazon.com/codestar/>.
2. Elija Start project (Comenzar proyecto).

Si no ve Start project (Comenzar proyecto) y se le dirige a la página con el listado de proyectos, significa que se ha creado el rol de servicio.

3. En Create service role (Crear rol de servicio), elija Yes, create role (Sí, crear rol).
4. Salga del asistente. Volverá a este punto más tarde.

Paso 3: configurar los permisos de IAM del usuario

Además del usuario administrativo, puede utilizarla AWS CodeStar como usuario de IAM, usuario federado, usuario raíz o como rol asumido. Para obtener información sobre lo que AWS CodeStar

pueden hacer los usuarios de IAM frente a los usuarios federados, consulte [AWS CodeStar IAMFunciones](#)

Si no ha configurado ningún usuario de IAM, consulte [Usuario de IAM](#).

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en: AWS IAM Identity Center

Cree un conjunto de permisos. Siga las instrucciones de [Creación de un conjunto de permisos](#) en la Guía del usuario de AWS IAM Identity Center .

- Usuarios administrados en IAM a través de un proveedor de identidades:

Cree un rol para la federación de identidades. Siga las instrucciones descritas en [Creación de un rol para un proveedor de identidad de terceros \(federación\)](#) en la Guía del usuario de IAM.

- Usuarios de IAM:

- Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en [Creación de un rol para un usuario de IAM](#) en la Guía del usuario de IAM.

- (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en [Adición de permisos a un usuario \(consola\)](#) de la Guía del usuario de IAM.

Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar

Muchos AWS CodeStar proyectos utilizan AWS CodeDeploy o AWS Elastic Beanstalk implementan código en las instancias de Amazon EC2. Para obtener acceso a instancias de Amazon EC2 asociadas con el proyecto, cree un par de claves de Amazon EC2 para el usuario de IAM. El usuario de IAM debe disponer de permisos para crear y administrar claves de Amazon EC2 (por ejemplo, permisos para realizar las acciones `ec2:CreateKeyPair` y `ec2:ImportKeyPair`). Para obtener más información, consulte [Pares de claves de Amazon EC2](#).

Paso 5: Abra la consola AWS CodeStar

Inicie sesión en la AWS CodeStar consola y AWS Management Console, a continuación, ábrala en <https://console.aws.amazon.com/codestar/>.

Siguientes pasos

¡Enhorabuena! Ha completado la configuración. Para empezar a trabajar con AWS CodeStar ella, consulte [Introducción a AWS CodeStar](#).

Introducción a AWS CodeStar

En este tutorial, usará AWS CodeStar para crear una aplicación web. Este proyecto incluye código de muestra en un repositorio de origen, una cadena de herramientas de implementación continua y un panel de proyecto en el que puede ver y supervisar el proyecto.

Si sigue los pasos que se indican a continuación, podrá:

- Crear un proyecto en AWS CodeStar.
- Explorar el proyecto.
- Confirmar un cambio de código.
- Ver el cambio de código implementado automáticamente.
- Añadir otras personas para trabajar en el proyecto.
- Eliminar los recursos del proyecto cuando ya no los necesite.

Note

Si no lo ha hecho todavía, deberá completar primero los pasos de [Configuración AWS CodeStar](#), incluido el [Paso 2: Crear el rol de AWS CodeStar servicio](#). Debe haber iniciado sesión con una cuenta de un usuario administrador en IAM. Para crear un proyecto, debe iniciar sesión en la AWS Management Console mediante un usuario de IAM que tenga la política **AWSCodeStarFullAccess**.

Temas

- [Paso 1: crear un proyecto de AWS CodeStar](#)
- [Paso 2: añadir información de visualización para su perfil de usuario de AWS CodeStar](#)
- [Paso 3: ver el proyecto](#)
- [Paso 4: confirmar un cambio](#)
- [Paso 5: añadir más miembros del equipo](#)
- [Paso 6: eliminación](#)
- [Paso 7: preparar el proyecto para un entorno de producción](#)
- [Pasos siguientes](#)
- [Tutorial: creación y administración de un proyecto sin servidor en AWS CodeStar](#)

- [Tutorial: Crear un proyecto en AWS CodeStar con la AWS CLI](#)
- [Tutorial: crear un proyecto de una habilidad de Alexa en AWS CodeStar](#)
- [Tutorial: creación de un proyecto con un repositorio de código fuente de GitHub](#)

Paso 1: crear un proyecto de AWS CodeStar

En este paso, creará un proyecto de desarrollo de software de JavaScript (Node.js) para una aplicación web. Utilizará una plantilla de proyecto de AWS CodeStar para crear el proyecto.

Note

La plantilla de proyecto AWS CodeStar utilizada en este tutorial emplea las siguientes opciones:

- Categoría de la aplicación: aplicación web
- Lenguaje de programación: Node.js
- Servicio de AWS: Amazon EC2

Si selecciona otras opciones, puede que su experiencia no coincida con lo que se documenta en este tutorial.

Para crear un proyecto en AWS CodeStar

1. Inicie sesión en la AWS Management Console y, a continuación, abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.

Asegúrese de que ha iniciado sesión en la región de AWS donde desea crear el proyecto y sus recursos. Por ejemplo, para crear un proyecto en Este de EE. UU. (Ohio), asegúrese de haber seleccionado esa región de AWS. Para obtener más información sobre las regiones de AWS en las que AWS CodeStar está disponible, consulte [Regiones y puntos de conexión](#) en la Referencia general de AWS .

2. En la página AWS CodeStar, seleccione Crear proyecto.
3. En la página Elegir una plantilla de proyecto, seleccione el tipo de proyecto en la lista de plantillas de proyectos de AWS CodeStar. Puede utilizar la barra de filtros para restringir las opciones. Por ejemplo, en el caso de un proyecto de aplicación web escrito en Node.js que se


implementará en instancias de Amazon EC2, seleccione las casillas de verificación Aplicación web, Node.js y Amazon EC2. A continuación, elija entre las plantillas disponibles para ese conjunto de opciones.

Para obtener más información, consulte [Plantillas de proyecto de AWS CodeStar](#).

4. Elija Siguiente.
5. En el campo de entrada de texto Nombre del proyecto, introduzca un nombre para el proyecto, como *Mi primer proyecto*. El ID del proyecto, el ID del proyecto se deriva del nombre de dicho proyecto, pero se limita a 15 caracteres.

Por ejemplo, el ID predeterminado de un proyecto con el nombre *Mi primer proyecto* sería *mi-primer-proye*. Este ID de proyecto es la base de los nombres de todos los recursos asociados al proyecto. AWS CodeStar utiliza este ID de proyecto como parte de la dirección URL del repositorio de código y para los nombres de roles de acceso de seguridad y políticas relacionados en IAM. Una vez creado el proyecto, el ID del proyecto no puede modificarse. Para editar el ID del proyecto antes de crearlo, en ID del proyecto, introduzca el ID que desee utilizar.

Para obtener más información sobre los límites de los nombres de proyectos y los ID de proyectos, consulte [Límites en AWS CodeStar](#).

 Note

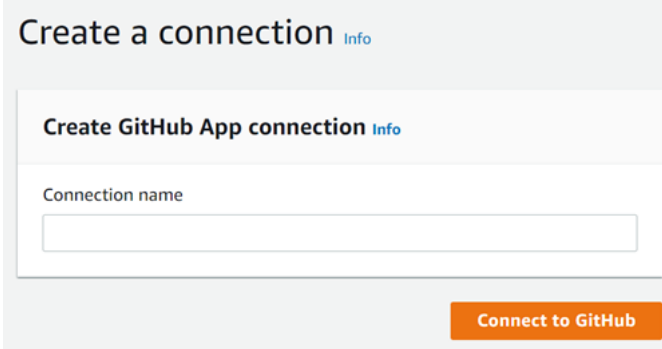
Los ID de proyecto deben ser únicos de la cuenta de AWS dentro de una región de AWS.

6. Elija el proveedor del repositorio: AWS CodeCommit o GitHub.
7. Si ha elegido AWS CodeCommit para Nombre del repositorio, acepte el nombre del repositorio de AWS CodeCommit predeterminado o escriba otro diferente. A continuación, vaya al paso 9.
8. Si elige GitHub, debe elegir o crear un recurso de conexión. Si ya tiene una conexión, selecciónela en el campo de búsqueda. De lo contrario, cree una conexión nueva ahora. Seleccione Conectarse a GitHub.

Se mostrará la página Crear una conexión.

Note

Para crear una conexión, debe tener una cuenta de GitHub. Si va a crear una conexión para una organización, debe ser el propietario de la organización.



The screenshot shows a web interface for creating a connection. At the top, it says "Create a connection" with a small "Info" link. Below that is a section titled "Create GitHub App connection" also with an "Info" link. Underneath, there is a text input field labeled "Connection name". At the bottom right of this section is an orange button that says "Connect to GitHub".

- a. En Cree una conexión de aplicación de GitHub, en el campo de entrada de texto Nombre de la conexión, escriba un nombre para la conexión. Seleccione Conectarse a GitHub.

Aparecerá la página Conectarse a GitHub, donde se muestra el campo Aplicaciones de GitHub.

- b. En Aplicaciones de GitHub, elija la instalación de una aplicación o elija Instalar una aplicación nueva para crear una.

Note

Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya instaló la aplicación AWS Connector for GitHub, elija la aplicación y omita este paso.

- c. En la página Instalar el conector de AWS para GitHub, elija la cuenta donde desee instalar la aplicación.

Note

Si instaló la aplicación previamente, puede elegir Configurar para dirigirse a una página de modificación para la instalación de la aplicación o puede utilizar el botón Atrás para volver a la consola.

- d. Si aparece la página Confirmar contraseña para continuar, introduzca su contraseña de GitHub y, a continuación, seleccione Iniciar sesión.
- e. En la página Instalar el conector de AWS para GitHub, deje los valores predeterminados y seleccione Instalar.
- f. En la página Conectarse a GitHub, el ID de instalación para la nueva instalación aparece en el campo de entrada de texto Aplicaciones de GitHub.

Una vez creada la conexión, en la página de creación del proyecto de CodeStar, aparece el mensaje Listo para conectarse.


Note

Puede ver la conexión en la sección Configuración de la consola de Herramientas para desarrolladores. Para obtener más información, consulte [Introducción a las conexiones](#).

Select a repository provider


CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).




The GitHub repository provider now uses CodeStar Connections

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or



Ready to connect
Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

[Redacted]
▼

Repository name

The name of the new repository.

Repository description

An optional description of the new repository.

Public

- g. Para Propietario del repositorio, seleccione la organización de GitHub o su cuenta personal de GitHub.
- h. Para Nombre del repositorio, acepte el nombre del repositorio de GitHub predeterminado o escriba otro diferente.
- i. Elija Público o Privado.

Note

Si desea utilizar AWS Cloud9 como entorno de desarrollo, debe elegir Público.

- j. (Opcional) En Descripción del repositorio, escriba una descripción para el repositorio de GitHub.

Note

Si selecciona una plantilla de proyecto de habilidades de Alexa, deberá conectar una cuenta de desarrollador de Amazon. Para obtener más información acerca de cómo trabajar con proyectos de habilidades de Alexa, consulte [Tutorial: crear un proyecto de una habilidad de Alexa en AWS CodeStar](#).

9. Si su proyecto está desplegado en instancias de Amazon EC2 y desea realizar cambios, configure las instancias de Amazon EC2 en Configuración de Amazon EC2. Por ejemplo, puede elegir entre los tipos de instancia disponibles para el proyecto.

Note

Los distintos tipos de instancia de Amazon EC2 proporcionan diferentes niveles de operatividad informática y podrían tener distintos costos asociados. Para obtener más información, consulte [Tipos de instancias de Amazon EC2](#) y [Precios de Amazon EC2](#). Si tiene más de una nube privada virtual (VPC) o varias subredes creadas en Amazon Virtual Private Cloud, también puede elegir la VPC y la subred que va a utilizar. Sin embargo, si elige un tipo de instancia de Amazon EC2 que no sea compatible con instancias dedicadas, no puede elegir una VPC cuya tenencia de instancias esté establecida en Dedicada.

Para obtener más información, consulte [¿Qué es Amazon VPC?](#) y [Conceptos básicos de las instancias dedicadas](#).

Para Par de claves, seleccione el par de claves de Amazon EC2 que ha creado en [Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar](#). Seleccione Confirmo que tengo acceso al archivo de clave privada.

10. Elija Siguiente.
11. Revise los recursos y los detalles de la configuración.
12. Seleccione Siguiente o Crear proyecto. (La selección mostrada depende de la plantilla del proyecto).

Es posible que el proyecto, que incluye el repositorio, tarde unos minutos en crearse.

- Una vez que el proyecto tenga un repositorio, puede utilizar la página Repositorio para configurar el acceso al mismo. Utilice los enlaces que se encuentran en Próximos pasos para configurar un IDE, configurar el seguimiento de problemas o añadir miembros del equipo a su proyecto.

Paso 2: añadir información de visualización para su perfil de usuario de AWS CodeStar

Al crear un proyecto, se le añade al equipo del proyecto como propietario. Si es la primera vez que utiliza AWS CodeStar, se le pedirá que proporcione:

- El nombre de visualización para mostrar a otros usuarios.
- La dirección de correo electrónico para mostrar a otros usuarios.

Esta información se utiliza en su perfil de usuario de AWS CodeStar. Los perfiles de usuario no son específicos de un proyecto, pero se limitan a una región de AWS. Debe crear un perfil de usuario en cada región de AWS a la que pertenezcan los proyectos. Cada perfil puede contener información diferente, si lo prefiere.

Escriba un nombre de usuario y una dirección de correo electrónico y, a continuación, elija Next (Siguiente).

Note

Este nombre de usuario y dirección de correo electrónico se utiliza en su perfil de usuario de AWS CodeStar. Si el proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio GitHub o problemas en Atlassian JIRA), dichos proveedores de recursos podrían tener sus propios perfiles de usuario, con nombres de usuario y direcciones de correo electrónico distintos. Para obtener más información, consulte la documentación del proveedor de recursos.

Paso 3: ver el proyecto

La página del proyecto de AWS CodeStar es donde tanto usted como su equipo pueden ver el estado de los recursos del proyecto, incluidas las últimas confirmaciones del proyecto, el estado de la

canalización de entrega continua y el desempeño de las instancias. Para ver más información sobre cualquiera de estos recursos, seleccione la página correspondiente en la barra de navegación.

En el nuevo proyecto, la barra de navegación contiene las siguientes páginas:

- La página Información general contiene información sobre la actividad del proyecto, los recursos del proyecto y el contenido README del proyecto.
- La página IDE es donde se conecta el proyecto a un entorno de desarrollo integrado (IDE) para modificar, probar y enviar los cambios en el código fuente. Contiene instrucciones para configurar los IDE tanto para GitHub como para los repositorios de AWS CodeCommit e información sobre los entornos de AWS Cloud9.
- La página Repositorio muestra los detalles del repositorio, incluidos el nombre, el proveedor, cuándo se modificó por última vez y las direcciones URL de clonación. También puede ver información sobre las confirmaciones más recientes, así como ver y crear solicitudes de extracción.
- La página Canalización muestra información de CI/CD sobre la canalización. Puede ver los detalles de la canalización, como el nombre, la acción más reciente y el estado. Puede ver el historial de la canalización y liberar un cambio. También puede ver el estado de los pasos individuales de la canalización.
- La página Monitorización muestra tanto Amazon EC2 como las métricas de AWS Lambda en función de la configuración del proyecto. Por ejemplo, se muestra el uso de la CPU de las instancias de Amazon EC2 implementadas mediante AWS Elastic Beanstalk o los recursos de CodeDeploy en la canalización. En los proyectos que utilizan AWS Lambda, se muestra la invocación y las métricas de error de la función de Lambda. Esta información se muestra por hora. Si ha utilizado la plantilla del proyecto de AWS CodeStar sugerida para este tutorial, debería ver un pico de actividad perceptible a medida que la aplicación se implemente por primera vez en dichas instancias. Puede actualizar la monitorización para ver los cambios en el estado de la instancia, lo que puede ayudarle a identificar problemas o la necesidad de más recursos.
- La página Problemas es para integrar el proyecto de AWS CodeStar con un proyecto de Atlassian JIRA. La configuración de este icono le permitirá a usted y al equipo del proyecto hacer un seguimiento de los problemas de JIRA desde el panel del proyecto.

En el panel de navegación del lateral izquierdo de la consola se puede navegar entre las páginas de Proyecto, Equipo y Configuración.

Paso 4: confirmar un cambio

En primer lugar, eche un vistazo a la aplicación de muestra que se incluye en el proyecto. Para ver el aspecto de la aplicación, seleccione Ver la aplicación desde cualquier parte de la navegación del proyecto. La aplicación web de muestra se visualizará en una nueva ventana o en la pestaña del navegador. Esta es la muestra de proyecto que ha diseñado e implementado AWS CodeStar.


Si desea ver el código, en la barra de navegación, seleccione Repositorio. Seleccione el enlace que aparece debajo de Nombre del repositorio y el repositorio del proyecto se abrirá en una nueva pestaña o ventana. Lea el contenido del archivo readme del repositorio (README .md) y examine el contenido de los archivos.

En este paso, realizará un cambio en el código y, a continuación, lo enviará al repositorio. Puede hacerlo de distintas maneras:

- Si el código del proyecto se almacena en un repositorio de CodeCommit o GitHub, puede utilizar AWS Cloud9 para trabajar con el código directamente desde el navegador web, sin necesidad de instalar ninguna herramienta. Para obtener más información, consulte [Crear un entorno de AWS Cloud9 para un proyecto](#).
- Si el código del proyecto está almacenado en un repositorio de CodeCommit y tiene Visual Studio o Eclipse instalado, puede utilizar el AWS Toolkit for Visual Studio o AWS Toolkit for Eclipse para conectarse con más facilidad al código. Para obtener más información, consulte [Utilizar un IDE con AWS CodeStar](#). Si no tiene Visual Studio o Eclipse instalado, entonces instale un cliente de Git y siga las instrucciones más adelante en este paso.
- Si el código del proyecto está almacenado en un repositorio de GitHub, puede utilizar las herramientas de su IDE para conectarse a GitHub.
 - Para Visual Studio, puede utilizar herramientas como GitHub Extension for Visual Studio. Para obtener información adicional, consulte la página [Overview](#) en el sitio web de GitHub Extension for Visual Studio y [Getting Started with GitHub for Visual Studio](#) en el sitio web de GitHub.
 - Para Eclipse, puede utilizar una herramienta como EGit for Eclipse. Para obtener más información, consulte la [EGit Documentation](#) en el sitio web de EGit.
 - Para ver otros IDE, consulte la documentación de su IDE.
- Para otros tipos de repositorios de código, consulte la documentación del proveedor del repositorio.

Las siguientes instrucciones muestran cómo realizar un cambio insignificante en la muestra.

Para configurar el equipo para confirmar los cambios (usuario de IAM)

 Note


Para este procedimiento se presupone que el código del proyecto está almacenado en un repositorio de CodeCommit. Para otros tipos de repositorios de código, consulte la documentación del proveedor del repositorio y, a continuación, pase al siguiente procedimiento, [Para clonar el repositorio del proyecto y hacer un cambio](#).

Si el código está almacenado en CodeCommit y ya está utilizando CodeCommit o ha utilizado la consola de AWS CodeStar para crear un entorno de desarrollo de AWS Cloud9 para el proyecto, no debe configurar nada más. Pase al siguiente procedimiento, [Para clonar el repositorio del proyecto y hacer un cambio](#).

1. [Instale Git](#) en el equipo local.
2. Inicie sesión en la AWS Management Console y abra la consola de IAM en <https://console.aws.amazon.com/iam/>.

Inicie sesión con el usuario de IAM que va a utilizar las credenciales de Git para las conexiones al repositorio del proyecto de AWS CodeStar en CodeCommit.

3. En el panel de navegación de la consola de IAM, elija Usuarios y, en la lista de usuarios, seleccione su usuario de IAM.
4. En la página de detalles del usuario, seleccione la pestaña Credenciales de seguridad, y en Credenciales de Git HTTPS para CodeCommit, seleccione Generar.

 Note

No puede elegir sus propias credenciales de inicio de sesión para las credenciales de Git. Para obtener más información, consulte [Usar credenciales de Git y HTTPS con CodeCommit](#).

5. Copie sus credenciales de inicio de sesión que IAM generó. Puede elegir Show (Mostrar) y, a continuación, copiar y pegar esta información en un archivo seguro en el equipo local o puede elegir Download credentials (Descargar credenciales) para descargar dicha información como archivo .CSV. Necesitará esta información para conectarse a CodeCommit.

Una vez que haya guardado las credenciales, elija Cerrar.

⚠ Important

Es la única forma de guardar las credenciales de inicio de sesión. Si no lo hace, podrá copiar el nombre de usuario de la consola de IAM, pero no podrá buscar la contraseña. Deberá restablecer la contraseña y, a continuación, guardarla.

Para configurar el equipo para confirmar los cambios (usuario federado)

Puede utilizar la consola para cargar archivos en el repositorio o puede utilizar Git para conectarse desde el equipo local. Si está utilizando un acceso federado, siga los pasos que se indican a continuación para utilizar Git para conectarse y clonar su repositorio desde el equipo local.

ℹ Note

Para este procedimiento se presupone que el código del proyecto está almacenado en un repositorio de CodeCommit. Para otros tipos de repositorios de código, consulte la documentación del proveedor del repositorio y, a continuación, pase al siguiente procedimiento, [Para clonar el repositorio del proyecto y hacer un cambio](#).

1. [Instale Git](#) en el equipo local.
2. [Instale la AWS CLI](#).
3. Configure sus credenciales de seguridad temporales para un usuario federado. Para obtener más información, consulte [Acceso temporal a los repositorios de CodeCommit](#). Las credenciales temporales constan de:
 - Clave de acceso de AWS
 - Clave secreta de AWS
 - Token de sesión

Para obtener más información sobre las credenciales temporales, consulte [Permisos para GetFederationToken](#).

4. Conéctese al repositorio utilizando el auxiliar de credenciales de la AWS CLI. Para obtener más información, consulte [Pasos para configurar conexiones HTTPS a repositorios de CodeCommit en Linux, macOS o Unix con el ayudante de credenciales de la CLI de AWS](#) o [Pasos para](#)

[configurar conexiones HTTPS a repositorios de CodeCommit en Windows con el ayudante de credenciales de la CLI de AWS.](#)

5. En el siguiente ejemplo se muestra cómo conectarse a un repositorio de CodeCommit y enviar una confirmación a dicho repositorio.

Ejemplo: Para clonar el repositorio del proyecto y hacer un cambio

Note

Este procedimiento muestra cómo clonar el repositorio del código del proyecto a su equipo, realizar un cambio en el archivo `index.html` del proyecto y, a continuación, introducir el cambio en el repositorio remoto. En este procedimiento se presupone que el código del proyecto está almacenado en un repositorio de CodeCommit y que está utilizando un cliente de Git desde la línea de comandos. Para otros tipos de herramientas o repositorios de código, consulte la documentación del proveedor acerca de cómo clonar el repositorio, cambiar el archivo y, a continuación, enviar el código.

1. Si utilizó la consola de AWS CodeStar para crear un entorno de desarrollo de AWS Cloud9 para el proyecto, abra el entorno de desarrollo y, a continuación, vaya al paso 3 de este procedimiento. Para abrir el entorno de desarrollo, consulte [Abrir un entorno de AWS Cloud9 para un proyecto](#).

Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación, seleccione Repositorio. En Clonar URL, elija el protocolo para el tipo de conexión que haya configurado para CodeCommit y, a continuación, copie el enlace. Por ejemplo, si ha seguido los pasos del procedimiento anterior para configurar las credenciales de Git para CodeCommit, seleccione HTTPS.

2. En el equipo local, abra un terminal o una ventana de línea de comandos y cambie los directorios a un directorio temporal. Ejecute el comando `git clone` para clonar el repositorio en su equipo. Pegue el enlace que ha copiado. Por ejemplo, para CodeCommit con HTTPS:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/my-first-projec
```

La primera vez que se conecte, se le pedirán las credenciales de inicio de sesión del repositorio. Para CodeCommit, introduzca las credenciales de inicio de sesión de Git que descargó en el procedimiento anterior.

3. Vaya al directorio clonado en su equipo y examine el contenido.
4. Abra el archivo `index.html` (en la carpeta pública) y realice un cambio en el archivo. Por ejemplo, añada un párrafo detrás de la etiqueta `<H2>` como:

```
<P>Hello, world!</P>
```

Guarde el archivo.

5. En el terminal o en la línea de comandos, añada el archivo modificado y, a continuación, confirme e introduzca el cambio:

```
git add index.html
git commit -m "Making my first change to the web app"
git push
```

6. En la página Repositorio, consulte los cambios en curso. Debería ver que el historial de confirmaciones del repositorio se actualiza con su confirmación, incluido el mensaje de confirmación. En la página Canalización, puede observar que la canalización recoge el cambio en el repositorio y comienza a crearlo e implementarlo. Una vez implementada la aplicación web, puede seleccionar Ver la aplicación para ver los cambios.

Note

Si se muestra Failed (Error) en alguna de las fases de canalización, consulte la siguiente ayuda para la resolución de problemas:

- Para la etapa Origen, consulte [Solución de problemas de AWS CodeCommit](#) en la Guía del usuario de AWS CodeCommit.
- Para la etapa de compilación, consulte [Solución de problemas de AWS CodeBuild](#) en la Guía del usuario de AWS CodeBuild.
- Para la etapa de implementación, consulte [Solución de problemas de AWS CloudFormation](#) en la Guía del usuario de AWS CloudFormation.

- Para los demás problemas, consulte [Solución de problemas AWS CodeStar](#).

Paso 5: añadir más miembros del equipo

Cada proyecto de AWS CodeStar ya está configurado con tres roles de AWS CodeStar. Cada rol ofrece su propio nivel de acceso al proyecto y sus recursos:

- **Propietario:** puede añadir y eliminar miembros del equipo, cambiar el panel del proyecto y eliminar el proyecto.
- **Colaborador:** puede cambiar el panel del proyecto y aportar código si el código está almacenado en CodeCommit, pero no puede añadir ni quitar miembros del equipo ni borrar el proyecto. Este es el rol que debe elegir para la mayoría de los miembros del equipo en un proyecto de AWS CodeStar.
- **Lector:** puede ver el panel del proyecto, el código del proyecto si dicho código está almacenado en CodeCommit y el estado del proyecto, pero no puede trasladar, añadir ni quitar iconos del panel del proyecto.

Important

Si su proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA), el acceso a dichos recursos lo controla el proveedor de recursos, no AWS CodeStar. Para obtener más información, consulte la documentación del proveedor de recursos.

Cualquier persona que tenga acceso a un proyecto de AWS CodeStar podría utilizar la consola de AWS CodeStar para acceder a los recursos que están fuera de AWS pero que están relacionados con el proyecto.

AWS CodeStar no permite que los miembros del equipo del proyecto participen en cualquier entorno de desarrollo de AWS Cloud9 de un proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#).

Para obtener más información acerca de los equipos y roles de proyectos, consulte [Trabajar con equipos de AWS CodeStar](#).

Para añadir un miembro del equipo a un proyecto de AWS CodeStar (consola)

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos y, a continuación, seleccione su proyecto.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.
4. En la página Miembros del equipo, elija Añadir miembro del equipo.
5. En Elegir usuario, realice una de las siguientes operaciones:
 - Si ya existe un usuario de IAM para la persona que desea añadir, seleccione a dicho usuario de IAM de la lista.

Note

Los usuarios que ya se hayan añadido a otro proyecto de AWS CodeStar aparecerán en la lista Usuarios de AWS CodeStar existentes.

En Rol del proyecto, elija el rol de AWS CodeStar (propietario, colaborador o lector) que desee otorgar a este usuario. Este es un rol de nivel de proyecto de AWS CodeStar que solo puede cambiar el propietario del proyecto. Cuando se aplica a un usuario de IAM, el rol proporciona todos los permisos necesarios para obtener acceso a los recursos del proyecto de AWS CodeStar. Aplica las políticas necesarias para crear y administrar credenciales de Git para código almacenado en CodeCommit en IAM o bien para cargar las claves de SSH de Amazon EC2 para el usuario en IAM.


Important

No puede proporcionar ni cambiar la información del nombre o del correo electrónico de visualización de un usuario de IAM a menos que haya iniciado sesión en la consola como dicho usuario. Para obtener más información, consulte [Administración de la información de visualización de su perfil de usuario de AWS CodeStar](#).

Seleccione Agregar el miembro del equipo.

- Si no existe un usuario de IAM para la persona que desea añadir al proyecto, seleccione Crear nuevo usuario de IAM. Se le redirigirá a la consola de IAM, donde podrá crear un nuevo usuario de IAM. Consulte [Creación de usuarios de IAM](#) en la Guía del usuario de

IAM para obtener más información. Tras crear su usuario de IAM, vuelva a la consola de AWS CodeStar, actualice la lista de usuarios y elija de la lista desplegable el usuario de IAM que creó. Introduzca el nombre de visualización de AWS CodeStar, la dirección de correo electrónico y el rol del proyecto que desee aplicar a este nuevo usuario y, a continuación, seleccione Agregar el miembro del equipo.

 Note

Para facilitar la administración, al menos un usuario debe tener asignado el rol de propietario del proyecto.

6. Envíe al nuevo miembro del equipo la siguiente información:

- Información de conexión para su proyecto de AWS CodeStar.
- Si el código fuente está almacenado en CodeCommit, [instrucciones para configurar el acceso con credenciales de Git](#) en el repositorio de CodeCommit desde los equipos locales.
- Información sobre cómo el usuario puede administrar su nombre que mostrar, dirección de correo electrónico y clave pública SSH de Amazon EC2, tal como se describe en [Trabajar con el perfil de usuario de AWS CodeStar](#).
- Contraseña de un solo uso e información de conexión, si el usuario es nuevo en AWS y ha creado un usuario de IAM para esa persona. La contraseña caducará la primera vez que el usuario inicie sesión. El usuario debe elegir una contraseña nueva.

Paso 6: eliminación

¡Enhorabuena! Ha terminado el tutorial. Si no desea seguir utilizando este proyecto y sus recursos, debe eliminarlo para evitar posibles cargos recurrentes en su cuenta de AWS.

Para eliminar un proyecto en AWS CodeStar

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos.
3. Seleccione el proyecto que desee eliminar y elija Eliminar.

O bien, abra el proyecto y seleccione Configuración en el panel de navegación del lado izquierdo de la consola. En la página de detalles del proyecto, seleccione Eliminar proyecto.

4. En la página Confirmación de eliminación, escriba eliminar. Mantenga seleccionada la opción Eliminar recursos si desea eliminar los recursos del proyecto. Elija Eliminar.

La eliminación de un proyecto puede tardar varios minutos. Una vez eliminado, el proyecto ya no aparece en la lista de proyectos en la consola de AWS CodeStar.

 Important

Si su proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio GitHub o problemas en Atlassian JIRA), dichos recursos no se eliminan, incluso si selecciona la casilla de verificación.

El proyecto no se pueden eliminar si las políticas administradas de AWS CodeStar se han asociado manualmente a roles que no son usuarios de IAM. Si ha asociado las políticas administradas del proyecto a un rol del usuario federado, primero deberá eliminar el proyecto. Para obtener más información, consulte [???](#).

Paso 7: preparar el proyecto para un entorno de producción

Una vez creado el proyecto, ya estará preparado para crear, probar e implementar código. Revise las siguientes consideraciones para mantener su proyecto en un entorno de producción:

- Aplique parches con regularidad y revise las prácticas recomendadas de seguridad para las dependencias que utiliza su aplicación. Para obtener más información, consulte [Prácticas recomendadas de seguridad para recursos de AWS CodeStar](#).
- Monitorice con regularidad la configuración del entorno sugerida por el lenguaje de programación para su proyecto.

Pasos siguientes

A continuación se muestran otros recursos que le ayudarán a saber más acerca de AWS CodeStar:

- El [Tutorial: creación y administración de un proyecto sin servidor en AWS CodeStar](#) utiliza un proyecto que crea e implementa un servicio web mediante el uso de lógica en AWS Lambda y al que se puede llamar mediante una API en Amazon API Gateway.
- [Plantillas de proyecto de AWS CodeStar](#) describe otros tipos de proyectos que puede crear.

- [Trabajar con equipos de AWS CodeStar](#) proporciona más información acerca de cómo habilitar a otras personas para que le ayuden a trabajar en sus proyectos.

Tutorial: creación y administración de un proyecto sin servidor en AWS CodeStar

En este tutorial, utilizará AWS CodeStar para crear un proyecto que utiliza el Modelo de aplicación sin servidor de AWS (AWS SAM) para crear y administrar recursos de AWS para un servicio web alojado en AWS Lambda.

AWS CodeStar utiliza AWS SAM, que se basa en AWS CloudFormation, para ofrecer un modo sencillo de crear y administrar recursos de AWS admitidos, incluidas las API de Amazon API Gateway, las funciones de AWS Lambda y las tablas de Amazon DynamoDB. (Este proyecto no utiliza ninguna tabla de Amazon DynamoDB).

Para obtener más información, consulte el repositorio de [AWS Serverless Application Model \(AWS SAM\)](#) en GitHub.

Requisitos previos: Complete los pasos de [Configuración AWS CodeStar](#).

Note

Es posible que se apliquen cargos a su cuenta de AWS por los costos relacionados con este tutorial, incluidos los costos de los servicios de AWS utilizados por AWS CodeStar. Para obtener más información, consulte [Precios de AWS CodeStar](#).

Temas

- [Información general](#)
- [Paso 1: creación del proyecto](#)
- [Paso 2: explorar recursos del proyecto](#)
- [Paso 3: probar el servicio web](#)
- [Paso 4: configurar la estación de trabajo para editar código de proyecto](#)
- [Paso 5: añadir lógica al servicio web](#)
- [Paso 6: probar el servicio web mejorado](#)

- [Paso 7: añadir una prueba de unidad al servicio web](#)
- [Paso 8: ver los resultados de pruebas de la unidad](#)
- [Paso 9: Eliminación](#)
- [Pasos siguientes](#)

Información general

En este tutorial, va a:

1. Utilizar AWS CodeStar para crear un proyecto que usa AWS SAM para compilar e implementar un servicio web basado en Python. Este servicio web está alojado en AWS Lambda y se puede acceder al mismo a través de Amazon API Gateway.
2. Explorar los recursos principales del proyecto, que incluyen:
 - El repositorio de AWS CodeCommit donde está almacenado el código fuente del proyecto. Este código fuente incluye la lógica del servicio web y define recursos de AWS relacionados.
 - La canalización de AWS CodePipeline que automatiza la construcción del código fuente. Esta canalización utiliza AWS SAM para crear e implementar una función en AWS Lambda, crear una API relacionada en Amazon API Gateway y conectar la API a la función.
 - La función que se implementa en AWS Lambda.
 - La API que se crea en Amazon API Gateway.
3. Probar el servicio web para confirmar que AWS CodeStar compila e implementa el servicio web según lo previsto.
4. Configurar la estación de trabajo local para trabajar con el código fuente del proyecto.
5. Cambiar el código fuente del proyecto utilizando su estación de trabajo local. Al añadir una función al proyecto y, a continuación, enviar los cambios al código fuente, AWS CodeStar vuelve a compilar e implementar el servicio web.
6. Probar el servicio web de nuevo para confirmar que AWS CodeStar se ha vuelto a compilar e implementar según lo previsto.
7. Escribir una prueba de unidad utilizando su estación de trabajo local para sustituir algunas de las pruebas manuales con una prueba automatizada. Al enviar la prueba de unidad, AWS CodeStar vuelve a compilar e implementar el servicio web y ejecuta la prueba de unidad.
8. Consultar los resultados de las pruebas de unidad.
9. Eliminar el proyecto. Este paso le ayuda a evitar cargos en su cuenta de AWS para los costos relacionados con este tutorial.

Paso 1: creación del proyecto

En este paso, utilice la consola de AWS CodeStar para crear un proyecto.

1. Inicie sesión en la AWS Management Console y abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.

Note

Debe iniciar sesión en la AWS Management Console con las credenciales asociadas al usuario de IAM que haya creado o identificado en [Configuración AWS CodeStar](#). Este usuario debe tener la política administrada **AWSCodeStarFullAccess** asociada.

2. Elija la región de AWS donde desea crear el proyecto y sus recursos.

Para obtener más información sobre las regiones de AWS en las que AWS CodeStar está disponible, consulte [Regiones y puntos de conexión](#) en la Referencia general de AWS .

3. Elija Crear proyecto.
4. En la página Elegir una plantilla de proyecto:
 - En Tipo de aplicación, seleccione Servicio web.
 - En Lenguaje de programación, seleccione Python.
 - En Servicios de AWS, seleccione AWS Lambda.
5. Seleccione la casilla que contiene sus selecciones. Elija Siguiente.
6. En Nombre del proyecto, escriba un nombre para el proyecto (por ejemplo, **My SAM Project**). Si usa un nombre distinto al del ejemplo, asegúrese de utilizarlo en todo el tutorial.

En ID del proyecto, AWS CodeStar elija un identificador relacionado para este proyecto (por ejemplo, my-sam-project). Si ve un ID de proyecto diferente, asegúrese de utilizarlo durante todo el tutorial.

Deje AWS CodeCommit seleccionado y no cambie el valor de Nombre del repositorio.

7. Elija Siguiente.
8. Revise la configuración y, a continuación, seleccione Crear presupuesto.

Si es la primera vez que utiliza AWS CodeStar en esta región de AWS, en Mostrar nombre y Correo electrónico, escriba el nombre que mostrar y la dirección de correo electrónico que desee que utilice AWS CodeStar para su usuario de IAM. Elija Siguiente.

9. Espere mientras AWS CodeStar crea el proyecto. Esto podría tardar varios minutos. No continúe hasta que vea el banner Proyecto provisionado al actualizar.

Paso 2: explorar recursos del proyecto

En este paso, se exploran cuatro de los recursos de AWS del proyecto para comprender cómo funciona el proyecto:

- El repositorio de AWS CodeCommit donde está almacenado el código fuente del proyecto. AWS CodeStar asigna al repositorio el nombre `my-sam-project`, donde `my-sam-project` es el nombre del proyecto.
- La canalización de AWS CodePipeline que utiliza CodeBuild y AWS SAM para automatizar la compilación y la implementación de la función de Lambda y de la API del servicio web en API Gateway. AWS CodeStar asigna a la canalización el nombre `my-sam-project--Pipeline`, donde `my-sam-project` es el ID del proyecto.
- La función de Lambda que contiene la lógica del servicio web. AWS CodeStar asigna a la función el nombre `awscodestar-my-sam-project-lambda>HelloWorld-ID_ALEATORIO`, donde:
 - `my-sam-project` es el ID del proyecto.
 - `HelloWorld` es el ID de la función tal y como se especifica en el archivo `template.yaml` del repositorio de AWS CodeCommit. Puede explorar este archivo más adelante.
 - **ID_ALEATORIO** es un ID aleatorio que AWS SAM asigna a la función para ayudar a garantizar su exclusividad.
- La API en API Gateway que facilita las tareas para llamar a la función de Lambda. AWS CodeStar asigna a la API el nombre `awscodestar-my-sam-project--lambda`, donde `my-sam-project` es el ID del proyecto.

Para explorar el repositorio de código fuente en CodeCommit

1. Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación, seleccione Repositorio.

2. Seleccione el enlace a su repositorio de CodeCommit (**My-SAM-Project**) en Detalles del repositorio.
3. En la consola de CodeCommit, en la página Código, se muestran los archivos de código fuente del proyecto:
 - `buildspec.yml`, que CodePipeline indica a CodeBuild que utilice durante la fase de compilación para comprimir el servicio web con AWS SAM.
 - `index.py`, que contiene la lógica de la función de Lambda. Esta función simplemente genera la cadena `Hello World` y una marca de tiempo en formato ISO.
 - `README.md`, que contiene información general sobre el repositorio.
 - `template-configuration.json`, que contiene el ARN del proyecto con marcadores de posición utilizados para etiquetar recursos con el ID del proyecto
 - `template.yml`, que utiliza AWS SAM para comprimir el servicio web y crear la API en API Gateway.

The screenshot shows the AWS CodeCommit console interface. At the top, there is a navigation bar with the AWS logo, 'Services', 'Resource Groups', and a notification icon. Below this, a sidebar on the left contains a 'Developer Tools' menu with a close button (X) and a 'CodeCommit' section. Under 'CodeCommit', there are links for 'Getting started', 'Repositories', 'Code' (highlighted in orange), 'Pull requests', 'Commits', 'Branches', 'Tags', and 'Settings'. Below these are sections for 'Build • CodeBuild', 'Deploy • CodeDeploy', and 'Pipeline • CodePipeline'. The main content area on the right shows the breadcrumb 'Developer Tools > CodeCommit > Repositories > My-SAM-Project' and the title 'My-SAM-Project'. Below the title is a table listing the files in the repository:

Name
tests
buildspec.yml
index.py
README.md
template-configuration.json
template.yml

Para ver el contenido de un archivo, elíjalo en la lista.

Para obtener más información sobre cómo utilizar la consola de CodeCommit, consulte la [Guía del usuario de AWS CodeCommit](#).

Para explorar la canalización en CodePipeline

1. Para ver información acerca de la canalización, abra el proyecto en la consola de AWS CodeStar y, en la barra de navegación, seleccione Canalización; a continuación, verá que la canalización contiene:
 - Una etapa Origen para obtener el código fuente desde CodeCommit.
 - Una etapa de compilación para crear el código fuente con CodeBuild.
 - Una etapa Implementar para implementar el código fuente compilado y los recursos de AWS con AWS SAM.
2. Para ver más información sobre la canalización, en Detalles de la canalización, elija la canalización para abrirla en la consola de CodePipeline.

Para obtener más información sobre la consola de CodePipeline, consulte la [Guía del usuario de AWS CodePipeline](#).

Para explorar la actividad del proyecto y los recursos del servicio de AWS en la página Información general

1. Abra el proyecto en la consola de AWS CodeStar. En la barra de navegación, seleccione Información general.
2. Revise las listas Actividad del proyecto y Recursos del proyecto.

Para explorar la función en Lambda

1. Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación lateral, seleccione Información general.
2. En Recursos del proyecto, seleccione el enlace en la columna ARN para la función de Lambda.

El código de la función se muestra en la consola de Lambda.

Para obtener más información acerca de la consola de Lambda, consulte la [Guía para desarrolladores de AWS Lambda](#).

Para explorar la API en API Gateway

1. Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación lateral, seleccione Información general.
2. En Recursos del proyecto, seleccione el enlace en la columna ARN para la API de Amazon API Gateway.

Los recursos de la API se muestran en la consola de API Gateway.

Para obtener más información sobre la consola de API Gateway, consulte la [Guía para desarrolladores de API Gateway](#).

Paso 3: probar el servicio web

En este paso, probará el servicio web que AWS CodeStar acaba de compilar e implementar.

1. Con el proyecto abierto en el paso anterior, en la barra de navegación, seleccione Canalización.
2. Asegúrese de que se muestre el estado Correcto en las etapas Fuente, Compilación e Implementación antes de continuar. Esto podría tardar varios minutos.

Note

Si se muestra Error en alguna de las etapas, consulte la siguiente ayuda para la solución de problemas:

- Para la etapa Origen, consulte [Solución de problemas de AWS CodeCommit](#) en la Guía del usuario de AWS CodeCommit.
- Para la etapa de compilación, consulte [Solución de problemas de AWS CodeBuild](#) en la Guía del usuario de AWS CodeBuild.
- Para la etapa de implementación, consulte [Solución de problemas de AWS CloudFormation](#) en la Guía del usuario de AWS CloudFormation.

- Para los demás problemas, consulte [Solución de problemas AWS CodeStar](#).

3. Seleccione Ver aplicación.

En la pestaña nueva que se abre en el navegador web, el servicio web muestra la siguiente salida de respuesta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

Paso 4: configurar la estación de trabajo para editar código de proyecto

En este paso, configurará la estación de trabajo local para editar el código fuente en el proyecto de AWS CodeStar. Su estación de trabajo local puede ser un equipo físico o virtual que se ejecuta en macOS, Windows o Linux.

1. Con su proyecto aún abierto del paso anterior:

- En la barra de navegación, seleccione IDE y, a continuación, expanda Acceder al código del proyecto.
- Seleccione Ver instrucciones debajo de la Interfaz de la línea de comandos.

Si tiene instalado Visual Studio o Eclipse, seleccione Ver instrucciones debajo de Visual Studio o Eclipse en su lugar, siga las instrucciones y, a continuación, pase a [Paso 5: añadir lógica al servicio web](#).

2. Siga las instrucciones para completar las siguientes tareas:

- a. Configure Git en su estación de trabajo.
- b. Utilice la consola de IAM para generar credenciales de Git para su usuario de IAM.
- c. Clone el repositorio de CodeCommit del proyecto en su estación de trabajo local.

3. En el panel de navegación izquierdo, seleccione Proyecto para volver a la información general del proyecto.

Paso 5: añadir lógica al servicio web

En este paso, utilice su estación de trabajo local para añadir lógica al servicio web. En concreto, añada una función de Lambda y, a continuación, conéctela a la API en API Gateway.

1. En su estación de trabajo local, vaya al directorio que contiene el repositorio del código fuente clonado.
2. En dicho directorio, cree un archivo llamado `hello.py`. Añada el siguiente código y luego guarde el archivo:

```
import json

def handler(event, context):
    data = {
        'output': 'Hello ' + event["pathParameters"]["name"]
    }
    return {
        'statusCode': 200,
        'body': json.dumps(data),
        'headers': {'Content-Type': 'application/json'}
    }
```

El código anterior simplemente genera la cadena Hello junto la cadena que envía el intermediario a la función.

3. En el mismo directorio, abra el archivo `template.yml`. Añada el siguiente código al final del archivo y, a continuación, guárdelo:

```
Hello:
  Type: AWS::Serverless::Function
  Properties:
    FunctionName: !Sub 'awscodestar-${ProjectId}-lambda-Hello'
    Handler: hello.handler
    Runtime: python3.7
    Role:
      Fn::GetAtt:
        - LambdaExecutionRole
        - Arn
    Events:
      GetEvent:
        Type: Api
        Properties:
```

```
Path: /hello/{name}
Method: get
```

AWS SAM utiliza este código para crear una función en Lambda, añadir un nuevo método y ruta a la API en API Gateway y, a continuación, conectar este método y ruta a la nueva función.

Note

La sangría del código anterior es importante. Si no añade código exactamente como se muestra, es posible que el proyecto no se cree correctamente.

4. Ejecute `git add .` para añadir cambios en el archivo en el área provisional del repositorio clonado. No olvide el punto (`.`), que añade todos los archivos modificados.

Note

Si utiliza Visual Studio o Eclipse en lugar de la línea de comando, las instrucciones para el uso de Git podrían ser diferentes. Consulte la documentación de Eclipse o Visual Studio.

5. Ejecute `git commit -m "Added hello.py and updated template.yaml."` para confirmar sus archivos provisionales en el repositorio clonado
6. Ejecute `git push` para enviar la confirmación al repositorio remoto.

Note

Es posible que se le pidan las credenciales de inicio de sesión que se generaron anteriormente. Para evitar que se le pida cada vez que interactúe con el repositorio remoto, considere la posibilidad de instalar y configurar un administrador de credenciales de Git. Por ejemplo, en macOS o Linux, puede ejecutar `git config credential.helper 'cache --timeout 900'` en el terminal para que no las solicite antes de transcurridos 15 minutos. También puede ejecutar `git config credential.helper 'store --file ~/.git-credentials'` para que nunca se las pida de nuevo. Git almacena sus credenciales en texto sin formato en un archivo de su directorio de inicio. Para obtener más información, consulte [Git Tools - Credential Storage](#) en el sitio web de Git.

Una vez que AWS CodeStar detecta el envío, ordena a CodePipeline que utilice CodeBuild y AWS SAM para volver a compilar e implementar el servicio web. Puede ver el progreso de la implementación en la página Canalización.

AWS SAM asigna a la nueva función el nombre `awscodestar-my-sam-project-lambda-Hello-ID_ALEATORIO`, donde:

- `my-sam-project` es el ID del proyecto.
- `Hello` es el ID de la función tal como se especifica en el archivo `template.yaml`.
- `ID_ALEATORIO` es un ID aleatorio que AWS SAM asigna a la función para que sea exclusiva.

Paso 6: probar el servicio web mejorado

En este paso, pruebe el servicio web mejorado que AWS CodeStar ha creado e implementado, en función de la lógica que añadió en el paso anterior.

1. Con el proyecto todavía abierto en la consola de AWS CodeStar, en la barra de navegación, seleccione Canalización.
2. Asegúrese de que la canalización se haya vuelto a ejecutar y que se muestre el estado Correcto en las etapas Fuente, Compilación e Implementación antes de continuar. Esto podría tardar varios minutos.

Note

Si se muestra Failed (Error) en alguna de las etapas, consulte la siguiente ayuda para la solución de problemas:

- Para la etapa Origen, consulte [Solución de problemas de AWS CodeCommit](#) en la Guía del usuario de AWS CodeCommit.
- Para la etapa de compilación, consulte [Solución de problemas de AWS CodeBuild](#) en la Guía del usuario de AWS CodeBuild.
- Para la etapa de implementación, consulte [Solución de problemas de AWS CloudFormation](#) en la Guía del usuario de AWS CloudFormation.

- Para los demás problemas, consulte [Solución de problemas AWS CodeStar](#).

3. Seleccione Ver aplicación.

En la pestaña nueva que se abre en el navegador web, el servicio web muestra la siguiente salida de respuesta:

```
{"output": "Hello World", "timestamp": "2017-08-30T15:53:42.682839"}
```

4. En el cuadro de dirección de la pestaña, añada la ruta **/hello/** y su nombre al final de la URL (por ejemplo, https://ID_API.execute-api.ID_REGIÓN.amazonaws.com/Prod/hello/SU_NOMBRE) y después pulse Intro.

Si su nombre es Mary, el servicio web de salida muestra la siguiente salida de respuesta:

```
{"output": "Hello Mary"}
```

Paso 7: añadir una prueba de unidad al servicio web

En este paso, utilice la estación de trabajo local para añadir una prueba que AWS CodeStar ejecutará en el servicio web. Esta prueba sustituye las pruebas manuales que realizó antes.

1. En su estación de trabajo local, vaya al directorio que contiene el repositorio del código fuente clonado.
2. En dicho directorio, cree un archivo llamado `hello_test.py`. Añada el siguiente código y luego guarde el archivo.

```
from hello import handler

def test_hello_handler():

    event = {
        'pathParameters': {
            'name': 'testname'
        }
    }

    context = {}
```

```
expected = {
  'body': '{"output": "Hello testname"}',
  'headers': {
    'Content-Type': 'application/json'
  },
  'statusCode': 200
}

assert handler(event, context) == expected
```

Esta prueba comprueba si la salida de la función de Lambda está en el formato previsto. En caso afirmativo, la prueba se ejecuta satisfactoriamente. De lo contrario, la prueba falla.

3. En el mismo directorio, abra el archivo `buildspec.yml`. Sustituya el contenido del archivo por el siguiente código y, a continuación, guárdelo.

```
version: 0.2

phases:
  install:
    runtime-versions:
      python: 3.7

    commands:
      - pip install pytest
      # Upgrade AWS CLI to the latest version
      - pip install --upgrade awscli

  pre_build:
    commands:
      - pytest

  build:
    commands:
      # Use AWS SAM to package the application by using AWS CloudFormation
      - aws cloudformation package --template template.yml --s3-bucket
      $S3_BUCKET --output-template template-export.yml

      # Do not remove this statement. This command is required for AWS CodeStar
      projects.
```

```
# Update the AWS Partition, AWS Region, account ID and project ID in the
project ARN on template-configuration.json file so AWS CloudFormation can tag
project resources.
- sed -i.bak 's/\${PARTITION}\$/'\${PARTITION}']/g;s/\${AWS_REGION}
\$/'\${AWS_REGION}']/g;s/\${ACCOUNT_ID}\$/'\${ACCOUNT_ID}']/g;s/\${PROJECT_ID}\
$/'\${PROJECT_ID}']/g' template-configuration.json

artifacts:
  type: zip
  files:
    - template-export.yml
    - template-configuration.json
```

Esta especificación de compilación indica a CodeBuild que instale pytest, el marco de pruebas de Python, en su entorno. CodeBuild utiliza pytest para ejecutar la prueba unitaria. El resto de la especificación de compilación es la misma que antes.

4. Utilice Git para introducir estos cambios en el repositorio remoto.

```
git add .

git commit -m "Added hello_test.py and updated buildspec.yml."

git push
```

Paso 8: ver los resultados de pruebas de la unidad

En este paso, verá si la prueba de unidad se ha realizado con éxito o ha fallado.

1. Con el proyecto todavía abierto en la consola de AWS CodeStar, en la barra de navegación, seleccione Canalización.
2. Asegúrese de que la canalización se haya vuelto a ejecutar antes de continuar. Esto podría tardar varios minutos.

Si la prueba de unidad se ha realizado correctamente, se muestra Correcto en la etapa Compilar.

3. Para ver los detalles del resultado de la prueba unitaria, en la etapa de compilación, seleccione el enlace de CodeBuild.

4. En la consola de CodeBuild, en la página Proyecto de compilación: my-sam-project, en Historial de compilaciones, seleccione el enlace en la columna Ejecución de la compilación de la tabla.
5. En la página my-sam-project:**ID_COMPILACIÓN**, en Registros de compilación, elija el enlace Ver el registro completo.
6. En la consola de Registros de Amazon CloudWatch, consulte la salida del registro para ver un resultado de prueba similar al siguiente. En el siguiente resultado de prueba, la prueba se ha superado:

```
...
===== test session starts =====
platform linux2 -- Python 2.7.12, pytest-3.2.1, py-1.4.34, pluggy-0.4.0
rootdir: /codebuild/output/src123456789/src, inifile:
collected 1 item

hello_test.py .

===== 1 passed in 0.01 seconds =====
...
```

Si la prueba no se ha superado, debería haber detalles en la salida de registro para ayudarle a solucionar el error.

Paso 9: Eliminación

En este paso, elimine el proyecto para evitar cargos continuos relacionados con este proyecto.

Si desea seguir utilizando este proyecto, puede omitir este paso, pero es posible que se aplique un cargo en su cuenta de AWS.

1. Con el proyecto todavía abierto en la consola de AWS CodeStar, en la barra de navegación, seleccione Configuración.
2. En Detalles del proyecto, seleccione Eliminar proyecto.
3. Escriba **delete**, marque la casilla Eliminar recursos y, a continuación, seleccione Eliminar.

⚠ Important

Si desmarca esta casilla, el registro del proyecto se eliminará de AWS CodeStar, pero se conservarán numerosos recursos de AWS del proyecto. Es posible que se aplique un cargo en su cuenta de AWS.

Si todavía hay un bucket de Amazon S3 que AWS CodeStar ha creado para este proyecto, siga los pasos que se indican a continuación para eliminarlo:

1. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3/>.
2. En la lista de buckets, seleccione el icono junto a `aws-codestar-ID_REGIÓN-ID_CUENTA-my-sam-project--pipe`, donde:
 - **ID_REGIÓN** es el ID de la región de AWS del proyecto que acaba de eliminar.
 - **ID_CUENTA** es el ID de la cuenta de AWS.
 - `my-sam-project` es el ID del proyecto que acaba de eliminar.
3. Elija Vaciar bucket. Escriba el nombre del bucket y después elija Confirmar.
4. Seleccione Eliminar bucket. Escriba el nombre del bucket y después elija Confirmar.

Pasos siguientes

Ahora que ha completado este tutorial, le recomendamos que revise los siguientes recursos:

- El tutorial [Introducción a AWS CodeStar](#) utiliza un proyecto que crea e implementa una aplicación web basada en Node.js que se ejecuta en una instancia de Amazon EC2.
- [Plantillas de proyecto de AWS CodeStar](#) describe otros tipos de proyectos que puede crear.
- [Trabajar con equipos de AWS CodeStar](#) muestra cómo otros pueden ayudarle a trabajar en sus proyectos.

Tutorial: Crear un proyecto en AWS CodeStar con la AWS CLI

En este tutorial se muestra cómo utilizar la AWS CLI para crear un proyecto de AWS CodeStar con código fuente de muestra y una plantilla de cadena de herramientas de muestra. AWS CodeStar aprovisiona la infraestructura de AWS y recursos de IAM especificados en una plantilla de cadena

de herramientas de AWS CloudFormation. El proyecto administra los recursos de cadena de herramientas para compilar e implementar el código fuente.

AWS CodeStar utiliza AWS CloudFormation para compilar e implementar el código de muestra. Este código de muestra crea un servicio web que está alojado en AWS Lambda y al que se puede acceder desde Amazon API Gateway.

Requisitos previos:

- Realice los pasos que se indican en [Configuración AWS CodeStar](#).
- Tiene que haber creado un bucket de almacenamiento de Amazon S3. En este tutorial, debe cargar el código fuente de muestra y la plantilla de la cadena de herramientas en esta ubicación.

Note

Es posible que se apliquen cargos a su cuenta de AWS por los costos relacionados con este tutorial, incluidos los costos de los servicios de AWS utilizados por AWS CodeStar. Para obtener más información, consulte [Precios de AWS CodeStar](#).

Temas

- [Paso 1: Descargar y revisar el código fuente de muestra](#)
- [Paso 2: Descargar la plantilla de la cadena de herramientas de muestra](#)
- [Paso 3: Comprobar la plantilla de la cadena de herramientas en AWS CloudFormation](#)
- [Paso 4: Cargar el código fuente y la plantilla de la cadena de herramientas](#)
- [Paso 5: Crear un proyecto en AWS CodeStar](#)

Paso 1: Descargar y revisar el código fuente de muestra

Hay un archivo .zip disponible para su descarga para este tutorial. Contiene código fuente de muestra para una [aplicación de muestra](#) de Node.js en la plataforma de computación Lambda. Cuando el código fuente se coloca en el repositorio, la carpeta y los archivos aparecen tal como se muestra a continuación:

```
tests/  
app.js  
buildspec.yml
```

```
index.js
package.json
README.md
template.yml
```

Los siguientes elementos del proyecto están representados en su código fuente de muestra:

- `tests/`: pruebas unitarias configuradas para este proyecto de CodeBuild del proyecto. Esta carpeta se incluye en el código de muestra, pero no es necesaria para crear un proyecto.
- `app.js`: código fuente de la aplicación para el proyecto.
- `buildspec.yml`: instrucciones de compilación de la etapa de compilación del recurso de CodeBuild. Este archivo es necesario para una plantilla de cadena de herramientas con un recurso de CodeBuild.
- `package.json`: información sobre las dependencias para el código fuente de la aplicación.
- `README.md`: archivo readme del proyecto incluido en todos los proyectos de AWS CodeStar. Este archivo se incluye en el código de muestra, pero no es necesario para crear un proyecto.
- `template.yml`: archivo de plantilla de la infraestructura o archivo de plantilla SAM incluido en todos los proyectos de AWS CodeStar. Esto es diferente de la plantilla de la cadena de herramientas `.yml` que cargará más adelante en este tutorial. Este archivo se incluye en el código de muestra, pero no es necesario para crear un proyecto.

Paso 2: Descargar la plantilla de la cadena de herramientas de muestra

La plantilla de la cadena de herramientas de muestra incluida para este tutorial crea un repositorio (CodeCommit), una canalización (CodePipeline) y un contenedor de compilación (CodeBuild) y utiliza AWS CloudFormation para implementar el código fuente en una plataforma de Lambda. Además de estos recursos, también hay roles de IAM que puede utilizar para definir el ámbito de los permisos de su entorno de tiempo de ejecución, un bucket de Amazon S3 que CodePipeline utiliza para almacenar los artefactos de implementación y una regla de CloudWatch Events que se utiliza para desencadenar las implementaciones de canalización al enviar un código al repositorio. Para seguir [las prácticas recomendadas de AWS IAM](#), reduzca el ámbito de las políticas de sus roles de la cadena de herramientas definidos en este ejemplo.

Descargue y descomprima la plantilla de AWS CloudFormation de muestra en formato [YAML](#).

Al ejecutar el comando `create-project` más adelante en el tutorial, esta plantilla crea los siguientes recursos de la cadena de herramientas personalizadas de AWS CloudFormation. Para obtener más


información acerca de los recursos creados en este tutorial, consulte los siguientes temas de la Guía del usuario de AWS CloudFormation:

- El recurso [AWS::CodeCommit::Repository](#) de AWS CloudFormation crea un repositorio de CodeCommit.
- El recurso [AWS::CodeBuild::Project](#) de AWS CloudFormation crea un proyecto de compilación de CodeBuild.
- El recurso [AWS::CodeDeploy::Application](#) de AWS CloudFormation crea una aplicación de CodeDeploy.
- El recurso [AWS::CodePipeline::Pipeline](#) de AWS CloudFormation crea una canalización de CodePipeline.
- El recurso [AWS::S3::Bucket](#) de AWS CloudFormation crea el bucket de artefactos de su canalización.
- El recurso [AWS::S3::BucketPolicy](#) de AWS CloudFormation crea la política del bucket del artefacto para el bucket del artefacto de la canalización.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de trabajador de IAM de CodeBuild que otorga permisos a AWS CodeStar para administrar su proyecto de compilación de CodeBuild.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de trabajador de IAM de CodePipeline que otorga permisos a AWS CodeStar para crear la canalización.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de trabajador de IAM de AWS CloudFormation que otorga permisos a AWS CodeStar para crear la pila de recursos.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de trabajador de IAM de AWS CloudFormation que otorga permisos a AWS CodeStar para crear la pila de recursos.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de trabajador de IAM de AWS CloudFormation que otorga permisos a AWS CodeStar para crear la pila de recursos.
- El recurso [AWS::Events::Rule](#) de AWS CloudFormation crea el rol de CloudWatch Events que monitoriza el repositorio para enviar eventos.
- El recurso [AWS::IAM::Role](#) de AWS CloudFormation crea el rol de IAM de CloudWatch Events.

Paso 3: Comprobar la plantilla de la cadena de herramientas en AWS CloudFormation

Antes de cargar la plantilla de la cadena de herramientas, puede probar la plantilla de la cadena de herramientas en AWS CloudFormation y solucionar los errores.

1. Guarde la plantilla actualizada en el equipo local y abra la consola de AWS CloudFormation. Elija Create Stack. Debería ver los nuevos recursos en la lista.
2. Revise la pila para ver si se han producido errores al crearla.
3. Tras finalizar la prueba, elimine la pila.

 Note

Asegúrese de eliminar la pila y todos los recursos creados en AWS CloudFormation. De lo contrario, al crear el proyecto, se podrían producir errores con los nombres de recursos ya en uso.

Paso 4: Cargar el código fuente y la plantilla de la cadena de herramientas

Para crear un proyecto de AWS CodeStar, primero debe empaquetar el código fuente en un archivo .zip y colocarlo en Amazon S3. AWS CodeStar inicializa el repositorio con este contenido. Especifique esta ubicación en su archivo de entrada al ejecutar el comando para crear su proyecto en la AWS CLI.

Asimismo, debe cargar su archivo `toolchain.yml` y colocarlo en Amazon S3. Especifique esta ubicación en su archivo de entrada al ejecutar el comando para crear su proyecto en la AWS CLI.

Para cargar el código fuente y la plantilla de la cadena de herramientas

1. La siguiente estructura de archivos de ejemplo muestra los archivos de origen y la plantilla de la cadena de herramientas listos para ser comprimido y cargado. El código de muestra incluye el archivo `template.yml`. Recuerde que este archivo es diferente del archivo `toolchain.yml`.

```
ls
src toolchain.yml

ls src/
README.md    app.js        buildspec.yml  index.js      package.json
template.yml  tests
```

2. Cree el archivo .zip para los archivos de código fuente.

```
cd src; zip -r "../src.zip" *; cd ../
```

3. Utilice el comando `cp` e incluya los archivos como parámetros.

Los siguientes comandos cargan el archivo `.zip` y `toolchain.yml` en Amazon S3.

```
aws s3 cp src.zip s3://MyBucket/src.zip
aws s3 cp toolchain.yml s3://MyBucket/toolchain.yml
```

Configuración del bucket de Amazon S3 para compartir el código fuente

- Dado que va a almacenar el código fuente y la cadena de herramientas en Amazon S3, puede utilizar las políticas de bucket de Amazon S3 y los ACL de objetos para asegurarse de que otros usuarios de IAM o cuentas de AWS puedan crear proyectos a partir de las muestras. AWS CodeStar garantiza que cualquier usuario que cree un proyecto personalizado tendrá acceso a la cadena de herramientas y al origen que desee utilizar.

Para permitir que cualquier persona utilice la muestra, ejecute los siguientes comandos:

```
aws s3api put-object-acl --bucket MyBucket --key toolchain.yml --acl public-read
aws s3api put-object-acl --bucket MyBucket --key src.zip --acl public-read
```

Paso 5: Crear un proyecto en AWS CodeStar

Siga estos pasos para crear su proyecto.

Important

Asegúrese de configurar la región preferida de AWS en la AWS CLI. El proyecto se crea en la región de AWS configurada en la AWS CLI.

1. Ejecute el comando `create-project` e incluya el parámetro `--generate-cli-skeleton`:

```
aws codestar create-project --generate-cli-skeleton
```

En el resultado se muestran datos con formato JSON. Copie los datos en un archivo (por ejemplo, `input.json`) en la ubicación del equipo o instancia local en la que haya instalado la AWS CLI. Modifique los datos copiados como se indica a continuación y guarde los resultados.

Este archivo de entrada está configurado para un proyecto llamado MyProject con el nombre de bucket myBucket.

- Asegúrese de proporcionar el parámetro `roleArn`. Para las plantillas personalizadas, como la plantilla de ejemplo de este tutorial, debe proporcionar un rol. Este rol debe tener permisos para crear todos los recursos especificados en [Paso 2: Descargar la plantilla de la cadena de herramientas de muestra](#).
- Asegúrese de indicar el parámetro `ProjectId` bajo `stackParameters`. La plantilla de muestra que se proporciona para este tutorial requiere dicho parámetro.

```
{
  "name": "MyProject",
  "id": "myproject",
  "description": "Sample project created with the CLI",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "MyBucket",
          "bucketKey": "src.zip"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "myproject"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
        "bucketName": "MyBucket",
        "bucketKey": "toolchain.yml"
      }
    }
  },
  "roleArn": "role_ARN",
  "stackParameters": {
    "ProjectId": "myproject"
  }
}
```

```
}  
}
```

2. Cambie al directorio que contiene el archivo que acaba de guardar y ejecute de nuevo el comando `create-project`. Incluya el parámetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Si el comando se ejecuta correctamente, aparecerán datos similares a los siguientes en el resultado:

```
{  
  "id": "project-ID",  
  "arn": "arn"  
}
```

- El resultado contiene información acerca del nuevo proyecto:
 - El valor `id` representa el ID del proyecto.
 - El valor `arn` representa el ARN del proyecto.
- 4. Para comprobar el estado de creación del proyecto, utilice el comando `describe-project`. Incluya el parámetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

En el resultado se muestra información similar a la siguiente:

```
{  
  "name": "MyProject",  
  "id": "myproject",  
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",  
  "description": "",  
  "createdTimeStamp": 1539700079.472,  
  "stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-  
myproject/stack-ID",  
  "status": {  
    "state": "CreateInProgress"  
  }  
}
```


- El resultado contiene información acerca del nuevo proyecto:
 - El valor `id` representa el ID único del proyecto.
 - El valor `state` representa el estado de la creación del proyecto, como, por ejemplo, `CreateInProgress` o `CreateComplete`.

Durante la creación del proyecto, puede [añadir miembros al equipo](#) o [configurar el acceso](#) al repositorio de su proyecto desde la línea de comandos o su IDE favorito.

Tutorial: crear un proyecto de una habilidad de Alexa en AWS CodeStar

AWS CodeStar es un servicio de desarrollo de AWS basado en la nube que proporciona las herramientas necesarias para desarrollar, compilar e implementar aplicaciones rápidamente en AWS. Con AWS CodeStar puede configurar toda su cadena de herramientas de entrega continua en minutos, lo que le permite empezar a publicar su código más rápidamente. Las plantillas de proyectos de habilidades de Alexa disponibles en AWS CodeStar le permiten crear una sencilla habilidad de Alexa “Hello World” desde su cuenta de AWS con solo unos clics. Con las plantillas también se crea una canalización de implementación básica que permite comenzar con un flujo de trabajo de integración continua (CI) para desarrollar habilidades.

Uno de los principales beneficios de crear habilidades de Alexa en AWS CodeStar es que puede comenzar a desarrollar habilidades en AWS y conectar su cuenta de desarrollador de Amazon al proyecto para implementar dichas habilidades en la fase de desarrollo directamente desde AWS. El otro es que se incluye una canalización de implementación (CI) con un repositorio que contiene todo el código fuente para el proyecto. Puede configurar este repositorio con el IDE que prefiera para crear habilidades con herramientas que ya conoce.

Requisitos previos

- Cree una cuenta de desarrollador de Amazon en <https://developer.amazon.com>. El registro es gratuito. La cuenta tiene sus habilidades de Alexa.
- Si no dispone de una cuenta de AWS, utilice el siguiente procedimiento para crearla.

Para inscribirse en AWS

1. Abra <https://aws.amazon.com/> y, a continuación, elija Crear una cuenta de AWS.

Note

Si ha iniciado previamente sesión en la AWS Management Console con las credenciales de Usuario raíz de la cuenta de AWS, elija Sign in to a different account (Iniciar sesión en una cuenta distinta). Si ha iniciado previamente sesión en la consola con las credenciales de IAM, seleccione Iniciar sesión con las credenciales de Usuario raíz de la cuenta de AWS. A continuación, seleccione Crear una nueva cuenta de AWS.

2. Siga las instrucciones en línea.

Important

Después de crear el proyecto de habilidad de Alexa, haga todos los cambios solo en el repositorio del proyecto. Le recomendamos no editar la habilidad directamente con cualquier otro kit de herramientas de habilidades de Alexa, como la CLI o la consola para desarrolladores de ASK. Estas herramientas no se integran con el repositorio del proyecto. Si las utiliza, el código de habilidades y de repositorio se desincronizará.

Paso 1: crear el proyecto y conectar su cuenta de desarrollador de Amazon

En este tutorial, creará una habilidad con Node.js que se ejecuta en AWS Lambda. La mayoría de los pasos son los mismos para otros lenguajes, aunque el nombre de la habilidad sea distinto. Consulte los detalles de la plantilla de proyecto específica que elija en el archivo README.md del repositorio del proyecto.

1. Inicie sesión en la AWS Management Console y, a continuación, abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Elija la región de AWS donde desea crear el proyecto y sus recursos. El tiempo de ejecución de la habilidad de Alexa está disponible solo en las siguientes regiones de AWS:
 - Asia-Pacífico (Tokio)
 - UE (Irlanda)
 - Este de EE. UU. (Norte de Virginia)
 - Oeste de EE. UU. (Oregón)

3. Elija Crear proyecto.
4. En la página Elegir una plantilla de proyecto:
 - a. En Categoría de aplicación, elija Habilidad de Alexa.
 - b. En Lenguajes de programación, elija Node.js.
5. Seleccione la casilla que contenga sus selecciones.
6. En Nombre del proyecto, escriba un nombre para el proyecto (por ejemplo, **My Alexa Skill**). Si usa un nombre distinto, asegúrese de que sea el mismo durante la realización de todo el tutorial. AWS CodeStar inserta un valor en ID de proyecto relacionado con el nombre (en este caso, sería, my-alexa-skill, por ejemplo). Si ve un ID de proyecto diferente, asegúrese de utilizarlo durante todo el tutorial.
7. En este tutorial, elija AWS CodeCommit para el repositorio y no cambie el valor de Nombre del repositorio.
8. Elija Conectar la cuenta de desarrollador de Amazon para vincular su cuenta y alojar la habilidad. Si no tiene una cuenta de desarrollador de Amazon, cree una cuenta y complete el registro primero desde [Amazon Developers](#).
9. Inicie sesión con sus credenciales de desarrollador de Amazon. Seleccione Permitir y, a continuación, seleccione Confirmar para completar la conexión.
10. Si tiene varios ID de proveedor asociados a su cuenta de desarrollador de Amazon, elija el que desee usar en este proyecto. Asegúrese de utilizar una cuenta que tenga asignada el rol de administrador o desarrollador.
11. Elija Siguiente.
12. (Opcional) Si es la primera vez que utiliza AWS CodeStar en esta región de AWS, escriba el nombre de visualización y la dirección de correo electrónico que desee que AWS CodeStar utilice para su usuario de IAM. Elija Siguiente.
13. Espere mientras AWS CodeStar crea el proyecto. Esto podría tardar varios minutos. No continúe hasta que vea el banner Proyecto aprovisionado.

Paso 2: probar la habilidad en el simulador de Alexa

En el primer paso, AWS CodeStar creó una habilidad por usted y la implementó en la fase de desarrollo de la habilidad de Alexa. Ahora va a probar dicha habilidad en el simulador de Alexa.

1. En el proyecto de la consola de AWS CodeStar, seleccione Ver aplicación. Esto abre una pestaña nueva en el simulador de Alexa.

2. Inicie sesión con las credenciales de desarrollador de Amazon de la cuenta que conectó a su proyecto en el paso 1.
3. En Test (Prueba), elija Development (Desarrollo) para habilitar la prueba.
4. Escriba `ask hello node hello`. El nombre de invocación predeterminado de su habilidad es `hello node`.
5. Su habilidad debería responder `Hello World!`.

Cuando la habilidad está activada en el simulador de Alexa, también puede invocarla en cualquier dispositivo con Alexa activado que esté registrado en su cuenta de desarrollador de Amazon. Para probar la habilidad en un dispositivo, diga Alexa, dile a "hello node" que salude.

Para obtener más información acerca del simulador de Alexa, consulte [Test Your Skill in the Developer Console](#).

Paso 3: explorar los recursos del proyecto

Como parte de la creación del proyecto, AWS CodeStar también creó recursos de AWS en su nombre. Entre estos recursos se incluye un repositorio del proyecto mediante CodeCommit, una canalización de implementación mediante CodePipeline y una función de AWS Lambda. Puede acceder a estos recursos desde la barra de navegación. Por ejemplo, si selecciona Repositorio, se muestran detalles sobre el repositorio de CodeCommit. Puede ver el estado de implementación de la canalización en la página Canalización. Puede ver una lista completa de los recursos de AWS que se han creado como parte de su proyecto; para ello, seleccione Información general en la barra de navegación. En la lista se incluyen enlaces a cada recurso.

Paso 4: haga un cambio a la respuesta de la habilidad

En este paso, hará un pequeño cambio en la respuesta de la habilidad para comprender el ciclo de iteración.

1. En el panel de navegación, seleccione Repositorio. Seleccione el enlace que aparece debajo de Nombre del repositorio y el repositorio del proyecto se abrirá en una nueva pestaña o ventana. Este repositorio contiene la especificación de la compilación (`buildspec.yml`), la pila de la aplicación de AWS CloudFormation (`template.yml`), el archivo `readme` y el código fuente de la habilidad en el [formato de paquete de habilidades \(estructura del proyecto\)](#).
2. Vaya al archivo `lambda > personalizado > index.js` (en el caso de Node.js.). Este archivo contiene el código de gestión de solicitudes, que utiliza el [SDK de ASK](#).

3. Elija Editar.
4. Sustituya la cadena `Hello World!` de la línea 24 por la cadena `Hello. How are you?`.
5. Desplácese hasta el final del archivo Escriba el nombre del autor y la dirección de correo electrónico, así como un mensaje de confirmación opcional.
6. Elija Confirmar cambios para confirmar los cambios realizados al repositorio.
7. Regrese al proyecto en AWS CodeStar y consulte la página Canalización. Debería ver la canalización implementándose.
8. Cuando la canalización termine de implementarse, pruebe la habilidad de nuevo en el simulador de Alexa. La habilidad debería responder `Hello. How are you?`.

Paso 5: configuración de la estación de trabajo local para conectarla al repositorio del proyecto

Anteriormente, realizó un pequeño cambio en el código fuente directamente en la consola de CodeCommit. En este paso, configurará el repositorio del proyecto desde la estación de trabajo local para poder editar y administrar el código desde la línea de comandos o el IDE de su preferencia. En los siguientes pasos, se explica cómo configurar las herramientas de línea de comandos.

1. Desplácese hasta el panel del proyecto en AWS CodeStar, de ser necesario.
2. En la barra de navegación, seleccione IDE.
3. En Acceder al código del proyecto, seleccione Ver instrucciones en la Interfaz de la línea de comandos.
4. Siga las instrucciones para completar las siguientes tareas:
 - a. Instale Git en la estación de trabajo local desde un sitio web como [Git Downloads](#).
 - b. Instale la CLI de AWS. Para obtener más información, consulte [Instalar la interfaz de la línea de comandos de AWS](#).
 - c. Configure la AWS CLI con la clave de acceso de su usuario de IAM y su clave secreta. Para obtener información, consulte [Configuración de la AWS CLI](#).
 - d. Clone el repositorio de CodeCommit del proyecto en su estación de trabajo local. Para obtener más información, consulte [Conectarse a un repositorio de CodeCommit](#).

Pasos siguientes

Con este tutorial ha aprendido a crear una habilidad sencilla. Para adquirir más práctica desarrollando habilidades, consulte los recursos siguientes.

- En el vídeo [How Alexa Skills Work](#), entre otros vídeos del canal Alexa Developers de YouTube, se explican en mayor detalle los aspectos fundamentales de las habilidades.
- Para conocer mejor las partes de su habilidad, puede consultar el [formato de paquete de la habilidad](#), los [esquemas del manifiesto de la habilidad](#) y los [esquemas del modelo de interacción](#).
- Convierta su idea en una habilidad: lea la documentación del [kit de habilidades de Alexa](#) y de los [SDK de ASK](#).

Tutorial: creación de un proyecto con un repositorio de código fuente de GitHub

Con AWS CodeStar, puede configurar su repositorio para crear, revisar y fusionar solicitudes de extracción con el equipo del proyecto.

En este tutorial, se explica cómo crear un proyecto con un ejemplo de código fuente de una aplicación web en un repositorio de GitHub, con una canalización que despliega los cambios y con instancias de EC2 en las que la aplicación está alojada en la nube. Una vez creado el proyecto, en este tutorial se muestra cómo crear y fusionar una solicitud de extracción de GitHub que realiza un cambio en la página de inicio de la aplicación web.

Temas

- [Paso 1: crear el proyecto y crear el repositorio de GitHub](#)
- [Paso 2: ver el código fuente](#)
- [Paso 3: crear una solicitud de extracción de GitHub](#)

Paso 1: crear el proyecto y crear el repositorio de GitHub

En este paso, utilice la consola para crear el proyecto y para crear una conexión al nuevo repositorio de GitHub. Para acceder al repositorio de GitHub, cree un recurso de conexión que AWS CodeStar utiliza para administrar la autorización con GitHub. Al crear el proyecto, se aprovisionan sus recursos adicionales para el usuario.

1. Inicie sesión en la AWS Management Console y, a continuación, abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Elija la región de AWS donde desee crear el proyecto y sus recursos.
3. En la página AWS CodeStar, seleccione Crear proyecto.
4. En la página Elegir una plantilla de proyecto, marque las casillas de verificación Aplicación web, Node.js y Amazon EC2. A continuación, elija entre las plantillas disponibles para ese conjunto de opciones.

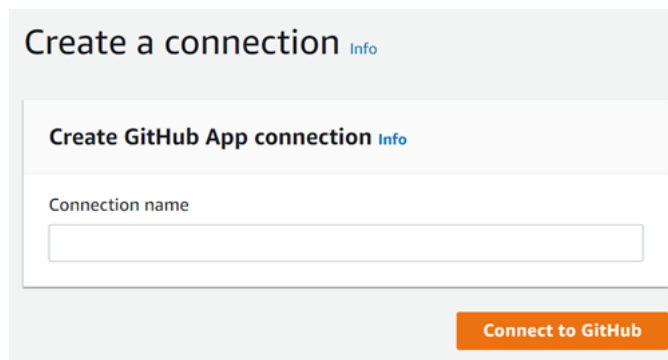
Para obtener más información, consulte [Plantillas de proyecto de AWS CodeStar](#).

5. Elija Siguiente.
6. En Project name (Nombre del proyecto), escriba un nombre para el proyecto (por ejemplo, **MyTeamProject**). Si elige otro nombre, asegúrese de utilizarlo durante todo el tutorial.
7. En Repositorio de proyectos, elija GitHub.
8. Si elige GitHub, tendrá que elegir o crear un recurso de conexión. Si ya tiene una conexión, selecciónela en el campo de búsqueda. De lo contrario, se creará una nueva conexión en este paso. Seleccione Conectarse a GitHub.

Se mostrará la página Crear una conexión.

Note

Para crear una conexión, debe tener una cuenta de GitHub. Si va a crear una conexión para una organización, debe ser el propietario de la organización.



Create a connection [Info](#)

Create GitHub App connection [Info](#)


Connection name

[Connect to GitHub](#)

- a. En Cree una conexión de aplicación de GitHub, el nombre de la conexión aparece en Nombre de la conexión. Seleccione Conectarse a GitHub.


Aparecerá la página Conectarse a GitHub, donde se muestra el campo Aplicaciones de GitHub.

- b. En GitHub Apps (Aplicaciones GitHub), elija la instalación de una aplicación o elija Install a new app (Instalar una aplicación nueva) para crear una.

 Note

Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya instaló la aplicación AWS Connector for GitHub, elija la aplicación y omita este paso.


- c. En la página Instalar el conector de AWS para GitHub, elija la cuenta donde desee instalar la aplicación.

 Note

Si instaló la aplicación previamente, puede elegir Configurar para dirigirse a una página de modificación para la instalación de la aplicación o puede utilizar el botón Atrás para volver a la consola.

- d. Si aparece la página Confirmar contraseña para continuar, introduzca su contraseña de GitHub y, a continuación, seleccione Iniciar sesión.
- e. En la página Instalar AWS Connector for GitHub, deje los valores predeterminados y elija Instalar.
- f. En la página Conectarse a GitHub, el ID de instalación para la nueva instalación aparece en Aplicaciones de GitHub.

Una vez que la conexión se haya creado correctamente, en la página de creación del proyecto de CodeStar, aparecerá el mensaje Listo para conectarse.


 Note

Puede ver la conexión en la sección Configuración de la consola de Herramientas para desarrolladores. Para obtener más información, consulte [Introducción a las conexiones](#).

Select a repository provider


CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).



i The GitHub repository provider now uses CodeStar Connections

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or

✔ **Ready to connect**

Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

[Redacted]
▼

Repository name

The name of the new repository.

cs-dk-gh

Repository description

An optional description of the new repository.

Public

- g. Para Propietario del repositorio, seleccione la organización de GitHub o su cuenta personal de GitHub.
- h. Para Nombre del repositorio, acepte el nombre del repositorio de GitHub predeterminado o escriba otro diferente.
- i. Elija Público o Privado.

i Note

Si desea utilizar AWS Cloud9 como entorno de desarrollo, debe elegir un repositorio público.

- j. (Opcional) En Descripción del repositorio, escriba una descripción para el repositorio de GitHub.

9. Configure sus instancias de Amazon EC2 en Configuración de Amazon EC2 si su proyecto se ha implementado en instancias de Amazon EC2 y desea realizar cambios. Por ejemplo, puede elegir entre los tipos de instancia disponibles para el proyecto.

Para Par de claves, seleccione el par de claves de Amazon EC2 que ha creado en [Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar](#). Seleccione Confirmando que tengo acceso al archivo de clave privada.

10. Seleccione Siguiente.
11. Revise los recursos y los detalles de la configuración.
12. Seleccione Siguiente o Crear proyecto. (La selección mostrada depende de la plantilla del proyecto).

Espere unos minutos mientras se crea el proyecto.

13. Una vez creado el proyecto, seleccione Ver la aplicación para ver la aplicación web.

Paso 2: ver el código fuente

En este paso, verá el código fuente y las herramientas que puede utilizar para el repositorio de código fuente.

1. En el panel de navegación del proyecto, seleccione Repositorio.

Para ver una lista de las confirmaciones en GitHub, seleccione Ver las confirmaciones. Esto abrirá el historial de confirmaciones en GitHub.

Para ver los problemas, seleccione la pestaña Problemas del proyecto. Para crear un nuevo problema en GitHub, seleccione Crear problema de GitHub. Esto abrirá el formulario de problemas del repositorio en GitHub.

2. En la pestaña Repositorio, seleccione el enlace que aparece debajo de Nombre del repositorio, y el repositorio del proyecto se abrirá en una nueva pestaña o ventana. Este repositorio contiene el código fuente de su proyecto.

Paso 3: crear una solicitud de extracción de GitHub

En este paso, se realizará un cambio menor en el código fuente y se creará una solicitud de extracción.

1. En GitHub, cree una nueva ramificación de características en el repositorio. Elija el campo desplegable de la ramificación principal e introduzca una nueva ramificación en el campo denominado `feature-branch`. Seleccione Crear la ramificación. La ramificación se creará y se extraerá para el usuario.
2. En GitHub, realice algún cambio en la ramificación `feature-branch`. Abra la carpeta pública y, a continuación, abra el archivo `index.html`.
3. En la consola de AWS CodeStar, en Solicitudes de extracción, para crear una solicitud de extracción en GitHub, seleccione Crear la solicitud de extracción. Esto abrirá el formulario de solicitud de extracción de repositorios en GitHub. En GitHub, seleccione el icono del lápiz para editar el archivo.

Después `Congratulations!`, agregue la cadena `Well done, <name>!` y sustituya `<name>` por su nombre. Seleccione Confirmar cambios. El cambio se confirmará en la ramificación de características.

4. En la consola de AWS CodeStar, seleccione el proyecto. Seleccione la pestaña Repositorio. En Solicitudes de extracción, seleccione Crear la solicitud de extracción.

El formulario se abrirá en GitHub. Deje la ramificación principal en la ramificación base. En Comparar con, elija la ramificación de características. Observe las diferencias.
5. En GitHub, seleccione Crear la solicitud de extracción. Se creará una solicitud de extracción denominada `Update index.html`.
6. En la consola de AWS CodeStar, consulta la nueva solicitud de extracción. Seleccione Combinar cambios para confirmar los cambios en el repositorio y combinar la solicitud de extracción con la ramificación principal de su repositorio.
7. Regrese al proyecto en AWS CodeStar y consulte la página Canalización. Debería ver la canalización implementándose.
8. Una vez creado el proyecto, seleccione Ver la aplicación para ver la aplicación web.

Plantillas de proyecto de AWS CodeStar

Las plantillas de proyectos de AWS CodeStar le permiten comenzar con una aplicación de ejemplo e implementarla con los recursos de AWS creados de modo que sea compatible con su proyecto de desarrollo. Cuando se elige una plantilla de proyecto de AWS CodeStar, el tipo de aplicación, el lenguaje de programación y la plataforma de computación se aprovisionan automáticamente para el usuario. Después de crear proyectos con las aplicaciones web, los servicios web, skills de Alexa y páginas web estáticas, puede sustituir la aplicación de ejemplo por la suya.

Una vez que AWS CodeStar crea el proyecto, puede modificar los recursos de AWS que admiten la entrega de su aplicación. AWS CodeStar trabaja con AWS CloudFormation para permitir que se utilice el código para crear servicios de soporte y plataformas con o sin servidor en la nube. AWS CloudFormation le permite modelar toda la infraestructura en un archivo de texto.

Temas

- [Archivos de proyectos de AWS CodeStar y recursos](#)
- [Introducción: elija una plantilla del proyecto](#)
- [Cómo hacer cambios en su proyecto de AWS CodeStar](#)

Archivos de proyectos de AWS CodeStar y recursos

Un proyecto de AWS CodeStar es una combinación de código fuente y de los recursos creados para implementar el código. El conjunto de recursos que le ayuda a crear, publicar e implementar el código se denomina recursos de la cadena de herramientas. Al crear el proyecto, una plantilla de AWS CloudFormation aprovisiona los recursos de la cadena de herramientas en una canalización de integración e implementación continuas (CI/CD).

Puede utilizar AWS CodeStar para crear proyectos en dos formas, en función de su nivel de experiencia con la creación de recursos de AWS:

- Si utiliza la consola para crear un proyecto, AWS CodeStar crea sus recursos de la cadena de herramientas, incluido el repositorio, y rellena el repositorio con código de aplicación de muestra y archivos del proyecto. Utilice la consola para configurar rápidamente proyectos de muestra en función de una serie de opciones de proyecto preconfiguradas.
- Cuando se utiliza la CLI para crear un proyecto, proporciona la plantilla de AWS CloudFormation que crea los recursos de la cadena de herramientas y también proporciona el código fuente de la

aplicación. Utilice la interfaz de línea de comandos (CLI) para permitir que AWS CodeStar cree su proyecto a partir de su plantilla y, a continuación, rellene el repositorio con su código de muestra.

Un proyecto de AWS CodeStar proporciona un solo punto de administración. Puede utilizar el asistente Create project (Crear proyecto) en la consola para configurar un proyecto de muestra. A continuación, puede utilizarlo como una plataforma de colaboración para administrar permisos y recursos para su equipo. Para obtener más información, consulte [¿Qué es AWS CodeStar?](#). Si utiliza la consola para crear un proyecto, el código fuente se suministra como código de muestra y se crean automáticamente los recursos de la cadena de herramientas de CI/CD

Al crear un proyecto en la consola, AWS CodeStar aprovisiona los siguientes recursos:

- Un repositorio de código en GitHub o CodeCommit.
- En el repositorio del proyecto, un archivo README.md que proporciona detalles de archivos y directorios.
- En el repositorio del proyecto, un archivo `template.yml` que almacena la definición de la pila del tiempo de ejecución de la aplicación. Este archivo se utiliza para añadir o modificar los recursos del proyecto que no sean recursos de la cadena de herramientas, como, por ejemplo, los recursos de AWS que se utilizan para las notificaciones, el soporte para la base de datos, la supervisión y el seguimiento.
- Los servicios y recursos de AWS creados en relación con la canalización, como el bucket del artefacto de Amazon S3, Eventos de Amazon CloudWatch y roles de servicios relacionados.
- Una aplicación de muestra funcional con código fuente completo y un punto de conexión de HTTP pública.
- Un recurso informático de AWS, en función del tipo de plantilla del proyecto de AWS CodeStar:
 - Una función Lambda.
 - Una instancia de Amazon EC2.
 - Un entorno de AWS Elastic Beanstalk.
- A partir del 6 de diciembre de 2018 PDT:
 - Un límite de permisos, que es una política de IAM especializada para controlar el acceso a los recursos del proyecto. El límite de permisos está asociado de forma predeterminada a roles en el proyecto de ejemplo. Para obtener más información, consulte [Límite de permisos de IAM para roles de trabajador](#).

- Un rol de IAM de AWS CloudFormation para crear recursos de proyecto mediante AWS CloudFormation que incluye permisos para todos los recursos de AWS CloudFormation admitidos, incluidos roles de IAM.
- Un rol de IAM de cadena de herramientas.
- Roles de ejecución para Lambda definidos en la pila de aplicación y que se pueden modificar.
- Antes del 6 de diciembre de 2018 PDT:
 - Un rol de IAM de AWS CloudFormation para crear recursos de proyecto con soporte para un conjunto limitado de recursos de AWS CloudFormation.
 - Un rol de IAM para crear un recurso de CodePipeline.
 - Un rol de IAM para crear un recurso de CodeBuild.
 - Un rol de IAM para crear un recurso de CodeDeploy, si es aplicable a su tipo de proyecto.
 - Un rol de IAM para crear la aplicación web de Amazon EC2, si es aplicable a su tipo de proyecto.
 - Un rol de IAM para crear un recurso de CloudWatch Events.
 - Un rol de ejecución para Lambda que se modifica de forma dinámica para incluir un conjunto parcial de recursos.

El proyecto incluye páginas detalladas que muestran el estado y contienen enlaces a la administración del equipo y enlaces a instrucciones de configuración para los IDE o para el repositorio, así como un historial de confirmaciones de los cambios en el código fuente en el repositorio. También puede seleccionar herramientas para conectarse a herramientas de seguimiento externas, como, por ejemplo, Jira.

Introducción: elija una plantilla del proyecto

Cuando elija un proyecto de AWS CodeStar en la consola, elige desde un conjunto de opciones preconfiguradas con código de muestra y los recursos para empezar rápidamente. Estas opciones se denominan plantillas de proyecto. Cada plantilla del proyecto de AWS CodeStar consta de un lenguaje de programación, un tipo de aplicación y plataforma de computación. La combinación que seleccione determina la plantilla del proyecto.

Elegir una plataforma de computación de plantillas

Cada plantilla configura uno de los siguientes tipos de plataformas de computación:

- Al elegir un proyecto de AWS Elastic Beanstalk, realice la implementación en un entorno de AWS Elastic Beanstalk en instancias de Amazon Elastic Compute Cloud en la nube.
- Al elegir un proyecto de Amazon EC2, AWS CodeStar crea instancias de Linux EC2 para alojar su aplicación en la nube. Los miembros del equipo de su proyecto pueden acceder a las instancias y su equipo utiliza el par de claves que proporciona a SSH en sus instancias de Amazon EC2. AWS CodeStar también tiene una SSH administrada que utiliza permisos del miembro del equipo para administrar conexiones de pares de claves.
- Al elegir AWS Lambda, AWS CodeStar crea un entorno sin servidor al que se accede a través de Amazon API Gateway, sin instancias o servidores que mantener.

Elija un tipo de aplicación de plantilla

Cada plantilla configura uno de los siguientes tipos de aplicaciones:

- Servicios web

Un servicio web se utiliza para tareas que se ejecutan en segundo plano como, por ejemplo, llamar a API. Una vez que AWS CodeStar crea su proyecto de servicio web de muestra, puede elegir la dirección URL del punto de conexión para ver "Hello World", pero el uso principal de este tipo de aplicación no es como interfaz de usuario (IU). Las plantillas de proyectos de AWS CodeStar de esta categoría admiten el desarrollo en Ruby, Java, ASP.NET, PHP, Node.js entre otros.

- Aplicación web

Una aplicación web incluye una IU. Una vez que AWS CodeStar crea su proyecto de aplicación web de muestra, puede elegir la dirección URL del punto de conexión para ver una aplicación web interactiva. Las plantillas de proyectos de AWS CodeStar de esta categoría admiten el desarrollo en Ruby, Java, ASP.NET, PHP, Node.js entre otros.

- Página web estática

Elija esta plantilla si desea un proyecto para un sitio web HTML. Las plantillas de proyectos de AWS CodeStar de esta categoría admiten el desarrollo en HTML5.

- Habilidad de Alexa

Seleccione esta plantilla si quiere crear una habilidad de Alexa con una función AWS Lambda. Al crear el proyecto de habilidad, AWS CodeStar devuelve un Nombre de recurso de Amazon (ARN)

que se puede utilizar como punto de conexión del servicio. Para obtener más información, consulte [Alojar una habilidad personalizada como una función de AWS Lambda](#).

Note

Las funciones de Lambda para las habilidades de Alexa se admiten solo en las regiones Este de EE. UU. (Norte de Virginia), Oeste de EE. UU. (Oregón), UE (Irlanda) y Asia-Pacífico (Tokio).

- Regla de configuración

Elija esta plantilla si desea que un proyecto tenga una regla de AWS Config que le permita automatizar las reglas en recursos de AWS en su cuenta. La función devuelve un ARN que puede utilizar como punto de conexión de servicio para la regla.

Elegir un lenguaje de programación de la plantilla

Cuando elija una plantilla de proyecto, seleccione un lenguaje de programación, como, por ejemplo, Ruby, Java, ASP.NET, PHP, Node.js y mucho más.

Cómo hacer cambios en su proyecto de AWS CodeStar

Puede actualizar su proyecto modificando:

- Código de muestra y recursos del lenguaje de programación para su aplicación.
- Los recursos que componen la infraestructura donde se almacena e implementa su aplicación (sistemas operativos, aplicaciones y servicios de soporte, los parámetros de implementación y la plataforma de computación en la nube). Puede modificar recursos de la aplicación en el archivo `template.yml`. Este es el archivo de AWS CloudFormation que crea un modelo de su entorno en tiempo de ejecución de la aplicación.

Note

Si está trabajando en un proyecto de habilidades de Alexa en AWS CodeStar, no podrá realizar cambios en dicha habilidad fuera del repositorio del código fuente de AWS CodeStar

(CodeCommit o GitHub). Si edita la habilidad en el portal de desarrolladores de Alexa, el cambio no se aplica al repositorio fuente y las versiones no se sincronizan.

Cambiar código fuente de aplicación y enviar los cambios

Para modificar código fuente de muestra, scripts y otros archivos de código fuente de la aplicación, edite archivos en el repositorio de código fuente de la siguiente manera:

- Mediante el modo Editar en CodeCommit o GitHub.
- Abriendo el proyecto en un IDE, como por ejemplo AWS Cloud9.
- Clonando el repositorio a nivel local y confirmando y enviando, continuación, los cambios. Para obtener más información, consulte [Paso 4: confirmar un cambio](#).

Cambiar recursos de aplicaciones con el archivo Template.yml

En lugar de modificar manualmente el recurso de una infraestructura, utilice AWS CloudFormation para modelar e implementar los recursos del tiempo de ejecución de la aplicación.

Puede modificar o añadir un recurso de aplicación, como, por ejemplo, una función Lambda, en su pila de tiempo de ejecución editando el archivo `template.yml` en su repositorio del proyecto. Puede añadir cualquier recurso que esté disponible como recurso de AWS CloudFormation.

Para cambiar el código o la configuración de una función de AWS Lambda, consulte [Añadir un recurso a un proyecto](#).

Modifique el archivo `template.yml` en el repositorio del proyecto para agregar el tipo de recursos de AWS CloudFormation que son recursos de la aplicación. Al añadir el recurso de una aplicación a la sección `Resources` del archivo `template.yml`, AWS CloudFormation y AWS CodeStar crean el recurso por usted. Para obtener una lista de recursos de AWS CloudFormation y las propiedades necesarias, consulte [Referencia de tipos de recursos de AWS](#). Para obtener más información, consulte este ejemplo en [Paso 1: editar el rol de trabajador de CloudFormation en IAM](#).

AWS CodeStar le permite implementar prácticas recomendadas mediante la configuración y el modelado del entorno del tiempo de ejecución de la aplicación.

Cómo administrar permisos para cambiar los recursos de aplicaciones

Cuando se utiliza AWS CloudFormation para añadir recursos de la aplicación de tiempo de ejecución, como, por ejemplo, una función Lambda, el rol de empleado de AWS CloudFormation puede utilizar los permisos que ya tiene. Para algunos recursos de la aplicación de tiempo de ejecución, deberá ajustar manualmente los permisos del rol de trabajador de AWS CloudFormation antes de editar el archivo `template.yml`.

Para ver un ejemplo sobre cómo cambiar los permisos del rol del empleado de AWS CloudFormation, consulte [Paso 5: Añadir permisos a nivel de recursos con una política insertada](#).

Prácticas recomendadas de AWS CodeStar

AWS CodeStar se integra con una serie de productos y servicios. En las siguientes secciones se describen las prácticas recomendadas para AWS CodeStar y los productos y servicios relacionados.

Temas

- [Prácticas recomendadas de seguridad para recursos de AWS CodeStar](#)
- [Prácticas recomendadas para configurar las versiones de dependencias](#)
- [Prácticas recomendadas de monitorización y registro para recursos de AWS CodeStar](#)

Prácticas recomendadas de seguridad para recursos de AWS CodeStar

Debería aplicar parches con regularidad y revisar las prácticas recomendadas de seguridad para las dependencias que utiliza su aplicación. Utilice estas prácticas recomendadas de seguridad para actualizar su código de muestra y mantener su proyecto en un entorno de producción:

- Realice el seguimiento de los anuncios continuos de seguridad y de actualizaciones para su entorno.
- Antes de implementar el proyecto, siga las prácticas recomendadas desarrolladas para su entorno.
- Revise las dependencias de su entorno de forma periódica y actualice según sea necesario.
- Cada plantilla de AWS CodeStar contiene instrucciones de configuración para su lenguaje de programación. Consulte el archivo README .md en el repositorio de origen de su proyecto.
- Como práctica recomendada para aislar los recursos del proyecto, gestione el acceso con privilegios mínimos a los recursos de AWS mediante una estrategia de varias cuentas, tal como se presenta en [Seguridad en AWS CodeStar](#).

Prácticas recomendadas para configurar las versiones de dependencias

El código fuente de muestra en su proyecto de AWS CodeStar utiliza las dependencias que se enumeran en el archivo package .json del repositorio de origen. Como práctica recomendada,

defina siempre sus dependencias para que apunten a una versión específica. Esto es lo que se conoce como asignar la versión. No se recomienda establecer la versión en `latest` ya que puede introducir cambios que pueden interrumpir su aplicación sin previo aviso.

Prácticas recomendadas de monitorización y registro para recursos de AWS CodeStar

Puede utilizar características de registro de AWS para determinar las acciones que los usuarios han realizado en su cuenta y los recursos que se han utilizado. Los archivos de registro muestran:

- La fecha y la hora de las acciones.
- La dirección IP de origen de una acción.
- Las acciones que han fallado debido a permisos inadecuados.

AWS CloudTrail se puede utilizar para registrar llamadas a la API AWS y eventos relacionados que realice una cuenta de AWS o bien que se realicen en su nombre. Para obtener más información, consulte [Registro de llamadas a la API de AWS CodeStar con AWS CloudTrail](#).

Trabajar con proyectos en AWS CodeStar

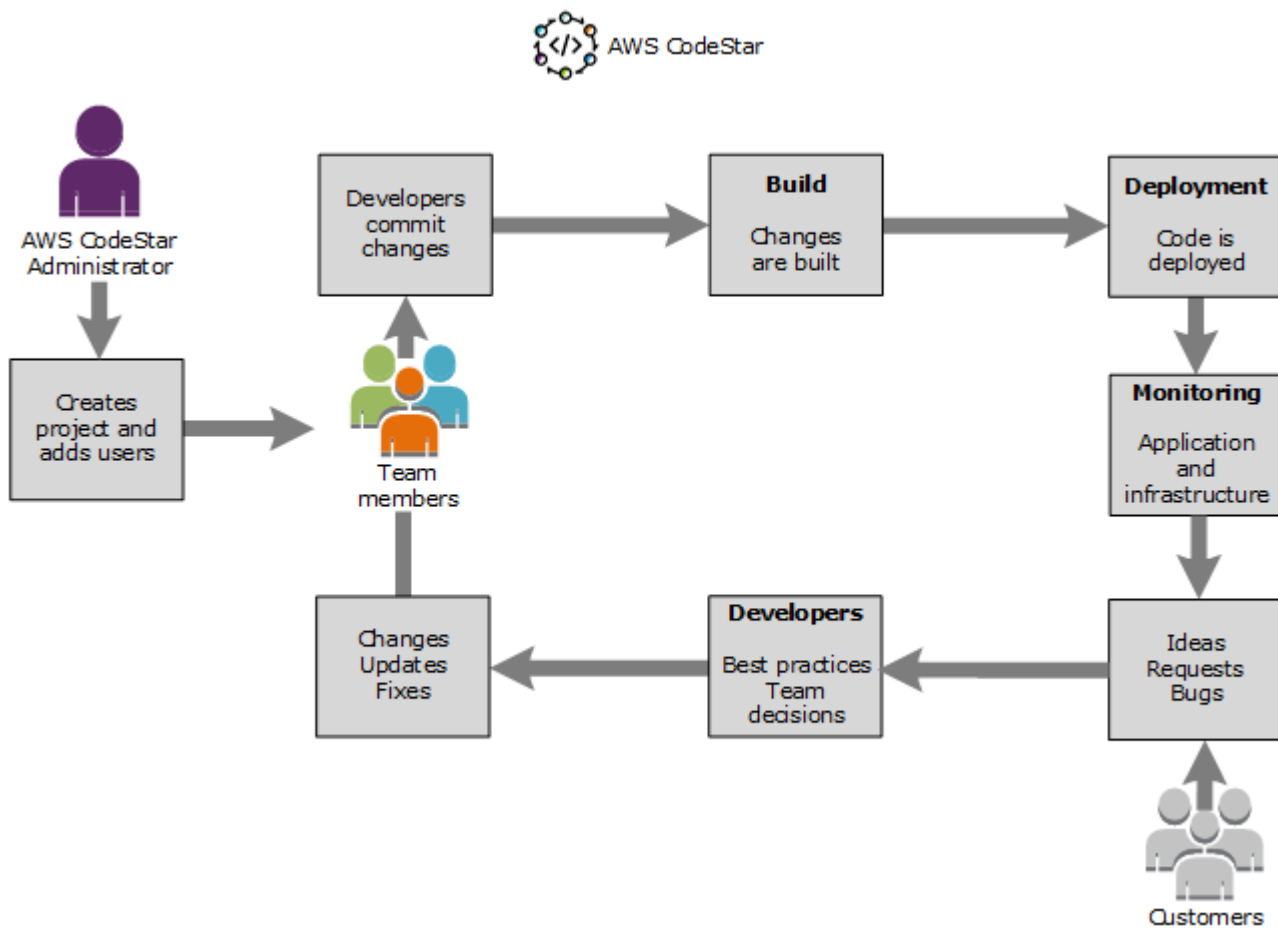
Al utilizar un proyecto de plantilla de AWS CodeStar, puede crear rápidamente un proyecto que ya esté configurado con los recursos que necesite, entre los que se incluyen:

- Repositorio de origen
- Entorno de compilación
- Recursos de implementación y alojamiento
- Lenguaje de programación

La plantilla incluso incluye ejemplos de código fuente para que pueda empezar a trabajar con su proyecto inmediatamente.

Una vez que tenga un proyecto, puede añadir o eliminar recursos, personalizar el panel del proyecto y monitorizar el progreso.

En el siguiente diagrama se muestra un flujo de trabajo básico en un proyecto de AWS CodeStar.



El flujo de trabajo básico en el diagrama muestra a un desarrollador con la política `AWSCodeStarFullAccess` aplicada que crea un proyecto y añade en él a los miembros del equipo. Juntos escriben, crean, prueban e implementan código. El panel del proyecto proporciona herramientas que se pueden utilizar en tiempo real para ver la actividad de la aplicación y supervisar las compilaciones, el flujo de código a través de la canalización de implementación y mucho más. El equipo utiliza el icono de la wiki del equipo para compartir información, prácticas recomendadas y enlaces. Integran el software de seguimiento de problemas para que les ayude a hacer un seguimiento del progreso y las tareas. Como los clientes proporcionan solicitudes y comentarios, el equipo añade esta información al proyecto y la integra en la planificación y el desarrollo del proyecto. A medida que crece el proyecto, el equipo añade más miembros de equipo para respaldar su base de código.

Crear un proyecto en AWS CodeStar

Utilice la consola de AWS CodeStar para crear un proyecto. Si utiliza una plantilla de proyectos, esta configurará los recursos necesarios. La plantilla también incluye código de muestra que puede utilizar para empezar a desarrollar código.

Para crear un proyecto, inicie sesión en la AWS Management Console con un usuario de IAM que tenga la política `AWSCodeStarFullAccess` o los permisos equivalentes. Para obtener más información, consulte [Configuración AWS CodeStar](#).

Note

Antes de completar los procedimientos en este tema, debe completar los pasos descritos en [Configuración AWS CodeStar](#).

Temas

- [Crear un proyecto en AWS CodeStar \(consola\)](#)
- [Crear un proyecto en AWS CodeStar \(AWS CLI\)](#)

Crear un proyecto en AWS CodeStar (consola)

Utilice la consola de AWS CodeStar para crear un proyecto.

Para crear un proyecto en AWS CodeStar

1. Inicie sesión en la AWS Management Console y, a continuación, abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.

Asegúrese de que ha iniciado sesión en la región de AWS donde desea crear el proyecto y sus recursos. Por ejemplo, para crear un proyecto en Este de EE. UU. (Ohio), asegúrese de haber seleccionado esa región de AWS. Para obtener más información sobre las regiones de AWS en las que AWS CodeStar está disponible, consulte [Regiones y puntos de conexión](#) en la Referencia general de AWS .

2. En la página AWS CodeStar, seleccione Crear proyecto.
3. En la página Elegir una plantilla de proyecto, seleccione el tipo de proyecto en la lista de plantillas de proyectos de AWS CodeStar. Puede utilizar la barra de filtros para restringir las


opciones. Por ejemplo, en el caso de un proyecto de aplicación web escrito en Node.js que se implementará en instancias de Amazon EC2, seleccione las casillas de verificación Aplicación web, Node.js y Amazon EC2. A continuación, elija entre las plantillas disponibles para ese conjunto de opciones.

Para obtener más información, consulte [Plantillas de proyecto de AWS CodeStar](#).

4. Elija Siguiente.
5. En el campo de entrada de texto Nombre del proyecto, introduzca un nombre para el proyecto, como *Mi primer proyecto*. El ID del proyecto, el ID del proyecto se deriva del nombre de dicho proyecto, pero se limita a 15 caracteres.

Por ejemplo, el ID predeterminado de un proyecto con el nombre *Mi primer proyecto* sería *mi-primer-proye*. Este ID de proyecto es la base de los nombres de todos los recursos asociados al proyecto. AWS CodeStar utiliza este ID de proyecto como parte de la dirección URL del repositorio de código y para los nombres de roles de acceso de seguridad y políticas relacionados en IAM. Una vez creado el proyecto, el ID del proyecto no puede modificarse. Para editar el ID del proyecto antes de crearlo, en ID del proyecto, introduzca el ID que desee utilizar.

Para obtener más información sobre los límites de los nombres de proyectos y los ID de proyectos, consulte [Límites en AWS CodeStar](#).

 Note

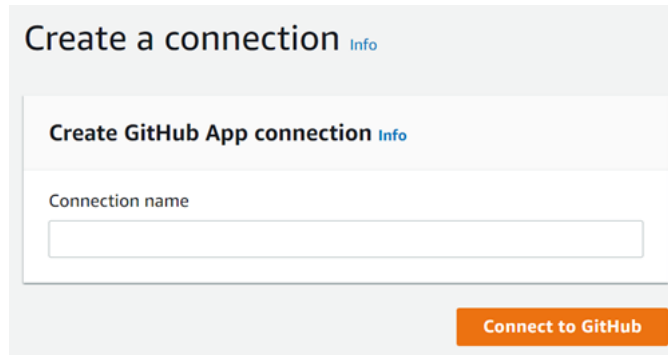
Los ID de proyecto deben ser únicos de la cuenta de AWS dentro de una región de AWS.

6. Elija el proveedor del repositorio: AWS CodeCommit o GitHub.
7. Si ha elegido AWS CodeCommit para Repository name (Nombre del repositorio), acepte el nombre del repositorio de AWS CodeCommit predeterminado o escriba otro diferente. A continuación, vaya al paso 9.
8. Si elige GitHub, debe elegir o crear un recurso de conexión. Si ya tiene una conexión, selecciónela en el campo de búsqueda. De lo contrario, cree una conexión nueva ahora. Seleccione Conectarse a GitHub.

Se mostrará la página Crear una conexión.

Note

Para crear una conexión, debe tener una cuenta de GitHub. Si va a crear una conexión para una organización, debe ser el propietario de la organización.



- a. En Cree una conexión de aplicación de GitHub, en el campo de entrada de texto Nombre de la conexión, escriba un nombre para la conexión. Seleccione Conectarse a GitHub.


Aparecerá la página Conectarse a GitHub, donde se muestra el campo Aplicaciones de GitHub.

- b. En Aplicaciones de GitHub, elija la instalación de una aplicación o elija Instalar una aplicación nueva para crear una.

Note

Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya instaló la aplicación AWS Connector for GitHub, elija la aplicación y omita este paso.


- c. En la página Instalar el conector de AWS para GitHub, elija la cuenta donde desee instalar la aplicación.

 Note

Si instaló la aplicación previamente, puede elegir Configurar para dirigirse a una página de modificación para la instalación de la aplicación o puede utilizar el botón Atrás para volver a la consola.

- d. Si aparece la página Confirmar contraseña para continuar, introduzca su contraseña de GitHub y, a continuación, seleccione Iniciar sesión.
- e. En la página Instalar el conector de AWS para GitHub, deje los valores predeterminados y seleccione Instalar.
- f. En la página Conectarse a GitHub, el ID de instalación para la nueva instalación aparece en el campo de entrada de texto Aplicaciones de GitHub.

Una vez creada la conexión, en la página de creación del proyecto de CodeStar, aparece el mensaje Listo para conectarse.


 Note

Puede ver la conexión en la sección Configuración de la consola de Herramientas para desarrolladores. Para obtener más información, consulte [Introducción a las conexiones](#).

Select a repository provider


CodeCommit

Use a new AWS CodeCommit repository for your project.



GitHub

Use a new GitHub source repository for your project (requires an existing GitHub account).




The GitHub repository provider now uses CodeStar Connections

To use a GitHub repository in CodeStar, create a connection. The connection will use GitHub Apps to access your repository. Use the following options to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection or create a new one and then return to this task.

or

 **Ready to connect**

Your Github connection is ready for use.

Repository owner

The owner of the new repository. This can be a personal GitHub account or a GitHub organization.

Repository name

The name of the new repository.

Repository description

An optional description of the new repository.

Public

- g. Para Propietario del repositorio, seleccione la organización de GitHub o su cuenta personal de GitHub.
- h. Para Nombre del repositorio, acepte el nombre del repositorio de GitHub predeterminado o escriba otro diferente.
- i. Elija Público o Privado.

Note


Si desea utilizar AWS Cloud9 como entorno de desarrollo, debe elegir Público.

- j. (Opcional) En Descripción del repositorio, escriba una descripción para el repositorio de GitHub.

 Note

Si selecciona una plantilla de proyecto de habilidades de Alexa, deberá conectar una cuenta de desarrollador de Amazon. Para obtener más información acerca de cómo trabajar con proyectos de habilidades de Alexa, consulte [Tutorial: crear un proyecto de una habilidad de Alexa en AWS CodeStar](#).

9. Si su proyecto está desplegado en instancias de Amazon EC2 y desea realizar cambios, configure las instancias de Amazon EC2 en Configuración de Amazon EC2. Por ejemplo, puede elegir entre los tipos de instancia disponibles para el proyecto.

 Note

Los distintos tipos de instancia de Amazon EC2 proporcionan diferentes niveles de operatividad informática y podrían tener distintos costos asociados. Para obtener más información, consulte [Tipos de instancias de Amazon EC2](#) y [Precios de Amazon EC2](#). Si tiene más de una nube privada virtual (VPC) o varias subredes creadas en Amazon Virtual Private Cloud, también puede elegir la VPC y la subred que va a utilizar. Sin embargo, si elige un tipo de instancia de Amazon EC2 que no sea compatible con instancias dedicadas, no puede elegir una VPC cuya tenencia de instancias esté establecida en Dedicada.

Para obtener más información, consulte [¿Qué es Amazon VPC?](#) y [Conceptos básicos de las instancias dedicadas](#).

Para Par de claves, seleccione el par de claves de Amazon EC2 que ha creado en [Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar](#). Seleccione Confirmo que tengo acceso al archivo de clave privada.

10. Elija Siguiente.
11. Revise los recursos y los detalles de la configuración.
12. Seleccione Siguiente o Crear proyecto. (La selección mostrada depende de la plantilla del proyecto).

Es posible que el proyecto, que incluye el repositorio, tarde unos minutos en crearse.

- Una vez que el proyecto tenga un repositorio, puede utilizar la página Repositorio para configurar el acceso al mismo. Utilice los enlaces que se encuentran en Próximos pasos para configurar un IDE, configurar el seguimiento de problemas o añadir miembros del equipo a su proyecto.

Durante la creación del proyecto, puede [agregar miembros al equipo](#) o [configurar el acceso](#) al repositorio de su proyecto desde la línea de comandos o su IDE favorito.

Crear un proyecto en AWS CodeStar (AWS CLI)

Un proyecto de AWS CodeStar es una combinación de código fuente y de los recursos creados para implementar el código. El conjunto de recursos que le ayuda a crear, publicar e implementar el código se denomina recursos de la cadena de herramientas. Al crear el proyecto, una plantilla de AWS CloudFormation aprovisiona los recursos de la cadena de herramientas en una canalización de integración e implementación continuas (CI/CD).

Cuando se usa la consola para crear un proyecto, la plantilla de la cadena de herramientas se crea automáticamente. Cuando se utiliza la AWS CLI para crear un proyecto, usted debe crear la plantilla de la cadena de herramientas que crea los recursos de la cadena de herramientas.

Una cadena de herramientas completa requiere los siguientes recursos recomendados:

- Un repositorio de CodeCommit o de GitHub que contenga el código fuente.
- Una canalización de CodePipeline configurada para estar atenta a los cambios en el repositorio.
 - Al utilizar CodeBuild para ejecutar pruebas de integración o de unidad, le recomendamos que agregue una etapa de compilación a la canalización para crear artefactos de compilación.
 - Le recomendamos que añada una etapa de implementación a la canalización que utiliza CodeDeploy o AWS CloudFormation para implementar el artefacto de compilación y el código fuente a la infraestructura de tiempo de ejecución.

Note

Dado que CodePipeline requiere al menos dos etapas en una canalización y la primera etapa debe ser la etapa de origen, añada una etapa de compilación o de implementación como segunda etapa.


Las cadenas de herramientas de AWS CodeStar se definen como una [plantilla de CloudFormation](#).

Para ver un tutorial en el que se explica esta tarea y se configuran los recursos de muestra, consulte [Tutorial: Crear un proyecto en AWS CodeStar con la AWS CLI](#).

Requisitos previos:

Al crear un proyecto, debe proporcionar los siguientes parámetros en un archivo de entrada. Si no se proporcionan estos parámetros, AWS CodeStar creará un proyecto vacío.

- Código fuente. Si este parámetro se incluye en la solicitud, también deberá incluir una plantilla de la cadena de herramientas.
 - El código fuente debe incluir el código de la aplicación necesario para ejecutar el proyecto.
 - El código fuente debe incluir los archivos de configuración necesarios, como, por ejemplo, un archivo `buildspec.yml` para un proyecto de CodeBuild o un archivo `appspec.yml` para una implementación de CodeDeploy.
 - Puede incluir elementos opcionales en el código fuente como un archivo `README` o `template.yml` para los recursos de AWS de la cadena de herramientas.
- Plantilla de la cadena de herramientas. La plantilla de la cadena de herramientas aprovisiona los recursos de AWS y los roles de IAM para administrarlos en el proyecto.
- Ubicaciones de origen. Si especifica el código fuente y una plantilla de la cadena de herramientas para el proyecto, deberá proporcionar una ubicación. Cargue los archivos de origen y la plantilla de la cadena de herramientas al bucket de Amazon S3. AWS CodeStar recupera los archivos y los utiliza para crear el proyecto.

 Important

Asegúrese de configurar la región de AWS preferida en la AWS CLI. El proyecto se crea en la región de AWS configurada en la AWS CLI.

1. Ejecute el comando `create-project` e incluya el parámetro `--generate-cli-skeleton`:

```
aws codestar create-project --generate-cli-skeleton
```


En el resultado se muestran datos con formato JSON. Copie los datos en un archivo (por ejemplo, `input.json`) en la ubicación del equipo o instancia local en la que haya instalado la AWS CLI. Modifique los datos copiados como se indica a continuación y guarde los resultados.

```
{
  "name": "project-name",
  "id": "project-id",
  "description": "description",
  "sourceCode": [
    {
      "source": {
        "s3": {
          "bucketName": "s3-bucket-name",
          "bucketKey": "s3-bucket-object-key"
        }
      },
      "destination": {
        "codeCommit": {
          "name": "codecommit-repository-name"
        },
        "gitHub": {
          "name": "github-repository-name",
          "description": "github-repository-description",
          "type": "github-repository-type",
          "owner": "github-repository-owner",
          "privateRepository": true,
          "issuesEnabled": true,
          "token": "github-personal-access-token"
        }
      }
    }
  ],
  "toolchain": {
    "source": {
      "s3": {
        "bucketName": "s3-bucket-name",
        "bucketKey": "s3-bucket-object-key"
      }
    },
    "roleArn": "service-role-arn",
    "stackParameters": {
      "KeyName": "key-name"
    }
  },
  "tags": {
    "KeyName": "key-name"
  }
}
```

```
}
```

Sustituya lo siguiente:

- *project-name*: obligatorio. Nombre fácil de este proyecto de AWS CodeStar.
- *project-id*: obligatorio. ID de proyecto de este proyecto de AWS CodeStar.

 Note

Debe tener un ID de proyecto único al crear un proyecto. Se mostrará un error si envía un archivo de entrada con un ID de proyecto que ya existe.

- *description*: opcional. Descripción de este proyecto de AWS CodeStar.
- *sourceCode*: opcional. Información de configuración para el código fuente proporcionado para el proyecto. Actualmente, solo se admite un único objeto `sourceCode`. Cada objeto `sourceCode` contiene información sobre la ubicación desde la que AWS CodeStar obtiene código fuente y el destino donde se rellena el código fuente.
- *source*: obligatorio. Define la ubicación donde se ha cargado el código fuente. El único origen admitido es Amazon S3. AWS CodeStar obtiene el código fuente y lo incluye en el repositorio después de que el usuario cree el proyecto.
 - *S3*: opcional. La ubicación de Amazon S3 del código fuente.
 - *bucket-name*: bucket que contiene el código fuente.
 - *bucket-key*: prefijo del bucket y clave de objeto que apunta al archivo `.zip` que contiene el código fuente (por ejemplo, `src.zip`).
- *destination*: opcional. Ubicaciones de destino donde el código fuente se rellena cuando se crea el proyecto. Los destinos admitidos para el código fuente son CodeCommit y GitHub.

Solo puede proporcionar una de estas dos opciones:

- *codeCommit*: el único atributo obligatorio es el nombre del repositorio de CodeCommit que debe contener el código fuente. Este repositorio debe estar en la plantilla de la cadena de herramientas.

Note

Para CodeCommit, debe proporcionar el nombre del repositorio que se definió en la pila de la cadena de herramientas. AWS CodeStar inicializa este repositorio con el código fuente que ha facilitado en Amazon S3.

- **github**: este objeto representa la información necesaria para crear el repositorio de GitHub y propagarlo con código fuente. Si elige un repositorio de GitHub, los siguientes valores son obligatorios.

Note

Para GitHub, no puede especificar un repositorio de GitHub existente. AWS CodeStar crea uno para el usuario y lo rellena con el código fuente que ha cargado en Amazon S3. AWS CodeStar utiliza la siguiente información para crear el repositorio en GitHub.


- **name**: obligatorio. Nombre de su repositorio de GitHub.
- **description**: obligatorio. Descripción de su repositorio de GitHub.
- **type**: obligatorio. Tipo de repositorio de GitHub. Los valores válidos son User (usuario) u Organization (organización).
- **owner**: obligatorio. Nombre de usuario de GitHub para el propietario del repositorio. Si el repositorio debe ser propiedad de una organización de GitHub, proporcione el nombre de la organización.
- **privateRepository**: obligatorio. Si desea que este repositorio sea privado o público. Los valores válidos son true (verdadero) o false (falso).
- **issuesEnabled**: obligatorio. Si desea habilitar problemas en GitHub con este repositorio. Los valores válidos son true (verdadero) o false (falso).
- **token**: opcional. Se trata de un token de acceso personal que utiliza AWS CodeStar para acceder a su cuenta de GitHub. Este token deben contener los siguientes ámbitos: repo, user y admin:repo_hook. Para recuperar un token de acceso personal de GitHub, consulte la página sobre [cómo crear un token de acceso personal para la línea de comandos](#) en el sitio web de GitHub.

Note

Si utiliza la CLI para crear un proyecto con un repositorio de origen de GitHub, AWS CodeStar utiliza el token para acceder al repositorio a través de las aplicaciones de OAuth. Si utiliza la consola para crear un proyecto con un repositorio de origen de GitHub, AWS CodeStar utiliza un recurso de conexión que accede al repositorio con las aplicaciones de GitHub.

- ***toolchain***: información sobre la cadena de herramientas CI/CD que se debe configurar cuando se crea el proyecto. Esta información incluye la ubicación en la que ha cargado la plantilla de la cadena de herramientas. La plantilla crea la pila de AWS CloudFormation que contiene los recursos de la cadena de herramientas. Esto también incluye las anulaciones de parámetros a los que AWS CloudFormation hace referencia y el rol que se va a usar para crear la pila. AWS CodeStar recupera la plantilla y utiliza AWS CloudFormation para ejecutarla.
- ***source***: obligatorio. La ubicación de la plantilla de la cadena de herramientas. Amazon S3 es la única ubicación de origen admitida.
 - ***S3***: opcional. Ubicación de Amazon S3 donde se ha cargado la plantilla de la cadena de herramientas.
 - ***bucket-name***: el nombre del bucket de Amazon S3.
 - ***bucket-key***: prefijo del bucket y clave de objeto que apunta al archivo .yaml o .json que contiene la plantilla de la cadena de herramientas (por ejemplo, files/toolchain.yaml).
 - ***stackParameters***: opcional. Contiene los pares de valor de clave que se transfieren a AWS CloudFormation. Estos son los parámetros, si los hay, que la plantilla de la cadena de herramientas tiene configurados como referencia.
 - ***role***: opcional. Rol que se utiliza para crear los recursos de la cadena de herramientas en la cuenta. El rol es obligatorio, tal como se indica a continuación:
 - Si no se proporciona el rol, AWS CodeStar utiliza el rol de servicio predeterminado creado para su cuenta si la cadena de herramientas es una plantilla de inicio rápido de AWS CodeStar. Si no hay ningún rol de servicio en la cuenta, puede crear uno. Para obtener más información, consulte [Paso 2: Crear el rol de AWS CodeStar servicio](#).

- Debe proporcionar el rol que debe si va a cargar y utilizar su propia plantilla de cadena de herramientas personalizada. Puede crear un rol que se base en el rol de servicio y la instrucción de política de AWS CodeStar. Para ver un ejemplo de esta instrucción de política, consulte [AWSCodeStarServiceRole Política](#).
- **tags**: opcional. Etiquetas asociadas al proyecto de AWS CodeStar.

 Note

Estas etiquetas no se asocian a los recursos incluidos en el proyecto.

2. Cambie al directorio que contiene el archivo que acaba de guardar y ejecute de nuevo el comando `create-project`. Incluya el parámetro `--cli-input-json`.

```
aws codestar create-project --cli-input-json file://input.json
```

3. Si el comando se ejecuta correctamente, aparecerán datos similares a los siguientes en el resultado:

```
{
  "id": "project-ID",
  "arn": "arn"
}
```

- El resultado contiene información acerca del nuevo proyecto:
 - El valor `id` representa el ID del proyecto.
 - El valor `arn` representa el ARN del proyecto.
4. Para comprobar el estado de creación del proyecto, utilice el comando `describe-project`. Incluya el parámetro `--id`.

```
aws codestar describe-project --id <project_ID>
```

En el resultado se muestra información similar a la siguiente:

```
{
  "name": "MyProject",
  "id": "myproject",
  "arn": "arn:aws:codestar:us-east-1:account_ID:project/myproject",
  "description": ""
}
```

```
"createdTimeStamp": 1539700079.472,  
  "stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-  
myproject/stack-ID",  
  "status": {  
    "state": "CreateInProgress"  
  }  
}
```

- El resultado contiene información acerca del nuevo proyecto:
 - El valor `state` representa el estado de la creación del proyecto, como, por ejemplo, `CreateInProgress` o `CreateComplete`.

Durante la creación del proyecto, puede [agregar miembros al equipo](#) o [configurar el acceso](#) al repositorio de su proyecto desde la línea de comandos o su IDE favorito.

Utilizar un IDE con AWS CodeStar

Cuando integra un entorno de desarrollo integrado (IDE) con AWS CodeStar, puede seguir escribiendo y desarrollando el código en su entorno de preferencia. Los cambios que realice se incluyen en el proyecto de AWS CodeStar cada vez que confirme y envíe su código.

The screenshot shows an IDE interface with a code editor on the left and a commit message dialog on the right. The code editor displays the following HTML code:

```

48     <nav class="website-nav">
49         <ul>
50             <li><a class="home-link" href="https://aws.amazon.com/">
51             <li><a href="https://aws.amazon.com/what-is-cloud-comput
52             <li><a href="https://aws.amazon.com/solutions/">Services
53             <li><a href="https://aws.amazon.com/contact-us/">Contact
54         </ul>
55     </nav>
56 </header>
57
58     <div class="message">
59         <a class="twitter-link" href="http://twitter.com/home/?status=I
60         <div class="text">
61             <h1>Congratulations!</h1>
62             <h2>You just created a Node.js web application</h2>
63             <h3>And I made a change in Eclipse!</h3>
64         </div>
65     </div>
66 </div>
67
68 <footer>
69     <p class="footer-contents">Designed and developed with <a href="http

```

The commit message dialog shows the following information:

- Commit Message:** Updated index.html with a new h3
- Author:** Mary Major <mary_major@example.com>
- Committer:** Mary Major <mary_major@example.com>

Buttons for "Commit and Push..." and "Commit" are visible at the bottom of the dialog.

Temas

- [Usar AWS Cloud9 con AWS CodeStar](#)
- [Utilizar Eclipse con AWS CodeStar](#)
- [Utilizar Visual Studio con AWS CodeStar](#)

Usar AWS Cloud9 con AWS CodeStar

Puede utilizar AWS Cloud9 para realizar cambios en el código y desarrollar software en un proyecto de AWS CodeStar. AWS Cloud9 es un IDE online, al que se accede a través de un navegador web. El IDE ofrece una completa experiencia de edición de código, con soporte para varios lenguajes de

programación y depuradores de tiempo de ejecución, así como un terminal integrado. En segundo plano, una instancia de Amazon EC2 aloja un entorno de desarrollo de AWS Cloud9. Este entorno proporciona el IDE de AWS Cloud9 y el acceso a los archivos de códigos del proyecto de AWS CodeStar. Para obtener más información, consulte la Guía del usuario de [AWS Cloud9](#).

Puede utilizar tanto la consola de AWS CodeStar como la consola de AWS Cloud9 para crear entornos de desarrollo de AWS Cloud9 para proyectos que almacenen su código en CodeCommit. Para proyectos de AWS CodeStar que almacenan su código en GitHub, solo puede utilizar la consola de AWS Cloud9. En este tema se describe cómo utilizar ambas consolas.

Para utilizar AWS Cloud9, necesita:

- Un usuario de IAM que se haya añadido a un proyecto de AWS CodeStar como miembro del equipo.
- Si el proyecto de AWS CodeStar almacena su código fuente en CodeCommit, las credenciales de AWS para el usuario de IAM.

Temas

- [Crear un entorno de AWS Cloud9 para un proyecto](#)
- [Abrir un entorno de AWS Cloud9 para un proyecto](#)
- [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#)
- [Eliminar un entorno de AWS Cloud9 de un proyecto](#)
- [Uso de GitHub con AWS Cloud9](#)
- [Recursos adicionales](#)

Crear un entorno de AWS Cloud9 para un proyecto

Siga estos pasos para crear un entorno de desarrollo de AWS Cloud9 para un proyecto de AWS CodeStar.

1. Siga los pasos que se indican en [Creación de un proyecto](#) si desea crear un proyecto nuevo.
2. Abra el proyecto en la consola de AWS CodeStar. En la barra de navegación, seleccione IDE. Seleccione Crear entorno y, a continuación, utilice los pasos que se describen a continuación.

⚠ Important

Si el proyecto se encuentra en una región de AWS donde no se admita AWS Cloud9, no verá las opciones de AWS Cloud9 en la pestaña IDE de la barra de navegación. Sin embargo, puede utilizar la consola de AWS Cloud9 para crear un entorno de desarrollo, abrir el entorno nuevo y, a continuación, conectarlo al repositorio de AWS CodeCommit del proyecto. Omita los siguientes pasos y consulte [Creación de un entorno](#), [Apertura de un entorno](#), y la [Muestra de AWS CodeCommit](#) en la Guía del usuario de AWS Cloud9. Para obtener la lista de las regiones de AWS compatibles, consulte [AWS Cloud9](#) en la Referencia general de Amazon Web Services.

En Crear entorno de AWS Cloud9, personalice los valores predeterminados del proyecto.


1. Para cambiar el tipo predeterminado de instancia de Amazon EC2 con el fin de alojar el entorno, en Tipo de instancia, elija el tipo de instancia.
2. AWS Cloud9 utiliza Amazon Virtual Private Cloud (Amazon VPC) en su cuenta de AWS para comunicarse con la instancia. En función de cómo esté configurado Amazon VPC en su cuenta de AWS, realice una de las siguientes operaciones.

¿La cuenta tiene una VPC con al menos una subred en esa VPC?	¿La VPC que desea que utilice AWS Cloud9 es la VPC predeterminada de la cuenta?	¿La VPC tiene una única subred?	Haga lo siguiente
No	—	—	Si no existe una VPC, créela. Expanda Network settings (Ajustes de red). En Red (VPC), elija Crear nueva VPC y luego siga las instrucciones de la página. Para obtener más información, consulte Crear una Amazon VPC para AWS Cloud9 en la Guía del usuario de AWS Cloud9.

¿La cuenta tiene una VPC con al menos una subred en esa VPC?	¿La VPC que desea que utilice AWS Cloud9 es la VPC predeterminada de la cuenta?	¿La VPC tiene una única subred?	Haga lo siguiente
			Si existe una VPC, pero no hay ninguna subred, cree una. Expanda Ajustes de red. En Red (VPC), elija Crear subred y luego siga las instrucciones. Para obtener más información, consulte la página sobre cómo crear una subred para AWS Cloud9 en la Guía del usuario de AWS Cloud9.
Sí	Sí	Sí	Avance hasta el paso 4 de este procedimiento. (AWS Cloud9 usa la VPC predeterminada con su única subred).
Sí	Sí	No	En Subred, elija la subred que desee que AWS Cloud9 utilice en la VPC predeterminada previamente seleccionada.
Sí	No	Yes o No	En Red (VPC), elija la VPC que desee que AWS Cloud9 utilice. En Subred, elija la subred que desee que AWS Cloud9 utilice en esa VPC.

Para obtener más información, consulte [Configuración de Amazon VPC para entornos de desarrollo de AWS Cloud9](#) en la Guía del usuario de AWS Cloud9.

- Introduzca un Nombre del entorno y, si lo desea, añada una Descripción del entorno.

 Note

Los nombres de entorno deben ser único para cada usuario.

4. Para cambiar el periodo de tiempo predeterminado después del cual AWS Cloud9 cierra el entorno cuando no se ha utilizado, expanda Configuración de ahorro de costos y, a continuación, cambie la configuración.
5. Seleccione Crear entorno.

Para abrir el entorno, consulte [Abrir un entorno de AWS Cloud9 para un proyecto](#).

Puede utilizar estos pasos para crear más de un entorno para un proyecto. Por ejemplo, es posible que desee utilizar un entorno para trabajar en una parte del código y otro entorno para trabajar en la misma parte del código con diferentes ajustes.

Abrir un entorno de AWS Cloud9 para un proyecto

Siga estos pasos para abrir un entorno de desarrollo de AWS Cloud9 creado para un proyecto de AWS CodeStar.

1. Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación, elija IDE.

Important

Si el código fuente del proyecto se almacena en GitHub, no verá IDE en la barra de navegación. Sin embargo, puede utilizar la consola de AWS Cloud9 para abrir un entorno existente. Omita el resto de este procedimiento y consulte [Opening an Environment \(Apertura de un entorno\)](#) en la Guía del usuario de AWS Cloud9 y [Uso de GitHub con AWS Cloud9](#).

2. En Sus entornos de AWS Cloud9 o Entornos de AWS Cloud9 compartidos, elija Abrir IDE para el entorno que desea abrir.

Puede utilizar el IDE de AWS Cloud9 para empezar a trabajar con código en el repositorio de AWS CodeCommit del proyecto de inmediato. Para obtener más información, consulte [La ventana Entorno](#), [El editor, pestañas y paneles](#) y [El terminal](#) en la Guía del usuario de AWS Cloud9 y [Comandos básicos de Git](#) en la Guía del usuario de AWS CodeCommit.

Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto

Después de crear un entorno de desarrollo de AWS Cloud9 para un proyecto de AWS CodeStar, puede invitar a otros usuarios en su cuenta de AWS, incluidos miembros del equipo del proyecto,

para obtener acceso a ese mismo entorno. Esto resulta especialmente útil para la programación en parejas, en la que dos programadores se turnan para codificar y ofrecer consejos mientras comparten pantalla o mientras están sentados en la misma estación de trabajo. Los miembros del entorno puede utilizar el IDE de AWS Cloud9 compartido para ver cambios de código de cada miembro resaltados en el editor de código y enviar mensajes de texto a otros miembros mientras codifican.

Añadir un miembro del equipo a un proyecto no permite de manera automática que el miembro participe en cualquier entorno de desarrollo de AWS Cloud9 relacionado para el proyecto. Para invitar a un miembro del equipo de proyectos a que obtenga acceso a un entorno para un proyecto, debe determinar el rol de acceso del miembro al entorno correcto, aplicar políticas administradas de AWS al usuario e invitar al usuario a su entorno. Para obtener más información, consulte [Acerca de los roles de acceso de los miembros del entorno](#) e [Invitar a un usuario de IAM a su entorno](#) en la Guía del usuario de AWS Cloud9.

Cuando invita a un miembro del equipo de un proyecto para que obtenga acceso a un entorno para un proyecto, la consola de AWS CodeStar muestra el entorno a ese miembro del equipo. El entorno se muestra en la lista Entornos compartidos en la pestaña IDE en la consola de AWS CodeStar del proyecto. Para mostrar esta lista, el miembro del equipo tiene que abrir el proyecto en la consola y, a continuación, elegir IDE en la barra de navegación.

Important

Si el código fuente del proyecto se almacena en GitHub, no verá IDE en la barra de navegación. Sin embargo, puede utilizar la consola de AWS Cloud9 para invitar a otros usuarios en su cuenta de AWS, incluidos los miembros del equipo del proyecto, a que obtenga acceso a un entorno. Para ello, consulte [Uso de GitHub con AWS Cloud9](#) en esta guía y consulte [Acerca de los roles de acceso de los miembros del entorno](#) e [Invitar a un usuario de IAM a su entorno](#) en la Guía del usuario de AWS Cloud9.

También puede invitar a un usuario que no es un miembro del equipo de proyectos a que obtenga acceso a un entorno. Por ejemplo, es posible que desee que un usuario trabaje en el código de un proyecto pero no tiene ningún otro acceso a ese proyecto. Para invitar a este tipo de usuarios, consulte [Acerca de los roles de acceso de los miembros del entorno](#) e [Invitar a un usuario de IAM a su entorno](#) en la Guía del usuario de AWS Cloud9. Cuando invita a un usuario que no es miembro del equipo de proyectos para que obtenga acceso a un entorno para un proyecto, ese usuario puede

utilizar la consola de AWS Cloud9 para obtener acceso al entorno. Para obtener más información, consulte [Abrir un entorno](#) en la Guía del usuario de AWS Cloud9.

Eliminar un entorno de AWS Cloud9 de un proyecto

Cuando elimina un proyecto y todos sus recursos de AWS de AWS Cloud9, todos los entornos de desarrollo de AWS CodeStar creados con la consola de AWS CodeStar también se eliminan y no se pueden recuperar. Puede eliminar un entorno de desarrollo de un proyecto sin eliminar el proyecto.

1. Con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación, elija IDE.

Important

Si el código fuente del proyecto se almacena en GitHub, no verá IDE en la barra de navegación. Sin embargo, puede utilizar la consola de AWS Cloud9 para eliminar un entorno de desarrollo. Omita el resto de este procedimiento y consulte [Eliminación de un entorno](#) en la Guía del usuario de AWS Cloud9.

2. Elija el entorno que desee eliminar en los entornos de Cloud9 y seleccione Eliminar.
3. Escriba **delete** para confirmar la eliminación del entorno de desarrollo y, a continuación, seleccione Eliminar.

Warning

Una vez que se ha eliminado, no es posible recuperar un entorno de desarrollo. Todos los cambios en el código sin confirmar en el entorno se perderán.

Uso de GitHub con AWS Cloud9

Para proyectos de AWS CodeStar que tienen su código fuente almacenado en GitHub, la consola de AWS CodeStar no permite trabajar con entornos de desarrollo de AWS Cloud9 directamente. Sin embargo, puede utilizar la consola de AWS Cloud9 para trabajar con código fuente en repositorios de GitHub.

1. Utilice la consola de AWS Cloud9 para crear un entorno de desarrollo de AWS Cloud9. Para obtener más información, consulte [Creación de un entorno](#) en la Guía del usuario de AWS Cloud9.

2. Utilice la consola de AWS Cloud9 para abrir un entorno de desarrollo. Para obtener más información, consulte [Apertura de un entorno](#) en la Guía del usuario de AWS Cloud9.
3. En el IDE, utilice una sesión de terminal para conectarse al repositorio de GitHub (un proceso conocido como clonación). Si una sesión de terminal no se está ejecutando, en la barra de menús en el IDE, elija Ventana, Terminal nuevo). Para los comandos que se va a utilizar para clonar el repositorio GitHub, consulte [Clonación de un repositorio](#) en el sitio web de ayuda de GitHub.

Para ir a la página principal del repositorio GitHub, con el proyecto abierto en la consola de AWS CodeStar, en la barra de navegación lateral, elija Código.

4. Utilice la ventana Entorno y las pestañas del editor en el IDE para ver, cambiar y guardar código. Para obtener más información, consulte [La ventana Entorno](#) y [El editor, pestañas y paneles](#) en la Guía del usuario de AWS Cloud9.
5. Utilice Git en la sesión de terminal del IDE para enviar los cambios al repositorio y para recibir periódicamente los cambios en el código que realicen otras personas del repositorio. Para obtener más información, consulte [Envío a un repositorio remoto](#) y [Obtención de un repositorio remoto](#) en el sitio web de ayuda de GitHub. Para ver los comandos de Git, consulte [Hoja informativa](#) en el sitio web de ayuda de GitHub.

Note

Para evitar que Git le solicite sus credenciales de GitHub cada vez que inserte o extraiga código del repositorio, puede utilizar un ayudante de credenciales. Para obtener más información, consulte la sección [Almacenamiento en caché de la contraseña de GitHub en Git](#) en el sitio web de ayuda de GitHub.

Recursos adicionales

Para obtener más información acerca del uso de AWS Cloud9, consulte la siguiente Guía de usuario de AWS Cloud9:

- [Tutorial](#)
- [Trabajo con entornos](#)
- [Uso del IDE](#)
- [Ejemplos](#)

Utilizar Eclipse con AWS CodeStar

Puede utilizar Eclipse para realizar cambios en el código y desarrollar software en un proyecto de AWS CodeStar. Puede editar el código de su proyecto de AWS CodeStar con Eclipse y, a continuación, confirmar e insertar los cambios en el repositorio de origen del proyecto de AWS CodeStar.

Note

La información de este tema se aplica únicamente a proyectos de AWS CodeStar que almacenan su código fuente en CodeCommit. Si su proyecto de AWS CodeStar almacena su código fuente en GitHub, puede utilizar una herramienta como EGit for Eclipse. Para obtener más información, consulte la [EGit Documentation](#) en el sitio web de EGit.

Si el proyecto de AWS CodeStar almacena su código fuente en CodeCommit, debe instalar una versión del AWS Toolkit for Eclipse que admita AWS CodeStar. Asimismo, debe ser un miembro del equipo del proyecto de AWS CodeStar, con rol de propietario o de colaborador.

Para utilizar Eclipse, también necesita:

- Un usuario de IAM que se haya añadido a un proyecto de AWS CodeStar como miembro del equipo.
- Si el proyecto de AWS CodeStar almacena su código fuente en CodeCommit, las [credenciales de Git](#) (credenciales de inicio de sesión) para el usuario de IAM.
- Permisos suficientes para instalar Eclipse y AWS Toolkit for Eclipse en su equipo local.

Temas

- [Paso 1: Instalar AWS Toolkit for Eclipse](#)
- [Paso 2: Importar el proyecto de AWS CodeStar a Eclipse](#)
- [Paso 3: Editar el código del proyecto de AWS CodeStar en Eclipse](#)

Paso 1: Instalar AWS Toolkit for Eclipse

El Kit de herramientas para Eclipse es un paquete de software que puede añadir a Eclipse. Se instala y administra de la misma forma que otros paquetes de software en Eclipse. El kit de herramientas de AWS CodeStar forma parte del Kit de herramientas para Eclipse.

Para instalar el Kit de herramientas para Eclipse con el módulo de AWS CodeStar

1. Instale Eclipse en el equipo local. Las versiones compatibles de Eclipse son Luna, Marte y Neon.
2. Descargue e instale el Kit de herramientas para Eclipse. Para obtener más información, consulte la [Guía de introducción a AWS Toolkit for Eclipse](#).
3. En Eclipse, seleccione Help (Ayuda) y, a continuación, elija Install New Software (Instalar software nuevo).
4. En Available Software (Software disponible), seleccione Add (Añadir).
5. En Add Repository (Añadir repositorio), seleccione Archive (Archivado), busque la ubicación en la que guardó el archivo .zip y abra el archivo. Deje el campo Name (Nombre) en blanco y elija OK (Aceptar).
6. En Softwares disponibles, elija Seleccionar todos para seleccionar tanto las Herramientas de administración principales de AWS como las Herramientas para desarrolladores y, a continuación, elija Siguiente.
7. En Install Details (Detalles de la instalación), elija Next (Siguiente).
8. En Review Licenses (Revisar licencias), lea los acuerdos de licencia. Elija I accept the terms of the license agreement (Acepto los términos del acuerdo de licencia) y elija Finish (Finalizar). Reinicie Eclipse.

Paso 2: Importar el proyecto de AWS CodeStar a Eclipse

Una vez instalado el Kit de herramientas para Eclipse, podrá importar proyectos de AWS CodeStar y editar, confirmar y enviar código desde el IDE.

Note

Puede añadir varios proyectos de AWS CodeStar a un único espacio de trabajo en Eclipse; pero si lo hace, debe actualizar sus credenciales de proyecto al cambiar de un proyecto a otro.

Para importar un proyecto de AWS CodeStar

1. En el menú de AWS, seleccione Importar proyecto de AWS CodeStar. También puede elegir File (Archivo) y luego elegir Import (Importar). En Select, expanda AWS y, a continuación, seleccione AWS CodeStar Project.

Elija Siguiente.

2. En Selección de proyectos de AWS CodeStar, elija su perfil de AWS y la región de AWS en la que se aloje el proyecto de AWS CodeStar. Si no tiene un perfil de AWS configurado con una clave de acceso y una clave secreta en su equipo, seleccione Configurar cuentas de AWS y siga las instrucciones.

En Seleccionar proyecto y repositorio de AWS CodeStar, seleccione su proyecto de AWS CodeStar. En Configurar credenciales de Git, escriba las credenciales de inicio de sesión que ha generado para obtener acceso al repositorio del proyecto. (Si no dispone de credenciales de Git, consulte [Introducción](#). Elija Siguiente.

AWS CodeStar Project Selection

Select the AWS CodeStar project you want to checkout from the remote host.

Select AWS account and region:

Select Account: [Configure AWS accounts...](#)

Select Region:

Select AWS CodeStar project and repository:

Project Name	Project ID	Project Description
My First Project	my-first-projec	AWS CodeStar created project

Select repository:

Configure Git credentials:

You can manually copy and paste Git credentials for AWS CodeCommit below. Alternately, you can import them from a downloaded .csv file. To learn how to generate Git credentials, see [Create Git Credentials for HTTPS Connections to AWS CodeCommit](#).

User name:

Password:

Show password

3. Todas las ramificaciones del repositorio del proyecto están seleccionadas de forma predeterminada. Si no desea importar una o varias ramificaciones, desmarque las casillas y, a continuación, seleccione Siguiente.
4. En Local Destination (Destino local), elija un destino en el cual el asistente de importación creará el repositorio local en su equipo y luego seleccione Finish (Finalizar).
5. En Project Explorer (Explorador de proyectos), expanda el árbol del proyecto para buscar los archivos del proyecto de AWS CodeStar.

Paso 3: Editar el código del proyecto de AWS CodeStar en Eclipse

Después de importar un proyecto de AWS CodeStar a un espacio de trabajo de Eclipse, puede editar el código del proyecto, guardar los cambios y confirmar y enviar su código al repositorio de origen del proyecto. Este es el mismo proceso que debe seguir para cualquier repositorio Git que emplee el complemento EGit para Eclipse. Para obtener más información, consulte la [Guía del usuario de EGit](#) en el sitio web de Eclipse.

Para editar el código del proyecto y realizar el primer envío al repositorio de origen para un proyecto de AWS CodeStar

1. En Project Explorer (Explorador de proyectos), expanda el árbol del proyecto para buscar los archivos del proyecto de AWS CodeStar.
2. Edite uno o varios archivos y guarde los cambios.
3. Cuando esté preparado para confirmar los cambios, abra el menú contextual para dicho archivo, elija Team (Equipo) y luego seleccione Commit (Confirmar).

Puede omitir este paso si la ventana Git Staging (Espacio provisional de Git) está abierta en la vista del proyecto.

4. En Git Staging (Espacio provisional de Git), introduzca los cambios moviendo los archivos modificados a Staged Changes (Cambios almacenados). Escriba un mensaje de confirmación en Commit Message (Confirmar mensaje) y, a continuación, seleccione Commit and Push (Confirmar y enviar).

La información de este tema se aplica únicamente a proyectos de AWS CodeStar que almacenan su código fuente en CodeCommit. Si su proyecto de AWS CodeStar almacena su código fuente en GitHub, puede utilizar una herramienta como GitHub Extension for Visual Studio. Para obtener información adicional, consulte la página [Overview](#) en el sitio web de GitHub Extension for Visual Studio y [Getting Started with GitHub for Visual Studio](#) en el sitio web de GitHub.

Para utilizar Visual Studio para editar el código en el repositorio de origen de un proyecto de AWS CodeStar, debe instalar una versión de AWS Toolkit for Visual Studio que admita AWS CodeStar. Debe ser miembro del equipo del proyecto de AWS CodeStar, con rol de propietario o de colaborador.

Para utilizar Visual Studio, también necesita:

- Un usuario de IAM que se haya añadido a un proyecto de AWS CodeStar como miembro del equipo.
- Las credenciales de AWS para el usuario de IAM (por ejemplo, su clave de acceso y su clave secreta).
- Permisos suficientes para instalar Visual Studio y AWS Toolkit for Visual Studio en su equipo local.

El Kit de herramientas para Visual Studio es un paquete de software que puede agregar a Visual Studio. Se instala y administra de la misma forma que otros paquetes de software en Visual Studio.

Para instalar el Kit de herramientas de Visual Studio con el módulo de AWS CodeStar y configurar el acceso al repositorio del proyecto

1. Instale Visual Studio en su equipo local.
2. Descargue e instale el Kit de herramientas de Visual Studio y guarde el archivo .zip en una carpeta local o en un directorio local. En la página Introducción al AWS Toolkit for Visual Studio, escriba o importe sus credenciales de AWS y, a continuación, seleccione Guardar y cerrar.
3. En Visual Studio, abra Team Explorer. En Proveedores de servicios alojados, busque CodeCommit y seleccione Conectar.
4. En Manage Connections, seleccione Clone. Elija el repositorio del proyecto y la carpeta de su equipo local en la que desea clonar el repositorio y, a continuación, seleccione OK (Aceptar).

5. Cuando se le pida que cree credenciales de Git, seleccione Yes. El conjunto de herramientas intenta crear credenciales en su nombre. Guarde el archivo de credenciales en un lugar seguro. Esta es la única oportunidad que tendrá para guardar estas credenciales. Si el conjunto de herramientas no puede crear credenciales en su nombre, o si selecciona No, debe crear y proporcionar sus propias credenciales de Git. Para obtener más información, consulte [Para configurar el equipo para confirmar los cambios \(usuario de IAM\)](#) o siga las instrucciones online.

Cuando haya terminado de clonar el proyecto, ya puede empezar a editar el código en Visual Studio y confirmar y enviar los cambios al repositorio del proyecto en CodeCommit.

Cambiar los recursos de AWS en un proyecto de AWS CodeStar

Después de crear un proyecto en AWS CodeStar, puede cambiar el conjunto predeterminado de recursos de AWS que añade AWS CodeStar al proyecto.

Cambios de recursos admitidos

En la siguiente tabla se enumeran los cambios admitidos en los recursos de AWS predeterminados en un proyecto de AWS CodeStar.

Cambio	Notas
Añadir una etapa a AWS CodePipeline.	Consulte Añadir una etapa a AWS CodePipeline .
Cambiar la configuración del entorno de Elastic Beanstalk.	Consulte Cambiar la configuración del entorno de AWS Elastic Beanstalk .
Cambiar un código o la configuración de la función de AWS Lambda, su rol de IAM relacionado o su API en Amazon API Gateway.	Consulte Cambiar una función de AWS Lambda en código fuente .
Añadir un recurso a un proyecto de AWS Lambda y expandir permisos para crear y obtener acceso al nuevo recurso.	Consulte Añadir un recurso a un proyecto .
Añadir el desvío de tráfico con CodeDeploy para una función de AWS Lambda.	Consulte Desviar el tráfico para un proyecto de AWS Lambda .

Cambio	Notas
Añadir compatibilidad con AWS X-Ray	Consulte Habilitar el seguimiento para un proyecto .
Edite el archivo buildspec.yml del proyecto para añadir una etapa de compilación de pruebas de unidad y que AWS CodeBuild se ejecute.	Consulte Paso 7: añadir una prueba de unidad al servicio web en el tutorial de proyecto sin servidor.
Añada su propio rol de IAM a su proyecto.	Consulte Añadir un rol de IAM a un proyecto .
Cambiar una definición de rol de IAM.	Para roles definidos en la pila de la aplicación. No puede cambiar los roles definidos en la cadena de herramientas o en las pilas de AWS CloudFormation.
Modifique su proyecto de Lambda para añadir un punto de conexión.	
Modifique su proyecto de EC2 para añadir un punto de conexión.	
Modifique su proyecto de Elastic Beanstalk para añadir un punto de conexión.	
Edite su proyecto para añadir una etapa Prod y un punto de conexión.	Consulte Añadir una etapa Prod y un punto de conexión a un proyecto .
Utilice de manera segura los parámetros de SSM en un proyecto de AWS CodeStar.	Consulte the section called “Utilice de forma segura los parámetros de SSM en un proyecto AWS CodeStar” .

No se admiten los cambios siguientes.

- Cambiar a otro destino de implementación (por ejemplo, implementar en AWS Elastic Beanstalk en lugar de en AWS CodeDeploy).
- Añadir un nombre de punto de conexión web sencillo.

- Cambiar el nombre del repositorio de CodeCommit (para un proyecto de AWS CodeStar conectado a CodeCommit).
- Para un proyecto de AWS CodeStar conectado a GitHub, desconecte el repositorio de GitHub y, a continuación, vuelva a conectar el repositorio a dicho proyecto o conecte cualquier otro repositorio a dicho proyecto. Puede utilizar la consola de CodePipeline (no la consola de AWS CodeStar) para desconectar y volver a conectar a GitHub en la etapa Origen de una canalización. No obstante, si vuelve a conectar la etapa Origen a un repositorio de GitHub distinto, entonces en el panel de AWS CodeStar del proyecto, la información en los iconos Repositorio y Problemas podría ser incorrecta o estar obsoleta. La desconexión del repositorio de GitHub no elimina la información de dicho repositorio de los iconos del historial de confirmación y de los problemas de GitHub en el panel del proyecto de AWS CodeStar. Para eliminar esta información, utilice el sitio web de GitHub para deshabilitar el acceso a GitHub desde el proyecto de AWS CodeStar. Para revocar el acceso, en el sitio web de GitHub, utilice la sección Authorized OAuth Apps (Aplicaciones de OAuth autorizadas) de la página de configuración de su perfil de cuenta de GitHub.
- Desconecte el repositorio de CodeCommit (para un proyecto de AWS CodeStar conectado a CodeCommit) y, a continuación, vuelva a conectar el repositorio a dicho proyecto o conecte cualquier otro repositorio a dicho proyecto.

Añadir una etapa a AWS CodePipeline

Puede añadir una nueva etapa a la canalización que AWS CodeStar crea en un proyecto. Para obtener más información, consulte [Editar una canalización en AWS CodePipeline](#) en la Guía del usuario de AWS CodePipeline.

Note

Si la nueva etapa depende de algún recurso de AWS que AWS CodeStar no ha creado, la canalización se podría interrumpir. Esto se debe a que el rol de IAM que ha creado AWS CodeStar para AWS CodePipeline puede no tener acceso a esos recursos de forma predeterminada.

Para intentar dar a AWS CodePipeline acceso a recursos de AWS que AWS CodeStar no ha creado, le recomendamos que cambie el rol de IAM que AWS CodeStar ha creado. No es compatible porque AWS CodeStar podría eliminar los cambios de su rol de IAM al llevar a cabo las comprobaciones periódicas de actualizaciones en el proyecto.

Cambiar la configuración del entorno de AWS Elastic Beanstalk

Puede cambiar la configuración de un entorno de Elastic Beanstalk que AWS CodeStar crea en un proyecto. Por ejemplo, es posible que desee cambiar el entorno de Elastic Beanstalk predeterminado del proyecto de AWS CodeStar de Instancia única a Equilibrio de carga. Para ello, edite el archivo `template.yml` en el repositorio del proyecto. Es posible que también necesite cambiar los permisos para los roles de trabajo de su proyecto. Después de insertar el cambio de plantilla, AWS CodeStar y AWS CloudFormation aprovisionan los recursos por usted.

Para obtener más información sobre la edición del archivo `template.yml`, consulte [Cambiar recursos de aplicaciones con el archivo `Template.yml`](#). Para obtener más información acerca de los entornos de Elastic Beanstalk, consulte [Consola de administración del entorno de AWS Elastic Beanstalk](#) en la Guía para desarrolladores de AWS Elastic Beanstalk.

Cambiar una función de AWS Lambda en código fuente

Puede cambiar el código o la configuración de una función de Lambda, o su rol de IAM o la API de API Gateway que AWS CodeStar crea en un proyecto. Para ello, recomendamos que utilice el AWS Serverless Application Model (AWS SAM) junto con el archivo `template.yaml` en el repositorio de CodeCommit del proyecto. Este archivo `template.yaml` define el nombre de la función, el controlador, el tiempo de ejecución, el rol de IAM y la API en API Gateway. Para obtener más información, consulte la sección [How to Create Serverless Applications Using AWS SAM \(Cómo crear aplicaciones sin servidor mediante SAM\)](#) en el sitio web de GitHub.

Habilitar el seguimiento para un proyecto

AWS X-Ray ofrece el seguimiento, que puede utilizar para analizar el comportamiento del rendimiento de las aplicaciones distribuidas (por ejemplo, las latencias de los tiempos de respuesta). Después de agregar los seguimientos a su proyecto de AWS CodeStar, puede utilizar la consola de AWS X-Ray para ver vistas de la aplicación y tiempos de respuesta.

Note

Puede utilizar estos pasos para los siguientes proyectos, creados con los siguientes cambios admitidos por el proyecto:

- Cualquier proyecto de Lambda.

- Para proyectos de Amazon EC2 o Elastic Beanstalk creados después del 3 de agosto de 2018, AWS CodeStar ha provisionado un archivo `/template.yml` en el repositorio del proyecto.

Cada plantilla de proyecto de AWS CodeStar incluye un archivo de AWS CloudFormation que modela las dependencias de tiempo de ejecución de AWS de su aplicación, como las tablas de base de datos y las funciones de Lambda. Este archivo está almacenado en el repositorio de origen en el archivo `/template.yml`.

Puede modificar este archivo para agregar el seguimiento añadiendo el recurso AWS X-Ray en la sección `Resources`. A continuación, modifique los permisos de IAM del proyecto para permitir que AWS CloudFormation cree el recurso. Para obtener más información sobre los elementos de la plantilla y el formateo, consulte [Referencia de tipos de recursos de AWS](#).

Estos son los pasos generales a seguir para personalizar la plantilla.

1. [Paso 1: Editar el rol de trabajador en IAM para seguimiento](#)
2. [Paso 2: Modificar el archivo `template.yml` para el seguimiento](#)
3. [Paso 3: Confirmar y enviar el cambio de la plantilla para el seguimiento](#)
4. [Paso 4: Monitorizar la actualización de la pila de AWS CloudFormation para el seguimiento](#)

Paso 1: Editar el rol de trabajador en IAM para seguimiento

Debe haber iniciado sesión como administrador para llevar a cabo los pasos 1 y 4. En este paso se muestra un ejemplo de edición de permisos para un proyecto de Lambda.

Note

Puede omitir este paso si su proyecto se ha provisionado con una política de límite de permisos.

Para proyectos creados después del 6 de diciembre de 2018 PDT, AWS CodeStar provisionó su proyecto con una política de límite de permisos.

1. Inicie sesión en la AWS Management Console y abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.

2. Cree un proyecto o elija un proyecto existente con un `template.yml` file y, a continuación, abra la página Recursos del proyecto.
3. En los Recursos del proyecto, localice el rol de IAM creado para el rol de CodeStarWorker/Lambda en la lista de recursos. El nombre del rol sigue el siguiente formato: `role/CodeStarWorker-Project_name-lambda-Function_name`. Elija el ARN para el rol.
4. El rol se abrirá en la consola de IAM. Seleccione Attach policies (Asociar políticas). Busque la política `AWSXrayWriteOnlyAccess`, seleccione la casilla situada junto a la misma y, luego, elija Attach Policy (Asociar política).

Paso 2: Modificar el archivo `template.yml` para el seguimiento

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Elija el proyecto sin servidor y, a continuación, abra la página Code (Código). En la parte superior del repositorio, localice y edite el archivo `template.yml`. En Resources, pegue el recurso en la sección Properties.

Tracing: Active

En este ejemplo se muestra una plantilla modificada:

```
Resources:
  GetHelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.get
      Runtime: nodejs4.3
      Tracing: Active # Enable X-Ray tracing for the function
    Role:
      Fn::ImportValue:
        !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
```

Paso 3: Confirmar y enviar el cambio de la plantilla para el seguimiento

- Confirme y envíe los cambios realizados en el archivo `template.yml`.

Note

Esto iniciará la canalización. Si confirma los cambios antes de actualizar los permisos de IAM, se iniciará la canalización, la actualización de la pila de AWS CloudFormation se topará con errores y la actualización de la pila se revertirá. Si esto ocurre, corrija los permisos y, a continuación, reinicie la canalización.

Paso 4: Monitorizar la actualización de la pila de AWS CloudFormation para el seguimiento

1. La actualización de la pila de AWS CloudFormation se inicia cuando la canalización del proyecto inicia la etapa de implementación. Para ver el estado de la actualización de la pila, en el panel de AWS CodeStar, elija la etapa AWS CloudFormation en la canalización.

Si la actualización de la pila en AWS CloudFormation devuelve errores, consulte las instrucciones de la solución de problemas [AWS CloudFormation: Restauración de creación de pila para permisos ausentes](#). Si faltan permisos del rol de trabajador, edite la política asociada al rol de trabajador de Lambda del proyecto. Consulte [Paso 1: Editar el rol de trabajador en IAM para seguimiento](#).

2. Utilice el panel para ver la correcta finalización de la canalización. El seguimiento ya está habilitado en la aplicación.
3. Compruebe que el seguimiento está habilitado revisando los detalles de la función en la consola de Lambda.
4. Elija el punto de conexión de la aplicación para el proyecto. Se realiza un seguimiento de esta interacción con la aplicación. Puede ver la información de seguimiento en la consola de AWS X-Ray.

Trace list					
ID	Age	Method	Response	Response time	URL
...315e2d41	4.7 min		200	270 ms	
...88c0c37c	12.8 sec		200	23.0 ms	

Añadir un recurso a un proyecto

Cada plantilla de AWS CodeStar para todos los proyectos dispone de un archivo de AWS CloudFormation que modela las dependencias de tiempo de ejecución de AWS de su aplicación, como las tablas de base de datos y las funciones de Lambda. Este archivo está almacenado en el repositorio de origen en el archivo `/template.yml`.

Note

Puede utilizar estos pasos para los siguientes proyectos, creados con los siguientes cambios admitidos por el proyecto:

- Cualquier proyecto de Lambda.
- Para proyectos de Amazon EC2 o Elastic Beanstalk creados después del 3 de agosto de 2018, AWS CodeStar ha aprovisionado un archivo `/template.yml` en el repositorio del proyecto.

Puede modificar este archivo añadiendo recursos de AWS CloudFormation en la sección `Resources`. Al modificar el archivo `template.yml` permite que AWS CodeStar y AWS CloudFormation puedan añadir el nuevo recurso al proyecto. Algunos recursos requieren que añada otros permisos a la política para el rol de trabajador de CloudFormation del proyecto. Para obtener más información sobre los elementos de la plantilla y el formateo, consulte [Referencia de tipos de recursos de AWS](#).

Después de determinar qué recursos debe agregar a su proyecto, estos son los pasos generales a seguir para personalizar una plantilla. Para obtener una lista de recursos de AWS CloudFormation y las propiedades necesarias, consulte [Referencia de tipos de recursos de AWS](#).

1. [Paso 1: editar el rol de trabajador de CloudFormation en IAM](#) (si es necesario)
2. [Paso 2: modificar el archivo `template.yml`](#)
3. [Paso 3: confirmar y enviar el cambio en la plantilla](#)
4. [Paso 4: Monitorizar la actualización de la pila de AWS CloudFormation](#)
5. [Paso 5: Añadir permisos a nivel de recursos con una política insertada](#)

Siga los pasos en esta sección para modificar la plantilla del proyecto de AWS CodeStar para añadir un recurso y, a continuación, expanda los permisos del rol de trabajador de CloudFormation del proyecto en IAM. En este ejemplo, el recurso [AWS::SQS::Queue](#) se añade al archivo `template.yml`. Este cambio lanza una respuesta automatizada en AWS CloudFormation que añade una cola de Amazon Simple Queue Service al proyecto.

Paso 1: editar el rol de trabajador de CloudFormation en IAM

Debe haber iniciado sesión como administrador para seguir los pasos 1 y 5.

Note

Puede omitir este paso si su proyecto se ha provisionado con una política de límite de permisos.

Para proyectos creados después del 6 de diciembre de 2018 PDT, AWS CodeStar provisionó su proyecto con una política de límite de permisos.

1. Inicie sesión en la AWS Management Console y abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Cree un proyecto o elija un proyecto existente con un `template.yml` file y, a continuación, abra la página Project resources (Recursos del proyecto).
3. En Project Resources (Recursos del proyecto), localice el rol de IAM creado para el rol de CodeStarWorker/AWS CloudFormation en la lista de recursos. El nombre del rol sigue el siguiente formato: `role/CodeStarWorker-Project_name-CloudFormation`.
4. El rol se abrirá en la consola de IAM. En la pestaña Permissions (Permisos), en Inline Policies (Políticas insertadas), expanda la fila de su política de rol de servicio, y elija Edit Policy (Editar política).
5. Elija la pestaña JSON para editar la política.

Note

La política asociada al rol de trabajador es `CodeStarWorkerCloudFormationRolePolicy`.

6. En el campo JSON, añada la siguiente instrucción de la política al elemento Statement.

```
{
  "Action": [
    "sqs:CreateQueue",
    "sqs>DeleteQueue",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes",
    "sqs:ListQueues",
    "sqs:GetQueueUrl"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
```

7. Elija Review policy (Revisar política) para asegurarse de que la política no contiene errores y, a continuación, elija Save changes (Guardar cambios).

Paso 2: modificar el archivo template.yml

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Elija el proyecto sin servidor y, a continuación, abra la página Code (Código). En la parte superior del repositorio, anote la ubicación de `template.yml`.
3. Utilice un IDE, la consola o la línea de comandos en el repositorio local para editar el archivo `template.yml` en el repositorio. Pegue el recurso en la sección Resources. En este ejemplo, cuando se copia el siguiente texto, se agrega la sección Resources.

```
Resources:
  TestQueue:
    Type: AWS::SQS::Queue
```

En este ejemplo se muestra una plantilla modificada:

```
Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
    Events:
      GetEvent:
        Type: Api
        Properties:
          Path: /
          Method: get
      PostEvent:
        Type: Api
        Properties:
          Path: /
          Method: post
  TestQueue:
    Type: AWS::SQS::Queue
```

Paso 3: confirmar y enviar el cambio en la plantilla

- Confirme y envíe los cambios realizados en el archivo `template.yml` que ha guardado en el paso 2.

Note

Esto iniciará la canalización. Si confirma los cambios antes de actualizar los permisos de IAM, se iniciará la canalización y la actualización de la pila de AWS CloudFormation se topará con errores por lo que la actualización de la pila se revertirá. Si esto ocurre, corrija los permisos y, a continuación, reinicie la canalización.

Paso 4: Monitorizar la actualización de la pila de AWS CloudFormation

1. Cuando la canalización del proyecto inicie la fase de implementación, se iniciará la actualización de la pila de AWS CloudFormation. Puede elegir la etapa AWS CloudFormation de la canalización en el panel de AWS CodeStar para ver la actualización de la pila.

Solución de problemas:

La actualización de la pila falla si faltan los permisos a nivel de recursos necesarios. Vea el estado de error en la vista del panel de AWS CodeStar para la canalización del proyecto.

Elija el enlace CloudFormation en la etapa de implementación de su canalización para solucionar el error en la consola de AWS CloudFormation. En la consola, en la lista Events (Eventos), seleccione su proyecto para ver los detalles de creación de la pila. Hay un mensaje que contiene los detalles del error. En este ejemplo, falta el permiso `sqs:CreateQueue`.

08:37:11 UTC-0700	UPDATE_ROLLBACK_COMPLETE	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
08:37:11 UTC-0700	DELETE_COMPLETE	AWS::SQS::Queue	TestQueue	
08:37:09 UTC-0700	UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	
08:37:06 UTC-0700	UPDATE_COMPLETE	AWS::Lambda::Function	HelloWorld	
08:37:03 UTC-0700	UPDATE_ROLLBACK_IN_PROGRESS	AWS::CloudFormation::Stack	awscodestar-dk-sqs-red-lambda	The following resource(s) failed to create: [TestQueue]. The following resource(s) failed to update: [HelloWorld].
08:37:02 UTC-0700	UPDATE_FAILED	AWS::Lambda::Function	HelloWorld	Resource update cancelled
08:37:01 UTC-0700	CREATE_FAILED	AWS::SQS::Queue	TestQueue	API: sqs:CreateQueue Access to the resource https://sqs.us-west-2.amazonaws.com/ is denied.
08:37:01 UTC-0700	CREATE_IN_PROGRESS	AWS::SQS::Queue	TestQueue	

Añada los permisos que faltan editando la política asociada al rol de trabajador de AWS CloudFormation del proyecto. Consulte [Paso 1: editar el rol de trabajador de CloudFormation en IAM](#).

- Después de ejecutar correctamente la canalización, los recursos se crean en la pila de AWS CloudFormation. En la lista Resources (Recursos) en AWS CloudFormation, visualice el recurso creado para el proyecto. En este ejemplo, la cola TestQueue aparece en la sección Resources (Recursos).

La URL de la cola está disponible en AWS CloudFormation. La URL de la cola tiene este formato:

```
https://{REGION_ENDPOINT}/queue. |api-domain|/{YOUR_ACCOUNT_NUMBER}/
{YOUR_QUEUE_NAME}
```

Para obtener más información, consulte la sección sobre el [envío de un mensaje de Amazon SQS](#), sobre la [entrada de un mensaje de la cola de Amazon SQS](#) y sobre la [eliminación de un mensaje de la cola de Amazon SQS](#).

Paso 5: Añadir permisos a nivel de recursos con una política insertada

Otorgue a los miembros del equipo acceso a su nuevo recurso añadiendo la política insertada adecuada al rol del usuario. No todos los recursos requieren permisos. Para seguir los siguientes pasos, debe iniciar sesión en la consola como usuario raíz, usuario administrador en la cuenta, usuario de IAM o usuario federado con la política administrada `AdministratorAccess` asociada o equivalente.

Para utilizar el editor de política de JSON para crear una política

1. Inicie sesión en la AWS Management Console y abra la consola de IAM en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegación de la izquierda, elija Políticas (Políticas).

Si es la primera vez que elige Políticas (Políticas), aparecerá la página Welcome to Managed Policies (Bienvenido a políticas administradas). Elija Get Started (Comenzar).

3. En la parte superior de la página, seleccione Crear política.
4. En la sección Editor de políticas, seleccione la opción JSON.
5. Ingrese el siguiente documento de política JSON:

```
{
  "Action": [
    "sqs:CreateQueue",
    "sqs>DeleteQueue",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes",
    "sqs:ListQueues",
    "sqs:GetQueueUrl"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
```

6. Elija Siguiente.

Note

Puede alternar entre las opciones Visual y JSON del editor en todo momento. No obstante, si realiza cambios o selecciona Siguiente en la opción Visual del editor, es posible que IAM reestructure la política, con el fin de optimizarla para el editor visual. Para obtener más información, consulte [Reestructuración de política](#) en la Guía del usuario de IAM.

7. En la página Revisar y crear, escriba el Nombre de la política y la Descripción (opcional) para la política que está creando. Revise los Permisos definidos en esta política para ver los permisos que concede la política.
8. Elija Create Policy (Crear política) para guardar la nueva política.

Añadir un rol de IAM a un proyecto

A partir del 6 de diciembre de 2018 PDT, puede definir sus propios roles y políticas en la pila de la aplicación (template.yml). Para mitigar los riesgos del escalado de privilegios y acciones destructivas, debe establecer el límite de permisos específico del proyecto para cada entidad de IAM que cree. Si tiene un proyecto de Lambda con varias funciones, una práctica recomendada consiste en crear un rol de IAM para cada función.

Para añadir un rol de IAM a su proyecto

1. Edite el archivo `template.yml` para su proyecto.
2. En la sección `Resources:`, añada su recurso de IAM, utilizando el formato del siguiente ejemplo:

```
SampleRole:
  Description: Sample Lambda role
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Effect: Allow
          Principal:
            Service: [lambda.amazonaws.com]
          Action: sts:AssumeRole
    ManagedPolicyArns:
      - arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole
    PermissionsBoundary: !Sub 'arn:${AWS::Partition}:iam:${AWS::AccountId}:policy/CodeStar_${ProjectId}_PermissionsBoundary'
```

3. Lance los cambios a través de la canalización y verifique el éxito.

Añadir una etapa Prod y un punto de conexión a un proyecto

Utilice los procedimientos de esta sección para añadir una nueva etapa de producción (Prod) a la canalización y una etapa de aprobación manual entre las etapas de implementación y producción de la canalización. Esto crea una pila de recursos adicionales cuando se ejecuta la canalización del proyecto.

Note

Puede utilizar estos procedimientos si:

- Para proyectos creados después del 3 de agosto de 2018, AWS CodeStar aprovisiona su proyecto de Amazon EC2, Elastic Beanstalk o Lambda con un archivo `/template.yml` en el repositorio del proyecto.
- Para proyectos creados después del 6 de diciembre de 2018 PDT, AWS CodeStar aprovisionó su proyecto con una política de límite de permisos.

Todos los proyectos de AWS CodeStar utilizan un archivo de plantilla de AWS CloudFormation que modela las dependencias de tiempo de ejecución de AWS de la aplicación, como las instancias de Linux y funciones de Lambda. El archivo `/template.yml` está almacenado en su repositorio de origen.

En el archivo `/template.yml`, utilice el parámetro `Stage` para añadir una pila de recursos para una nueva etapa en la canalización del proyecto.

Stage:

Type: String

Description: The name for a project pipeline stage, such as Staging or Prod, for which resources are provisioned and deployed.

Default: ''

El parámetro `Stage` se aplica a todos los recursos designados con el ID de proyecto al que se hace referencia en el recurso. Por ejemplo, el siguiente nombre de rol es un recurso designado en la plantilla:

```
RoleName: !Sub 'CodeStar-${ProjectId}-WebApp${Stage}'
```

Requisitos previos

Utilice las opciones de plantilla en la consola de AWS CodeStar para crear un proyecto.

Asegúrese de que el usuario de IAM tenga los siguientes permisos:

- `iam:PassRole` en el rol de AWS CloudFormation del proyecto.
- `iam:PassRole` en el rol de la cadena de herramientas del proyecto.
- `cloudformation:DescribeStacks`
- `cloudformation:ListChangeSets`

Solo para proyectos de Elastic Beanstalk o Amazon EC2:

- `codedeploy:CreateApplication`
- `codedeploy:CreateDeploymentGroup`
- `codedeploy:GetApplication`
- `codedeploy:GetDeploymentConfig`
- `codedeploy:GetDeploymentGroup`
- `elasticloadbalancing:DescribeTargetGroups`

Temas

- [Paso 1: crear un nuevo grupo de implementación en CodeDeploy \(solo proyectos de Amazon EC2\)](#)
- [Paso 2: añadir una nueva etapa de canalización a la etapa Prod](#)
- [Paso 3: añadir una etapa de aprobación manual](#)
- [Paso 4: enviar un cambio y monitorizar la actualización de la pila de AWS CloudFormation](#)

Paso 1: crear un nuevo grupo de implementación en CodeDeploy (solo proyectos de Amazon EC2)

Puede elegir la aplicación de CodeDeploy y, a continuación, añadir un nuevo grupo de implementación asociado a la nueva instancia.

Note

Si su proyecto es un proyecto de Lambda o Elastic Beanstalk, puede omitir este paso.

1. Abra la consola de CodeDeploy en <https://console.aws.amazon.com/codedeploy>.
2. Elija la aplicación de CodeDeploy que se generó para su proyecto cuando se creó en AWS CodeStar.
3. En Deployment groups (Grupos de implementaciones), elija Create deployment group (Crear grupo de implementaciones).
4. En Deployment group name (Nombre de grupo de implementación), escriba **<project-id>-prod-Env**.
5. En Service role (Rol de servicio), elija el rol de trabajador de cadena de herramientas para su proyecto de AWS CodeStar.
6. En Deployment type (Tipo de implementación), elija In-place (In situ).
7. En Environment configuration (Configuración de entorno), elija la pestaña Amazon EC2 Instances (Instancias de Amazon EC2).
8. En el grupo de etiquetas, en Key (Clave), elija `aws:cloudformation:stack-name`. En Value (Valor), elija `awscodestar-<projectid>-infrastructure-prod` (la pila que debe crear la acción GenerateChangeSet).
9. En Deployment settings (Configuración de implementación), elija `CodeDeployDefault.AllAtOnce`.
10. Borre Choose a load balancer (Elegir un balanceador de carga).
11. Elija Create deployment group.

Ahora se ha creado el segundo grupo de implementación.

Paso 2: añadir una nueva etapa de canalización a la etapa Prod

Añadir una etapa con el mismo conjunto de acciones de implementación que la etapa de implementación del proyecto. Por ejemplo, la nueva etapa Prod para un proyecto de Amazon EC2 debe tener las mismas acciones que en la etapa de implementación creada para el proyecto.

Para copiar parámetros y campos desde la etapa de implementación

1. Desde el panel del proyecto de AWS CodeStar, seleccione Detalles de la canalización para abrir la canalización en la consola de CodePipeline.
2. Elija Editar.
3. En la etapa de implementación, elija Editar etapa.
4. Seleccione el icono de edición en la acción GenerateChangeSet. Anote los valores de los campos siguientes. Utilizará estos valores cuando cree una nueva acción.
 - Nombre de pila
 - Cambiar nombre de conjunto
 - Plantilla
 - Template configuration (Configuración de plantilla)
 - Artefactos de entrada
5. Expanda Avanzado y en Parámetros, copie los parámetros del proyecto. Pegue estos parámetros en la nueva acción. Por ejemplo, copie los parámetros que se muestran aquí en formato JSON:

- Proyectos de Lambda:

```
{
  "ProjectId": "MyProject"
}
```

- Proyectos de Amazon EC2:

```
{
  "ProjectId": "MyProject",
  "InstanceType": "t2.micro",
  "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-EXAMPLEY5VSFS",
  "ImageId": "ami-EXAMPLE1",
  "KeyPairName": "my-keypair",
  "SubnetId": "subnet-EXAMPLE",
  "VpcId": "vpc-EXAMPLE1"
}
```

- Proyectos de Elastic Beanstalk:

```
{
  "ProjectId": "MyProject",
  "InstanceType": "t2.micro",
  "KeyPairName": "my-keypair",
  "SubnetId": "subnet-EXAMPLE",
  "VpcId": "vpc-EXAMPLE",
  "SolutionStackName": "64bit Amazon Linux 2018.03 v3.0.5 running Tomcat 8 Java
8",
  "EBTrustRole": "CodeStarWorker-myproject-EBService",
  "EBInstanceProfile": "awscodestar-myproject-EBInstanceProfile-11111EXAMPLE"
}
```

6. En el panel de edición de etapa, elija Cancelar.

Para crear una acción GenerateChangeSet en su nueva etapa Prod


Note

Después de añadir la nueva acción, pero aún en el modo de edición, si vuelve a abrir la acción para su edición, es posible que no se muestren algunos campos. También puede aparecer el siguiente error: Stack stack-name does not exist (La pila "nombre de pila" no existe)

Este error no le impide guardar la canalización. Sin embargo, para restaurar los campos que faltan, debe eliminar la nueva acción y añadirla de nuevo. Después de guardar y ejecutar la canalización, se reconoce la pila y el error no vuelve a aparecer.

1. Si la canalización no se muestra aún, desde su panel del proyecto de AWS CodeStar, seleccione Pipeline Details (Detalles de canalización) para abrir la canalización en la consola.
2. Elija Editar.
3. En la parte inferior del diagrama, seleccione + Add stage (Añadir etapa).
4. Escriba el nombre de la etapa (por ejemplo, **Prod**) y, a continuación, elija + Add action group (+Añadir grupo de acción).
5. En Nombre de la acción, escriba un nombre (por ejemplo, **GenerateChangeSet**).
6. En Proveedor de acción, elija AWS CloudFormation.
7. En Action mode (Modo acción), elija Create or replace a change set (Crear o reemplazar un conjunto de cambios).


8. En Stack name (Nombre de la pila), escriba un nuevo nombre para la pila AWS CloudFormation que va a crear esta acción. Comience por un nombre que sea idéntico al nombre de la pila de implementación y, a continuación, añada **-prod**:
 - Proyectos de Lambda: `awscodestar-<project_name>-lambda-prod`
 - Proyectos de Amazon EC2 y Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`
9. En Change set name (Cambiar nombre del conjunto), escriba el mismo nombre de conjunto que se indica en la etapa de implementación existente (por ejemplo, **pipeline-changeset**).
10. En Input artifacts (Artefactos de entrada), seleccione el artefacto de compilación.
11. En Template (Plantilla), escriba el mismo nombre de plantilla de cambio que se indica en la etapa de implementación existente (por ejemplo, **<project-ID>-BuildArtifact::template.yml**).
12. En Template configuration (Configuración de plantilla), especifique el mismo nombre de archivo de plantilla de configuración que se indica en la etapa de implementación existente (por ejemplo, **<project-ID>-BuildArtifact::template-configuration.json**).
13. En Capabilities (Capacidades), elija `CAPABILITY_NAMED_IAM`.
14. En Role name (Nombre de rol), elija el rol de trabajador de AWS CloudFormation de su proyecto.
15. Expanda Advanced (Avanzado) y en Parameters (Parámetros), pegue los parámetros de su proyecto. Incluya el parámetro Stage, que se muestra aquí en formato JSON, para un proyecto de Amazon EC2:

 Note

El nombre de la pila debe empezar por **awscodestar-<project_name>-** exactamente o la creación de la pila genera un error.

```
{  
  
  "ProjectId": "MyProject",  
  "InstanceType": "t2.micro",  
  "WebAppInstanceProfile": "awscodestar-MyProject-WebAppInstanceProfile-  
EXAMPLEY5VSFS",  
  "ImageId": "ami-EXAMPLE1",  
  "KeyPairName": "my-keypair",  
  "SubnetId": "subnet-EXAMPLE",
```


```
"VpcId": "vpc-EXAMPLE1",  
"Stage": "Prod"  
}
```

 Note

Asegúrese de pegar todos los parámetros del proyecto, no solo los parámetros nuevos o los parámetros que desea cambiar.

16. Seleccione Guardar.

17. En el panel AWS CodePipeline, elija Guardar cambio de canalización y, a continuación, Guardar cambio.

 Note


Es posible que aparezca un mensaje donde se informe de que se están eliminando y añadiendo recursos de detección de cambios. Confirme el mensaje y continúe con el siguiente paso de este tutorial.

Consulte la canalización actualizada.

Para crear una acción ExecuteChangeSet en la nueva etapa Prod

1. Si aún no ve la canalización, desde el panel de su proyecto de AWS CodeStar, elija Pipeline Details (Detalles de canalización) para abrir su canalización en la consola.
2. Elija Editar.
3. En la nueva etapa de producción, después de la nueva acción GenerateChangeSet, elija + Add action group (+ Añadir grupo de acción).
4. En Nombre de la acción, escriba un nombre (por ejemplo, **ExecuteChangeSet**).
5. En Proveedor de acción, elija AWS CloudFormation.
6. En Action mode (Modo de acción), elija Execute a change set (Ejecutar un conjunto de cambios).
7. En Stack name (Nombre de pila), escriba el nuevo nombre de la pila de AWS CloudFormation que escribió en la acción GenerateChangeSet (por ejemplo, **awscodestar-`<project-ID>`-infrastructure-prod**).

8. En Cambiar nombre del conjunto, escriba el mismo nombre de conjunto de cambios utilizado en la etapa de implementación (por ejemplo, **pipeline-changeset**).
9. Seleccione Listo.
10. En el panel AWS CodePipeline, elija Guardar cambio de canalización y, a continuación, Guardar cambio.

 Note

Es posible que aparezca un mensaje donde se informe de que se están eliminando y añadiendo recursos de detección de cambios. Confirme el mensaje y continúe con el siguiente paso de este tutorial.

Consulte la canalización actualizada.

Para crear una acción de implementación de CodeDeploy en la nueva etapa Prod (solo proyectos de Amazon EC2)

1. Después de las nuevas acciones en su etapa Prod, elija + Action (+Acción).
2. En Nombre de la acción, escriba un nombre (por ejemplo, **Deploy**).
3. En Proveedor de acción, elija AWS CodeDeploy.
4. En Nombre de aplicación, elija el nombre de la aplicación de CodeDeploy para el proyecto.
5. En Grupo de implementación, seleccione el nombre del nuevo grupo de implementación de CodeDeploy que creó en el paso 2.
6. En Artefactos de entrada, elija el mismo artefacto de compilación utilizado en la etapa existente.
7. Seleccione Listo.
8. En el panel AWS CodePipeline, elija Guardar cambio de canalización y, a continuación, Guardar cambio. Consulte la canalización actualizada.

Paso 3: añadir una etapa de aprobación manual

Como práctica recomendada, añada una etapa de aprobación manual delante de su nueva etapa de producción.

1. En la parte superior izquierda, elija Editar.

2. En el diagrama de la canalización, entre las etapas de implementación Deploy y Prod, elija + Add stage (+ Añadir etapa).
3. En Edit stage (Editar etapa), escriba un nombre de etapa (por ejemplo, **Approval**) y, a continuación, elija + Add action group (+ Añadir grupo de acciones).
4. En Nombre de la acción, escriba un nombre (por ejemplo, **Approval**).
5. En Approval type, elija Manual approval.
6. (Opcional) En Configuración, en ARN de tema de SNS, seleccione el tema de SNS que ha creado y al que se ha suscrito.
7. Elija Añadir acción.
8. En el panel AWS CodePipeline, elija Guardar cambio de canalización y, a continuación, Guardar cambio. Consulte la canalización actualizada.
9. Para enviar los cambios y comenzar una compilación de canalización, seleccione Release change (Publicar modificación) y, a continuación, Release (Publicar).

Paso 4: enviar un cambio y monitorizar la actualización de la pila de AWS CloudFormation

1. Mientras la canalización esté en ejecución, puede seguir los pasos que se indican a continuación para seguir la creación de la pila y el punto de conexión para la nueva etapa.
2. Cuando la canalización comienza la etapa de implementación, se inicia la actualización de la pila de AWS CloudFormation. Puede elegir la etapa AWS CloudFormation en la canalización en el panel de AWS CodeStar para ver la notificación de actualización de la pila. Para ver los datos de creación de la pila, en la consola, elija su proyecto en la lista Events (Eventos).
3. Después de completar correctamente la canalización, los recursos se crean en la pila de AWS CloudFormation. En la consola de AWS CloudFormation, elija la pila de infraestructura para su proyecto. Los nombres de pila siguen este formato:
 - Proyectos de Lambda: `awscodestar-<project_name>-lambda-prod`
 - Proyectos de Amazon EC2 y Elastic Beanstalk: `awscodestar-<project_name>-infrastructure-prod`

En la lista Recursos de la consola de AWS CloudFormation, consulte el recurso creado para el proyecto. En este ejemplo, la nueva instancia de Amazon EC2 aparece en la sección Recursos.

4. Acceda al punto de conexión para su etapa de producción:

- Para un proyecto de Elastic Beanstalk, abra la nueva pila en la consola de AWS CloudFormation y amplíe Recursos. Seleccione la aplicación de Elastic Beanstalk. Al hacerlo, se abrirá la consola de Elastic Beanstalk. Seleccione Environments (Entornos). Elija la URL en URL para abrir el punto de conexión en un navegador.
 - Para un proyecto de Lambda, abra la nueva pila en la consola de AWS CloudFormation y amplíe Recursos. Elija el recurso de API Gateway. El enlace se abrirá en la consola de API Gateway. Elija Etapas. Elija la URL en Invocar URL para abrir el punto de conexión en un navegador.
 - Para un proyecto de Amazon EC2, seleccione la nueva instancia de Amazon EC2 en su lista de recursos del proyecto en la consola de AWS CodeStar. El enlace se abrirá en la página Instancia de la consola de Amazon EC2. Elija la pestaña Description (Descripción), copie la URL en Public DNS (IPv4) (DNS pública (IPv4)) y abra la URL en un navegador.
5. Compruebe que el cambio se implementa.

Utilice de forma segura los parámetros de SSM en un proyecto AWS CodeStar

Muchos clientes almacenan secretos, como las credenciales, en parámetros de [Almacén de parámetros de Systems Manager](#). Ahora puede utilizar estos parámetros de forma segura en un AWS CodeStar proyecto. Por ejemplo, es posible que desee utilizar los parámetros SSM en las especificaciones de compilación CodeBuild o al definir los recursos de la aplicación en su conjunto de cadenas de herramientas (template.yml).

Para utilizar los parámetros de SSM en un CodeStar proyecto de AWS, debe etiquetar manualmente los parámetros con el ARN del CodeStar proyecto de AWS. También debe proporcionar los permisos adecuados al rol de trabajador de la CodeStar cadena de herramientas de AWS para acceder a los parámetros que ha etiquetado.

Antes de empezar

- [Cree un nuevo parámetro de Systems Manager](#) o identifique uno existente que contenga la información a la que desee acceder.
- Identifique qué CodeStar proyecto de AWS quiere usar o [cree uno nuevo](#).
- Tome nota del ARN del CodeStar proyecto. Debe tener un aspecto similar al siguiente:
`arn:aws:codestar:region-id:account-id:project/project-id`

Etiquete un parámetro con el ARN CodeStar del proyecto AWS

Consulte la página sobre [cómo etiquetar parámetros de Systems Manager](#) para obtener instrucciones detalladas.

1. En Clave, introduzca `awscodestar:projectArn`.
2. En Valor, introduzca el ARN del proyecto de CodeStar: `arn:aws:codestar:region-id:account-id:project/project-id`
3. Seleccione Guardar.

Ahora puede hacer referencia al parámetro de SSM en su archivo `template.yml`. Si desea utilizarlo con un rol de trabajador de la cadena de herramientas, deberá conceder permisos adicionales.

Otorgue permisos para usar parámetros etiquetados en su cadena de herramientas de CodeStar proyectos de AWS

Note

Estos pasos solo se aplican a los proyectos creados después del 6 de diciembre de 2018 PDT .

1. Abra el panel de CodeStar proyectos de AWS correspondiente al proyecto que desee usar.
2. Haga clic en Project (Proyecto) para ver la lista de recursos creados y busque el rol de trabajador de la cadena de herramientas. Se trata de un recurso de IAM con nombre con el formato: `role/CodeStarWorker-project-id-ToolChain`.
3. Haga clic en el ARN para abrirlo en la consola de IAM.
4. Ubique `ToolChainWorkerPolicy` y amplíelo, si es necesario.
5. Haga clic en Edit Policy (Editar política).
6. En Action: añada la línea siguiente:

```
ssm:GetParameter*
```

7. Haga clic en Review policy (Revisar política) y después en Save changes (Guardar cambios).

Para los proyectos creados antes del 6 de diciembre de 2018 PDT, tendrá que añadir los siguientes permisos a los roles de trabajador para cada servicio.

```
{
  "Action": [
    "ssm:GetParameter*"
  ],
  "Resource": "*",
  "Effect": "Allow",
  "Condition": {
    "StringEquals": {
      "ssm:ResourceTag/awscodestar:projectArn": "arn:aws:codestar:region-
id:account-id:project/project-id"
    }
  }
}
```

Desviar el tráfico para un proyecto de AWS Lambda

AWS CodeDeploy admite las implementaciones de versiones de las funciones de AWS Lambda en sus proyectos de AWS CodeStar sin servidor. La implementación de AWS Lambda desvía el tráfico entrante de una función Lambda existente a una versión actualizada de la función Lambda. Le recomendamos que pruebe una función Lambda actualizada mediante la implementación de una versión independiente y, a continuación, restaurando la implementación de la primera versión si es necesario.

Siga los pasos que se describen en esta sección para modificar la plantilla del proyecto de AWS CodeStar y actualizar los permisos de IAM del rol CodeStarWorker. Esta tarea inicia una respuesta automatizada en AWS CloudFormation que crea funciones de AWS Lambda asociadas y, a continuación, indica AWS CodeDeploy para desviar el tráfico a un entorno actualizado.

Note

Siga estos pasos solo si ha creado el proyecto de AWS CodeStar antes del 12 de diciembre de 2018.

AWS CodeDeploy tiene tres opciones de implementación que le permiten desviar el tráfico a versiones de su función AWS Lambda en su aplicación:

- **Valores controlados:** el tráfico se desvía en dos incrementos. Puede elegir opciones "canary" predefinidas que especifiquen el porcentaje de tráfico desviado a la versión actualizada de la función Lambda en el primer incremento y el intervalo, en minutos, antes de que el tráfico restante se desvíe en el segundo incremento.
- **Lineal:** el tráfico se desvía en incrementos iguales con el mismo número de minutos entre incrementos. Puede elegir opciones lineales predefinidas que especifiquen el porcentaje de tráfico desviado en cada incremento y el número de minutos entre cada incremento. El tráfico se desvía en incrementos iguales con el mismo número de minutos entre incrementos. Puede elegir opciones lineales predefinidas que especifiquen el porcentaje de tráfico desviado en cada incremento y el número de minutos entre cada incremento.
- **A la vez:** todo el tráfico se desvía a la vez desde la función Lambda original a la versión de la función Lambda actualizada.

Tipo de preferencia de implementación

Canary10Percent30Minutes

Canary10Percent5Minutes

Canary10Percent10Minutes

Canary10Percent15Minutes

Linear10PercentEvery10Minutes

Linear10PercentEvery1Minute

Linear10PercentEvery2Minutes

Linear10PercentEvery3Minutes

AllAtOnce

Para obtener más información acerca de las implementaciones de AWS CodeDeploy en una plataforma de computación de AWS Lambda, consulte [Implementaciones en una plataforma de computación de AWS Lambda](#).

Para obtener más información acerca de AWS SAM, consulte [AWS Serverless Application Model \(AWS SAM\)](#) en GitHub.

Requisitos previos:

Al crear un proyecto sin servidor, seleccione cualquier plantilla con la plataforma de computación Lambda. Debe haber iniciado sesión como administrador para llevar a cabo los pasos 4 a 6.

Paso 1: Modificar la plantilla de SAM para añadir los parámetros de implementación de la versión de AWS Lambda

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Cree un proyecto o elija un proyecto existente con un archivo `template.yml` y, a continuación, abra la página Code (Código). En la parte superior del repositorio, anote la ubicación de la plantilla de SAM denominada `template.yml` que debe modificarse.
3. Abra el archivo `template.yml` en su IDE o repositorio local. Copie el siguiente texto para añadir una sección `Globals` al archivo. El texto de muestra de este tutorial elige la opción `Canary10Percent5Minutes`.

```
Globals:
  Function:
    AutoPublishAlias: live
    DeploymentPreference:
      Enabled: true
      Type: Canary10Percent5Minutes
```

En este ejemplo se muestra una plantilla modificada después de añadir la sección `Globals`:

```
AWSTemplateFormatVersion: 2010-09-09
Transform:
- AWS::Serverless-2016-10-31
- AWS::CodeStar

Parameters:
  ProjectId:
    Type: String
    Description: CodeStar projectId used to associate new resources to team members

Globals:
  Function:
    AutoPublishAlias: live
    DeploymentPreference:
      Enabled: true
      Type: Canary10Percent5Minutes

Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.6
      Role:
        Fn::ImportValue:
          !Join ['-', [!Ref 'ProjectId', !Ref 'AWS::Region', 'LambdaTrustRole']]
      Events:
```

Para obtener más información, consulte la guía de referencia [Globals Section](#) para plantillas de SAM.

Paso 2: Editar el rol de AWS CloudFormation para añadir permisos

1. Inicie sesión en la AWS Management Console y abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.

Note

Debe iniciar sesión en la AWS Management Console con las credenciales asociadas al usuario de IAM que haya creado o identificado en [Configuración AWS CodeStar](#). Este usuario debe tener la política administrada de AWS denominada **AWSCodeStarFullAccess** asociada.

2. Elija el proyecto sin servidor existente y, a continuación, abra la página Recursos del proyecto.
3. En Recursos, elija el rol de IAM creado para el rol CodeStarWorker/AWS CloudFormation. El rol se abrirá en la consola de IAM.
4. En la pestaña Permissions, en Inline Políticas, en la fila de su política de rol de servicio, elija Edit Policy. Elija la pestaña JSON para editar la política en formato JSON.

 Note

El rol de servicio se llama CodeStarWorkerCloudFormationRolePolicy.

5. En el campo JSON, añada las siguientes instrucciones de la política al elemento Statement. Sustituya los marcadores de posición de *region* e *id* por su región e ID de cuenta.

```
{
  "Action": [
    "s3:GetObject",
    "s3:GetObjectVersion",
    "s3:GetBucketVersioning"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::codepipeline*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "lambda:*"
  ],
  "Resource": [
    "arn:aws:lambda:region:id:function:*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "apigateway:*"
  ],
  "Resource": [
    "arn:aws:apigateway:region::*"
  ],
  "Effect": "Allow"
}
```

```
"Effect": "Allow"
},
{
  "Action": [
    "iam:GetRole",
    "iam:CreateRole",
    "iam>DeleteRole",
    "iam:PutRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam>DeleteRolePolicy",
    "iam:DetachRolePolicy"
  ],
  "Resource": [
    "arn:aws:iam::id:role/*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "codedeploy:CreateApplication",
    "codedeploy>DeleteApplication",
    "codedeploy:RegisterApplicationRevision"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:application:*"
  ],
  "Effect": "Allow"
}
```

```

},
{
  "Action": [
    "codedeploy:CreateDeploymentGroup",
    "codedeploy:CreateDeployment",
    "codedeploy>DeleteDeploymentGroup",
    "codedeploy:GetDeployment"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:deploymentgroup:*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "codedeploy:GetDeploymentConfig"
  ],
  "Resource": [
    "arn:aws:codedeploy:region:id:deploymentconfig:*"
  ],
  "Effect": "Allow"
}

```

6. Elija Revisar política para asegurarse de que la política no contiene errores. Si no surgen errores, elija Guardar cambios.

Paso 3: Confirmar y enviar el cambio en la plantilla para iniciar el desvío de la versión de AWS Lambda

1. Confirme y envíe los cambios realizados en el archivo `template.yml` que ha guardado en el paso 1.

Note

Esto iniciará la canalización. Si confirma los cambios antes de actualizar los permisos de IAM, se iniciará la canalización y la actualización de la pila de AWS CloudFormation se topará con errores por lo que se revertirá la actualización de la pila. Si esto ocurre, corrija los permisos y, a continuación, reinicie la canalización.

2. La actualización de la pila de AWS CloudFormation se inicia cuando la canalización del proyecto inicia la etapa de implementación. Para ver la notificación de la actualización de la pila

cuando se inicie la implementación, en el panel de AWS CodeStar, seleccione la etapa AWS CloudFormation en la canalización.

Durante la actualización de la pila, AWS CloudFormation actualiza automáticamente los recursos del proyecto tal y como se indica a continuación:

- AWS CloudFormation procesa el archivo `template.yml` mediante la creación de funciones Lambda, enlaces de eventos y recursos.
- AWS CloudFormation llama a Lambda para crear la versión nueva de la función.
- AWS CloudFormation crea un archivo AppSpec y llama a AWS CodeDeploy para desviar el tráfico.

Para obtener más información acerca de la publicación de funciones de Lambda asociadas en SAM, consulte la referencia de la plantilla [AWS Serverless Application Model \(SAM\)](#). Para obtener más información acerca de los enlaces de eventos y los recursos en el archivo AppSpec de AWS CodeDeploy, consulte la [sección “recursos” de AppSpec \(solo para implementaciones de AWS Lambda\)](#) y la [sección “enlaces” de AppSpec para una implementación de AWS Lambda](#).

3. Después de completar correctamente la canalización, los recursos se crean en la pila de AWS CloudFormation. En la página Proyecto, en la lista Recursos del proyecto, consulte la aplicación de AWS CodeDeploy, el grupo de implementación de AWS CodeDeploy y los recursos del rol de servicio de AWS CodeDeploy creados para el proyecto.
4. Para crear una nueva versión, realice un cambio en la función Lambda en el repositorio. La nueva implementación se inicia y desvía el tráfico de acuerdo con el tipo de implementación indicado en la plantilla de SAM. Para ver el estado del tráfico que se desvía a la nueva versión, en la página Proyecto, en la lista Recursos del proyecto, seleccione el enlace a la implementación de AWS CodeDeploy.
5. Para ver los detalles de cada revisión, en Revisiones, elija el enlace al grupo de implementaciones de AWS CodeDeploy.
6. En su directorio de trabajo local, puede realizar cambios en la función de AWS Lambda y confirmar el cambio en el repositorio del proyecto. AWS CloudFormation admite AWS CodeDeploy para que administre la próxima revisión de la misma manera. Para obtener más información acerca de volver a implementar, detener o revertir una implementación de Lambda, consulte la sección [Implementaciones en una plataforma de computación de AWS Lambda](#).

Llevar su proyecto de AWS CodeStar a producción

Después de crear la aplicación mediante un proyecto de AWS CodeStar y ver lo que AWS CodeStar proporciona, tal vez desee llevar su proyecto a producción. Una forma de hacerlo es replicar los recursos de AWS de la aplicación fuera de AWS CodeStar. Seguirá necesitando un repositorio, un proyecto de compilación, una canalización y una implementación, pero en lugar de que AWS CodeStar los cree para el usuario, este los podrá volver a crear mediante AWS CloudFormation.

Note

Puede ser útil crear o ver un proyecto similar mediante alguno de los inicios rápidos de AWS CodeStar, y utilizar ese inicio rápido como plantilla para su propio proyecto con el fin de asegurarse de que incluya los recursos y las políticas que necesita.

Un proyecto de AWS CodeStar es una combinación de código fuente y de los recursos creados para implementar el código. El conjunto de recursos que le ayuda a crear, publicar e implementar el código se denomina recursos de la cadena de herramientas. Al crear el proyecto, una plantilla de AWS CloudFormation aprovisiona los recursos de la cadena de herramientas en una canalización de integración e implementación continuas (CI/CD).

Cuando se usa la consola para crear un proyecto, la plantilla de la cadena de herramientas se crea automáticamente. Cuando se utiliza la AWS CLI para crear un proyecto, usted debe crear la plantilla de la cadena de herramientas que crea los recursos de la cadena de herramientas.

Una cadena de herramientas completa requiere los siguientes recursos recomendados:

1. Un repositorio de CodeCommit o de GitHub que contenga el código fuente.
2. Una canalización de CodePipeline configurada para estar atenta a los cambios en el repositorio.
 - a. Si utiliza AWS CodeBuild para ejecutar pruebas de integración o de unidad, le recomendamos que agregue una etapa de compilación a la canalización para crear artefactos de compilación.
 - b. Le recomendamos que añada una etapa de implementación a la canalización que utiliza CodeDeploy o AWS CloudFormation para implementar el artefacto de compilación y el código fuente a la infraestructura de tiempo de ejecución.

Note

Dado que CodePipeline requiere al menos dos etapas en una canalización y la primera etapa debe ser la etapa de origen, añada una etapa de compilación o de implementación como segunda etapa.

Temas

- [Creación de un repositorio de GitHub](#)

Creación de un repositorio de GitHub

Puede crear un repositorio de GitHub definiéndolo en su plantilla de cadena de herramientas. Asegúrese de que ya ha creado una ubicación para el archivo ZIP que contiene su código fuente, para que el código se pueda cargar en el repositorio. Además, debe haber creado un token de acceso personal en GitHub para que AWS pueda conectarse a GitHub en su nombre. Además del token de acceso personal para GitHub, también debe tener el permiso `s3:GetObject` para el objeto Code que pase.

Para especificar un repositorio público de GitHub, añada código similar al siguiente a su plantilla de cadena de herramientas en AWS CloudFormation.

```
GitHubRepo:
  Condition: CreateGitHubRepo
  Description: GitHub repository for application source code
  Properties:
    Code:
      S3:
        Bucket: MyCodeS3Bucket
        Key: MyCodeS3BucketKey
    EnableIssues: true
    IsPrivate: false
    RepositoryAccessToken: MyGitHubPersonalAccessToken
    RepositoryDescription: MyAppCodeRepository
    RepositoryName: MyAppSource
    RepositoryOwner: MyGitHubUserName
  Type: AWS::CodeStar::GitHubRepository
```

Este código especifica la siguiente información:

- La ubicación del código que va a incluir, que debe ser un bucket de Amazon S3.
- Si desea habilitar la funcionalidad de problemas en el repositorio de GitHub.
- Si el repositorio de GitHub es privado.
- El token de acceso personal de GitHub que creó.
- Descripción, nombre y propietario del repositorio que va a crear.

Para obtener detalles completos sobre la información que debe especificar, consulte [AWS::CodeStar::GitHubRepository](#) en la Guía del usuario de AWS CloudFormation.

Trabajar con etiquetas de proyectos en AWS CodeStar

Puede asociar etiquetas con proyectos en AWS CodeStar. Las etiquetas pueden ayudarle a administrar sus proyectos. Por ejemplo, podría agregar una etiqueta con una clave `Release` y un valor `Beta` a cualquier proyecto en el que esté trabajando su organización para una versión beta.

Añadir una etiqueta a un proyecto

1. Con el proyecto abierto en la consola de AWS CodeStar, en el panel de navegación lateral, seleccione Configuración.
2. En Etiquetas, seleccione Editar.
3. En Clave, escriba un nombre para la etiqueta. En Valor, escriba el valor de la etiqueta.
4. Opcional: seleccione Agregar etiqueta para agregar más etiquetas.
5. Cuando haya acabado de agregar etiquetas, seleccione Guardar.

Eliminar una etiqueta de un proyecto

1. Con el proyecto abierto en la consola de AWS CodeStar, en el panel de navegación lateral, seleccione Configuración.
2. En Etiquetas, seleccione Editar.
3. En Etiquetas, busque la etiqueta que desee eliminar y, a continuación, seleccione Eliminar etiqueta.

4. Seleccione Guardar.

Obtener una lista de etiquetas para un proyecto

Utilice la AWS CLI ejecutar el comando de `AWS CodeStarlist-tags-for-project`, especificando el nombre del proyecto:

```
aws codestar list-tags-for-project --id my-first-projec
```

Si se ejecuta correctamente, aparece un listado de etiquetas en la salida, similar a la siguiente:

```
{
  "tags": {
    "Release": "Beta"
  }
}
```

Eliminar un proyecto de AWS CodeStar

Si ya no necesita un proyecto, puede eliminarlo y eliminar sus recursos para no incurrir en cargos adicionales en AWS. Cuando se elimina un proyecto, todos los miembros del equipo se elimina de ese proyecto. Los roles de proyecto se eliminan de sus usuarios de IAM, pero sus perfiles de usuario en AWS CodeStar no cambian. Puede utilizar la consola de AWS CodeStar o la AWS CLI para eliminar un proyecto. Para eliminar un proyecto, se necesita el rol de servicio de AWS CodeStar `aws-codestar-service-role`, que debe estar sin modificar y ser admitido por AWS CodeStar.

Important

La eliminación de un proyecto en AWS CodeStar no se puede deshacer. De forma predeterminada, todos los recursos de AWS del proyecto se eliminan de su cuenta de AWS, incluidos:

- El repositorio de CodeCommit para el proyecto junto con todo lo que esté almacenado en ese repositorio.
- Los roles de proyecto de AWS CodeStar y las políticas de IAM asociadas configuradas para el proyecto y sus recursos.
- Cualquier instancia de Amazon EC2 creada para el proyecto.

- La aplicación de implementación y los recursos asociados, como por ejemplo:
 - Una aplicación de CodeDeploy y los grupos de implementación asociados.
 - Una función de AWS Lambda y las API de API Gateway asociadas.
 - Una aplicación de AWS Elastic Beanstalk y el entorno asociado.
- La canalización de implementación continua del proyecto en CodePipeline.
- Las pilas de AWS CloudFormation asociadas con el proyecto.
- Cualquier entorno de desarrollo de AWS Cloud9 creado con la consola de AWS CodeStar. Todos los cambios en el código sin confirmar en los entornos se perderán.

Para eliminar todos los recursos del proyecto junto con el proyecto, seleccione la casilla de verificación Eliminar recursos. Si borra esta opción, el proyecto se eliminará en AWS CodeStar y los roles de proyecto que permitían el acceso a esos recursos se eliminarán en IAM, pero todos los demás recursos se retendrán. Podría seguir incurriendo en cargo para esos recursos en AWS. Si decide que ya no desea uno o varios de estos recursos, debe eliminarlos manualmente. Para obtener más información, consulte [Eliminación de un AWS CodeStar proyecto: se ha eliminado un proyecto, pero aún existen recursos](#).

Si decide mantener los recursos cuando elimina un proyecto, se recomienda copiar la lista de recursos de la página de detalles del proyecto. De esta forma, tendrá un registro de todos los recursos que ha mantenido, aunque el proyecto ya no exista.

Temas

- [Eliminar un proyecto en AWS CodeStar \(consola\)](#)
- [Eliminar un proyecto en AWS CodeStar \(AWS CLI\)](#)

Eliminar un proyecto en AWS CodeStar (consola)

Puede utilizar la consola de AWS CodeStar para eliminar un proyecto.

Para eliminar un proyecto en AWS CodeStar

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos.
3. Seleccione el proyecto que desee eliminar y elija Eliminar.

O bien, abra el proyecto y seleccione Configuración en el panel de navegación del lado izquierdo de la consola. En la página de detalles del proyecto, seleccione Eliminar proyecto.

4. En la página Confirmación de eliminación, escriba eliminar. Mantenga seleccionada la opción Eliminar recursos si desea eliminar los recursos del proyecto. Elija Eliminar.

La eliminación de un proyecto puede tardar varios minutos. Una vez eliminado, el proyecto ya no aparece en la lista de proyectos en la consola de AWS CodeStar.

Important

Si su proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio GitHub o problemas en Atlassian JIRA), dichos recursos no se eliminan, incluso si selecciona la casilla de verificación.

El proyecto no se pueden eliminar si las políticas administradas de AWS CodeStar se han asociado manualmente a roles que no son usuarios de IAM. Si ha asociado las políticas administradas del proyecto a un rol del usuario federado, primero deberá eliminar el proyecto. Para obtener más información, consulte [???](#).

Eliminar un proyecto en AWS CodeStar (AWS CLI)

Puede utilizar la AWS CLI para eliminar un proyecto.

Para eliminar un proyecto en AWS CodeStar

1. En un terminal (Linux, macOS, o Unix) o en un símbolo del sistema (Windows), ejecute el comando `delete-project`, incluido el nombre del proyecto. Por ejemplo, para eliminar un proyecto con el ID *my-2nd-project*:

```
aws codestar delete-project --id my-2nd-project
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "projectArn": "arn:aws:codestar:us-east-2:111111111111:project/my-2nd-project"
}
```

Los proyectos no se eliminan inmediatamente.

2. Ejecute el comando `describe-project`, incluido el nombre del proyecto. Por ejemplo, para comprobar el estado de un proyecto con el ID `my-2nd-project`:

```
aws codestar describe-project --id my-2nd-project
```

si el proyecto aún no se ha eliminado, este comando devuelve resultados similares a los siguientes:

```
{
  "name": "my project",
  "id": "my-2nd-project",
  "arn": "arn:aws:codestar:us-west-2:123456789012:project/my-2nd-project",
  "description": "My second CodeStar project.",
  "createdTimeStamp": 1572547510.128,
  "status": {
    "state": "CreateComplete"
  }
}
```

Si se ha eliminado el proyecto, este comando devuelve resultados similares a los siguientes:

```
An error occurred (ProjectNotFoundException) when calling the DescribeProject
operation: The project ID was not found: my-2nd-project. Make sure that the
project ID is correct and then try again.
```

3. Ejecute el comando `list-projects` y compruebe que el proyecto eliminado ya no aparece en la lista de proyectos asociados a su cuenta de AWS.

```
aws codestar list-projects
```

Trabajar con equipos de AWS CodeStar

Después de crear un proyecto de desarrollo, conceda acceso a otras personas para poder trabajar juntos. En AWS CodeStar, cada proyecto tiene un equipo de proyecto. Un usuario puede pertenecer a varios proyectos de AWS CodeStar y tener distintos roles de AWS CodeStar (y, por lo tanto, permisos diferentes) en cada uno de ellos. En la consola de AWS CodeStar, los usuarios ven todos los proyectos asociados a su cuenta de AWS pero solo pueden ver y trabajar en los proyectos en los que son miembros del equipo.

Los miembros del equipo pueden elegir un nombre sencillo para ellos mismos. También pueden añadir una dirección de correo electrónico para que otros miembros del equipo puedan ponerse en contacto con ellos. Los miembros del equipo que no son propietarios no pueden cambiar su rol de AWS CodeStar para el proyecto.

Cada proyecto en AWS CodeStar tiene tres roles:

Roles y permisos en un proyecto de AWS CodeStar

Nombre de función	Ver el panel y el estado del proyecto	Añadir/Eliminar/ Obtener acceso a los recursos del proyecto	Añadir/Eliminar miembros del equipo	Eliminar proyecto
Propietario	x	x	x	x
Contributor	x	x		
Lector	x			

- **Propietario:** puede añadir y eliminar otros miembros del equipo, aportar código al repositorio del proyecto si el código está almacenado en CodeCommit, conceder o denegar a otros miembros del equipo el acceso remoto a alguna instancia de Amazon EC2 asociada al proyecto que se ejecute en Linux, configurar el panel del proyecto y eliminar el proyecto.
- **Colaborador:** puede añadir y eliminar recursos del panel, como un icono de JIRA, aportar código al repositorio del proyecto si el código está almacenado en CodeCommit e interactuar totalmente con el panel. No puede añadir ni eliminar miembros del equipo, conceder ni denegar el acceso remoto a los recursos ni eliminar el proyecto. Este es el rol que debe elegir para la mayoría de los miembros del equipo.

- **Lector:** puede ver el panel del proyecto, el código si está almacenado en CodeCommit y, en los iconos del panel, el estado del proyecto y sus recursos.

⚠ Important

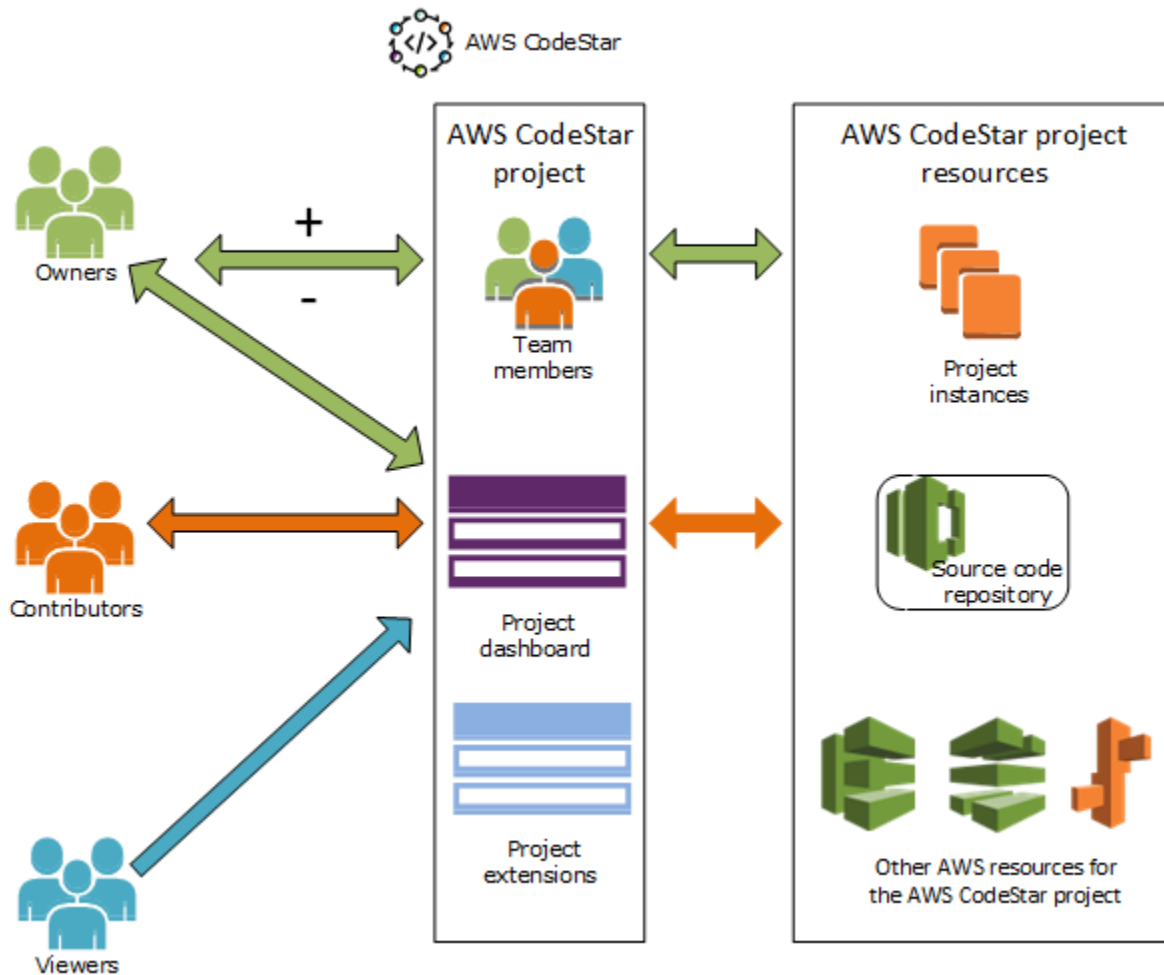
Si su proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA), el acceso a dichos recursos lo controla el proveedor de recursos, no AWS CodeStar. Para obtener más información, consulte la documentación del proveedor de recursos.

Cualquier persona que tenga acceso a un proyecto de AWS CodeStar puede utilizar la consola de AWS CodeStar para acceder a los recursos que están fuera de AWS pero que están relacionados con el proyecto.

AWS CodeStar no permite automáticamente que los miembros del equipo del proyecto participen en cualquier entorno de desarrollo de AWS Cloud9 para un proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#).

Se asocia una política de IAM con cada rol del proyecto. Esta política está personalizada para el proyecto con el fin de reflejar sus recursos. Para obtener más información sobre estas políticas, consulte [AWS CodeStarEjemplos de políticas basadas en la identidad](#).

En el siguiente diagrama se muestra la relación entre cada rol y un proyecto de AWS CodeStar.



Temas

- [Añadir miembros de equipo a un proyecto de AWS CodeStar](#)
- [Administrar permisos para los miembros del equipo de AWS CodeStar](#)
- [Eliminación de miembros del equipo de un proyecto de AWS CodeStar](#)

Añadir miembros de equipo a un proyecto de AWS CodeStar

Si tiene el rol de propietario en un proyecto de AWS CodeStar o aplica la política `AWSCodeStarFullAccess` a su usuario de IAM, puede añadir otros usuarios de IAM al equipo del proyecto. Es un proceso sencillo que aplica un rol de AWS CodeStar (propietario, colaborador o lector) al usuario. Estos roles son por proyecto y están personalizados. Por ejemplo, un miembro colaborador del equipo en el proyecto A podría tener permisos para recursos diferentes de los de un miembro colaborador del equipo en el proyecto B. Un miembro del equipo solo puede tener un rol en

un proyecto. Una vez que ha añadido un miembro al equipo, este puede interactuar inmediatamente con el proyecto en el nivel definido por el rol.

Entre las ventajas de los roles de AWS CodeStar y la pertenencia al equipo se incluyen:

- No tiene que configurar manualmente los permisos en IAM para los miembros del equipo.
- Puede cambiar fácilmente el nivel de acceso de un miembro del equipo a un proyecto.
- Los usuarios pueden acceder a los proyectos en la consola de AWS CodeStar solo si son miembros del equipo.
- El acceso del usuario a un proyecto viene definido por el rol.

Para obtener más información acerca de los equipos y roles de AWS CodeStar, consulte [Trabajar con equipos de AWS CodeStar](#) y [Trabajar con el perfil de usuario de AWS CodeStar](#).

Para añadir un miembro del equipo a un proyecto, debe tener el rol de propietario de AWS CodeStar para el proyecto o aplicar la política `AWSCodeStarFullAccess`.

Important

Añadir un miembro del equipo no afecta al acceso de dicho miembro del equipo a los recursos que están fuera de AWS (por ejemplo, un repositorio GitHub o problemas en Atlassian JIRA). Estos permisos de acceso los controla el proveedor de recursos, no AWS CodeStar. Para obtener más información, consulte la documentación del proveedor de recursos.

Cualquier usuario que tenga acceso a un proyecto de AWS CodeStar puede utilizar la consola de AWS CodeStar para acceder a los recursos que están fuera de AWS pero que están relacionados con el proyecto.

Añadir un miembro del equipo a un proyecto no permite de manera automática que el miembro participe en cualquier entorno de desarrollo de AWS Cloud9 relacionado con el proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#). Conceder a los usuarios federados acceso a un proyecto implica asociar manualmente la política administrada de propietario, colaborador o lector de AWS CodeStar al rol asumido por el usuario federado. Para obtener más información, consulte [Acceso de usuarios federados a AWS CodeStar](#).

Temas

- [Añadir un miembro del equipo \(consola\)](#)
- [Añadir y ver miembros del equipo \(AWS CLI\)](#)

Añadir un miembro del equipo (consola)

Puede utilizar la consola de AWS CodeStar para añadir un miembro del equipo a su proyecto. Si ya existe un usuario de IAM para la persona que desee añadir, puede añadir el usuario de IAM. De lo contrario, puede crear un usuario de IAM para esa persona al añadirla al proyecto.

Para añadir un miembro del equipo a un proyecto de AWS CodeStar (consola)

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos y, a continuación, seleccione su proyecto.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.
4. En la página Miembros del equipo, elija Añadir miembro del equipo.
5. En Elegir usuario, realice una de las siguientes operaciones:
 - Si ya existe un usuario de IAM para la persona que desea añadir, seleccione a dicho usuario de IAM de la lista.

Note

Los usuarios que ya se hayan añadido a otro proyecto de AWS CodeStar aparecerán en la lista Usuarios de AWS CodeStar existentes.

En Rol del proyecto, elija el rol de AWS CodeStar (propietario, colaborador o lector) que desee otorgar a este usuario. Este es un rol de nivel de proyecto de AWS CodeStar que solo puede cambiar el propietario del proyecto. Cuando se aplica a un usuario de IAM, el rol proporciona todos los permisos necesarios para obtener acceso a los recursos del proyecto de AWS CodeStar. Aplica las políticas necesarias para crear y administrar credenciales de Git para código almacenado en CodeCommit en IAM o bien para cargar las claves de SSH de Amazon EC2 para el usuario en IAM.

⚠ Important

No puede proporcionar ni cambiar la información del nombre o del correo electrónico de visualización de un usuario de IAM a menos que haya iniciado sesión en la consola como dicho usuario. Para obtener más información, consulte [Administración de la información de visualización de su perfil de usuario de AWS CodeStar](#).

Seleccione Agregar el miembro del equipo.

- Si no existe un usuario de IAM para la persona que desea añadir al proyecto, seleccione Crear nuevo usuario de IAM. Se le redirigirá a la consola de IAM, donde podrá crear un nuevo usuario de IAM. Consulte [Creación de usuarios de IAM](#) en la Guía del usuario de IAM para obtener más información. Tras crear su usuario de IAM, vuelva a la consola de AWS CodeStar, actualice la lista de usuarios y elija de la lista desplegable el usuario de IAM que creó. Introduzca el nombre de visualización de AWS CodeStar, la dirección de correo electrónico y el rol del proyecto que desee aplicar a este nuevo usuario y, a continuación, seleccione Agregar el miembro del equipo.

ℹ Note

Para facilitar la administración, al menos un usuario debe tener asignado el rol de propietario del proyecto.

6. Envíe al nuevo miembro del equipo la siguiente información:

- Información de conexión para su proyecto de AWS CodeStar.
- Si el código fuente está almacenado en CodeCommit, [instrucciones para configurar el acceso con credenciales de Git](#) en el repositorio de CodeCommit desde los equipos locales.
- Información sobre cómo el usuario puede administrar su nombre que mostrar, dirección de correo electrónico y clave pública SSH de Amazon EC2, tal como se describe en [Trabajar con el perfil de usuario de AWS CodeStar](#).
- Contraseña de un solo uso e información de conexión, si el usuario es nuevo en AWS y ha creado un usuario de IAM para esa persona. La contraseña caducará la primera vez que el usuario inicie sesión. El usuario debe elegir una contraseña nueva.

Añadir y ver miembros del equipo (AWS CLI)

Puede usar la AWS CLI para añadir miembros del equipo al equipo de su proyecto. También puede ver información acerca de todos los miembros del equipo en su proyecto.

Para añadir un miembro del equipo

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `associate-team-member` con los parámetros `--project-id`, `--user-arn` y `--project-role`. También puede especificar si el usuario tiene o no acceso remoto a instancias del proyecto incluyendo los parámetros `--remote-access-allowed` o `--no-remote-access-allowed`. Por ejemplo:

```
aws codestar associate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/Jane_Doe --project-role Contributor --remote-access-
allowed
```

Este comando no devuelve ningún resultado.

Para ver todos los miembros del equipo (AWS CLI)

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `list-team-members` con el parámetro `--project-id`. Por ejemplo:

```
aws codestar list-team-members --project-id my-first-projec
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "teamMembers":[
    {"projectRole":"Owner","remoteAccessAllowed":true,"userArn":"arn:aws:iam::111111111111:use
Mary_Major"},
    {"projectRole":"Contributor","remoteAccessAllowed":true,"userArn":"arn:aws:iam::1111111111
Jane_Doe"},
    {"projectRole":"Contributor","remoteAccessAllowed":true,"userArn":"arn:aws:iam::1111111111
John_Doe"},
```

```
{ "projectRole": "Viewer", "remoteAccessAllowed": false, "userArn": "arn:aws:iam::111111111111:u
John_Stiles" }
]
```

Administrar permisos para los miembros del equipo de AWS CodeStar

Puede cambiar los permisos de los miembros del equipo cambiando su rol de AWS CodeStar. A cada miembro del equipo solo se le pueden asignar un rol en un proyecto de AWS CodeStar, pero se pueden asignar muchos usuarios al mismo rol. No puede utilizar la consola de AWS CodeStar ni la AWS CLI para administrar permisos.

Important

Para cambiar un rol para un miembro del equipo, debe tener el rol de propietario de AWS CodeStar en ese proyecto o tener la política `AWSCodeStarFullAccess` aplicada.

Cambiar los permisos de un miembro del equipo no afecta al acceso de dicho miembro del equipo a los recursos que están fuera de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA). Estos permisos de acceso los controla el proveedor de recursos, no AWS CodeStar. Para obtener más información, consulte la documentación del proveedor de recursos.

Cualquier persona que tenga acceso a un proyecto de AWS CodeStar puede utilizar la consola de AWS CodeStar para obtener acceso a los recursos que están fuera de AWS pero que están relacionados con dicho proyecto.

Cambiar el rol de un miembro del equipo de un proyecto no permite o evita de forma automática que dicho miembro participe en cualquier entorno de implementación de AWS Cloud9 del proyecto. Para permitir o evitar que un miembro del equipo participe en un entorno compartido, consulte [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#).

También puede conceder permisos para que los usuarios obtengan acceso remoto a cualquier instancia de Amazon EC2 que ejecute Linux asociada con el proyecto. Después de conceder este permiso, el usuario debe cargar una clave pública de SSH que está asociada a su perfil de usuario

de AWS CodeStar en todos los proyectos del equipo. Para poder conectarse correctamente a instancias de Linux, el usuario debe tener configurada SSH y la clave privada en el equipo local.

Temas

- [Administrar permisos de equipo \(consola\)](#)
- [Administrar permisos de equipo \(AWS CLI\)](#)

Administrar permisos de equipo (consola)

Puede usar la consola de AWS CodeStar para administrar los roles de los miembros del equipo. También puede administrar el acceso remoto de los miembros a las instancias de Amazon EC2 asociadas con su proyecto.

Para cambiar el rol de un miembro del equipo

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos y, a continuación, seleccione su proyecto.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.
4. En la página Miembros del equipo, seleccione al miembro del equipo y, a continuación, seleccione Editar.
5. En Rol de proyecto, elija el rol de AWS CodeStar (propietario, colaborador o lector) que desee otorgar a este usuario.

Para obtener más información acerca de los roles de AWS CodeStar y sus permisos, consulte [Trabajar con equipos de AWS CodeStar](#).

Seleccione Editar el miembro del equipo.

Para conceder a un miembro del equipo permisos de acceso remoto a instancias de Amazon EC2

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos y, a continuación, seleccione su proyecto.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.
4. En la página Miembros del equipo, seleccione al miembro del equipo y, a continuación, seleccione Editar.

5. Seleccione la casilla Otorgar acceso SSH a las instancias del proyecto y, a continuación, seleccione Editar el miembro del equipo.
6. (Opcional) Notifique a los miembros del equipo que deben cargar una clave pública SSH para sus usuarios de AWS CodeStar si aún no lo han hecho. Para obtener más información, consulte [Agregue una clave pública a su perfil AWS CodeStar de usuario](#).

Administrar permisos de equipo (AWS CLI)

Puede utilizar la AWS CLI para administrar el rol de proyecto asignado a un miembro del equipo. Puede utilizar los mismos comandos de la AWS CLI para administrar el acceso remoto de ese miembro del equipo a instancias de Amazon EC2 asociadas con su proyecto.

Para administrar los permisos de un miembro del equipo

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `update-team-member` con los parámetros `--project-id`, `-user-arn` y `--project-role`. También puede especificar si el usuario tiene o no acceso remoto a instancias del proyecto incluyendo los parámetros `--remote-access-allowed` o `--no-remote-access-allowed`. Por ejemplo, para actualizar el rol de proyecto de un usuario de IAM llamado `John_Doe` y cambiar sus permisos por los de lector sin acceso remoto a las instancias de Amazon EC2 del proyecto:

```
aws codestar update-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe --project-role Viewer --no-remote-access-
allowed
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "projectRole": "Viewer",
  "remoteAccessAllowed": false,
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

Eliminación de miembros del equipo de un proyecto de AWS CodeStar

Después de eliminar a un usuario de un proyecto de AWS CodeStar, dicho usuario seguirá apareciendo en el historial de confirmaciones del repositorio del proyecto, pero ya no tendrá acceso al repositorio de CodeCommit ni a ningún otro recurso del proyecto como, por ejemplo, la canalización del proyecto. (La excepción a esta regla es un usuario de IAM que tenga otras políticas aplicadas que le otorguen acceso a dichos recursos). El usuario no podrá acceder al panel del proyecto, y el proyecto dejará de aparecer en la lista de proyectos visible para el usuario en el panel de AWS CodeStar. Puede utilizar la consola de AWS CodeStar o la AWS CLI para eliminar los miembros del equipo de su equipo del proyecto.

Important

Aunque al eliminar a un miembro del equipo de un proyecto se le deniega el acceso remoto a las instancias de Amazon EC2 del proyecto, no cierra ninguna de las sesiones de SSH activas del usuario.

Quitar a un miembro del equipo no afecta al acceso de dicho miembro del equipo a los recursos que están fuera de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA). Estos permisos de acceso los controla el proveedor de recursos, no AWS CodeStar. Para obtener más información, consulte la documentación del proveedor de recursos.

Eliminar a un miembro del equipo de proyecto no elimina automáticamente los entornos de desarrollo de AWS Cloud9 vinculados con dicho miembro del equipo ni evita que participe en cualquier entorno de desarrollo de AWS Cloud9 relacionado al que ha sido invitado.

Para eliminar un entorno de desarrollo, consulte [Eliminar un entorno de AWS Cloud9 de un proyecto](#). Para evitar que un miembro del equipo participe en un entorno compartido, consulte [Compartir un entorno de AWS Cloud9 con un miembro del equipo del proyecto](#).

Para eliminar a un miembro del equipo de un proyecto, usted debe tener el rol de propietario de AWS CodeStar de dicho proyecto o la política `AWSCodeStarFullAccess` aplicada a su cuenta.

Temas

- [Eliminar miembros del equipo \(consola\)](#)
- [Eliminar miembros del equipo \(AWS CLI\)](#)

Eliminar miembros del equipo (consola)

Puede utilizar la consola de AWS CodeStar para eliminar los miembros del equipo de su equipo del proyecto.

Para eliminar un miembro del equipo de un proyecto

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. En el panel de navegación, seleccione Proyectos y, a continuación, seleccione su proyecto.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.
4. En la página Miembros del equipo, seleccione al miembro del equipo y, a continuación, seleccione Eliminar.

Eliminar miembros del equipo (AWS CLI)

Puede utilizar la AWS CLI para eliminar los miembros del equipo de su equipo del proyecto.

Para eliminar un miembro del equipo

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `disassociate-team-member` con `--project-id` y `-user-arn`. Por ejemplo:

```
aws codestar disassociate-team-member --project-id my-first-projec --user-arn
arn:aws:iam:111111111111:user/John_Doe
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "projectId": "my-first-projec",
  "userArn": "arn:aws:iam::111111111111:user/John_Doe"
}
```

Trabajar con el perfil de usuario de AWS CodeStar

Su perfil de usuario de AWS CodeStar está asociado con su usuario de IAM. Este perfil contiene un nombre de visualización y una dirección de correo electrónico que se utiliza en todos los proyectos de AWS CodeStar a los que pertenece. Puede cargar una clave pública SSH que se asociará con su perfil. Esta clave pública forma parte del par de claves SSH pública y privada que se utiliza al conectarse a instancias de Amazon EC2 asociadas con los proyectos de AWS CodeStar a los que pertenece.

Note

La información de estos temas trata solo su perfil de usuario de AWS CodeStar. Si el proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA), dichos proveedores de recursos podrían utilizar sus propios perfiles de usuario, por lo que podrían tener configuraciones distintas. Para obtener más información, consulte la documentación del proveedor de recursos.

Temas

- [Administración de la información de visualización de su perfil de usuario de AWS CodeStar](#)
- [Agregue una clave pública a su perfil AWS CodeStar de usuario](#)

Administración de la información de visualización de su perfil de usuario de AWS CodeStar

Puede utilizar la consola de AWS CodeStar o la AWS CLI para cambiar el nombre de visualización y la dirección de correo electrónico de su perfil de usuario. Un perfil de usuario no es específico del proyecto. Se asocia a su usuario de IAM y se utiliza en los proyectos de AWS CodeStar a los que usted pertenece en un AWS. Si pertenece a proyectos de más de una región de AWS, tendrá perfiles de usuario independientes.

Solo puede gestionar su propio perfil de usuario en la consola de AWS CodeStar. Si tiene la política `AWSCodeStarFullAccess`, puede ver y administrar otros perfiles mediante la AWS CLI.

 Note


La información de este tema trata solo su perfil de usuario de AWS CodeStar. Si el proyecto utiliza recursos que no son de AWS (por ejemplo, un repositorio de GitHub o problemas en Atlassian JIRA), dichos proveedores de recursos podrían utilizar sus propios perfiles de usuario, por lo que podrían tener configuraciones distintas. Para obtener más información, consulte la documentación del proveedor de recursos.

Temas

- [Administrar el perfil de usuario \(consola\)](#)
- [Administrar perfiles de usuario \(AWS CLI\)](#)

Administrar el perfil de usuario (consola)

Puede gestionar su perfil de usuario en la consola de AWS CodeStar dirigiéndose a cualquier proyecto en el que sea miembro del equipo y cambiar la información del perfil. Dado que los perfiles de usuario son específicos del usuario pero no son específicos de un proyecto, los cambios en el perfil de usuario aparecerán en cada proyecto de cada región de AWS del que sea miembro del equipo.


 Important

Para utilizar la consola para modificar la información de visualización de un usuario, debe iniciar sesión con dicho usuario de IAM. Ningún otro usuario, ni siquiera los que tienen el rol de propietario de AWS CodeStar de un proyecto ni con la política `AWSCodeStarFullAccess` aplicada, puede cambiar la información de visualización.

Para cambiar la información de visualización en todos los proyectos de una región de AWS

1. Abra la consola de AWS CodeStar en <https://console.aws.amazon.com/codestar/>.
2. Seleccione Proyectos en el panel de navegación y, a continuación, seleccione un proyecto en el que sea miembro del equipo.
3. En el panel de navegación lateral del proyecto, seleccione Equipo.

4. En la página Miembros del equipo, seleccione el usuario de IAM y, a continuación, seleccione Editar.
5. Edite el nombre de visualización, la dirección de correo electrónico o ambos y, a continuación, seleccione Editar el miembro del equipo.

 Note

Se requieren tanto un nombre de visualización como una dirección de correo electrónico. Para obtener más información, consulte [Límites en AWS CodeStar](#).

Administrar perfiles de usuario (AWS CLI)

Puede utilizar la AWS CLI para crear y administrar su perfil de usuario en AWS CodeStar. También puede utilizar la AWS CLI para ver la información de su perfil de usuario y ver todos los perfiles de usuario configurados para su cuenta de AWS en una región de AWS.

Asegúrese de que su perfil de AWS se haya configurado para la región en la que desee crear, administrar o ver los perfiles de usuario.

Para crear un perfil de usuario

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `create-user-profile` con los parámetros `user-arn`, `display-name` y `email-address`. Por ejemplo:

```
aws codestar create-user-profile --user-arn arn:aws:iam:111111111111:user/John_Stiles --display-name "John Stiles" --email-address "john_stiles@example.com"
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "createdTimestamp":1.491439687681E9,"
  displayName":"John Stiles",
  "emailAddress":"john.stiles@example.com",
  "lastModifiedTimestamp":1.491439687681E9,
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

Para ver su información de visualización

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `describe-user-profile` con el parámetro `user-arn`. Por ejemplo:

```
aws codestar describe-user-profile --user-arn arn:aws:iam:111111111111:user/
Mary_Major
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "createdTimestamp":1.490634364532E9,
  "displayName":"Mary Major",
  "emailAddress":"mary.major@example.com",
  "lastModifiedTimestamp":1.491001935261E9,
  "sshPublicKey":"EXAMPLE=",
  "userArn":"arn:aws:iam::111111111111:user/Mary_Major"
}
```

Para cambiar su información de visualización

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `update-user-profile` con el parámetro `user-arn` y los parámetros de perfil que desee cambiar, como `display-name` o `email-address`. Por ejemplo, si un usuario con el nombre de visualización "Jane Doe" desea cambiar su nombre de visualización por "Jane Mary Doe":

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--display-name "Jane Mary Doe"
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Mary Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

```
}
```

Para enumerar todos los perfiles de usuario de una región de AWS en su cuenta de AWS

1. Abra un terminal o una ventana de comandos.
2. Ejecute el comando `aws codestar list-user-profiles`. Por ejemplo:

```
aws codestar list-user-profiles
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "userProfiles": [
    {
      "displayName": "Jane Doe",
      "emailAddress": "jane.doe@example.com",
      "sshPublicKey": "EXAMPLE1",
      "userArn": "arn:aws:iam::111111111111:user/Jane_Doe"
    },
    {
      "displayName": "John Doe",
      "emailAddress": "john.doe@example.com",
      "sshPublicKey": "EXAMPLE2",
      "userArn": "arn:aws:iam::111111111111:user/John_Doe"
    },
    {
      "displayName": "Mary Major",
      "emailAddress": "mary.major@example.com",
      "sshPublicKey": "EXAMPLE=",
      "userArn": "arn:aws:iam::111111111111:user/Mary_Major"
    },
    {
      "displayName": "John Stiles",
      "emailAddress": "john.stiles@example.com",
      "sshPublicKey": "",
      "userArn": "arn:aws:iam::111111111111:user/John_Stiles"
    }
  ]
}
```

Agregue una clave pública a su perfil AWS CodeStar de usuario

Puede cargar una clave pública de SSH como parte del par de claves pública y privada que va a crear y administrar. Puede utilizar este par de claves de SSH pública y privada para acceder a instancias de Amazon EC2 que ejecuten Linux. Si un propietario del proyecto le ha concedido permiso de acceso remoto, solo podrá acceder a las instancias asociadas al proyecto. Puede usar la AWS CodeStar consola o AWS CLI administrar su clave pública.

Important

AWS CodeStar El propietario del proyecto puede conceder a los propietarios, colaboradores y espectadores del proyecto acceso SSH a las instancias de Amazon EC2 del proyecto, pero solo la persona (propietario, colaborador o espectador) puede configurar la clave SSH. Para ello, el usuario debe haber iniciado sesión como propietario, colaborador o lector individual. AWS CodeStar no administra las claves SSH de los entornos. AWS Cloud9

Temas

- [Administrar la clave pública \(consola\)](#)
- [Administrar la clave pública \(AWS CLI\)](#)
- [Conectarse a la instancia de Amazon EC2 con la clave privada](#)

Administrar la clave pública (consola)

Aunque no puede generar un key pair público-privado en la consola, puede crear uno localmente y, a continuación, agregarlo o administrarlo como parte de su perfil de usuario a través de la AWS CodeStar consola.

Para administrar la clave de SSH pública

1. Ejecute el comando `ssh-keygen` desde un terminal o una ventana de emulador de Bash para generar un par de claves de SSH pública y privada en el equipo local. Puede generar una clave en cualquier formato que permita Amazon EC2. Para obtener información sobre los formatos aceptados, consulte [Importar su propia clave pública a Amazon EC2](#). Lo ideal sería generar una clave que sea SSH-2 RSA, en formato OpenSSH y que contenga 2 048 bits. La clave pública se almacena en un archivo con la extensión `.pub`.
2. Abra la AWS CodeStar consola en <https://console.aws.amazon.com/codestar/>.

- Elija un proyecto en el que sea miembro del equipo.
3. En el panel de navegación, seleccione Equipo.
 4. En la página Miembros del equipo, busque el nombre del usuario de IAM y, a continuación, seleccione Editar.
 5. En la página Editar el miembro del equipo, en Acceso remoto, habilite Permitir el acceso de SSH a las instancias del proyecto.
 6. En el cuadro Clave pública SSH, pegue la clave pública y, a continuación, seleccione Editar el miembro del equipo.

Note

Puede cambiar su clave pública eliminando la clave antigua en este campo y pegando una nueva. Del mismo modo, puede eliminar una clave pública; para ello, borre el contenido de este campo y, a continuación, seleccione Editar el miembro del equipo.

Al cambiar o eliminar una clave pública está cambiando su perfil de usuario. No es un cambio según cada proyecto. Dado que la clave está asociada a su perfil, cambiará (o se eliminará) en todos los proyectos en los que le ha concedido acceso remoto.

Al eliminar su clave pública se elimina su acceso a instancias de Amazon EC2 que ejecutan Linux en todos los proyectos en los que le han concedido acceso remoto. Sin embargo, no se cierra ninguna sesión SSH abierta con dicha clave. Asegúrese de cerrar las sesiones abiertas.

Administrar la clave pública (AWS CLI)

Puede utilizarla AWS CLI para administrar su clave pública SSH como parte de su perfil de usuario.

Para administrar la clave pública

1. Ejecute el comando `ssh-keygen` desde un terminal o una ventana de emulador de Bash para generar un par de claves de SSH pública y privada en el equipo local. Puede generar una clave en cualquier formato que permita Amazon EC2. Para obtener información sobre los formatos aceptados, consulte [Importar su propia clave pública a Amazon EC2](#). Lo ideal sería generar una clave que sea SSH-2 RSA, en formato OpenSSH y que contenga 2 048 bits. La clave pública se almacena en un archivo con la extensión `.pub`.

2. Para añadir o cambiar tu clave pública SSH en tu perfil de AWS CodeStar usuario, ejecuta el `update-user-profile` comando con el `--ssh-public-key` parámetro. Por ejemplo:

```
aws codestar update-user-profile --user-arn arn:aws:iam:111111111111:user/Jane_Doe
--ssh-key-id EXAMPLE1
```

Este comando devuelve un resultado similar al siguiente:

```
{
  "createdTimestamp":1.491439687681E9,
  "displayName":"Jane Doe",
  "emailAddress":"jane.doe@example.com",
  "lastModifiedTimestamp":1.491442730598E9,
  "sshPublicKey":"EXAMPLE1",
  "userArn":"arn:aws:iam::111111111111:user/Jane_Doe"
}
```

Conectarse a la instancia de Amazon EC2 con la clave privada

Asegúrese de que ya haya creado un par de claves de Amazon EC2. Añada su clave pública a su perfil de usuario en AWS CodeStar. Para crear un par de claves, consulte [Paso 4: crear un par de claves de Amazon EC2 para proyectos de AWS CodeStar](#). Para añadir la clave pública a su perfil de usuario, consulte las instrucciones indicadas anteriormente en este tema.

Para conectarse a una instancia de Linux de Amazon EC2 mediante la clave privada

1. Con el proyecto abierto en la AWS CodeStar consola, en el panel de navegación, selecciona Proyecto.
2. En Recursos del proyecto, elija el enlace ARN en la fila donde Tipo sea Amazon EC2 y Nombre empiece por instancia.
3. En la consola de Amazon EC2, seleccione Conectar.
4. Siga las instrucciones en el cuadro de diálogo Conéctese a la instancia.

Para el nombre de usuario, utilice `ec2-user`. Si utiliza un nombre de usuario incorrecto, no podrá conectarse a la instancia.

Para obtener más información, consulte los siguientes recursos en la Guía del usuario de Amazon EC2.

- [Conexión a la instancia de Linux mediante SSH](#)
- [Conexión a la instancia Linux desde Windows utilizando PuTTY](#)
- [Conexión a su instancia de Linux mediante MindTerm](#)

Seguridad en AWS CodeStar

La seguridad en la nube de AWS es la mayor prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y usted. El [modelo de responsabilidad compartida](#) la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los [Programas de conformidad de AWS](#). Para obtener información sobre los programas de conformidad que se aplican a AWS CodeStar, consulte [Servicios de AWS en el ámbito del programa de conformidad](#).
- Seguridad en la nube: su responsabilidad se determina según el servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza AWS CodeStar. En los siguientes temas, se le mostrará cómo configurar AWS CodeStar para satisfacer sus objetivos de seguridad y conformidad. También puede aprender a utilizar otros servicios de AWS que lo ayuden a monitorear y proteger los recursos de AWS CodeStar.

Al crear políticas personalizadas y utilizar los límites de permisos en AWS CodeStar, garantice el acceso a los privilegios mínimos; para ello conceda solo los permisos necesarios para realizar una tarea y limite los permisos a los recursos específicos. Para evitar que los miembros de otros proyectos accedan a los recursos de su proyecto, otorgue a los miembros de la organización permisos independientes para cada proyecto de AWS CodeStar. Como práctica recomendada, cree una cuenta de proyecto para cada miembro y, a continuación, asigne a esa cuenta un acceso basado en roles.

Por ejemplo, puede utilizar un servicio como AWS Control Tower con organizaciones de AWS para aprovisionar cuentas para cada rol de desarrollador en un grupo de DevOps. A continuación, puede asignar permisos a esas cuentas. Los permisos generales se aplican a la cuenta, pero el usuario tiene acceso limitado a los recursos ajenos al proyecto.

Para obtener más información sobre la administración del acceso con privilegios mínimos a los recursos de AWS mediante una estrategia de varias cuentas, consulte la [estrategia de varias cuentas de AWS para su zona de aterrizaje](#) en la Guía del usuario de AWS Control Tower.

Temas

- [Protección de datos en AWS CodeStar](#)
- [Identity and Access Management para AWS CodeStar](#)
- [Registro de llamadas a la API de AWS CodeStar con AWS CloudTrail](#)
- [Validación de la conformidad para AWS CodeStar](#)
- [Resiliencia en AWS CodeStar](#)
- [Seguridad de infraestructura en AWS CodeStar](#)

Protección de datos en AWS CodeStar

El modelo de [responsabilidad AWS compartida modelo](#) se aplica a la protección de datos en AWS CodeStar. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta todos los Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Usted también es responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulte la sección [Privacidad de datos FAQ](#). Para obtener información sobre la protección de datos en Europa, consulte el [modelo de responsabilidad AWS compartida y](#) la entrada del GDPR blog sobre AWS seguridad.

Con fines de protección de datos, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utilice la autenticación multifactorial (MFA) con cada cuenta.
- Use SSL/TLS para comunicarse con AWS los recursos. Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Configure API y registre la actividad del usuario con AWS CloudTrail.
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.

- Utilice servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita entre FIPS 140 y 3 módulos criptográficos validados para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un FIPS terminal. Para obtener más información sobre los FIPS puntos finales disponibles, consulte la [Norma federal de procesamiento de información \(\) FIPS 140-3](#).

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye cuando trabaja con CodeStar o Servicios de AWS utiliza la consola, API AWS CLI, o. AWS SDKs Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se puede emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, le recomendamos encarecidamente que no incluya la información sobre las credenciales URL para validar la solicitud a ese servidor.

Cifrado de datos en AWS CodeStar

De forma predeterminada, AWS CodeStar cifra la información que almacena sobre el proyecto. Todo lo que no sea su ID de proyecto, está cifrado en reposo, como el nombre del proyecto, la descripción y los correos electrónicos de los usuarios. Evita incluir información personal en tu proyectoIDs. AWS CodeStar también cifra la información en tránsito de forma predeterminada. No se requiere ninguna acción por parte del cliente ni para el cifrado en reposo ni para el cifrado en tránsito.

Identity and Access Management para AWS CodeStar

AWS Identity and Access Management (IAM) es una Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a AWS los recursos. IAMlos administradores controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar AWS CodeStar los recursos. IAMes una Servicio de AWS que puede utilizar sin coste adicional.

Temas

- [Público](#)
- [Autenticación con identidades](#)
- [Administración de acceso mediante políticas](#)
- [¿Cómo AWS CodeStar funciona con IAM](#)
- [AWS CodeStar Políticas y permisos a nivel de proyecto](#)

- [AWS CodeStarEjemplos de políticas basadas en la identidad](#)
- [Solución de problemas AWS CodeStar de identidad y acceso](#)

Público

La forma de usar AWS Identity and Access Management (IAM) varía según el trabajo en el que se realice AWS CodeStar.

Usuario del servicio: si utiliza el AWS CodeStar servicio para realizar su trabajo, el administrador le proporcionará las credenciales y los permisos que necesita. A medida que vaya utilizando más AWS CodeStar funciones para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarlo a solicitar los permisos correctos al administrador. Si no puede acceder a una función en AWS CodeStar, consulte [Solución de problemas AWS CodeStar de identidad y acceso](#).

Administrador de servicios: si está a cargo de AWS CodeStar los recursos de su empresa, probablemente tenga acceso total a ellos AWS CodeStar. Su trabajo consiste en determinar a qué AWS CodeStar funciones y recursos deben acceder los usuarios del servicio. A continuación, debe enviar solicitudes al IAM administrador para cambiar los permisos de los usuarios del servicio. Revise la información de esta página para comprender los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar IAM con AWS CodeStar, consulte [¿Cómo AWS CodeStar funciona con IAM](#).

IAM administrador: si es IAM administrador, puede que desee obtener más información sobre cómo puede redactar políticas para administrar el acceso a ellas AWS CodeStar. Para ver ejemplos de políticas AWS CodeStar basadas en la identidad que puede utilizar IAM, consulte [AWS CodeStarEjemplos de políticas basadas en la identidad](#)

Autenticación con identidades

La autenticación es la forma de iniciar sesión AWS con sus credenciales de identidad. Debe estar autenticado (con quien haya iniciado sesión AWS) como IAM usuario o asumiendo un IAM rol.

Usuario raíz de la cuenta de AWS

Puede iniciar sesión AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad. AWS IAM Identity Center Los usuarios (IAM Identity Center), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, el administrador

configuró previamente la federación de identidades mediante roles. IAM Cuando accede AWS mediante la federación, asume indirectamente un rol.

Según el tipo de usuario que sea, puede iniciar sesión en el portal AWS Management Console o en el de AWS acceso. Para obtener más información sobre cómo iniciar sesión AWS, consulte [Cómo iniciar sesión Cuenta de AWS en su](#) Guía del AWS Sign-In usuario.

Si accede AWS mediante programación, AWS incluye un kit de desarrollo de software (SDK) y una interfaz de línea de comandos (CLI) para firmar criptográficamente sus solicitudes con sus credenciales. Si no utilizas AWS herramientas, debes firmar las solicitudes tú mismo. Para obtener más información sobre cómo usar el método recomendado para firmar las solicitudes usted mismo, consulte [Firmar AWS API las solicitudes](#) en la Guía del IAM usuario.

Independientemente del método de autenticación que use, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, le AWS recomienda que utilice la autenticación multifactorial (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte [Autenticación multifactorial](#) en la Guía del AWS IAM Identity Center usuario y [Uso de la autenticación multifactorial \(MFA\) AWS en](#) la Guía del IAM usuario.

Cuenta de AWS usuario root

Al crear una Cuenta de AWS, comienza con una identidad de inicio de sesión que tiene acceso completo a todos Servicios de AWS los recursos de la cuenta. Esta identidad se denomina usuario Cuenta de AWS raíz y se accede a ella iniciando sesión con la dirección de correo electrónico y la contraseña que utilizaste para crear la cuenta. Recomendamos encarecidamente que no utilice el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de tareas que requieren que inicie sesión como usuario root, consulte [Tareas que requieren credenciales de usuario root](#) en la Guía del IAM usuario.

IAMusuarios y grupos

Un [IAMusuario](#) es una identidad dentro de ti Cuenta de AWS que tiene permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos utilizar credenciales temporales en lugar de crear IAM usuarios con credenciales de larga duración, como contraseñas y claves de acceso. Sin embargo, si tiene casos de uso específicos que requieren credenciales a largo plazo con IAM los usuarios, le recomendamos que rote las claves de acceso. Para obtener más información, consulte [Rotar las claves de acceso con regularidad para los casos de uso que requieran credenciales de larga duración](#) en la Guía del IAM usuario.

Un [IAM grupo](#) es una identidad que especifica un conjunto de IAM usuarios. No puede iniciar sesión como grupo. Puede usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos para grandes conjuntos de usuarios. Por ejemplo, puede asignar un nombre a un grupo IAMAdmins y concederle permisos para administrar IAM los recursos.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales de larga duración permanentes; no obstante, los roles proporcionan credenciales temporales. Para obtener más información, consulte [Cuándo crear un IAM usuario \(en lugar de un rol\)](#) en la Guía del IAM usuario.

IAM Funciones

Un [IAM rol](#) es una identidad dentro de tu Cuenta de AWS que tiene permisos específicos. Es similar a un IAM usuario, pero no está asociado a una persona específica. Puede asumir temporalmente un IAM rol en el AWS Management Console [cambiando de rol](#). Puede asumir un rol llamando a una AWS API operación AWS CLI o utilizando una operación personalizada URL. Para obtener más información sobre los métodos de uso de roles, consulte [Uso de IAM roles](#) en la Guía del IAM usuario.

IAM los roles con credenciales temporales son útiles en las siguientes situaciones:

- **Acceso de usuario federado:** para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información sobre los roles para la federación, consulte [Creación de un rol para un proveedor de identidad externo](#) en la Guía del IAM usuario. Si usa IAM Identity Center, configura un conjunto de permisos. Para controlar a qué pueden acceder sus identidades después de autenticarse, IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM. Para obtener información acerca de los conjuntos de permisos, consulte [Conjuntos de permisos](#) en la Guía del usuario de AWS IAM Identity Center .
- **Permisos IAM de usuario temporales:** un IAM usuario o rol puede asumir un IAM rol para asumir temporalmente diferentes permisos para una tarea específica.
- **Acceso multicuenta:** puedes usar un IAM rol para permitir que alguien (un responsable de confianza) de una cuenta diferente acceda a los recursos de tu cuenta. Los roles son la forma principal de conceder acceso entre cuentas. Sin embargo, con algunos Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para conocer

la diferencia entre las funciones y las políticas basadas en recursos para el acceso multicuenta, consulta el tema sobre el acceso a los [recursos entre cuentas IAM en](#) la Guía del IAM usuario.

- Acceso entre servicios: algunos Servicios de AWS utilizan funciones en otros. Servicios de AWS Por ejemplo, cuando realizas una llamada en un servicio, es habitual que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado al servicio.
- Sesiones de acceso directo (FAS): cuando utilizas un IAM usuario o un rol para realizar acciones en AWS ellas, se te considera director. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos del principal que llama a un Servicio de AWS, junto con los que solicitan, Servicio de AWS para realizar solicitudes a los servicios descendentes. FAS las solicitudes solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información detallada sobre la política a la hora de realizar FAS solicitudes, consulte [Reenviar las sesiones de acceso](#).
- Función de servicio: una función de servicio es una [IAM función](#) que un servicio asume para realizar acciones en su nombre. Un IAM administrador puede crear, modificar y eliminar un rol de servicio desde dentro IAM. Para obtener más información, consulte [Crear un rol para delegar permisos Servicio de AWS en un rol](#) en el IAM Manual del usuario.
- Función vinculada a un servicio: una función vinculada a un servicio es un tipo de función de servicio que está vinculada a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.
- Aplicaciones que se ejecutan en Amazon EC2: puedes usar un IAM rol para administrar las credenciales temporales de las aplicaciones que se ejecutan en una EC2 instancia y que realizan AWS CLI o AWS API solicitan. Esto es preferible a almacenar las claves de acceso dentro de la EC2 instancia. Para asignar un AWS rol a una EC2 instancia y ponerlo a disposición de todas sus aplicaciones, debe crear un perfil de instancia adjunto a la instancia. Un perfil de instancia contiene el rol y permite que los programas que se ejecutan en la EC2 instancia obtengan credenciales temporales. Para obtener más información, consulte [Uso de un IAM rol para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del IAM usuario.

Para saber si se deben usar IAM roles o IAM usuarios, consulte [Cuándo crear un IAM rol \(en lugar de un usuario\)](#) en la Guía del IAM usuario.

Administración de acceso mediante políticas

El acceso se controla AWS creando políticas y adjuntándolas a AWS identidades o recursos. Una política es un objeto AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando un director (usuario, usuario raíz o sesión de rol) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan AWS como JSON documentos. Para obtener más información sobre la estructura y el contenido de los documentos de JSON políticas, consulte [Descripción general de JSON las políticas](#) en la Guía del IAM usuario.

Los administradores pueden usar AWS JSON las políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Para conceder a los usuarios permiso para realizar acciones en los recursos que necesitan, un IAM administrador puede crear IAM políticas. A continuación, el administrador puede añadir las IAM políticas a las funciones y los usuarios pueden asumir las funciones.

IAM las políticas definen los permisos para una acción independientemente del método que se utilice para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la acción `iam:GetRole`. Un usuario con esa política puede obtener información sobre el rol de AWS Management Console AWS CLI, el o el AWS API.

Políticas basadas en identidad

Las políticas basadas en la identidad son documentos de política de JSON permisos que se pueden adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte [Creación de IAM políticas](#) en la Guía del usuario. IAM

Las políticas basadas en identidades pueden clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puede adjuntar a varios usuarios, grupos y funciones de su empresa. Cuenta de AWS Las políticas administradas incluyen políticas AWS administradas y políticas administradas por el cliente. Para saber cómo elegir entre una política

gestionada o una política integrada, consulte [Elegir entre políticas gestionadas y políticas integradas en la Guía del IAM](#) usuario.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de JSON política que se adjuntan a un recurso. Algunos ejemplos de políticas basadas en recursos son las políticas de confianza de IAM roles y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe [especificar una entidad principal](#) en una política en función de recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No puede usar políticas AWS administradas desde una política IAM basada en recursos.

Listas de control de acceso () ACLs

Las listas de control de acceso (ACLs) controlan qué directores (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de JSON políticas.

Amazon S3 AWS WAF y Amazon VPC son ejemplos de servicios compatibles ACLs. Para obtener más información ACLs, consulte la [descripción general de la lista de control de acceso \(ACL\)](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite tipos de políticas adicionales y menos comunes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Límites de permisos: un límite de permisos es una función avanzada en la que se establecen los permisos máximos que una política basada en la identidad puede conceder a una IAM entidad (IAM usuario o rol). Puede establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites

de los permisos, consulte los [límites de los permisos para IAM las entidades](#) en la Guía del IAM usuario.

- Políticas de control de servicios (SCPs): SCPs son JSON políticas que especifican los permisos máximos para una organización o unidad organizativa (OU) AWS Organizations. AWS Organizations es un servicio para agrupar y administrar de forma centralizada varios de los Cuentas de AWS que son propiedad de su empresa. Si habilitas todas las funciones de una organización, puedes aplicar políticas de control de servicios (SCPs) a una o a todas tus cuentas. SCP limita los permisos de las entidades en las cuentas de los miembros, incluidas las de cada una Usuario raíz de la cuenta de AWS. Para obtener más información sobre Organizations SCPs, consulte las [políticas de control de servicios](#) en la Guía del AWS Organizations usuario.
- Políticas de sesión: las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidades del rol y las políticas de la sesión. Los permisos también pueden proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información, consulte [las políticas de sesión](#) en la Guía del IAM usuario.

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para saber cómo se AWS determina si se debe permitir una solicitud cuando se trata de varios tipos de políticas, consulte la [lógica de evaluación de políticas](#) en la Guía del IAM usuario.

¿Cómo AWS CodeStar funciona con IAM

Antes de administrar el acceso a AWS CodeStar, debe comprender qué IAM funciones están disponibles para su uso AWS CodeStar. IAM Para obtener una visión general de cómo AWS CodeStar funcionan con otros AWS servicios IAM, consulte [AWS los servicios con los que funcionan IAM](#) en la Guía del IAM usuario.

Temas

- [AWS CodeStar Políticas basadas en la identidad](#)
- [AWS CodeStar Políticas basadas en recursos](#)
- [Autorización basada en etiquetas de AWS CodeStar](#)
- [AWS CodeStar IAM Funciones](#)

- [IAM acceso del usuario a AWS CodeStar](#)
- [Acceso de usuarios federados a AWS CodeStar](#)
- [Uso de credenciales temporales con AWS CodeStar](#)
- [Roles vinculados a servicios](#)
- [Roles de servicio](#)

AWS CodeStar Políticas basadas en la identidad

Con las políticas IAM basadas en la identidad, puede especificar las acciones y los recursos permitidos o denegados y las condiciones en las que se permiten o deniegan las acciones. AWS CodeStar crea varias políticas basadas en la identidad en su nombre, que le permiten AWS CodeStar crear y gestionar recursos dentro del ámbito de un proyecto. AWS CodeStar AWS CodeStar admite acciones, recursos y claves de condición específicos. Para obtener información sobre todos los elementos que se utilizan en una JSON política, consulte la [Referencia sobre los elementos de la IAM JSON política](#) en la Guía del IAM usuario.

Acciones

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El `Action` elemento de una JSON política describe las acciones que puede utilizar para permitir o denegar el acceso en una política. Las acciones de política suelen tener el mismo nombre que la AWS API operación asociada. Hay algunas excepciones, como las acciones que solo permiten permisos y que no tienen una operación coincidente. API También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Las acciones políticas AWS CodeStar utilizan el siguiente prefijo antes de la acción: `codestar:`. Por ejemplo, para permitir que un IAM usuario específico edite los atributos de un AWS CodeStar proyecto, como su descripción, puede utilizar la siguiente declaración de política:

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

```
"Action" : [
  "codestar:UpdateProject"
],
"Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
}
]
}
```

Las instrucciones de la política deben incluir un elemento `Action` o un elemento `NotAction`. AWS CodeStar define su propio conjunto de acciones que describen las tareas que puede realizar con este servicio.

Para especificar varias acciones en una única instrucción, sepárelas con comas del siguiente modo:

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

Puede utilizar caracteres comodín para especificar varias acciones (*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `List`, incluya la siguiente acción:

```
"Action": "codestar:List*"
```

Para ver una lista de AWS CodeStar acciones, consulte [las acciones definidas por AWS CodeStar](#) en la Guía del IAM usuario.

Recursos

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` JSON de política especifica el objeto o los objetos a los que se aplica la acción. Las instrucciones deben contener un elemento `Resource` o `NotResource`. Como práctica recomendada, especifique un recurso mediante su [nombre de recurso de Amazon \(ARN\)](#). Puede hacerlo para acciones que admitan un tipo de recurso específico, conocido como permisos de nivel de recurso.

Para las acciones que no admiten permisos de nivel de recurso, como las operaciones de descripción, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

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

El recurso AWS CodeStar del proyecto tiene lo siguienteARN:

```
arn:aws:codestar:region:account:project/resource-specifier
```

Para obtener más información sobre el formato deARNs, consulte [Amazon Resource Names \(ARNs\) y AWS Service Namespaces](#).

Por ejemplo, lo siguiente especifica el nombre del AWS CodeStar proyecto *my-first-projec* registrado 111111111111 en la AWS cuenta de la región: AWS us-east-2

```
arn:aws:codestar:us-east-2:111111111111:project/my-first-projec
```

A continuación se especifica cualquier AWS CodeStar proyecto que comience con el nombre my-proj registrado 111111111111 en la AWS cuenta de la AWS regiónus-east-2:

```
arn:aws:codestar:us-east-2:111111111111:project/my-proj*
```

Algunas AWS CodeStar acciones, como la de enumerar proyectos, no se pueden realizar en un recurso. En dichos casos, debe utilizar el carácter comodín (*).

```
"LisProjects": "*"
```

Para ver una lista de los tipos de AWS CodeStar recursos y sus respectivos tiposARNs, consulte [los recursos definidos por AWS CodeStar](#) en la Guía del IAM usuario. Para saber con qué acciones puede especificar cada recurso, consulte [Acciones definidas por AWS CodeStar](#). ARN

Claves de condición

AWS CodeStar no proporciona ninguna clave de condición específica del servicio, pero sí admite el uso de algunas claves de condición globales. Para ver todas las claves de condición AWS globales, consulte las claves de [contexto de condición AWS globales](#) en la Guía del IAMusuario.

Ejemplos

Para ver ejemplos de políticas AWS CodeStar basadas en la identidad, consulte [AWS CodeStarEjemplos de políticas basadas en la identidad](#)

AWS CodeStar Políticas basadas en recursos

AWS CodeStar no admite políticas basadas en los recursos.

Autorización basada en etiquetas de AWS CodeStar

Puedes adjuntar etiquetas a los AWS CodeStar proyectos o pasarles etiquetas en una solicitud. AWS CodeStar Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el [elemento de condición](#) de una política utilizando las claves de condición `codestar:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`. Para obtener más información sobre el etiquetado de AWS CodeStar recursos, consulte [the section called “Trabajar con etiquetas de proyectos”](#).

Para ver un ejemplo de política basada en la identidad para limitar el acceso a un AWS CodeStar proyecto en función de las etiquetas de ese proyecto, consulte [Visualización de proyectos de AWS CodeStar basados en etiquetas](#)

AWS CodeStar IAMFunciones

Un [IAMrol](#) es una entidad de tu AWS cuenta que tiene permisos específicos.

Se puede utilizar AWS CodeStar como [IAMusuario](#), usuario federado, usuario raíz o rol asumido. Todos los tipos de usuarios con los permisos adecuados pueden administrar los permisos del proyecto para sus AWS recursos, pero AWS CodeStar administran los permisos del proyecto automáticamente para IAM los usuarios. [IAMlas políticas](#) y los [roles](#) otorgan permisos y acceso a ese usuario en función del rol del proyecto. Puede usar la IAM consola para crear otras políticas que asignen AWS CodeStar y otros permisos a un IAM usuario.

Por ejemplo, puede que quiera permitir a un usuario ver pero no cambiar un proyecto de AWS CodeStar . En este caso, se añade el IAM usuario a un AWS CodeStar proyecto con el rol de espectador. Cada AWS CodeStar proyecto tiene un conjunto de políticas que te ayudan a controlar el acceso al proyecto. Además, puedes controlar a qué usuarios tienen acceso AWS CodeStar.

AWS CodeStar el acceso se gestiona de forma diferente para IAM los usuarios y los usuarios federados. Solo se pueden añadir IAM usuarios a los equipos. Para conceder a IAM los usuarios permisos para acceder a los proyectos, añada al usuario al equipo del proyecto y asigne un rol

al usuario. Para conceder permisos a los usuarios federados para acceder a los proyectos, debe adjuntar manualmente la política gestionada del rol del AWS CodeStar proyecto al rol del usuario federado.

En la siguiente tabla se resumen las herramientas disponibles para cada tipo de acceso.

Función de permisos	IAM usuario	Usuario federado	Usuario raíz
SSH administración de claves para acceso remoto para proyectos de Amazon EC2 y Elastic Beanstalk	✓		
AWS CodeCommit SSH acceso	✓		
IAM permisos de usuario gestionados por AWS CodeStar	✓		
Permisos de proyectos administrados manualmente		✓	✓
Los usuarios pueden añadirse al proyecto como miembros del equipo	✓		

IAM acceso del usuario a AWS CodeStar

Al añadir un IAM usuario a un proyecto y elegir un rol para el usuario, AWS CodeStar se aplica automáticamente la política correspondiente al IAM usuario. En el IAM caso de los usuarios, no es necesario adjuntar o administrar directamente las políticas o los permisos IAM. Para obtener información sobre cómo añadir un IAM usuario a un AWS CodeStar proyecto, consulte [Añadir miembros de equipo a un proyecto de AWS CodeStar](#). Para obtener información sobre cómo eliminar un IAM usuario de un AWS CodeStar proyecto, consulte [Eliminación de miembros del equipo de un proyecto de AWS CodeStar](#).

Adjunte una política en línea a un usuario IAM

Al añadir un usuario a un proyecto, AWS CodeStar se adjunta automáticamente la política gestionada al proyecto que coincida con la función del usuario. No debes adjuntar manualmente una política AWS CodeStar gestionada para un proyecto a un IAM usuario. Con la excepción de `AWSCodeStarFullAccess`, no recomendamos adjuntar políticas que cambien los permisos

de un IAM usuario en un AWS CodeStar proyecto. Si decide crear y adjuntar sus propias políticas, consulte [Añadir y quitar permisos de IAM identidad](#) en la Guía del IAM usuario.

Acceso de usuarios federados a AWS CodeStar

En lugar de crear un IAM usuario o utilizar el usuario raíz, puede utilizar las identidades de los usuarios AWS Directory Service, el directorio de usuarios de su empresa, un proveedor de identidades web o IAM los usuarios que asuman funciones. Esto se conoce como usuarios federados.

Conceda a los usuarios federados acceso a su AWS CodeStar proyecto adjuntando manualmente a la función del usuario las políticas gestionadas descritas en [Políticas y permisos a AWS CodeStar nivel de proyecto](#). IAM Debe adjuntar la política de propietario, colaborador o espectador después de AWS CodeStar crear los recursos y funciones del proyecto. IAM

Requisitos previos:

- Debe tener configurado un proveedor de identidad. Por ejemplo, puedes configurar un proveedor de SAML identidad y configurar la AWS autenticación a través del proveedor. Para obtener más información sobre la configuración de un proveedor de identidad, consulte [Creación de proveedores de IAM identidad](#). Para obtener más información sobre la SAML federación, consulte [Acerca de la federación SAML basada en 2.0](#).
- Tiene que haber creado un rol que asuma un usuario federado cuando se solicita acceso a través de un [proveedor de identidad](#). Se debe adjuntar a la función una política de STS confianza que permita a los usuarios federados asumir la función. Para obtener más información, consulte [Usuarios y roles federados](#) en la Guía del IAM usuario.
- Debe haber creado su AWS CodeStar proyecto y conocer su ID.

Para obtener más información sobre cómo crear un rol para proveedores de identidad, consulte [Creación de un rol para un proveedor de identidad de terceros \(federación\)](#).

Adjunte la política `AWSCodeStarFullAccess` gestionada al rol del usuario federado

Otorgue permisos a un usuario federado para crear un proyecto asociándole la política administrada `AWSCodeStarFullAccess`. Para realizar estos pasos, debe haber iniciado sesión en la consola como usuario raíz, usuario administrador de la cuenta o usuario o IAM usuario federado con la política `AdministratorAccess` gestionada asociada o equivalente.

Note

Después de crear el proyecto, los permisos de propietario del proyecto no se aplican automáticamente. Use un rol con permisos administrativos para la cuenta y asocie la política administrada de propietario, tal y como se describe en [Adjunte la política de su proyecto gestionada por AWS CodeStar espectadores, colaboradores o propietarios a la función del usuario federado](#).

1. Abra la IAM consola. En el panel de navegación, seleccione Políticas.
2. Escriba `AWSCodeStarFullAccess` en el campo de búsqueda. El nombre de la política se muestra con un tipo de política Administrada por AWS . Puede ampliar la política para ver los permisos en la instrucción de la política.
3. Seleccione el círculo junto a la política y en Acciones de la política, elija Asociar.
4. En la página Resumen, elija la pestaña Entidades asociadas. Elija Asociar.
5. En la página Asociar política, filtre por el rol del usuario federado en el campo de búsqueda. Seleccione la casilla situada junto al nombre del rol y, a continuación, elija Asociar política. La pestaña Entidades asociadas muestra el nuevo adjunto.

Adjunte la política de su proyecto gestionada por AWS CodeStar espectadores, colaboradores o propietarios a la función del usuario federado

Conceda a los usuarios federados acceso a su proyecto asociando la política administrada de propietario, lector o colaborador de al rol del usuario. La política administrada ofrece el nivel adecuado de permisos. A diferencia de IAM los usuarios, debe adjuntar y separar manualmente las políticas administradas para los usuarios federados. Esto equivale a asignar permisos de proyecto a los miembros del equipo en. AWS CodeStar Para realizar estos pasos, debe haber iniciado sesión en la consola como usuario root, como usuario administrador de la cuenta o como IAM usuario o usuario federado con la política `AdministratorAccess` gestionada asociada o equivalente.

Requisitos previos:

- Tiene que haber creado un rol o tiene que existir un rol que asuma el usuario federado.
- Debe saber qué nivel de permisos desea conceder. Las políticas administradas asociada a los roles de propietario, colaborador y lector proporcionan permisos basados en roles al proyecto.

- AWS CodeStar El proyecto debe haberse creado. La política gestionada no estará disponible IAM hasta que se cree el proyecto.
1. Abra la IAM consola. En el panel de navegación, seleccione Políticas.
 2. Escriba el ID de proyecto en el campo de búsqueda. El nombre de la política del proyecto se muestra con un tipo de política Administrada por el cliente. Puede ampliar la política para ver los permisos en la instrucción de la política.
 3. Elija una de estas políticas administradas. Seleccione el círculo junto a la política y en Acciones de la política, elija Asociar.
 4. En la página Resumen, elija la pestaña Entidades asociadas. Elija Asociar.
 5. En la página Asociar política, filtre por el rol del usuario federado en el campo de búsqueda. Seleccione la casilla situada junto al nombre del rol y, a continuación, elija Asociar política. La pestaña Entidades asociadas muestra el nuevo adjunto.

Separe una política AWS CodeStar gestionada de la función del usuario federado

Antes de eliminar el AWS CodeStar proyecto, debe separar manualmente las políticas administradas que haya asociado a la función de un usuario federado. Para realizar estos pasos, debe haber iniciado sesión en la consola como usuario raíz, usuario administrador de la cuenta o usuario o IAM usuario federado con la política `AdministratorAccess` gestionada asociada o equivalente.

1. Abra la IAM consola. En el panel de navegación, seleccione Políticas.
2. Escriba el ID de proyecto en el campo de búsqueda.
3. Seleccione el círculo junto a la política y en Acciones de la política, elija Asociar.
4. En la página Resumen, elija la pestaña Entidades asociadas.
5. Filtre por el rol de usuario federado en el campo de búsqueda. Elija Desasociar.

Adjunte una política AWS Cloud9 gestionada al rol del usuario federado

Si utiliza un entorno de AWS Cloud9 desarrollo, conceda a los usuarios federados el acceso a él adjuntando la política `AWSCloud9User` gestionada a la función del usuario. A diferencia de IAM los usuarios, debe adjuntar y separar manualmente las políticas administradas para los usuarios federados. Para realizar estos pasos, debe haber iniciado sesión en la consola como usuario raíz, usuario administrador de la cuenta o usuario o IAM usuario federado con la política `AdministratorAccess` gestionada asociada o equivalente.

Requisitos previos:

- Tiene que haber creado un rol o tiene que existir un rol que asuma el usuario federado.
- Debe saber qué nivel de permisos desea conceder:
 - La política administrada `AWSCloud9User` permite al usuario hacer lo siguiente:
 - Cree sus propios entornos AWS Cloud9 de desarrollo.
 - Obtener información sobre sus entornos.
 - Cambiar la configuración de sus entornos.
 - La política administrada `AWSCloud9Administrator` permite al usuario hacer lo siguiente para sí mismo o para otros:
 - Crear entornos.
 - Obtener información sobre entornos.
 - Eliminar entornos.
 - Cambiar la configuración de los entornos.

1. Abra la IAM consola. En el panel de navegación, seleccione Políticas.
2. Escriba el nombre de la política en el campo de búsqueda. El nombre de la política administrada se muestra con un tipo de política Administrada por AWS . Puede ampliar la política para ver los permisos en la instrucción de la política.
3. Elija una de estas políticas administradas. Seleccione el círculo junto a la política y en Acciones de la política, elija Asociar.
4. En la página Resumen, elija la pestaña Entidades asociadas. Elija Asociar.
5. En la página Asociar política, filtre por el rol del usuario federado en el campo de búsqueda. Seleccione la casilla situada junto al nombre del rol y, a continuación, elija Attach policy (Asociar política). La pestaña Entidades asociadas muestra el nuevo adjunto.

Separe una política AWS Cloud9 gestionada de la función del usuario federado

Si utiliza un entorno de AWS Cloud9 desarrollo, puede eliminar el acceso de un usuario federado al mismo separando la política que concede el acceso. Para llevar a cabo estos pasos, debe haber iniciado sesión en la consola como usuario raíz, usuario administrador de la cuenta o usuario o IAM usuario federado con la política `AdministratorAccess` gestionada asociada o equivalente.

1. Abra la IAM consola. En el panel de navegación, seleccione Políticas.

2. Escriba el nombre del proyecto en el campo de búsqueda.
3. Seleccione el círculo junto a la política y en Acciones de la política, elija Asociar.
4. En la página Resumen, elija la pestaña Entidades asociadas.
5. Filtre por el rol de usuario federado en el campo de búsqueda. Elija Desasociar.

Uso de credenciales temporales con AWS CodeStar

Puede usar credenciales temporales para iniciar sesión con la federación, asumir un IAM rol o asumir un rol multicuenta. Para obtener credenciales de seguridad temporales, puede llamar a AWS STS API operaciones como [AssumeRole](#) o [GetFederationToken](#).

AWS CodeStar admite el uso de credenciales temporales, pero la funcionalidad de los miembros del AWS CodeStar equipo no funciona para el acceso federado. AWS CodeStar La funcionalidad de miembro del equipo solo permite añadir un IAM usuario como miembro del equipo.

Roles vinculados a servicios

Los [roles vinculados al servicio](#) permiten a AWS los servicios acceder a los recursos de otros servicios para completar una acción en tu nombre. Los roles vinculados al servicio aparecen en tu IAM cuenta y son propiedad del servicio. Un administrador de puede ver, pero no editar, los permisos de los roles vinculados a servicios.

AWS CodeStar no admite funciones vinculadas al servicio.

Roles de servicio

Esta característica permite que un servicio asuma un [rol de servicio](#) en su nombre. Este rol permite que el servicio obtenga acceso a los recursos de otros servicios para completar una acción en su nombre. Los roles de servicio aparecen en su IAM cuenta y son propiedad de la cuenta. Esto significa que un administrador puede cambiar los permisos de este rol. Sin embargo, hacerlo podría deteriorar la funcionalidad del servicio.

AWS CodeStar admite funciones de servicio. AWS CodeStar utiliza un rol de `aws-codestar-service-role` cuando crea y administra los recursos del proyecto. Para obtener más información, consulte [los términos y conceptos de las funciones](#) en la Guía del IAM usuario.

⚠ Important

Para crear este rol de servicio, debe haber iniciado sesión como usuario administrador de o como cuenta raíz. Para obtener más información, consulte [Solo acceso por primera vez: sus credenciales de usuario raíz](#) y [Creación del primer usuario y grupo de administrador](#) en la Guía del IAM usuario.

Este rol se crea para usted la primera vez que crea un proyecto en AWS CodeStar. El rol de servicio actúa en su nombre para:

- Crear los recursos que elija al crear un proyecto.
- Muestra información sobre esos recursos en el panel de control AWS CodeStar del proyecto.

También actúa en su nombre al administrar los recursos de un proyecto. Para ver un ejemplo de esta instrucción de política, consulte [AWSCodeStarServiceRole Política](#).

Además, AWS CodeStar crea varios roles de servicio específicos del proyecto, según el tipo de proyecto. AWS CloudFormation y se crean funciones de cadena de herramientas para cada tipo de proyecto.

- AWS CloudFormation los roles AWS CodeStar permiten acceder AWS CloudFormation para crear y modificar pilas para tu AWS CodeStar proyecto.
- Los roles de la cadena de herramientas permiten acceder AWS CodeStar a otros AWS servicios para crear y modificar recursos para su AWS CodeStar proyecto.

AWS CodeStar Políticas y permisos a nivel de proyecto

Al crear un proyecto, AWS CodeStar crea las IAM funciones y políticas que necesita para administrar los recursos del proyecto. Las políticas se dividen en tres categorías:

- IAMpolíticas para los miembros del equipo del proyecto.
- IAMpolíticas para las funciones de los trabajadores.
- IAMpolíticas para una función de ejecución en tiempo de ejecución.

IAMPolíticas para los miembros del equipo

Al crear un proyecto, AWS CodeStar crea tres políticas administradas por el cliente para que el propietario, el colaborador y el espectador puedan acceder al proyecto. Todos los AWS CodeStar proyectos contienen IAM políticas para estos tres niveles de acceso. Estos niveles de acceso son específicos del proyecto y se definen mediante una política IAM gestionada con un nombre estándar, donde *project-id* es el ID del AWS CodeStar proyecto (por ejemplo, *my-first-projec*):

- CodeStar_*project-id*_Owner
- CodeStar_*project-id*_Contributor
- CodeStar_*project-id*_Viewer

Important

Estas políticas están sujetas a cambios por parte de AWS CodeStar. No deben editarse manualmente. Si desea añadir o cambiar permisos, adjunte políticas adicionales al IAM usuario.

A medida que agrega miembros del equipo (IAMusuarios) al proyecto y elige sus niveles de acceso, la política correspondiente se adjunta al IAM usuario, lo que le otorga el conjunto de permisos adecuado para actuar con los recursos del proyecto. En la mayoría de los casos, no es necesario adjuntar o administrar directamente las políticas o los permisosIAM. No se recomienda adjuntar manualmente una política de nivel de AWS CodeStar acceso a un IAM usuario. Si es absolutamente necesario, como complemento de una política de nivel de AWS CodeStar acceso, puede crear sus propias políticas administradas o integradas para aplicar su propio nivel de permisos a un IAM usuario.

Las políticas están estrechamente circunscritas a los recursos del proyecto y a acciones específicas. A medida que se añaden nuevos recursos a la infraestructura, AWS CodeStar intenta actualizar las políticas de los miembros del equipo para incluir permisos de acceso al nuevo recurso, si se trata de uno de los tipos de recursos compatibles.

Note

Las políticas de niveles de acceso de un AWS CodeStar proyecto se aplican únicamente a ese proyecto. Esto ayuda a garantizar que los usuarios solo puedan ver e interactuar con los

AWS CodeStar proyectos para los que tienen permisos, en el nivel que determine su función. Solo los usuarios que crean AWS CodeStar proyectos deben tener aplicada una política que permita el acceso a todos los AWS CodeStar recursos, independientemente del proyecto.

Todas las políticas de nivel de AWS CodeStar acceso varían en función de los AWS recursos asociados al proyecto al que estén asociados los niveles de acceso. A diferencia de otros servicios de AWS , estas políticas se personalizan cuando se crea el proyecto y se actualizan a medida que cambian los recursos del proyecto. Por lo tanto, no hay una política administrada canónica de propietario, colaborador o lector.

AWS CodeStar Política de roles de propietario

La política gestionada por el CodeStar_*project-id*_Owner cliente permite al usuario realizar todas las acciones AWS CodeStar del proyecto sin restricciones. Esta es la única política que permite a un usuario añadir o eliminar miembros del equipo. El contenido de la política varía según los recursos asociados al proyecto. Consulte [AWS CodeStar Política de roles de propietario](#) para ver un ejemplo.

Un IAM usuario con esta política puede realizar todas AWS CodeStar las acciones del proyecto, pero a diferencia de un IAM usuario con la `AWSCodeStarFullAccess` política, no puede crear proyectos. El alcance del `codestar:*` permiso está limitado a un recurso específico (el AWS CodeStar proyecto asociado a ese ID de proyecto).

AWS CodeStar Política de roles de colaborador

La política administrada por el cliente CodeStar_*project-id*_Contributor permite al usuario colaborar en el proyecto y cambiar el panel del proyecto, pero no le permite añadir ni eliminar miembros del equipo. El contenido de la política varía según los recursos asociados al proyecto. Consulte [Política del rol de Colaborador de AWS CodeStar](#) para ver un ejemplo.

AWS CodeStar Política de roles de espectadores

La política administrada por el cliente CodeStar_*project-id*_Viewer permite a un usuario ver un proyecto en AWS CodeStar, pero no permite cambiar los recursos ni añadir o eliminar miembros del equipo. El contenido de la política varía según los recursos asociados al proyecto. Consulte [AWS CodeStar Política de roles de espectadores](#) para ver un ejemplo.

IAMPolíticas para las funciones de los trabajadores

Si crea su AWS CodeStar proyecto después del 6 de diciembre de 2018PDT, AWS CodeStar crea dos roles de trabajador `CodeStar-project-id-ToolChain` y `CodeStar-project-id-CloudFormation`. Un rol de trabajador es un IAM rol específico del proyecto que se AWS CodeStar crea para pasarlo a un servicio. Otorga permisos para que el servicio pueda crear recursos y ejecutar acciones en el contexto del proyecto. AWS CodeStar El rol de trabajador de la cadena de herramientas tiene una relación de confianza establecida con los servicios de la cadena de herramientas CodeBuild, como CodeDeploy, y. CodePipeline A los miembros del equipo del proyecto (propietarios y colaboradores) se les concede acceso para transferir el rol de trabajador a servicios posteriores de confianza. Para ver un ejemplo de la instrucción de política insertada para este rol, consulte [AWS CodeStar Política sobre las funciones de los trabajadores de Toolchain \(después del 6 de diciembre de 2018PDT\)](#).

El rol de CloudFormation trabajador incluye permisos para los recursos seleccionados que admite AWS CloudFormation, así como permisos para crear IAM usuarios, roles y políticas en su pila de aplicaciones. También tiene una relación de confianza establecida con AWS CloudFormation. Para mitigar los riesgos de aumento de privilegios y acciones destructivas, la política de AWS CloudFormation funciones incluye una condición que exige el límite de permisos específico del proyecto para cada IAM entidad (usuario o función) creada en el conjunto de infraestructuras. Para ver un ejemplo de la instrucción de política insertada para este rol, consulte [AWS CloudFormation Política de roles de los trabajadores](#).

En el caso de AWS CodeStar los proyectos creados antes del 6 de diciembre de 2018, PDT AWS CodeStar crea funciones de trabajador individuales para los recursos de la cadena de herramientas CodePipeline CodeBuild, como y CloudWatch Events, y también crea una función de trabajador AWS CloudFormation que admite un conjunto limitado de recursos. Cada uno de estos roles tiene una relación de confianza establecida con el servicio correspondiente. A los miembros del equipo del proyecto (propietarios y colaboradores) y algunos de los demás roles de trabajador se les concede acceso para transferir el rol a servicios posteriores de confianza. Los permisos para los roles de trabajador se definen en una política insertada circunscrita a un conjunto básico de acciones que el rol puede llevar a cabo en un conjunto de recursos del proyecto. Estos permisos son estáticos. Incluyen permisos a los recursos que se incluyen en el proyecto en el momento de la creación, pero no se actualizan cuando se añaden nuevos recursos al proyecto. Para obtener ejemplos de estas instrucciones de política, consulte:

- [AWS CloudFormation Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)

- [AWS CodePipeline Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)
- [AWS CodeBuild Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)
- [Política de funciones de CloudWatch los trabajadores de Amazon Events \(antes del 6 de diciembre de 2018PDT\)](#)

IAMPolítica para la función de ejecución

Para los proyectos creados después del 6 de diciembre de 2018PDT, AWS CodeStar crea una función de ejecución genérica para el proyecto de ejemplo en la pila de aplicaciones. El rol se limita a los recursos del proyecto que utilizan la política de límites de permisos. A medida que amplíe el proyecto de ejemplo, podrá crear IAM funciones adicionales, y la política de AWS CloudFormation funciones exige que estas funciones se limiten según el límite de los permisos para evitar la escalada de privilegios. Para obtener más información, consulte [Añadir un rol de IAM a un proyecto](#).

Para los proyectos de Lambda creados antes del 6 de diciembre de 2018PDT, AWS CodeStar crea una función de ejecución de Lambda que tiene una política en línea adjunta con permisos para actuar sobre los recursos de la pila de proyectos. AWS SAM A medida que se agregan nuevos recursos a la SAM plantilla, AWS CodeStar intenta actualizar la política de funciones de ejecución de Lambda para incluir permisos para el nuevo recurso si es uno de los tipos de recursos compatibles.

Límite de permisos de IAM

Después del 6 de diciembre de 2018PDT, al crear un proyecto, AWS CodeStar crea una política administrada por el cliente y asigna esa política como [límite de IAM permisos](#) a las IAM funciones del proyecto. AWS CodeStar requiere que todas IAM las entidades creadas en la pila de aplicaciones tengan un límite de permisos. Un límite de permisos controla los permisos máximos que puede tener el rol, pero no proporciona ningún permiso al rol. Las políticas de permisos definen los permisos para el rol. Esto significa que, con independencia del número de permisos adicionales que se añadan a un rol, cualquier persona que utilice el rol no puede realizar más que las acciones incluidas en el límite de permisos. Para obtener información sobre cómo se evalúan las políticas de permisos y los límites de los permisos, consulte la [lógica de evaluación de políticas](#) en la Guía del IAM usuario.

AWS CodeStar utiliza un límite de permisos específico del proyecto para evitar que los privilegios se transfieran a recursos ajenos al proyecto. El límite de AWS CodeStar permisos incluye los recursos ARNs del proyecto. Para ver un ejemplo de esta instrucción de política, consulte [AWS CodeStar Política de límites de permisos](#).

La AWS CodeStar transformación actualiza esta política al agregar o quitar un recurso compatible del proyecto mediante la pila de aplicaciones (`template.yml`).

Agregue un límite de IAM permisos a los proyectos existentes

Si tienes un AWS CodeStar proyecto que se creó antes del 6 de diciembre de 2018PDT, debes añadir manualmente un límite de permisos a las IAM funciones del proyecto. Como práctica recomendada, le recomendamos que utilice un límite de recursos específico de un proyecto que incluya únicamente recursos en el proyecto para evitar el escalado de privilegios a recursos fuera del proyecto. Siga estos pasos para usar el límite de permisos AWS CodeStar administrados que se actualiza a medida que el proyecto evoluciona.

1. Inicie sesión en la AWS CloudFormation consola y busque la plantilla para la pila de cadenas de herramientas de su proyecto. Esta plantilla se llama `awscodestar-project-id`.
2. Seleccione la plantilla, elija Acciones y, a continuación, elija Ver/editar plantilla en Designer.
3. Localice la sección Resources e incluya el siguiente fragmento de código en la parte superior de la sección.

```
PermissionsBoundaryPolicy:
  Description: Creating an IAM managed policy for defining the permissions boundary
for an AWS CodeStar project
  Type: AWS::IAM::ManagedPolicy
  Properties:
    ManagedPolicyName: !Sub 'CodeStar_${ProjectId}_PermissionsBoundary'
    Description: 'IAM policy to define the permissions boundary for IAM entities
created in an AWS CodeStar project'
    PolicyDocument:
      Version: '2012-10-17'
      Statement:
        - Sid: '1'
          Effect: Allow
          Action: ['*']
          Resource:
            - !Sub 'arn:${AWS::Partition}:cloudformation:${AWS::Region}:
${AWS::AccountId}:stack/awscodestar-${ProjectId}-*'

```

Es posible que necesites IAM permisos adicionales para actualizar la pila desde la AWS CloudFormation consola.

4. (Opcional) Si quieres crear IAM funciones específicas para la aplicación, completa este paso. Desde la IAM consola, actualiza la política integrada asociada al AWS CloudFormation rol de tu proyecto para incluir el siguiente fragmento. Es posible que necesites IAM recursos adicionales para actualizar la política.

```

{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": "arn:aws:iam:::{AccountId}:role/CodeStar-{ProjectId}*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:CreateServiceLinkedRole",
    "iam:GetRole",
    "iam>DeleteRole",
    "iam>DeleteUser"
  ],
  "Resource": "*",
  "Effect": "Allow"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam:AttachUserPolicy",
    "iam:CreateRole",
    "iam:CreateUser",
    "iam>DeleteRolePolicy",
    "iam>DeleteUserPolicy",
    "iam:DetachUserPolicy",
    "iam:DetachRolePolicy",
    "iam:PutUserPermissionsBoundary",
    "iam:PutRolePermissionsBoundary"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "iam:PermissionsBoundary": "arn:aws:iam:::{AccountId}:policy/CodeStar_{ProjectId}_PermissionsBoundary"
    }
  }
},

```

```
    "Effect": "Allow"  
  }
```

5. Introduce un cambio en tu proceso de proyecto para AWS CodeStar actualizar el límite de los permisos con los permisos adecuados.

Para obtener más información, consulte [Añadir un rol de IAM a un proyecto](#).

AWS CodeStarEjemplos de políticas basadas en la identidad

De forma predeterminada, IAM los usuarios y los roles no tienen permiso para crear o modificar AWS CodeStar recursos. Tampoco pueden realizar tareas con AWS Management Console AWS CLI, o AWS API. El administrador debe crear IAM políticas que concedan a los usuarios y roles permisos para realizar API operaciones específicas en los recursos específicos que necesitan. A continuación, el administrador debe adjuntar esas políticas a los IAM usuarios o grupos que requieran esos permisos.

Para obtener información sobre cómo crear una política IAM basada en la identidad con estos documentos de JSON política de ejemplo, consulte [Creación de políticas en la JSON pestaña de la Guía del IAMusuario](#).

Temas

- [Prácticas recomendadas relativas a políticas](#)
- [AWSCodeStarServiceRole Política](#)
- [AWSCodeStarFullAccess Política](#)
- [AWS CodeStar Política de roles de propietario](#)
- [Política del rol de Colaborador de AWS CodeStar](#)
- [AWS CodeStar Política de roles de espectadores](#)
- [AWS CodeStar Política sobre las funciones de los trabajadores de Toolchain \(después del 6 de diciembre de 2018PDT\)](#)
- [AWS CloudFormation Política de roles de los trabajadores](#)
- [AWS CloudFormation Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)
- [AWS CodePipeline Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)

- [AWS CodeBuild Política sobre las funciones de los trabajadores \(antes del 6 de diciembre de 2018PDT\)](#)
- [Política de funciones de CloudWatch los trabajadores de Amazon Events \(antes del 6 de diciembre de 2018PDT\)](#)
- [AWS CodeStar Política de límites de permisos](#)
- [Listado de recursos para un proyecto](#)
- [Uso de la AWS CodeStar consola](#)
- [Permitir a los usuarios ver sus propios permisos](#)
- [Actualización de un proyecto de AWS CodeStar](#)
- [Añadir un miembro de equipo a un proyecto](#)
- [Listar los perfiles de usuario asociados a una AWS cuenta](#)
- [Visualización de proyectos de AWS CodeStar basados en etiquetas](#)
- [AWS CodeStar actualizaciones de las políticas AWS gestionadas](#)

Prácticas recomendadas relativas a políticas

Las políticas basadas en la identidad determinan si alguien puede crear AWS CodeStar recursos de tu cuenta, acceder a ellos o eliminarlos. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comience con las políticas AWS administradas y avance hacia los permisos con privilegios mínimos: para empezar a conceder permisos a sus usuarios y cargas de trabajo, utilice las políticas AWS administradas que otorgan permisos para muchos casos de uso comunes. Están disponibles en su Cuenta de AWS. Le recomendamos que reduzca aún más los permisos definiendo políticas administradas por el AWS cliente que sean específicas para sus casos de uso. Para obtener más información, consulte [las políticas AWS gestionadas](#) o [las políticas AWS gestionadas para las funciones laborales](#) en la Guía del IAM usuario.
- Aplique permisos con privilegios mínimos: cuando establezca permisos con IAM políticas, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Para obtener más información sobre cómo IAM aplicar permisos, consulte [Políticas y permisos IAM en](#) la IAM Guía del usuario.

- Utilice las condiciones en IAM las políticas para restringir aún más el acceso: puede añadir una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puede escribir una condición de política para especificar que todas las solicitudes deben enviarse mediante SSL. También puedes usar condiciones para conceder el acceso a las acciones del servicio si se utilizan a través de una acción específica Servicio de AWS, por ejemplo AWS CloudFormation. Para obtener más información, consulte [los elementos IAM JSON de la política: Condición](#) en la Guía del IAM usuario.
- Utilice IAM Access Analyzer para validar sus IAM políticas y garantizar permisos seguros y funcionales: IAM Access Analyzer valida las políticas nuevas y existentes para que se ajusten al lenguaje de las políticas (JSON) y IAM a las IAM mejores prácticas. IAM Access Analyzer proporciona más de 100 comprobaciones de políticas y recomendaciones prácticas para ayudarle a crear políticas seguras y funcionales. Para obtener más información, consulte la [validación de políticas de IAM Access Analyzer](#) en la Guía del IAM usuario.
- Requerir autenticación multifactorial (MFA): si se encuentra en una situación en la que se requieren IAM usuarios o un usuario raíz Cuenta de AWS, actívela MFA para aumentar la seguridad. Para solicitarlo MFA cuando se convoque a API las operaciones, añada MFA condiciones a sus políticas. Para obtener más información, consulte [Configuración del API acceso MFA protegido](#) en la Guía del IAM usuario.

Para obtener más información sobre las prácticas recomendadas IAM, consulte las [prácticas recomendadas de seguridad IAM en](#) la Guía del IAM usuario.

AWS CodeStar Service Role Política

La `aws-codestar-service-role` política se adjunta a la función de servicio que AWS CodeStar permite realizar acciones con otros servicios. La primera vez que inicie sesión AWS CodeStar, creará el rol de servicio. Solo necesita crearlo una vez. La política se asocia automáticamente al rol de servicio después de crearlo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ProjectEventRules",
      "Effect": "Allow",
      "Action": [
        "events:PutTargets",
        "events:RemoveTargets",
```

```

        "events:PutRule",
        "events>DeleteRule",
        "events:DescribeRule"
    ],
    "Resource": [
        "arn:aws:events::*:rule/awscodestar-*"
    ]
},
{
    "Sid": "ProjectStack",
    "Effect": "Allow",
    "Action": [
        "cloudformation:*Stack*",
        "cloudformation:CreateChangeSet",
        "cloudformation:ExecuteChangeSet",
        "cloudformation>DeleteChangeSet",
        "cloudformation:GetTemplate"
    ],
    "Resource": [
        "arn:aws:cloudformation::*:stack/awscodestar-*",
        "arn:aws:cloudformation::*:stack/awseb-*",
        "arn:aws:cloudformation::*:stack/aws-cloud9-*",
        "arn:aws:cloudformation:*:aws:transform/CodeStar*"
    ]
},
{
    "Sid": "ProjectStackTemplate",
    "Effect": "Allow",
    "Action": [
        "cloudformation:GetTemplateSummary",
        "cloudformation:DescribeChangeSet"
    ],
    "Resource": "*"
},
{
    "Sid": "ProjectQuickstarts",
    "Effect": "Allow",
    "Action": [
        "s3:GetObject"
    ],
    "Resource": [
        "arn:aws:s3:::awscodestar-*/*"
    ]
},

```

```

{
  "Sid": "ProjectS3Buckets",
  "Effect": "Allow",
  "Action": [
    "s3:*"
  ],
  "Resource": [
    "arn:aws:s3:::aws-codestar-*",
    "arn:aws:s3:::elasticbeanstalk-*"
  ]
},
{
  "Sid": "ProjectServices",
  "Effect": "Allow",
  "Action": [
    "codestar:*",
    "codecommit:*",
    "codepipeline:*",
    "codedeploy:*",
    "codebuild:*",
    "autoscaling:*",
    "cloudwatch:Put*",
    "ec2:*",
    "elasticbeanstalk:*",
    "elasticloadbalancing:*",
    "iam:ListRoles",
    "logs:*",
    "sns:*",
    "cloud9:CreateEnvironmentEC2",
    "cloud9>DeleteEnvironment",
    "cloud9:DescribeEnvironment*",
    "cloud9:ListEnvironments"
  ],
  "Resource": "*"
},
{
  "Sid": "ProjectWorkerRoles",
  "Effect": "Allow",
  "Action": [
    "iam:AttachRolePolicy",
    "iam:CreateRole",
    "iam>DeleteRole",
    "iam>DeleteRolePolicy",
    "iam:DetachRolePolicy",

```



```

        "iam:GetRole",
        "iam:PassRole",
        "iam:GetRolePolicy",
        "iam:PutRolePolicy",
        "iam:SetDefaultPolicyVersion",
        "iam:CreatePolicy",
        "iam>DeletePolicy",
        "iam:AddRoleToInstanceProfile",
        "iam:CreateInstanceProfile",
        "iam>DeleteInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile"
    ],
    "Resource": [
        "arn:aws:iam::*:role/CodeStarWorker*",
        "arn:aws:iam::*:policy/CodeStarWorker*",
        "arn:aws:iam::*:instance-profile/awscodestar-*"
    ]
},
{
    "Sid": "ProjectTeamMembers",
    "Effect": "Allow",
    "Action": [
        "iam:AttachUserPolicy",
        "iam:DetachUserPolicy"
    ],
    "Resource": "*",
    "Condition": {
        "ArnEquals": {
            "iam:PolicyArn": [
                "arn:aws:iam::*:policy/CodeStar_*"
            ]
        }
    }
},
{
    "Sid": "ProjectRoles",
    "Effect": "Allow",
    "Action": [
        "iam:CreatePolicy",
        "iam>DeletePolicy",
        "iam:CreatePolicyVersion",
        "iam>DeletePolicyVersion",
        "iam:ListEntitiesForPolicy",
        "iam:ListPolicyVersions",

```

```

        "iam:GetPolicy",
        "iam:GetPolicyVersion"
    ],
    "Resource": [
        "arn:aws:iam::*:policy/CodeStar_*"
    ]
},
{
    "Sid": "InspectServiceRole",
    "Effect": "Allow",
    "Action": [
        "iam:ListAttachedRolePolicies"
    ],
    "Resource": [
        "arn:aws:iam::*:role/aws-codestar-service-role",
        "arn:aws:iam::*:role/service-role/aws-codestar-service-role"
    ]
},
{
    "Sid": "IAMLinkRole",
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "iam:AWSServiceName": "cloud9.amazonaws.com"
        }
    }
},
{
    "Sid": "DescribeConfigRuleForARN",
    "Effect": "Allow",
    "Action": [
        "config:DescribeConfigRules"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Sid": "ProjectCodeStarConnections",
    "Effect": "Allow",

```

```

        "Action": [
            "codestar-connections:UseConnection",
            "codestar-connections:GetConnection"
        ],
        "Resource": "*"
    },
    {
        "Sid": "ProjectCodeStarConnectionsPassConnections",
        "Effect": "Allow",
        "Action": "codestar-connections:PassConnection",
        "Resource": "*",
        "Condition": {
            "StringEqualsIfExists": {
                "codestar-connections:PassedToService":
"codepipeline.amazonaws.com"
            }
        }
    }
]
}

```

AWSCodeStarFullAccess Política

En las [Configuración AWS CodeStar](#) instrucciones, adjuntó una política con el nombre `AWSCodeStarFullAccess` de su IAM usuario. Esta declaración de política permite al usuario realizar todas las acciones disponibles AWS CodeStar con todos los AWS CodeStar recursos disponibles asociados a la AWS cuenta. Esto incluye la creación y eliminación de proyectos. El siguiente ejemplo es un fragmento de una política de `AWSCodeStarFullAccess` representativa. La política real varía en función de la plantilla que seleccione al iniciar un nuevo AWS CodeStar proyecto.

AWS CloudFormation requiere `cloudformation::ListStacks` permiso cuando se llama `cloudformation::DescribeStacks` sin una pila de destino.

Detalles de los permisos

Esta política incluye permisos para poder hacer lo siguiente:

- `ec2`—Recupera información sobre EC2 las instancias para crear un AWS CodeStar proyecto.
- `cloud9`—Recuperar información sobre AWS Command Line Interface los entornos.
- `cloudformation`—Recuperar información sobre las pilas de AWS CodeStar proyectos.

- `codestar`—Realizar acciones dentro de un proyecto. AWS CodeStar

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CodeStarEC2",
      "Effect": "Allow",
      "Action": [
        "codestar:*",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets",
        "cloud9:DescribeEnvironment*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CodeStarCF",
      "Effect": "Allow",
      "Action": [
        "cloudformation:DescribeStack*",
        "cloudformation:ListStacks*",
        "cloudformation:GetTemplateSummary"
      ],
      "Resource": [
        "arn:aws:cloudformation:*:*:stack/awscodestar-*"
      ]
    }
  ]
}
```

Se recomienda no otorgar tanto acceso a todos los usuarios. En su lugar, puede añadir permisos a nivel de proyecto mediante los roles de proyecto gestionados por AWS CodeStar. Los roles otorgan niveles específicos de acceso a los AWS CodeStar proyectos y se denominan de la siguiente manera:

- Propietario
- Colaborador
- Lector

AWS CodeStar Política de roles de propietario

La política de roles de AWS CodeStar propietario permite al usuario realizar todas las acciones de un AWS CodeStar proyecto sin restricciones. AWS CodeStar aplica la CodeStar_*project-id*_Owner política a los miembros del equipo del proyecto con el nivel de acceso de propietario.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Owner"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],
  "Resource": [
    "arn:aws:iam::account-id:user/user-name"
  ]
}
...

```

Política del rol de Colaborador de AWS CodeStar

La política de roles de AWS CodeStar colaborador permite al usuario contribuir al proyecto y cambiar el panel de control del proyecto. AWS CodeStar aplica la CodeStar_*project-id*_Contributor política a los miembros del equipo del proyecto con el nivel de acceso de colaborador. Los usuarios con acceso de colaborador pueden colaborar al proyecto y cambiar el panel del proyecto, pero no pueden añadir o quitar miembros.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    "codestar:PutExtendedAccess",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam:account-id:policy/CodeStar_project-id_Contributor"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:*UserProfile",
    ...
  ],

```

```

"Resource": [
  "arn:aws:iam::account-id:user/user-name"
]
}
...

```

AWS CodeStar Política de roles de espectadores

La política de roles de AWS CodeStar espectador permite al usuario ver un proyecto en AWS CodeStar. AWS CodeStar aplica la CodeStar_*project-id*_Viewer política a los miembros del equipo del proyecto con el nivel de acceso de espectadores. Los usuarios con acceso como espectadores pueden ver un proyecto en los recursos AWS CodeStar, pero no pueden cambiarlos, ni añadir o eliminar miembros del equipo.

```

...
{
  "Effect": "Allow",
  "Action": [
    ...
    "codestar:Describe*",
    "codestar:Get*",
    "codestar:List*",
    ...
  ],
  "Resource": [
    "arn:aws:codestar:us-east-2:111111111111:project/project-id",
    "arn:aws:iam::account-id:policy/CodeStar_project-id_Viewer"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "codestar:DescribeUserProfile",
    "codestar:ListProjects",
    "codestar:ListUserProfiles",
    "codestar:VerifyServiceRole",
    ...
  ],
  "Resource": [
    "*"
  ]
},
{

```

```

"Effect": "Allow",
"Action": [
  "codestar:*UserProfile",
  ...
],
"Resource": [
  "arn:aws:iam::account-id:user/user-name"
]
}
...

```

AWS CodeStar Política sobre las funciones de los trabajadores de Toolchain (después del 6 de diciembre de 2018PDT)

Para AWS CodeStar los proyectos creados después del 6 de diciembre de 2018PDT, AWS CodeStar crea una política interna para un puesto de trabajador que crea recursos para su proyecto en otros AWS servicios. El contenido de la política depende del tipo de proyecto que está creando. La siguiente política es un ejemplo. Para obtener más información, consulte [IAMPolíticas para las funciones de los trabajadores](#).

```

{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject*",
        "codecommit:CancelUploadArchive",
        "codecommit:GetBranch",
        "codecommit:GetCommit",
        "codecommit:GetUploadArchiveStatus",
        "codecommit:GitPull",
        "codecommit:UploadArchive",
        "codebuild:StartBuild",
        "codebuild:BatchGetBuilds",
        "codebuild:StopBuild",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeChangeSet",

```



```

    "cloudformation:CreateChangeSet",
    "cloudformation>DeleteChangeSet",
    "cloudformation:ExecuteChangeSet",
    "codepipeline:StartPipelineExecution",
    "lambda:ListFunctions",
    "lambda:InvokeFunction",
    "sns:Publish"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
},
{
  "Action": [
    "kms:GenerateDataKey*",
    "kms:Encrypt",
    "kms:Decrypt"
  ],
  "Resource": [
    "*"
  ],
  "Effect": "Allow"
}
]
}

```

AWS CloudFormation Política de roles de los trabajadores

Para AWS CodeStar los proyectos creados después del 6 de diciembre de 2018PDT, AWS CodeStar crea una política interna para un rol de trabajador que crea AWS CloudFormation recursos para el AWS CodeStar proyecto. El contenido de la política depende del tipo de recursos necesarios para el

proyecto. La siguiente política es un ejemplo. Para obtener más información, consulte [IAMPolíticas para las funciones de los trabajadores](#).

```
{
  {
    "Statement": [
      {
        "Action": [
          "s3:PutObject",
          "s3:GetObject",
          "s3:GetObjectVersion"
        ],
        "Resource": [
          "arn:aws:s3:::aws-codestar-region-id-account-id-project-id",
          "arn:aws:s3:::aws-codestar-region-id-account-id-project-id/*"
        ],
        "Effect": "Allow"
      },
      {
        "Action": [
          "apigateway:DELETE",
          "apigateway:GET",
          "apigateway:PATCH",
          "apigateway:POST",
          "apigateway:PUT",
          "codedeploy:CreateApplication",
          "codedeploy:CreateDeployment",
          "codedeploy:CreateDeploymentConfig",
          "codedeploy:CreateDeploymentGroup",
          "codedeploy>DeleteApplication",
          "codedeploy>DeleteDeployment",
          "codedeploy>DeleteDeploymentConfig",
          "codedeploy>DeleteDeploymentGroup",
          "codedeploy:GetDeployment",
          "codedeploy:GetDeploymentConfig",
          "codedeploy:GetDeploymentGroup",
          "codedeploy:RegisterApplicationRevision",
          "codestar:SyncResources",
          "config>DeleteConfigRule",
          "config:DescribeConfigRules",
          "config>ListTagsForResource",
          "config:PutConfigRule",
          "config:TagResource",
```

```
"config:UntagResource",
"dynamodb:CreateTable",
"dynamodb>DeleteTable",
"dynamodb:DescribeContinuousBackups",
"dynamodb:DescribeTable",
"dynamodb:DescribeTimeToLive",
"dynamodb:ListTagsOfResource",
"dynamodb:TagResource",
"dynamodb:UntagResource",
"dynamodb:UpdateContinuousBackups",
"dynamodb:UpdateTable",
"dynamodb:UpdateTimeToLive",
"ec2:AssociateIamInstanceProfile",
"ec2:AttachVolume",
"ec2:CreateSecurityGroup",
"ec2:createTags",
"ec2:DescribeIamInstanceProfileAssociations",
"ec2:DescribeInstances",
"ec2:DescribeSecurityGroups",
"ec2:DescribeSubnets",
"ec2:DetachVolume",
"ec2:DisassociateIamInstanceProfile",
"ec2:ModifyInstanceAttribute",
"ec2:ModifyInstanceCreditSpecification",
"ec2:ModifyInstancePlacement",
"ec2:MonitorInstances",
"ec2:ReplaceIamInstanceProfileAssociation",
"ec2:RunInstances",
"ec2:StartInstances",
"ec2:StopInstances",
"ec2:TerminateInstances",
"events>DeleteRule",
"events:DescribeRule",
"events:ListTagsForResource",
"events:PutRule",
"events:PutTargets",
"events:RemoveTargets",
"events:TagResource",
"events:UntagResource",
"kinesis:AddTagsToStream",
"kinesis:CreateStream",
"kinesis:DecreaseStreamRetentionPeriod",
"kinesis>DeleteStream",
"kinesis:DescribeStream",
```

```
"kinesis:IncreaseStreamRetentionPeriod",
"kinesis:RemoveTagsFromStream",
"kinesis:StartStreamEncryption",
"kinesis:StopStreamEncryption",
"kinesis:UpdateShardCount",
"lambda:CreateAlias",
"lambda:CreateFunction",
"lambda>DeleteAlias",
"lambda>DeleteFunction",
"lambda>DeleteFunctionConcurrency",
"lambda:GetFunction",
"lambda:GetFunctionConfiguration",
"lambda:ListTags",
"lambda:ListVersionsByFunction",
"lambda:PublishVersion",
"lambda:PutFunctionConcurrency",
"lambda:TagResource",
"lambda:UntagResource",
"lambda:UpdateAlias",
"lambda:UpdateFunctionCode",
"lambda:UpdateFunctionConfiguration",
"s3:CreateBucket",
"s3>DeleteBucket",
"s3>DeleteBucketWebsite",
"s3:PutAccelerateConfiguration",
"s3:PutAnalyticsConfiguration",
"s3:PutBucketAcl",
"s3:PutBucketCORS",
"s3:PutBucketLogging",
"s3:PutBucketNotification",
"s3:PutBucketPublicAccessBlock",
"s3:PutBucketVersioning",
"s3:PutBucketWebsite",
"s3:PutEncryptionConfiguration",
"s3:PutInventoryConfiguration",
"s3:PutLifecycleConfiguration",
"s3:PutMetricsConfiguration",
"s3:PutReplicationConfiguration",
"sns:CreateTopic",
"sns>DeleteTopic",
"sns:GetTopicAttributes",
"sns:ListSubscriptionsByTopic",
"sns:ListTopics",
"sns:SetSubscriptionAttributes",
```

```

        "sns:Subscribe",
        "sns:Unsubscribe",
        "sqs:CreateQueue",
        "sqs>DeleteQueue",
        "sqs:GetQueueAttributes",
        "sqs:GetQueueUrl",
        "sqs:ListQueueTags",
        "sqs:TagQueue",
        "sqs:UntagQueue"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "lambda:AddPermission",
        "lambda:RemovePermission"
    ],
    "Resource": [
        "arn:aws:lambda:region-id:account-id:function:awscodestar-*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStar-project-id*"
    ],
    "Effect": "Allow"
},
{
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "codedeploy.amazonaws.com"
        }
    },
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CodeDeploy"
    ],
    ],

```

```

    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:CreateChangeSet"
    ],
    "Resource": [
      "arn:aws:cloudformation:region-id:aws:transform/Serverless-2016-10-31",
      "arn:aws:cloudformation:region-id:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "iam:CreateServiceLinkedRole",
      "iam:GetRole",
      "iam>DeleteRole",
      "iam>DeleteUser"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Condition": {
      "StringEquals": {
        "iam:PermissionsBoundary": "arn:aws:iam::account-id:policy/CodeStar_project-id_PermissionsBoundary"
      }
    },
    "Action": [
      "iam:AttachRolePolicy",
      "iam:AttachUserPolicy",
      "iam:CreateRole",
      "iam:CreateUser",
      "iam>DeleteRolePolicy",
      "iam>DeleteUserPolicy",
      "iam:DetachUserPolicy",
      "iam:DetachRolePolicy",
      "iam:PutUserPermissionsBoundary",
      "iam:PutRolePermissionsBoundary"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
},

```

```

    {
      "Action": [
        "kms:CreateKey",
        "kms:CreateAlias",
        "kms>DeleteAlias",
        "kms:DisableKey",
        "kms:EnableKey",
        "kms:UpdateAlias",
        "kms:TagResource",
        "kms:UntagResource"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Condition": {
        "StringEquals": {
          "ssm:ResourceTag/awscodestar:projectArn":
"arn:aws:codestar:project-id:account-id:project/project-id"
        }
      },
      "Action": [
        "ssm:GetParameter*"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}

```

AWS CloudFormation Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018PDT)

Si su AWS CodeStar proyecto se creó antes del 6 de diciembre de 2018PDT, AWS CodeStar cree una política en línea para un rol de AWS CloudFormation trabajador. A continuación se muestra un ejemplo de una instrucción de política.

```

{
  "Statement": [
    {
      "Action": [
        "s3:PutObject",

```

```

        "s3:GetObject",
        "s3:GetObjectVersion"
    ],
    "Resource": [
        "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe/*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "codestar:SyncResources",
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:AddPermission",
        "lambda:UpdateFunction",
        "lambda:UpdateFunctionCode",
        "lambda:GetFunction",
        "lambda:GetFunctionConfiguration",
        "lambda:UpdateFunctionConfiguration",
        "lambda:RemovePermission",
        "lambda:listTags",
        "lambda:TagResource",
        "lambda:UntagResource",
        "apigateway:*",
        "dynamodb:CreateTable",
        "dynamodb>DeleteTable",
        "dynamodb:DescribeTable",
        "kinesis:CreateStream",
        "kinesis>DeleteStream",
        "kinesis:DescribeStream",
        "sns:CreateTopic",
        "sns>DeleteTopic",
        "sns:ListTopics",
        "sns:GetTopicAttributes",
        "sns:SetTopicAttributes",
        "s3:CreateBucket",
        "s3>DeleteBucket",
        "config:DescribeConfigRules",
        "config:PutConfigRule",
        "config>DeleteConfigRule",
        "ec2:*",
        "autoscaling:*",
        "elasticloadbalancing:*"
    ]
}

```



```

        "elasticbeanstalk:*"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:CreateChangeSet"
    ],
    "Resource": [
      "arn:aws:cloudformation:us-east-1:aws:transform/Serverless-2016-10-31",
      "arn:aws:cloudformation:us-east-1:aws:transform/CodeStar"
    ],
    "Effect": "Allow"
  }
]
}

```

AWS CodePipeline Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018PDT)

Si su AWS CodeStar proyecto se creó antes del 6 de diciembre de 2018PDT, AWS CodeStar cree una política en línea para un rol de CodePipeline trabajador. A continuación se muestra un ejemplo de una instrucción de política.

```

{
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetBucketVersioning",
        "s3:PutObject"
      ],
    },
  ],
}

```

```

    "Resource": [
      "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe",
      "arn:aws:s3:::aws-codestar-us-east-1-account-id-project-id-pipe/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codecommit:CancelUploadArchive",
      "codecommit:GetBranch",
      "codecommit:GetCommit",
      "codecommit:GetUploadArchiveStatus",
      "codecommit:UploadArchive"
    ],
    "Resource": [
      "arn:aws:codecommit:us-east-1:account-id:project-id"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codebuild:StartBuild",
      "codebuild:BatchGetBuilds",
      "codebuild:StopBuild"
    ],
    "Resource": [
      "arn:aws:codebuild:us-east-1:account-id:project/project-id"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:DescribeStacks",
      "cloudformation:DescribeChangeSet",
      "cloudformation:CreateChangeSet",
      "cloudformation>DeleteChangeSet",
      "cloudformation:ExecuteChangeSet"
    ],
    "Resource": [
      "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/*"
    ],
    "Effect": "Allow"
  },

```

```

    {
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::account-id:role/CodeStarWorker-project-id-CloudFormation"
      ],
      "Effect": "Allow"
    }
  ]
}

```

AWS CodeBuild Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018PDT)

Si su AWS CodeStar proyecto se creó antes del 6 de diciembre de 2018PDT, AWS CodeStar cree una política en línea para un rol de CodeBuild trabajador. A continuación se muestra un ejemplo de una instrucción de política.

```

{
  "Statement": [
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe/*",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
        "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app/*"
      ],
      "Effect": "Allow"
    }
  ]
}

```

```

    },
    {
      "Action": [
        "codecommit:GitPull"
      ],
      "Resource": [
        "arn:aws:codecommit:us-east-1:account-id:project-id"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "kms:GenerateDataKey*",
        "kms:Encrypt",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:account-id:alias/aws/s3"
      ],
      "Effect": "Allow"
    }
  ]
}

```

Política de funciones de CloudWatch los trabajadores de Amazon Events (antes del 6 de diciembre de 2018PDT)

Si su AWS CodeStar proyecto se creó antes del 6 de diciembre de 2018PDT, AWS CodeStar cree una política en línea para un puesto de trabajador de CloudWatch eventos. A continuación se muestra un ejemplo de una instrucción de política.

```

{
  "Statement": [
    {
      "Action": [
        "codepipeline:StartPipelineExecution"
      ],
      "Resource": [
        "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline"
      ],
      "Effect": "Allow"
    }
  ]
}

```

}

AWS CodeStar Política de límites de permisos

Si crea un AWS CodeStar proyecto después del 6 de diciembre de 2018PDT, AWS CodeStar crea una política de límites de permisos para su proyecto. Esta política impide el escalado de privilegios a recursos fuera del proyecto. Se trata de una política dinámica que se actualiza a medida que el proyecto evoluciona. El contenido de la política depende del tipo de proyecto que está creando. La siguiente política es un ejemplo. Para obtener más información, consulte [Límite de permisos de IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3::*/AWSLogs/*/Config/*"
      ]
    },
    {
      "Sid": "2",
      "Effect": "Allow",
      "Action": [
        "*"
      ],
      "Resource": [
        "arn:aws:codestar:us-east-1:account-id:project/project-id",
        "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id-lambda/eefbbf20-c1d9-11e8-8a3a-500c28b4e461",
        "arn:aws:cloudformation:us-east-1:account-id:stack/awscodestar-project-id/4b80b3f0-c1d9-11e8-8517-500c28b236fd",
        "arn:aws:codebuild:us-east-1:account-id:project/project-id",
        "arn:aws:codecommit:us-east-1:account-id:project-id",
        "arn:aws:codepipeline:us-east-1:account-id:project-id-Pipeline",
        "arn:aws:execute-api:us-east-1:account-id:7rlst5mrgi",
        "arn:aws:iam:account-id:role/CodeStarWorker-project-id-CloudFormation",
        "arn:aws:iam:account-id:role/CodeStarWorker-project-id-CloudWatchEventRule",
        "arn:aws:iam:account-id:role/CodeStarWorker-project-id-CodeBuild",
        "arn:aws:iam:account-id:role/CodeStarWorker-project-id-CodePipeline",

```

```

    "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",
    "arn:aws:lambda:us-east-1:account-id:function:awscodestar-project-id-lambda-
GetHelloWorld-KFKTXYNH9573",
    "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-app",
    "arn:aws:s3::aws-codestar-us-east-1-account-id-project-id-pipe"
  ]
},
{
  "Sid": "3",
  "Effect": "Allow",
  "Action": [
    "apigateway:GET",
    "config:Describe*",
    "config:Get*",
    "config:List*",
    "config:Put*",
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:DescribeLogGroups",
    "logs:PutLogEvents"
  ],
  "Resource": [
    "*"
  ]
}
]
}

```

Listado de recursos para un proyecto

En este ejemplo, quieres conceder a un IAM usuario específico de tu AWS cuenta acceso para que muestre los recursos de un AWS CodeStar proyecto.

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListResources",
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}

```

```
]
}
```

Uso de la AWS CodeStar consola

No se requieren permisos específicos para acceder a la AWS CodeStar consola, pero no puedes hacer nada útil a menos que tengas la `AWSCodeStarFullAccess` política o una función a AWS CodeStar nivel de proyecto: propietario, colaborador o espectador. Para obtener más información sobre `AWSCodeStarFullAccess`, consulte [AWSCodeStarFullAccess Política](#). Para obtener más información sobre las políticas de nivel de proyecto, consulte [IAMPolíticas para los miembros del equipo](#).

No es necesario que concedas permisos mínimos de consola a los usuarios que solo realicen llamadas a la AWS CLI o a la AWS API. En su lugar, permita el acceso únicamente a las acciones que coincidan con la API operación que está intentando realizar.

Permitir a los usuarios ver sus propios permisos

En este ejemplo, se muestra cómo se puede crear una política que permita a IAM los usuarios ver las políticas integradas y administradas asociadas a su identidad de usuario. Esta política incluye permisos para completar esta acción en la consola o mediante programación mediante la tecla o. AWS CLI AWS API

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
```

```

    "Action": [
      "iam:GetGroupPolicy",
      "iam:GetPolicyVersion",
      "iam:GetPolicy",
      "iam:ListAttachedGroupPolicies",
      "iam:ListGroupPolicies",
      "iam:ListPolicyVersions",
      "iam:ListPolicies",
      "iam:ListUsers"
    ],
    "Resource": "*"
  }
]
}

```

Actualización de un proyecto de AWS CodeStar

En este ejemplo, quieres conceder a un IAM usuario específico de tu AWS cuenta acceso para editar los atributos de un AWS CodeStar proyecto, como su descripción.

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:UpdateProject"
      ],
      "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
    }
  ]
}

```

Añadir un miembro de equipo a un proyecto

En este ejemplo, quieres conceder a un IAM usuario específico la posibilidad de añadir miembros del equipo a un AWS CodeStar proyecto con el ID del proyecto *my-first-projec*, pero para negarle explícitamente a ese usuario la posibilidad de eliminar miembros del equipo:

```

{
  "Version": "2012-10-17",
  "Statement" : [

```



```

{
  "Effect" : "Allow",
  "Action" : [
    "codestar:AssociateTeamMember",
  ],
  "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
},
{
  "Effect" : "Deny",
  "Action" : [
    "codestar:DisassociateTeamMember",
  ],
  "Resource" : "arn:aws:codestar:us-east-2:project/my-first-projec"
}
]
}

```

Listar los perfiles de usuario asociados a una AWS cuenta

En este ejemplo, permite que un IAM usuario que tenga esta política adjunta muestre todos los perfiles de AWS CodeStar usuario asociados a una AWS cuenta:

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codestar:ListUserProfiles",
      ],
      "Resource" : "*"
    }
  ]
}

```

Visualización de proyectos de AWS CodeStar basados en etiquetas

Puedes usar las condiciones de tu política basada en la identidad para controlar el acceso a los AWS CodeStar proyectos en función de las etiquetas. Este ejemplo muestra cómo se puede crear una política que permita ver un proyecto. Sin embargo, los permisos solo se conceden si la etiqueta

de proyecto `Owner` tiene el valor del nombre de usuario de dicho usuario. Esta política también proporciona los permisos necesarios para llevar a cabo esta acción en la consola.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListProjectsInConsole",
      "Effect": "Allow",
      "Action": "codestar:ListProjects",
      "Resource": "*"
    },
    {
      "Sid": "ViewProjectIfOwner",
      "Effect": "Allow",
      "Action": "codestar:GetProject",
      "Resource": "arn:aws:codestar:*:*:project/*",
      "Condition": {
        "StringEquals": {"codestar:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```

Puedes adjuntar esta política a los IAM usuarios de tu cuenta. Si un usuario llamado `richard-roe` intenta ver un AWS CodeStar proyecto, el proyecto debe estar etiquetado `Owner=richard-roe` o `owner=richard-roe`. De lo contrario, se le deniega el acceso. La clave de la etiqueta de condición `Owner` coincide con los nombres de las claves de condición `Owner` y `owner` porque no distinguen entre mayúsculas y minúsculas. Para obtener más información, consulte [Elementos IAM JSON de política: condición](#) en la Guía del IAM usuario.

AWS CodeStar actualizaciones de las políticas AWS gestionadas

Consulte los detalles sobre las actualizaciones de las políticas AWS administradas AWS CodeStar desde que este servicio comenzó a rastrear estos cambios. Para recibir alertas automáticas sobre los cambios en esta página, suscríbase al RSS feed de la página del [historial del AWS CodeStar documento](#).

Cambio	Descripción	Fecha
AWSCodeStarFullAccessPolítica : actualice la AWSCodeStarFullAccess política	Se ha actualizado la política de roles de AWS CodeStar acceso. El resultado de la política es el mismo, pero la formación de nubes requiere ListStacks algo adicional DescribeStacks, algo que ya es obligatorio.	24 de marzo de 2023
AWSCodeStarServiceRolePolítica : actualice la AWSCodeStarServiceRole política	La política del rol de AWS CodeStar servicio se ha actualizado para corregir las acciones redundantes de la declaración de política. La política de roles de servicio permite al AWS CodeStar servicio realizar acciones en su nombre.	23 de septiembre de 2021
AWS CodeStar comenzó a rastrear los cambios	AWS CodeStar comenzó a realizar un seguimiento de los cambios de sus políticas AWS gestionadas.	23 de septiembre de 2021

Solución de problemas AWS CodeStar de identidad y acceso

Utilice la siguiente información como ayuda para diagnosticar y solucionar los problemas más comunes que pueden surgir al trabajar con AWS CodeStar y IAM.

Temas

- [No estoy autorizado a realizar ninguna acción en AWS CodeStar](#)
- [No estoy autorizado a realizar iam: PassRole](#)
- [Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis AWS CodeStar recursos](#)

No estoy autorizado a realizar ninguna acción en AWS CodeStar

Si AWS Management Console le indica que no está autorizado a realizar una acción, póngase en contacto con su administrador para obtener ayuda. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

El siguiente ejemplo de error se produce cuando el usuario IAM mateojackson intenta utilizar la consola para ver los detalles de un *widget* pero no tiene `codestar:GetWidget` permisos.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
codestar:GetWidget on resource: my-example-widget
```

En este caso, Mateo pide a su administrador que actualice sus políticas de forma que pueda obtener acceso al recurso *my-example-widget* mediante la acción `codestar:GetWidget`.

No estoy autorizado a realizar iam: PassRole

Si recibes un mensaje de error que indica que no estás autorizado a realizar la `iam:PassRole` acción, debes actualizar tus políticas para que puedas transferirle AWS CodeStar una función.

Algunos Servicios de AWS permiten transferir una función existente a ese servicio en lugar de crear una nueva función de servicio o una función vinculada al servicio. Para ello, debe tener permisos para transferir el rol al servicio.

El siguiente ejemplo de error se produce cuando un IAM usuario denominado marymajor intenta utilizar la consola para realizar una acción en ella. Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

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

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con AWS el administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis AWS CodeStar recursos

Puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admiten políticas basadas en recursos o listas de control de acceso (ACLs), puedes usar esas políticas para permitir que las personas accedan a tus recursos.

Para más información, consulte lo siguiente:

- Para saber si AWS CodeStar es compatible con estas funciones, consulte. [¿Cómo AWS CodeStar funciona con IAM](#)
- Para obtener información sobre cómo proporcionar acceso a los recursos de su propiedad, consulte [Proporcionar acceso a un IAM usuario en otro Cuenta de AWS de su propiedad](#) en la Guía del IAM usuario. Cuentas de AWS
- Para obtener información sobre cómo proporcionar acceso a tus recursos a terceros Cuentas de AWS, consulta Cómo permitir el [acceso a recursos que Cuentas de AWS son propiedad de terceros](#) en la Guía del IAM usuario.
- Para obtener información sobre cómo proporcionar acceso mediante la federación de identidades, consulte [Proporcionar acceso a usuarios autenticados externamente \(federación de identidades\)](#) en la Guía del IAM usuario.
- Para saber la diferencia entre el uso de roles y políticas basadas en recursos para el acceso entre cuentas, consulte el acceso a [recursos entre cuentas IAM en la Guía](#) del usuario. IAM

Registro de llamadas a la API de AWS CodeStar con AWS CloudTrail

AWS CodeStar se integra con AWS CloudTrail, un servicio que proporciona un registro de las acciones hechas por un usuario, un rol o un servicio de AWS en AWS CodeStar. CloudTrail captura las llamadas a la API de AWS CodeStar como eventos. Las llamadas capturadas incluyen las llamadas realizadas desde la consola de AWS CodeStar y las llamadas de código a las operaciones de la API de AWS CodeStar. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de S3, incluidos los eventos para AWS CodeStar. Si no configura un registro de seguimiento, puede ver los eventos más recientes de la consola de CloudTrail en el Historial de eventos. Mediante la información que CloudTrail recopila, se puede

determinar la solicitud que se envió a AWS CodeStar, la dirección IP desde la que se realizó la solicitud, quién la realizó, cuándo se realizó y otros detalles adicionales.

Para obtener más información acerca de CloudTrail, consulte la [Guía del usuario de AWS CloudTrail](#).

Información de AWS CodeStar en CloudTrail

CloudTrail se habilita en su cuenta de AWS cuando la crea. Cuando se produce una actividad en AWS CodeStar, esa actividad se registra en un evento de CloudTrail junto con otros eventos de servicio de AWS en Event history (Historial de eventos). Puede ver, buscar y descargar los últimos eventos de la cuenta de AWS. Para obtener más información, consulte [Ver eventos con el historial de eventos de CloudTrail](#).

Para mantener un registro continuo de eventos en la cuenta de AWS, incluidos los eventos de AWS CodeStar, cree un registro de seguimiento. De manera predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones de AWS. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de S3 especificado. También puede configurar otros servicios de AWS para analizar y actuar según los datos de eventos recopilados en los registros de CloudTrail. Para obtener más información, consulte lo siguiente:

- [Introducción a la creación de registros de seguimiento](#)
- [Servicios e integraciones compatibles con CloudTrail](#)
- [Configuración de notificaciones de Amazon SNS para CloudTrail](#)
- [Recibir archivos de registro de CloudTrail de varias regiones](#) y [Recibir archivos de registro de CloudTrail de varias cuentas](#)

Todas las acciones de AWS CodeStar las registra CloudTrail y se documentan en la [Referencia de la API de AWS CodeStar](#). Por ejemplo, las llamadas a las acciones `DescribeProject`, `UpdateProject` y `AssociateTeamMember` generan entradas en los archivos de log de CloudTrail.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales raíz o del usuario de IAM de .
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte el [Elemento `userIdentity` de CloudTrail](#).

Descripción de las entradas de archivos de registro de AWS CodeStar

Los archivos log de CloudTrail pueden contener una o varias entradas de log. Un evento representa una solicitud específica realizada desde un origen y contiene información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas públicas a la API, por lo que estas no aparecen en ningún orden específico.

En el siguiente ejemplo se muestra una entrada de registro de CloudTrail que ilustra una operación de `CreateProject` a la que se llama desde AWS CodeStar:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAJLIN20F3UBEXAMPLE:role-name",
    "arn": "arn:aws:sts::account-ID:assumed-role/role-name/role-session-name",
    "accountId": "account-ID",
    "accessKeyId": "ASIAJ44LFQS5XEXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2017-06-04T23:56:57Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAJLIN20F3UBEXAMPLE",
        "arn": "arn:aws:iam::account-ID:role/service-role/role-name",
        "accountId": "account-ID",
        "userName": "role-name"
      }
    },
    "invokedBy": "codestar.amazonaws.com"
  },
  "eventTime": "2017-06-04T23:56:57Z",
  "eventSource": "codestar.amazonaws.com",
  "eventName": "CreateProject",
  "awsRegion": "region-ID",
  "sourceIPAddress": "codestar.amazonaws.com",
  "userAgent": "codestar.amazonaws.com",
  "requestParameters": {
```

```

    "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "id": "project-ID",
    "stackId": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "description": "AWS CodeStar created project",
    "name": "project-name",
    "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name"
  },
  "responseElements": {
    "projectTemplateId": "arn:aws:codestar:region-ID::project-template/project-template-name",
    "arn": "arn:aws:codestar:us-east-1:account-ID:project/project-ID",
    "clientRequestToken": "arn:aws:cloudformation:region-ID:account-ID:stack/stack-name/additional-ID",
    "id": "project-ID"
  },
  "requestID": "7d7556d0-4981-11e7-a3bc-dd5daEXAMPLE",
  "eventID": "6b0d6e28-7a1e-4a73-981b-c8fdbEXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "account-ID"
}

```

Validación de la conformidad para AWS CodeStar

AWS CodeStar no está en el ámbito de los programas de conformidad de AWS.

Para obtener una lista de los servicios que AWS incluyen los programas de conformidad específicos, consulte los [servicios AWS incluidos en cada programa de conformidad](#). Para obtener información general, consulte [Programas de conformidad de AWS](#).

Puede descargar los informes de auditoría de terceros con AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#).

Resiliencia en AWS CodeStar

La infraestructura global de AWS se compone de regiones y zonas de disponibilidad de AWS. AWS Las regiones proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y

bases de datos que realizan una conmutación por error automática entre zonas de disponibilidad sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre las regiones y zonas de disponibilidad de AWS, consulte [Infraestructura global de AWS](#).

Seguridad de infraestructura en AWS CodeStar

Como servicio gestionado, AWS CodeStar está protegido por la seguridad de la red AWS global. Para obtener información sobre los servicios AWS de seguridad y cómo se protege la infraestructura, consulte [Seguridad AWS en la nube](#). Para diseñar su AWS entorno utilizando las mejores prácticas de seguridad de la infraestructura, consulte [Protección de infraestructuras en un marco](#) de buena AWS arquitectura basado en el pilar de la seguridad.

Utiliza las API llamadas AWS publicadas para acceder a CodeStar través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Cifre suites con perfecto secreto (PFS), como (Ephemeral Diffie-Hellman) o DHE ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben firmarse con un identificador de clave de acceso y una clave de acceso secreta que esté asociada a un director. IAM También puede utilizar [AWS Security Token Service](#) (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

De forma predeterminada, AWS CodeStar no aísla el tráfico del servicio. Los proyectos creados con Internet AWS CodeStar están abiertos a la Internet pública, a menos que modifique manualmente la configuración de acceso a través de AmazonEC2, API Gateway o Elastic Beanstalk. Esto se hace de forma intencionada. Puede modificar la configuración de acceso en AmazonEC2, API Gateway o Elastic Beanstalk en la medida que desee, lo que incluye impedir todo acceso a Internet.

AWS CodeStar no proporciona soporte para los VPC puntos finales (AWS PrivateLink) de forma predeterminada, pero puede configurar ese soporte directamente en los recursos del proyecto.

Límites en AWS CodeStar

En la siguiente tabla se describen los límites en AWS CodeStar. AWS CodeStar depende de otros servicios de AWS para los recursos del proyecto. Algunos de los límites del servicio se pueden cambiar. Para obtener más información sobre los límites que pueden cambiarse, consulte [Límites de servicio de AWS](#).

Número de proyectos	Máximo de 333 proyectos en una cuenta de AWS. Los límites reales varían según el nivel de las otras dependencias del servicio (por ejemplo, el número máximo de canalizaciones permitidas en CodePipeline para su cuenta de AWS).
Número de proyectos de AWS CodeStar al que puede pertenecer un usuario de IAM	Máximo de 10 por cada usuario de IAM individual.
ID de proyectos	<p>Los ID de proyectos deben ser únicos dentro de una cuenta de AWS. Los ID de proyectos deben tener al menos 2 caracteres y no pueden superar los 15 caracteres. Los caracteres permitidos son:</p> <ul style="list-style-type: none"> Letras de la a a la z inclusive. Número del 0 al 9 inclusive. El carácter especial - (signo menos). <p>Cualquier otro carácter como, por ejemplo, mayúsculas, espacios, . (punto), @ (signo de la arroba) o _ (guion bajo) no están permitidos.</p>
Nombres de proyectos	Los nombres de proyectos no puede superar 100 caracteres de longitud y no pueden empezar ni terminar con un espacio vacío.

Descripciones de proyectos	Cualquier combinación de caracteres con una longitud de entre 0 y 1024 caracteres. Las descripciones de proyectos son opcionales.
Miembros del equipo en un proyecto de AWS CodeStar	100
Nombre de visualización de un perfil de usuario	Cualquier combinación de caracteres con una longitud entre 0 y 100 caracteres. Los nombres de visualización deben incluir al menos un carácter. Ese carácter no puede ser un espacio. Los nombres de visualización no pueden empezar ni terminar con un espacio.
Dirección de correo electrónico de un perfil de usuario	La dirección de correo electrónico debe incluir una @ y terminar en una extensión de dominio válida.
Acceso federado, acceso de cuenta raíz o acceso temporal a AWS CodeStar	AWS CodeStar admite los usuarios federados y el uso de credenciales de acceso temporal. No se recomienda usar AWS CodeStar con una cuenta raíz.
Roles de IAM	Un máximo de 5120 caracteres en cualquier política administrada que se asocie a un rol de IAM.

Solución de problemas AWS CodeStar

La siguiente información puede ayudarle a solucionar problemas comunes en AWS CodeStar.

Temas

- [Error al crear el proyecto: el proyecto no se ha creado](#)
- [Creación de proyectos: aparece un error cuando intento editar la configuración de Amazon EC2 al crear un proyecto](#)
- [Eliminación de un AWS CodeStar proyecto: se ha eliminado un proyecto, pero aún existen recursos](#)
- [Fallo en la gestión del equipo: no se ha podido añadir un usuario de IAM a un equipo de un proyecto AWS CodeStar](#)
- [Error de acceso: un usuario federado no puede acceder a un proyecto AWS CodeStar](#)
- [Error de acceso: un usuario federado no puede acceder ni crear un entorno AWS Cloud9](#)
- [Error de acceso: un usuario federado puede crear un AWS CodeStar proyecto, pero no puede ver los recursos del proyecto](#)
- [Error del rol de servicio: el rol de servicio no se ha podido crear](#)
- [Error del rol de servicio: el rol de servicio no es válido o falta](#)
- [Problema con el rol del proyecto: las comprobaciones del estado de AWS Elastic Beanstalk salud fallan en las instancias de un AWS CodeStar proyecto](#)
- [Error del rol de proyecto: un rol de proyecto no es válido o falta](#)
- [Extensiones del proyecto: no se puede conectar a JIRA](#)
- [GitHub: No se puede acceder al historial de confirmaciones, los problemas o el código de un repositorio](#)
- [AWS CloudFormation: Restauración de creación de pila para permisos ausentes](#)
- [AWS CloudFormation no está autorizado a realizar la función de ejecución iam: PassRole on Lambda](#)
- [No se ha podido crear la conexión para un repositorio GitHub](#)

Error al crear el proyecto: el proyecto no se ha creado

Problema: cuando trata de crear un proyecto, ve un mensaje que indica que se ha producido un error al crearlo.

Soluciones posibles: las razones más comunes del error son:

- Ya existe un proyecto con ese ID en tu AWS cuenta, posiblemente en otra AWS región.
- El usuario de IAM con el que iniciaste sesión AWS Management Console no tiene los permisos necesarios para crear un proyecto.
- Al rol AWS CodeStar de servicio le faltan uno o varios permisos necesarios.
- Has alcanzado el límite máximo de uno o más recursos para un proyecto (por ejemplo, el límite de políticas gestionadas por el cliente en IAM, en los buckets de Amazon S3 o en los pipelines).
CodePipeline

Antes de crear un proyecto, verifique que haya aplicado la política `AWSCodeStarFullAccess` a su usuario de IAM. Para obtener más información, consulte [AWSCodeStarFullAccess Política](#).

Al crear un proyecto, asegúrese de que el ID es único y cumple los requisitos de AWS CodeStar . Asegúrese de seleccionar la casilla de verificación AWS CodeStar Desea obtener permiso para administrar AWS los recursos en su nombre.

Para solucionar otros problemas, abre la AWS CloudFormation consola, elige la pila del proyecto que has intentado crear y selecciona la pestaña Eventos. Puede que haya más de una pila para un proyecto. Los nombres de las pilas empezarán con `awscodestar-`, seguido del ID del proyecto. Las pilas pueden estar en la vista de filtro Deleted (Eliminado). Revise los mensajes de error en los eventos de pila y corrija el problema que se indica como causa de esos errores.

Creación de proyectos: aparece un error cuando intento editar la configuración de Amazon EC2 al crear un proyecto

Problema: al editar las opciones de configuración de Amazon EC2 durante la creación del proyecto, ve un mensaje de error o una opción no disponible y no puede continuar con la creación del proyecto.

Soluciones posibles: las razones más comunes de este mensaje de error son las siguientes:

- La VPC de la plantilla del AWS CodeStar proyecto (la VPC predeterminada o la que se usó cuando se editó la configuración de Amazon EC2) tiene una tenencia de instancias dedicada y el tipo de instancia no es compatible con las instancias dedicadas. Elija un tipo de instancia diferente o una Amazon VPC diferente.

- Tu AWS cuenta no tiene Amazon VPC. Puede que haya eliminado la VPC predeterminada y no haya creado otras. Abra la consola de Amazon VPC en <https://console.aws.amazon.com/vpc/>, elija las VPC y asegúrese de tener al menos una VPC configurada. En caso contrario, cree una. Para obtener más información, consulte [Introducción a Amazon Virtual Private Cloud](#) en la Guía de introducción a Amazon VPC.
- La Amazon VPC no tiene ninguna subred. Elija otra VPC o cree una subred para la VPC. Para obtener más información, consulte [Conceptos básicos de VPC y subredes](#).

Eliminación de un AWS CodeStar proyecto: se ha eliminado un proyecto, pero aún existen recursos

Problema: se eliminó un AWS CodeStar proyecto, pero los recursos creados para ese proyecto aún existen. De forma predeterminada, AWS CodeStar elimina los recursos del proyecto cuando se elimina el proyecto. Algunos recursos, como los buckets de Amazon S3, se retienen incluso cuando el usuario selecciona la casilla de verificación Eliminar recursos, ya que los buckets podrían contener datos.

Soluciones posibles: abra la [AWS CloudFormation consola](#) y busca una o más de las AWS CloudFormation pilas utilizadas para crear el proyecto. Los nombres de las pilas empezarán con `awscodestar-`, seguido del ID del proyecto. Las pilas pueden estar en la vista de filtro Deleted (Eliminado). Revise los eventos asociados con la pila para descubrir los recursos creados para el proyecto. Abra la consola de cada uno de esos recursos en la AWS región en la que creaste el AWS CodeStar proyecto y, a continuación, elimina los recursos manualmente.

Entre los recursos del proyecto que podrían conservarse se incluyen:

- Uno o varios buckets del proyecto en Amazon S3. A diferencia de otros recursos del proyecto, los depósitos de proyectos en Amazon S3 no se eliminan cuando se selecciona la casilla Eliminar AWS los recursos asociados junto con el AWS CodeStar proyecto.

Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.

- Un repositorio de origen para su proyecto en CodeCommit.

Abre la CodeCommit consola en <https://console.aws.amazon.com/codecommit/>.

- Una canalización para tu proyecto CodePipeline.

Abre la CodePipeline consola en <https://console.aws.amazon.com/codepipeline/>.

- Una aplicación y la implementación asociada se agrupan en CodeDeploy.

Abra la CodeDeploy consola en <https://console.aws.amazon.com/codedeploy/>.

- Aplicación y entornos asociados en AWS Elastic Beanstalk.

Abra la consola de Elastic Beanstalk en <https://console.aws.amazon.com/elasticbeanstalk/>.

- Función en AWS Lambda.

Abra la AWS Lambda consola en <https://console.aws.amazon.com/lambda/>.

- Una o más API en API Gateway.

Abra la consola de API Gateway en <https://console.aws.amazon.com/apigateway/>.

- Una o varias políticas de IAM o roles en IAM.

Inicie sesión en la consola de IAM AWS Management Console y ábrala en <https://console.aws.amazon.com/iam/>.

- Una instancia en Amazon EC2.

Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.

- Uno o más entornos de desarrollo en AWS Cloud9.

Para ver, acceder y gestionar los entornos de desarrollo, abra la AWS Cloud9 consola en <https://console.aws.amazon.com/cloud9/>.

Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos recursos no se eliminan, aunque la casilla Eliminar los AWS recursos asociados junto con el CodeStar proyecto esté seleccionada.

Fallo en la gestión del equipo: no se ha podido añadir un usuario de IAM a un equipo de un proyecto AWS CodeStar

Problema: al intentar añadir un usuario a un proyecto, ve un mensaje de error que indica que la adición ha fallado.

Soluciones posibles: el motivo más común de este error es que el usuario ha llegado al límite de las políticas administradas que se pueden aplicar a un usuario en IAM. También puede recibir este error

si no tiene el rol de propietario en el AWS CodeStar proyecto en el que intentó añadir el usuario, o si el usuario de IAM no existe o se ha eliminado.

Asegúrese de haber iniciado sesión como usuario propietario de ese AWS CodeStar proyecto. Para obtener más información, consulte [Añadir miembros de equipo a un proyecto de AWS CodeStar](#).

Para solucionar otros problemas, abra la consola de IAM, seleccione el usuario que haya intentado agregar y compruebe cuántas políticas administradas se aplican a ese usuario de IAM.

Para obtener más información, consulte [Limitaciones en las entidades y los objetos de IAM](#). Para obtener información sobre los límites que pueden cambiarse, consulte [Límites de los servicios de AWS](#).

Error de acceso: un usuario federado no puede acceder a un proyecto AWS CodeStar

Problema: un usuario federado no puede ver los proyectos en la AWS CodeStar consola.

Soluciones posibles: si ha iniciado sesión como un usuario federado, asegúrese de que tiene la política administrada apropiada asociada al rol que va a asumir para iniciar sesión. Para obtener más información, consulte [Adjunte la política de su proyecto gestionada por AWS CodeStar espectadores, colaboradores o propietarios a la función del usuario federado](#).

Añada usuarios federados a su AWS Cloud9 entorno adjuntando políticas manualmente. Consulte [Adjunte una política AWS Cloud9 gestionada al rol del usuario federado](#).

Error de acceso: un usuario federado no puede acceder ni crear un entorno AWS Cloud9

Problema: un usuario federado no puede ver ni crear un AWS Cloud9 entorno en la AWS Cloud9 consola.

Soluciones posibles: si ha iniciado sesión como un usuario federado, asegúrese de que tiene la política administrada apropiada asociada al rol de usuario federado.

Para añadir usuarios federados a su AWS Cloud9 entorno, debe adjuntar manualmente las políticas a la función del usuario federado. Consulte [Adjunte una política AWS Cloud9 gestionada al rol del usuario federado](#).

Error de acceso: un usuario federado puede crear un AWS CodeStar proyecto, pero no puede ver los recursos del proyecto

Problema: un usuario federado pudo crear un proyecto, pero no puede ver recursos del proyecto, como por ejemplo, la canalización del proyecto.

Posibles soluciones: si ha adjuntado la política **AWSCodeStarFullAccess** gestionada, tiene permisos para crear un proyecto en AWS CodeStar ella. Sin embargo, para obtener acceso a todos los recursos del proyecto, debe asociar la política administrada del propietario.

Una vez AWS CodeStar creados los recursos del proyecto, los permisos del proyecto para todos los recursos del proyecto están disponibles en las políticas gestionadas por el propietario, el contribuyente y el espectador. Para obtener acceso a todos los recursos, debe asociar manualmente la política del propietario a su rol. Consulte [Paso 3: configurar los permisos de IAM del usuario](#).

Error del rol de servicio: el rol de servicio no se ha podido crear

Problema: al intentar crear un proyecto en AWS CodeStar, aparece un mensaje en el que se le pide que cree el rol de servicio. Cuando elige la opción de crearlo, aparece un error.

Posibles soluciones: el motivo más común de este error es que ha iniciado sesión AWS con una cuenta que no tiene los permisos suficientes para crear el rol de servicio. Para crear el rol de AWS CodeStar servicio (`aws-codestar-service-role`), debe iniciar sesión como usuario administrativo o con una cuenta raíz. Cierre sesión en la consola y, a continuación, vuelva a iniciar sesión con un usuario de IAM que tenga aplicada la política administrada `AdministratorAccess`.

Error del rol de servicio: el rol de servicio no es válido o falta

Problema: al abrir la AWS CodeStar consola, aparece un mensaje en el que se indica que el rol de AWS CodeStar servicio no existe o no es válido.

Soluciones posibles: el motivo más común de este error es que un usuario administrativo ha editado o ha eliminado el rol de servicio (`aws-codestar-service-role`). Si se ha eliminado el rol de servicio, se le pide que lo cree. Para crear el rol, debe haber iniciado sesión como usuario administrativo o con una cuenta raíz. Si el rol ha sido editado, ya no es válido. Inicie sesión en la consola de IAM como usuario administrativo, busque el rol de servicio en la lista de roles y elimínelo. Cambie a la AWS CodeStar consola y siga las instrucciones para crear el rol de servicio.

Problema con el rol del proyecto: las comprobaciones del estado de AWS Elastic Beanstalk salud fallan en las instancias de un AWS CodeStar proyecto

Problema: si creó un AWS CodeStar proyecto que incluye Elastic Beanstalk antes del 22 de septiembre de 2017, es posible que no se realicen las comprobaciones de estado de Elastic Beanstalk. Si no ha cambiado la configuración de Elastic Beanstalk desde que creó el proyecto, se producirá un error en la comprobación de estado y el entorno aparecerá atenuado. A pesar del error en la comprobación de estado, la aplicación debe seguir ejecutándose según lo previsto. Si no ha cambiado la configuración de Elastic Beanstalk desde que creó el proyecto, se produce un error en la comprobación de estado y la aplicación podrían no ejecutarse correctamente.

Corrección: a uno o varios roles de IAM le faltan las declaraciones de la política de IAM necesarias. Añada las políticas que faltan a los roles afectados en su cuenta de AWS .

1. [Inicie sesión en la consola de IAM AWS Management Console y ábrala en https://console.aws.amazon.com/iam/.](https://console.aws.amazon.com/iam/)

(Si no puede hacerlo, póngase en contacto con el administrador de su AWS cuenta para obtener ayuda).

2. Seleccione Roles en el panel de navegación.
3. En la lista de funciones, elija CodeStarWorker**Project-ID -EB**, donde *Project-ID es el ID* de uno de los proyectos afectados. (Si no puede encontrar fácilmente un rol en la lista, escriba parte o todo el nombre del rol en el cuadro Search (Buscar).)
4. En la pestaña Permissions, elija Attach Policy.
5. En la lista de políticas, seleccione y.
AWSElasticBeanstalkEnhancedHealthAWSElasticBeanstalkService (Si no puede encontrar fácilmente una política en la lista, escriba parte o todo el nombre de la política en el cuadro Search (Buscar).)
6. Seleccione Asociar política.
7. Repita los pasos 3 a 6 para cada rol afectado que tenga un nombre que siga el patrón CodeStarWorker: **PROJECT-ID -EB**.

Error del rol de proyecto: un rol de proyecto no es válido o falta

Problema: al intentar añadir un usuario a un proyecto, ve un mensaje de error que indica que la adición ha fallado porque la política de un rol de proyecto falta o no es válida.

Soluciones posibles: el motivo más común de este error es que una o varias políticas del proyecto se han editado o se han eliminado de IAM. Las políticas del proyecto son exclusivas de los AWS CodeStar proyectos y no se pueden volver a crear. El proyecto no se pueden utilizar. Cree un proyecto en y AWS CodeStar, a continuación, migre los datos al nuevo proyecto. Clone el código de proyecto del repositorio del proyecto que no se puede utilizar e inserte ese código en el repositorio del nuevo proyecto. Copie la información de la wiki del equipo del antiguo proyecto en el proyecto nuevo. Añada usuarios al nuevo proyecto. Cuando esté seguro de que se han migrado todos los datos y ajustes, elimine el proyecto inservible.

Extensiones del proyecto: no se puede conectar a JIRA

Problema: cuando utilizas la extensión JIRA de Atlassian para intentar conectar un AWS CodeStar proyecto a una instancia de JIRA, aparece el siguiente mensaje: «La URL no es una URL de JIRA válida. Verify that the URL is correct" (La URL no es una URL de JIRA válida. Compruebe que la URL es correcta).

Soluciones posibles:

- Asegúrese de que la URL de JIRA es correcta y, a continuación, intente conectarse de nuevo.
- Su instancia de JIRA con alojamiento propio puede que no sea accesible desde la red pública de Internet. Póngase en contacto con el administrador de red para asegurarse de que se puede obtener acceso a su instancia de JIRA desde la red pública de Internet y, a continuación, intente conectarse de nuevo.

GitHub: No se puede acceder al historial de confirmaciones, los problemas o el código de un repositorio

Problema: en el panel de control de un proyecto que almacena su código GitHub, los cuadros Historial de confirmaciones y GitHubProblemas muestran un error de conexión, o si se selecciona Abrir en GitHub o Crear problema en estos mosaicos, se muestra un error.

Causas posibles:

- Es posible que el AWS CodeStar proyecto ya no tenga acceso al GitHub repositorio.
- Es posible que el repositorio se haya eliminado o se le haya cambiado el nombre GitHub.

AWS CloudFormation: Restauración de creación de pila para permisos ausentes

Después de añadir un recurso al archivo `template.yml`, vea la actualización de la pila de AWS CloudFormation para ver si hay algún mensaje de error. Se produce un error en la actualización de la pila si no se cumplen determinados criterios (por ejemplo, cuando faltan permisos a nivel de recursos necesarios).

Note

El 2 de mayo de 2019, actualizamos la política de funciones de los AWS CloudFormation trabajadores para todos los proyectos existentes. Esta actualización reduce el ámbito de permisos de acceso concedidos a la canalización de proyectos para mejorar la seguridad en los proyectos.

Para solucionar problemas, consulta el estado de la falla en la vista del AWS CodeStar panel de control de la cartera de tu proyecto.

A continuación, selecciona el CloudFormation enlace en la fase de implementación de la canalización para solucionar el error en la AWS CloudFormation consola. Para ver detalles de creación de la pila, expanda la lista Events (Eventos) para su proyecto y ver cualquier mensaje de error. El mensaje indica qué permisos faltan. Corrija la política de proceso de trabajo AWS CloudFormation y, a continuación, vuelva a ejecutar la canalización.

AWS CloudFormation no está autorizado a realizar la función de ejecución iam: PassRole on Lambda

Si tiene un proyecto creado antes del PDT del 6 de diciembre de 2018 que crea funciones Lambda, es posible que aparezca AWS CloudFormation un error como este:

```
User: arn:aws:sts::id:assumed-role/CodeStarWorker-project-id-CloudFormation/  
AWSCloudFormation is not authorized to perform: iam:PassRole on resource:
```

```
arn:aws:iam::id:role/CodeStarWorker-project-id-Lambda (Service: AWSLambdaInternal;  
Status Code: 403; Error Code: AccessDeniedException; Request ID: id)
```

Este error se produce porque su función de AWS CloudFormation trabajador no tiene permiso para transferir una función para aprovisionar la nueva función de Lambda.

Para corregir este error, tendrá que actualizar su política de roles de AWS CloudFormation trabajador con el siguiente fragmento.

```
{  
  "Action": [ "iam:PassRole" ],  
  "Resource": [  
    "arn:aws:iam::account-id:role/CodeStarWorker-project-id-Lambda",  
  ],  
  "Effect": "Allow"  
}
```

Después de actualizar la política, ejecute de nuevo la canalización.

También puede utilizar un rol personalizado para su función de Lambda si añade un límite de permisos al proyecto, como se describe en [Agregue un límite de IAM permisos a los proyectos existentes](#)

No se ha podido crear la conexión para un repositorio GitHub

Problema:

Dado que una conexión a un GitHub repositorio utiliza el AWS Connector GitHub, necesitas permisos de propietario de la organización o permisos de administrador del repositorio para crear la conexión.

Posibles correcciones: para obtener información sobre los niveles de permisos de un GitHub repositorio, consulta <https://docs.github.com/en/free-pro-team@latest/github/organization-setting-up-and-managing-organizations-and-teams/permission-levels-for-an>

Notas de la versión de la Guía de usuario de AWS CodeStar

En la tabla siguiente, se describen los cambios importantes de cada versión de la Guía del usuario de AWS CodeStar. Para obtener notificaciones sobre las actualizaciones de esta documentación, puede suscribirse a una fuente RSS.

Cambio	Descripción	Fecha
Actualizaciones de la política de acceso	Se ha actualizado la política de roles de acceso a AWS CodeStar. El resultado de la política es el mismo, pero cloudformation requiere ListStacks además de DescribeStacks, que ya es obligatorio. Para obtener más referencias sobre la política actualizada, consulte la Política de AWSCodeStarFullAccess .	24 de marzo de 2023
Actualizaciones de la política de roles de servicio	Se ha actualizado la política de roles de servicio de AWS CodeStar. Para obtener más referencias sobre la política actualizada, consulte la Política de AWSCodeStarServiceRole .	23 de septiembre de 2021
Utilizar un recurso de conexión para proyectos con un repositorio de código fuente de GitHub	Al utilizar la consola para crear un proyecto de AWS CodeStar con un repositorio de GitHub, se utiliza un recurso de conexión para administrar las acciones de GitHub. Las conexiones utilizan aplicaciones de GitHub, mientras que	27 de abril de 2021

la autorización anterior de GitHub utilizaba OAuth. Para ver un tutorial que muestra cómo crear un proyecto que utiliza una conexión a GitHub, consulte el [Tutorial: Crear un proyecto con un repositorio de código fuente de GitHub](#). En el tutorial también se explica cómo crear, revisar y combinar una solicitud de extracción para el repositorio de fuentes del proyecto.

[AWS CodeStar ahora es compatible con AWS Cloud9 en la región Oeste de EE. UU. \(Norte de California\)](#)

AWS CodeStar ahora es compatible con AWS Cloud9 en la región Oeste de EE. UU. (Norte de California). Para obtener más información, consulte [Configuración de Cloud9](#).

16 de febrero de 2021

[Actualizar la documentación para reflejar la nueva experiencia de la consola](#)

El 12 de agosto de 2020, el servicio de AWS CodeStar pasó a una nueva experiencia de usuario en la consola de AWS. La guía del usuario se ha actualizado a la nueva experiencia de la consola.

12 de agosto de 2020

[Los proyectos de AWS CodeStar se pueden crear con la CLI de AWS CodeStar](#)

Los proyectos de AWS CodeStar se pueden crear con el comando de la CLI. AWS CodeStar crea el proyecto y la infraestructura con código fuente y la plantilla de la cadena de herramientas que proporcione. Consulte [Crear un proyecto en AWS CodeStar \(CLI de AWS\)](#).

24 de octubre de 2018

[Todas las plantillas de proyectos de AWS CodeStar ahora incluyen un archivo de AWS CloudFormation para las actualizaciones de infraestructura](#)

AWS CodeStar funciona con AWS CloudFormation para permitirle utilizar código para crear servicios de soporte y servidores o plataformas sin servidor en la nube. El archivo de AWS CloudFormation ya está disponible para todos los tipos de plantilla de proyecto de AWS CodeStar (las plantillas con las plataformas de computación de Lambda, EC2 o Elastic Beanstalk). El archivo se almacena en `template.yml` en el repositorio de origen. Puede ver y modificar el archivo para añadir recursos a su proyecto. Consulte [Plantillas de proyecto](#).

3 de agosto de 2018

[Notificaciones de actualización de la Guía del usuario de AWS CodeStar ya disponibles a través de RSS](#)

La versión HTML de la Guía del usuario de AWS CodeStar ahora admite una fuente RSS de las actualizaciones que se documentan en la página Notas de la versión de las actualizaciones de la documentación. La fuente RSS incluye las actualizaciones realizadas a partir del 30 de junio de 2018. Las actualizaciones anunciadas anteriormente siguen estando disponibles en la página Notas de la versión de las actualizaciones de la documentación. Utilice el botón RSS del panel del menú superior para suscribirse a la fuente.

30 de junio de 2018

En la siguiente tabla se describen los cambios importantes de cada versión de la Guía del usuario de AWS CodeStar anteriores al 30 de junio de 2018.

Cambio	Descripción	Fecha de modificación
La Guía del usuario de AWS CodeStar ahora está disponible en GitHub.	Esta guía ya está disponible en GitHub. También puede utilizar GitHub para enviar comentarios y realizar solicitudes de cambio del contenido de esta guía. Para obtener más información, haga clic en el icono Edit on GitHub (Editar en GitHub) en la barra de navegación de la guía o consulte el repositorio de awsdocs/aws-codestar-user-guide en el sitio web de GitHub.	22 de febrero de 2018

Cambio	Descripción	Fecha de modificación
AWS CodeStar ya está disponible en Asia-Pacífico (Seúl)	AWS CodeStar ya está disponible en la región Asia-Pacífico (Seúl). Para obtener más información, consulte AWS CodeStar en la Referencia general de Amazon Web Services.	14 de febrero de 2018
AWS CodeStar ya está disponible en Asia-Pacífico (Tokio) y Canadá (centro)	AWS CodeStar ya está disponible en las regiones Asia-Pacífico (Tokio) y Canadá (centro). Para obtener más información, consulte AWS CodeStar en la Referencia general de Amazon Web Services.	20 de diciembre de 2017
AWS CodeStar ya admite AWS Cloud9	AWS CodeStar ya admite el uso de AWS Cloud9, un IDE online basado en un navegador web, para trabajar con el código del proyecto. Para obtener más información, consulte Usar AWS Cloud9 con AWS CodeStar . Para obtener una lista de las regiones de AWS compatibles, consulte AWS Cloud9 en la Referencia general de Amazon Web Services.	30 de noviembre de 2017
AWS CodeStar ya es compatible con GitHub	AWS CodeStar ahora es compatible con el almacenamiento de código de proyectos en GitHub. Para obtener más información, consulte la sección sobre crear un proyecto .	12 de octubre de 2017
AWS CodeStar ya disponible en Oeste de EE. UU. (Norte de California) y Europa (Londres)	AWS CodeStar ya está disponible en la región Oeste de EE. UU. (Norte de California) y Europa (Londres). Para obtener más información, consulte AWS CodeStar en la Referencia general de Amazon Web Services.	17 de agosto de 2017

Cambio	Descripción	Fecha de modificación
AWS CodeStar ya disponible en Asia-Pacífico (Sídney), Asia-Pacífico (Singapur) y Europa (Fráncfort)	AWS CodeStar ya está disponible en las regiones Asia-Pacífico (Sídney), Asia-Pacífico (Singapur) y Europa (Fráncfort). Para obtener más información, consulte AWS CodeStar en la Referencia general de Amazon Web Services.	25 de julio de 2017
AWS CloudTrail ya admite AWS CodeStar	AWS CodeStar ya se ha integrado con CloudTrail, un servicio que captura las llamadas a la API realizadas por AWS CodeStar o en su nombre, en su cuenta de AWS y que envía los archivos de registro al bucket de Amazon S3 que se especifique. Para obtener más información, consulte Registro de llamadas a la API de AWS CodeStar con AWS CloudTrail .	14 de junio de 2017
Versión inicial	Esta es la primera versión de la Guía del usuario de AWS CodeStar.	19 de abril de 2017

Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el [Glosario de AWS](#) en la Referencia de Glosario de AWS.