Guía del usuario

AWS CodeStar

AWS CodeStar: Guía del usuario

Table of Contents

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 version original de inglés, prevalecerá la version 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, crear e implementar aplicaciones rápidamente en AWS un AWS CodeStar proyecto. Un AWS CodeStar proyecto crea e integra AWS servicios para su cadena de herramientas de desarrollo de proyectos. Según la plantilla de AWS CodeStar proyecto que elijas, esa cadena de herramientas puede incluir el control de código fuente, la compilación, el despliegue, los servidores virtuales o los recursos sin servidor, etc. AWS CodeStar también administra los permisos necesarios para los usuarios del proyecto (denominados miembros del equipo). Al añadir usuarios como miembros del equipo a un AWS CodeStar proyecto, los propietarios del proyecto pueden conceder de forma rápida y sencilla a cada miembro del equipo el acceso adecuado a su función al proyecto y a sus recursos.

Temas

¿Con qué puedo hacer? AWS CodeStar

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

- Inicie nuevos proyectos de software AWS en cuestión de minutos con plantillas para aplicaciones web, 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 los recursos del proyecto están configurados para funcionar juntos.
- 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 a todos los AWS servicios utilizados en el proyecto, por lo que no es necesario crear ni gestionar políticas de IAM complejas.
- Visualice, opere y colabore en sus proyectos en un solo lugar: AWS CodeStar incluye un panel de control del proyecto que proporciona una visión general del proyecto, su cadena de herramientas y los 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 puedo empezar AWS CodeStar?

Para empezar con AWS CodeStar:

1. Prepárese para usarlo AWS CodeStar siguiendo los pasos que se indican enConfiguración AWS CodeStar.
2. Experimente AWS CodeStar siguiendo los pasos del Cómo empezar con AWS CodeStar tutorial.
3. Comparta el proyecto con otros desarrolladores siguiendo los pasos de Añadir miembros del equipo a un AWS CodeStar proyecto .
4. Integre su IDE favorito siguiendo los pasos que se indican en Utilice 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

Inscríbase en una Cuenta de AWS

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

Para suscribirte a una Cuenta de AWS

1. Abrir https://portal.aws.amazon.com/billing/registro.
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 un Cuenta de AWS, Usuario raíz de la cuenta de AWSse 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. En cualquier momento, puede ver la actividad de su cuenta actual y administrarla accediendo a 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 Consolecomo 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 Iniciar sesión como usuario raíz 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.

Inicio de sesión como usuario con acceso de administrador

- Para iniciar sesión con el usuario de IAM Identity Center, use 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. Funciones de AWS CodeStar IAM

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 gestionados 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 EC2 claves de Amazon para AWS CodeStar proyectos

Muchos AWS CodeStar proyectos utilizan AWS CodeDeploy o AWS Elastic Beanstalk implementan código en las EC2 instancias de Amazon. Para acceder a EC2 las instancias de Amazon asociadas a su proyecto, cree un EC2 key pair de Amazon para su usuario de IAM. Tu usuario de IAM debe tener permisos para crear y gestionar EC2 las claves de Amazon (por ejemplo, permiso para realizar las `ec2:ImportKeyPair` acciones `ec2:CreateKeyPair` y acciones). Para obtener más información, consulte Amazon EC2 Key Pairs.

Paso 5: Abre la AWS CodeStar consola

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

Siguientes pasos

¡Enhorabuena! Ha completado la configuración. Para empezar a trabajar con AWS CodeStar ella, consulteCómo empezar con AWS CodeStar.

Cómo empezar con AWS CodeStar

En este tutorial, se utiliza 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:

- Cree 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 AWS Management Console con un usuario de IAM que tenga la **`AWSCodeStarFullAccess`**política.

Temas

- Tutorial: creación y administración de un proyecto sin servidor en AWS CodeStar
- Tutorial: Cree un proyecto AWS CodeStar con AWS CLI
- Tutorial: Crea un proyecto de habilidades de Alexa en AWS CodeStar
- Tutorial: Crear un proyecto con un repositorio GitHub de fuentes

Paso 1: Crear un proyecto AWS CodeStar

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

> **Note**
>
> La plantilla de AWS CodeStar proyecto utilizada en este tutorial utiliza las siguientes opciones:
>
> - Categoría de la aplicación: aplicación web
> - Lenguaje de programación: Node.js
> - AWS Servicio: 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 y AWS Management Console, a continuación, abra la AWS CodeStar consola en https://console.aws.amazon.com/codestar/.

   Asegúrese de haber iniciado sesión en la AWS región en la que desea crear el proyecto y sus recursos. Por ejemplo, para crear un proyecto en EE. UU. Este (Ohio), asegúrese de haber seleccionado esa AWS región. Para obtener información sobre AWS las regiones donde AWS CodeStar está disponible, consulte Regiones y puntos finales en la Referencia AWS general.
2. En la página AWS CodeStar, seleccione Crear proyecto.
3. En la página Elija una plantilla de proyecto, elija el tipo de proyecto de la lista de plantillas de AWS CodeStar proyectos. Puede utilizar la barra de filtros para restringir las opciones. Por ejemplo, para implementar un proyecto de aplicación web escrito en Node.js en EC2 instancias

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

Para obtener más información, consulte AWS CodeStar Plantillas de proyectos.

4. Elija Next (Siguiente).
5. En el campo de entrada de texto del nombre del proyecto, introduzca un nombre para el proyecto, como*My First Project*. 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 denominado *My First Project* es *my-first-projec*. 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.

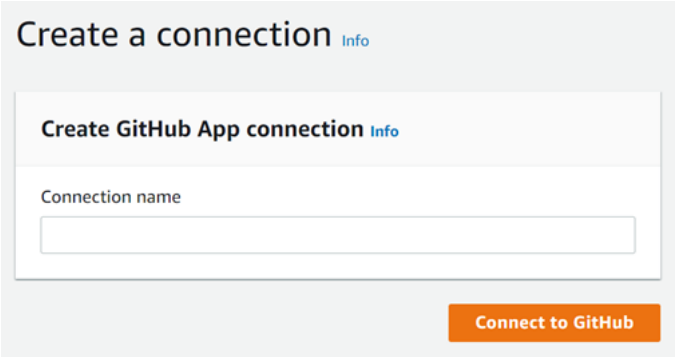
   Para obtener información sobre los límites de los nombres de los proyectos y los proyectos IDs, consulteLímites en AWS CodeStar.

   > **Note**
   >
   > El proyecto IDs debe ser único para su AWS cuenta en una AWS región.

6. Elige el proveedor del repositorio, AWS CodeCommito GitHub.
7. Si lo elige AWS CodeCommit, en Nombre del repositorio, acepte el nombre de AWS CodeCommit repositorio predeterminado o introduzca uno diferente. A continuación, vaya al paso 9.
8. Si lo desea 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. Selecciona Conectar a GitHub.

   Se mostrará la página Crear una conexión.

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



a. En Crear conexión a una GitHub aplicación, en el campo de texto de entrada del nombre de la conexión, introduzca un nombre para la conexión. Selecciona Conectar a GitHub.

   Aparece la GitHub página Conectar a y muestra el campo GitHub Aplicaciones.

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

> **Note**
>
> Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya ha instalado el AWS conector para la GitHub aplicación, elíjalo y omita este paso.

c. En la GitHub página Instalar AWS conector para, elige la cuenta en la que quieres 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.

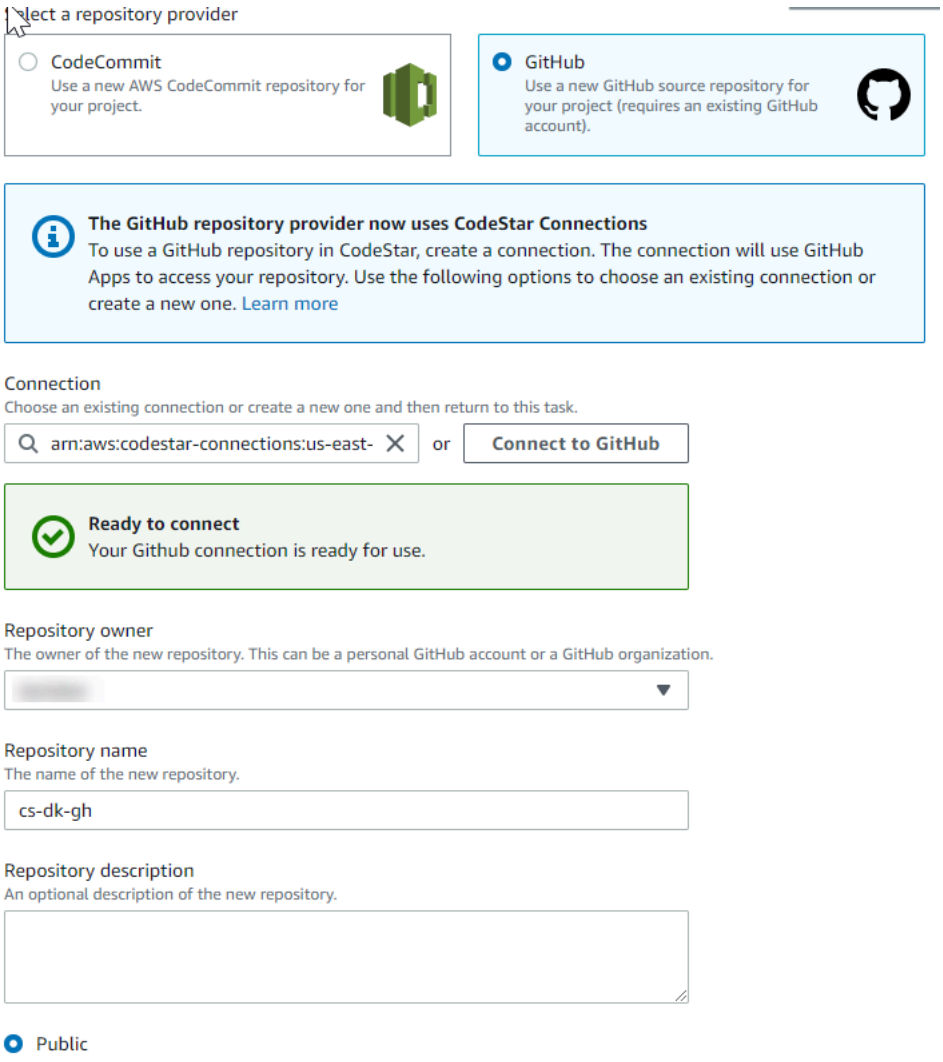
d. Si aparece la página Confirmar la contraseña para continuar, introduzca la GitHub contraseña y, a continuación, seleccione Iniciar sesión.

e. En la GitHub página Instalar el AWS conector para, mantenga los valores predeterminados y seleccione Instalar.

f. En la GitHub página Conectar a, el identificador de instalación de la nueva instalación aparece en el campo de entrada de texto GitHub Aplicaciones.

Una vez creada la conexión, en la página de CodeStar creación del proyecto, 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.



g. Como propietario del repositorio, elige la GitHub organización o tu GitHub cuenta personal.

h. En Nombre del repositorio, acepte el nombre del GitHub repositorio predeterminado o introduzca uno diferente.

i. Elija Público o Privado.

> **Note**
>
> Para usarlo AWS Cloud9 como entorno de desarrollo, debe elegir Público.

j. (Opcional) En la descripción del repositorio, introduzca una descripción para el GitHub repositorio.

> **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: Crea un proyecto de habilidades de Alexa en AWS CodeStar.

9. Si tu proyecto está desplegado en EC2 instancias de Amazon y deseas realizar cambios, configura tus EC2 instancias de Amazon en Amazon EC2 Configuration. Por ejemplo, puede elegir entre los tipos de instancia disponibles para el proyecto.

> **Note**
>
> Los distintos tipos de EC2 instancias de Amazon proporcionan distintos niveles de potencia informática y pueden tener costes asociados diferentes. Para obtener más información, consulte Tipos de EC2 instancias de Amazon y EC2 precios de Amazon.
> 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 eliges un tipo de EC2 instancia de Amazon que no sea compatible con las instancias dedicadas, no podrás elegir una VPC cuya tenencia de instancias esté configurada como Dedicada.
> Para obtener más información, consulte ¿Qué es Amazon VPC? y Conceptos básicos de las instancias dedicadas.

En Par de claves, elige el par de EC2 claves de Amazon en el que creastePaso 4: Crear un par de EC2 claves de Amazon para AWS CodeStar proyectos. Seleccione Confirmo 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).

    Es posible que el proyecto, que incluye el repositorio, tarde unos minutos en crearse.
13. 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 lo utilizas AWS CodeStar, se te pedirá que proporciones:

- 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 AWS CodeStar de usuario. Los perfiles de usuario no son específicos del proyecto, sino que se limitan a una AWS región. Debe crear un perfil de usuario en cada AWS región en la que pertenezca a 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 utilizan en su perfil de AWS CodeStar usuario. Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos proveedores de recursos pueden tener sus propios perfiles de usuario, con diferentes nombres de usuario y direcciones de correo

electrónico. Para obtener más información, consulte la documentación del proveedor de recursos.

Paso 3: ver el proyecto

La página de tu AWS CodeStar proyecto es donde tú y tu equipo veis el estado de los recursos de tu proyecto, incluidas las últimas confirmaciones del proyecto, el estado de tu proceso de entrega continua y el rendimiento de tus 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 de configuración IDEs para ambos GitHub AWS CodeCommit repositorios e información sobre sus AWS Cloud9 entornos.
- La página Repositorio muestra los detalles del repositorio, incluidos el nombre, el proveedor, cuándo se modificó por última vez y el clon URLs. 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 de monitorización muestra Amazon EC2 o AWS Lambda las métricas en función de la configuración del proyecto. Por ejemplo, muestra el uso de la CPU de todas las EC2 instancias de Amazon implementadas por AWS Elastic Beanstalk o CodeDeploy los recursos de tu canalización. En los proyectos que la utilizan AWS Lambda, muestra las métricas de invocación y error de la función Lambda. Esta información se muestra por hora. Si usaste la plantilla de AWS CodeStar proyecto sugerida para este tutorial, deberías observar un aumento notable en la actividad cuando la aplicación se implemente por primera vez en esas 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 de problemas sirve para integrar tu AWS CodeStar proyecto 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. Este es el ejemplo del proyecto que se AWS CodeStar creó e implementó.

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 está almacenado en un GitHub repositorio CodeCommit o repositorio, puede utilizarlo AWS Cloud9 para trabajar con el código directamente desde su navegador web, sin necesidad de instalar ninguna herramienta. Para obtener más información, consulte Cree un AWS Cloud9 entorno para un proyecto.
- Si el código del proyecto está almacenado en un CodeCommit repositorio y tienes Visual Studio o Eclipse instalados, puedes usar AWS Toolkit for Visual Studio o AWS Toolkit for Eclipse para conectarte más fácilmente al código. Para obtener más información, consulte Utilice 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 GitHub repositorio, puedes usar las herramientas del IDE para conectarte a él GitHub.
  - En el caso de Visual Studio, puede utilizar herramientas como la GitHub extensión para Visual Studio. Para obtener más información, consulte la página de información general en el sitio web de la GitHub extensión para Visual Studio y la sección GitHub Introducción a Visual Studio en el GitHub sitio web.

  - En el caso de Eclipse, puede utilizar una herramienta como EGit Eclipse. Para obtener más información, consulte la EGitdocumentación del sitio EGit web.
  - Para obtener más información IDEs, 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 CodeCommit y ya lo estás utilizando CodeCommit o has utilizado la AWS CodeStar consola para crear un entorno de AWS Cloud9 desarrollo para el proyecto, no necesitas más configuración. 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 consola de IAM AWS Management Console y ábrala en https://console.aws.amazon.com/iam/.

   Inicia sesión como el usuario de IAM que usará las credenciales de Git para las conexiones al repositorio de tu AWS CodeStar proyecto. CodeCommit
3. En el panel de navegación de la consola de IAM, seleccione Usuarios y, en la lista de usuarios seleccione su usuario de IAM.
4. En la página de detalles del usuario, selecciona la pestaña Credenciales de seguridad y, en Credenciales de Git HTTPS CodeCommit, selecciona Generar.

> **Note**
>
> No puede elegir sus propias credenciales de inicio de sesión para las credenciales de Git. Para obtener más información, consulta 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.

> **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. Instala el AWS CLI.

3. Configure sus credenciales de seguridad temporales para un usuario federado. Para obtener información, consulte Acceso temporal a los CodeCommit repositorios. Las credenciales temporales constan de:

   - AWS clave de acceso
   - AWS clave secreta
   - Token de sesión

   Para obtener más información sobre las credenciales temporales, consulte Permisos para GetFederationToken.
4. Conéctese a su repositorio mediante el asistente de AWS CLI credenciales. Para obtener más información, consulte Pasos de configuración para conexiones HTTPS a CodeCommit repositorios en Linux, macOS o Unix con el asistente de credenciales AWS CLI o Pasos de configuración para conexiones HTTPS a CodeCommit repositorios en Windows con el asistente de credenciales CLI AWS
5. El siguiente ejemplo muestra cómo conectarse a un CodeCommit repositorio y enviar una confirmación a él.

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, asumimos que el código de tu proyecto está almacenado en un CodeCommit repositorio y que utilizas un cliente 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 ha utilizado la AWS CodeStar consola para crear un entorno de AWS Cloud9 desarrollo 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 Abra un AWS Cloud9 entorno para un proyecto.

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

2. En el equipo local, abra un terminal o una ventana de línea de comandos y cambe 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 usar 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 ello 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 termina 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 AWS CodeStar proyecto ya está configurado con tres AWS CodeStar funciones. 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 CodeCommit, pero no puede añadir ni eliminar miembros del equipo ni eliminar el proyecto. Este es el rol que debes elegir para la mayoría de los miembros del equipo de un AWS CodeStar proyecto.
- Visor: puede ver el panel del proyecto, el código del proyecto si está almacenado y el estado del proyecto, pero no puede mover, añadir ni eliminar teselas del panel del proyecto. CodeCommit

> **Important**
>
> Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), el acceso a esos 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 AWS CodeStar proyecto podría utilizar la AWS CodeStar consola para acceder a recursos ajenos al proyecto AWS pero relacionados con él. AWS CodeStar no permite que los miembros del equipo del proyecto participen en ningún entorno de AWS Cloud9 desarrollo relacionado con un proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte Comparta un AWS Cloud9 entorno con un miembro del equipo del proyecto.

Para obtener más información acerca de los equipos y roles de proyectos, consulte Trabajando con AWS CodeStar equipos.

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

1. Abre la AWS CodeStar consola 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 han agregado a otro AWS CodeStar proyecto aparecen en la lista de AWS CodeStar usuarios existentes.

     En el rol del proyecto, elija el AWS CodeStar rol (propietario, colaborador o espectador) para 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 acceder a los recursos AWS CodeStar del proyecto. Aplica las políticas necesarias para crear y administrar las credenciales de Git para el código almacenado CodeCommit en IAM o para cargar las claves EC2 SSH de Amazon 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 Administre la información de visualización de su perfil de AWS CodeStar usuario .

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 el usuario de IAM, vuelve a la AWS CodeStar consola, actualiza la lista de usuarios y elige el usuario de IAM que creaste en la lista desplegable. Introduce el nombre AWS CodeStar para mostrar, la dirección de correo electrónico y el rol del proyecto que deseas aplicar a este nuevo usuario y, a continuación, selecciona Añadir 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 tu AWS CodeStar proyecto.
   - Si el código fuente está almacenado en CodeCommit, instrucciones para configurar el acceso con credenciales de Git al CodeCommit repositorio desde sus ordenadores locales.
   - Información sobre cómo el usuario puede gestionar su nombre visible, dirección de correo electrónico y clave EC2 SSH pública de Amazon, tal y como se describe enCómo trabajar con su perfil AWS CodeStar de usuario .
   - 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 quieres seguir usando este proyecto y sus recursos, debes eliminarlo para evitar posibles cargos continuos a tu AWS cuenta.

Para eliminar un proyecto en AWS CodeStar

1. Abre la AWS CodeStar consola 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 de la AWS CodeStar consola.

   > **Important**
   >
   > Si tu proyecto utiliza recursos ajenos a AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos recursos no se eliminan, aunque selecciones la casilla de verificación.
   > Tu proyecto no se puede eliminar si alguna política AWS CodeStar gestionada se ha asociado manualmente a funciones 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.

Siguientes pasos

Estos son otros recursos que le ayudarán a obtener información sobre AWS CodeStar:

- Tutorial: creación y administración de un proyecto sin servidor en AWS CodeStarUtiliza un proyecto que crea e implementa un servicio web mediante la lógica AWS Lambda y una API de Amazon API Gateway lo puede llamar.
- AWS CodeStar Plantillas de proyectos describe otros tipos de proyectos que puede crear.
- Trabajando con AWS CodeStar equipos 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, se utiliza AWS CodeStar para crear un proyecto que utilice el modelo de aplicaciones AWS sin servidor (AWS SAM) para crear y administrar AWS los recursos de un servicio web alojado en él. AWS Lambda

AWS CodeStar usa AWS SAM, que se basa en AWS CloudFormation, para proporcionar una forma simplificada de crear y administrar AWS los recursos compatibles, incluidos Amazon API Gateway APIs, AWS Lambda las funciones y las tablas de Amazon DynamoDB. (Este proyecto no utiliza ninguna tabla de Amazon DynamoDB).

Para obtener más información, consulte el Modelo de aplicaciones AWS sin servidor (AWS SAM) en. GitHub

Requisitos previos: Complete los pasos de Configuración AWS CodeStar.

> **Note**
>
> Es posible que se le cobren a su AWS cuenta los costos relacionados con este tutorial, incluidos los costos de los AWS servicios utilizados por AWS CodeStar. Para obtener más información, consulte AWS CodeStar Precios.

Temas

Descripción general

En este tutorial, va a:

1. Se utiliza AWS CodeStar para crear un proyecto que utilice AWS SAM para crear e implementar un servicio web basado en Python. Este servicio web está alojado AWS Lambda y se puede acceder a él a través de Amazon API Gateway.
2. Explorar los recursos principales del proyecto, que incluyen:
   - El AWS CodeCommit repositorio donde se almacena el código fuente del proyecto. Este código fuente incluye la lógica del servicio web y define recursos de AWS relacionados.
   - La AWS CodePipeline canalización que automatiza la creación del código fuente. Esta canalización utiliza AWS SAM para crear e implementar una función para AWS Lambda, crear una API relacionada en Amazon API Gateway y conectar la API a la función.

- La función en la que se implementa AWS Lambda.
- La API que se crea en Amazon API Gateway.

3. Pruebe el servicio web para confirmar que AWS CodeStar creó e implementó el servicio web según lo esperado.
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. Vuelva a probar el servicio web para confirmar que se AWS CodeStar reconstruyó y se reimplementó 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 realizar la prueba unitaria, AWS CodeStar se reconstruye y vuelve a implementar el servicio web y se ejecuta la prueba unitaria.
8. Consultar los resultados de las pruebas de unidad.
9. Eliminar el proyecto. Este paso te ayuda a evitar que se carguen a tu AWS cuenta los costes relacionados con este tutorial.

Paso 1: creación del proyecto

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

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

   > **Note**
   >
   > Debe iniciar sesión con las AWS Management Console credenciales asociadas al usuario de IAM que creó o con el que se identificó. Configuración AWS CodeStar Este usuario debe tener la política administrada **`AWSCodeStarFullAccess`** asociada.

2. Elija la AWS región en la que desea crear el proyecto y sus recursos.

   Para obtener información sobre AWS las regiones donde AWS CodeStar está disponible, consulte Regiones y puntos finales en la Referencia AWS general.

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 contenga sus selecciones. Elija Next (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.

   Para el identificador del proyecto, AWS CodeStar elige 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 Next (Siguiente).
8. Revise la configuración y, a continuación, seleccione Crear presupuesto.

   Si es la primera vez que lo usa AWS CodeStar en esta AWS región, en Nombre para mostrar y correo electrónico, introduzca el nombre para mostrar y la dirección de correo electrónico que desee usar AWS CodeStar para su usuario de IAM. Elija Next (Siguiente).
9. Espere mientras AWS CodeStar crea el proyecto. Esto podría tardar varios minutos. No continúe hasta que vea el banner Proyecto aprovisionado al actualizar.

Paso 2: explorar recursos del proyecto

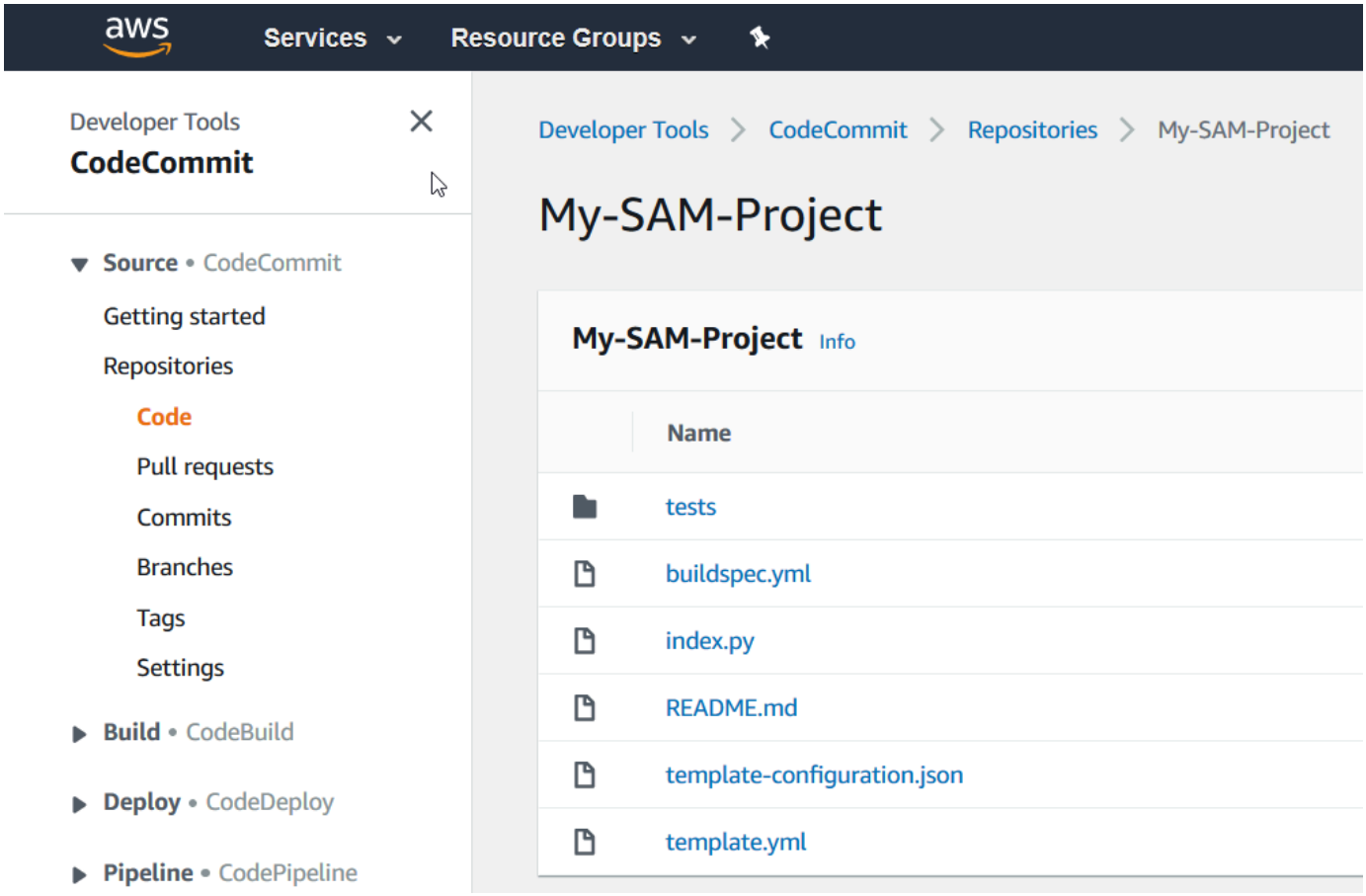
En este paso, explorarás cuatro de los AWS recursos del proyecto para entender cómo funciona:

- El AWS CodeCommit repositorio donde se almacena el código fuente del proyecto. AWS CodeStar da el nombre al repositorio my-sam-project, donde my-sam-projectestá el nombre del proyecto.
- La AWS CodePipeline canalización que utiliza CodeBuild un AWS SAM para automatizar la creación e implementación de la función Lambda y la API del servicio web en API Gateway. AWS CodeStar asigna a la canalización el nombre my-sam-project--Pipeline, que my-sam-projectes el ID del proyecto.
- La función Lambda que contiene la lógica del servicio web. AWS CodeStar da a la función el nombre awscodestar-my-sam-project-lambda- HelloWorld -, donde: ***`RANDOM_ID`***
  - my-sam-projectes el ID del proyecto.

  - HelloWorldes el identificador de la función tal como se especifica en el `template.yaml` archivo del AWS CodeCommit repositorio. Puede explorar este archivo más adelante.
  - *`RANDOM_ID`*es un identificador aleatorio que AWS SAM asigna a la función para garantizar su exclusividad.
- La API de API Gateway que facilita la llamada a la función Lambda. AWS CodeStar asigna a la API el nombre awscodestar-my-sam-project--lambda, que my-sam-projectes el ID del proyecto.

Para explorar el repositorio de código fuente en CodeCommit

1. Con el proyecto abierto en la AWS CodeStar consola, en la barra de navegación, selecciona Repositorio.
2. Elige el enlace a tu CodeCommit repositorio (**`My-SAM-Project`**) en Detalles del repositorio.
3. En la CodeCommit consola, en la página de códigos, se muestran los archivos de código fuente del proyecto:
   - `buildspec.yml`, que CodePipeline indica que se CodeBuild debe utilizar durante la fase de creación para empaquetar el servicio web mediante 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 AWS SAM utiliza para empaquetar el servicio web y crear la API en API Gateway.



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

Para obtener más información sobre el uso de la CodeCommit consola, consulte la Guía del AWS CodeCommit usuario.

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 Source (Fuente) para obtener el código fuente desde CodeCommit.
   - Una etapa Build (Compilar) para crear el código fuente con CodeBuild.
   - Una etapa de implementación para implementar el código fuente y AWS los recursos creados con AWS SAM.

2. Para ver más información sobre la canalización, en Detalles de la canalización, selecciona tu canalización para abrirla en la CodePipeline consola.

Para obtener información sobre el uso de la CodePipeline consola, consulta la Guía del AWS CodePipeline usuario.

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

1. Abre tu proyecto en la AWS CodeStar consola y, en la barra de navegación, selecciona Descripción 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 AWS CodeStar consola, en la barra de navegación lateral, selecciona Descripción 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 AWS CodeStar consola, en la barra de navegación lateral, selecciona Descripción 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, probarás el servicio web que AWS CodeStar acabas de crear 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. Clona el CodeCommit repositorio del proyecto en tu 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 usa este código para crear una función en Lambda, agregar un método y una ruta nuevos a la API en API Gateway y, a continuación, conectar este método y esta 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, indica que se debe CodePipeline usar un AWS SAM para reconstruir CodeBuild y volver a 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-, donde: *RANDOM_ID*

- my-sam-projectes el identificador del proyecto.
- Hello es el ID de la función tal como se especifica en el archivo `template.yaml`.
- *RANDOM_ID*es un identificador aleatorio que AWS SAM asigna a la función para que sea única.

Paso 6: probar el servicio web mejorado

En este paso, se prueba el servicio web mejorado que se AWS CodeStar creó e implementó, en función de la lógica que se agregó en el paso anterior.

1. Con el proyecto aún abierto en la AWS CodeStar consola, en la barra de navegación, selecciona Pipeline.
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 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 direcciones de la pestaña, añade la ruta **`/hello/`** y tu nombre al final de la URL (por ejemplo, https://***`API_ID`***.execute-api). ***`REGION_ID`***.amazonaws. com/Prod/hello/***`YOUR_FIRST_NAME`***) y, a continuación, presione Entrar.

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, utilizará su estación de trabajo local para agregar una prueba que AWS CodeStar se ejecute 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 CodeBuild que se instale pytest, el marco de pruebas de Python, en su entorno de compilación. CodeBuild usa 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 aún abierto en la AWS CodeStar consola, en la barra de navegación, selecciona Pipeline.
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 de los resultados de la prueba unitaria, en la etapa de creación, selecciona el CodeBuildenlace.
4. En la CodeBuild consola, en la my-sam-project página Construir proyecto:, en Historial de compilaciones, elija el enlace de la columna Construir ejecución de la tabla.
5. En la *BUILD_ID* página my-sam-project:, en Crear registros, selecciona el enlace Ver todo el registro.
6. En la consola de Amazon CloudWatch Logs, busque en el resultado del registro 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 quieres seguir utilizando este proyecto, puedes saltarte este paso, pero es posible que se sigan cobrando a tu AWS cuenta.

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

> **Important**
>
> Si desactivas esta casilla, se eliminará el registro del proyecto AWS CodeStar, pero se conservarán muchos de los AWS recursos del proyecto. Es posible que se sigan realizando cargos en tu AWS cuenta.

Si todavía hay un bucket de Amazon S3 AWS CodeStar creado para este proyecto, sigue estos pasos para eliminarlo. :

1. Abra la consola Amazon S3, en https://console.aws.amazon.com/s3/.
2. En la lista de buckets, elija el icono situado junto a aws-codestar- ***`REGION_ID`*** - - --pipe, donde: ***`ACCOUNT_ID`*** my-sam-project
   - *`REGION_ID`*es el ID de la AWS región del proyecto que acabas de eliminar.
   - *`ACCOUNT_ID`*es el ID AWS de tu cuenta.
   - my-sam-projectes el ID del proyecto que acabas 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.

Siguientes pasos

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

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

Tutorial: Cree un proyecto AWS CodeStar con AWS CLI

Este tutorial le muestra cómo usarlo AWS CLI para crear un AWS CodeStar proyecto con un ejemplo de código fuente y una plantilla de cadena de herramientas de ejemplo. AWS CodeStar aprovisiona la AWS infraestructura y los recursos de IAM especificados en una plantilla de cadena de AWS CloudFormation herramientas. El proyecto administra los recursos de cadena de herramientas para compilar e implementar el código fuente.

AWS CodeStar AWS CloudFormation se utiliza para crear e implementar el código de muestra. Este código de ejemplo crea un servicio web que está hospedado AWS Lambda y al que se puede acceder a través de 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 le cobren a su AWS cuenta los costos relacionados con este tutorial, incluidos AWS los servicios utilizados por AWS CodeStar. Para obtener más información, consulte AWS CodeStar Precios.

Temas

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 de unidad 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`: El archivo de plantilla de infraestructura o el archivo de plantilla SAM incluidos en todos los AWS CodeStar proyectos. 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 cadena de herramientas de ejemplo que se proporciona en este tutorial crea un repositorio (CodeCommit), una canalización (CodePipeline) y un contenedor de compilación (CodeBuild) y se utiliza AWS CloudFormation para implementar el código fuente en una plataforma Lambda. Además de estos recursos, también hay funciones de IAM que puede utilizar para determinar los permisos de su entorno de ejecución, un bucket de Amazon S3 que se CodePipeline utiliza para almacenar los artefactos de despliegue y una regla de CloudWatch eventos que se utiliza para activar las implementaciones en canalización al insertar código en su 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.

Descarga y descomprime la AWS CloudFormation plantilla 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 AWS::CodeCommit::Repository AWS CloudFormation recurso crea un CodeCommit repositorio.
- El AWS::CodeBuild::Project AWS CloudFormation recurso crea un proyecto de CodeBuild compilación.
- El AWS::CodeDeploy::Application AWS CloudFormation recurso crea una CodeDeploy aplicación.
- El AWS::CodePipeline::Pipeline AWS CloudFormation recurso crea una CodePipeline canalización.
- El AWS::S3::Bucket AWS CloudFormation recurso crea el depósito de artefactos de tu canalización.
- El AWS::S3::BucketPolicy AWS CloudFormation recurso crea la política de depósitos de artefactos para el depósito de artefactos de tu canalización.
- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de trabajador de CodeBuild IAM que otorga AWS CodeStar permisos para gestionar tu CodeBuild proyecto de construcción.
- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de trabajador de CodePipeline IAM que otorga AWS CodeStar permisos para crear tu canalización.
- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de trabajador de AWS CloudFormation IAM que otorga AWS CodeStar permisos para crear tu pila de recursos.

- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de trabajador de AWS CloudFormation IAM que otorga AWS CodeStar permisos para crear tu pila de recursos.
- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de trabajador de AWS CloudFormation IAM que otorga AWS CodeStar permisos para crear tu pila de recursos.
- El AWS::Events::Rule AWS CloudFormation recurso crea la regla de CloudWatch eventos que supervisa el repositorio en busca de eventos push.
- El AWS::IAM::Role AWS CloudFormation recurso crea el rol de IAM de CloudWatch eventos.

Paso 3: Pruebe su plantilla de 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 su ordenador local y abra la AWS CloudFormation consola. Elija Crear pila. 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 ella 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 AWS CodeStar proyecto, primero debe empaquetar el código fuente en un archivo.zip y colocarlo en Amazon S3. AWS CodeStar inicializa su repositorio con estos contenidos. 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. Esta ubicación se especifica en el archivo de entrada cuando se ejecuta el comando para crear el proyecto en 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

- Como está almacenando el código fuente y la cadena de herramientas en Amazon S3, puede utilizar las políticas y objetos de bucket de Amazon S3 ACLs para garantizar que otros usuarios o AWS cuentas de IAM puedan crear proyectos a partir de sus muestras. AWS CodeStar garantiza que cualquier usuario que cree un proyecto personalizado tenga acceso a la cadena de herramientas y a la fuente 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: Crea un proyecto en AWS CodeStar

Siga estos pasos para crear su proyecto.

> **Important**
>
> Asegúrese de configurar la AWS región preferida en AWS CLI. Su proyecto se crea en la AWS región configurada en 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 a un archivo (por ejemplo,*`input.json`*) en una ubicación de su equipo local o instancia en la que AWS CLI esté instalado. 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 agregar miembros al equipo o configurar el acceso al repositorio de su proyecto desde la línea de comandos o su IDE favorito.

Tutorial: Crea un proyecto de habilidades de Alexa en AWS CodeStar

AWS CodeStar es un servicio de desarrollo basado en la nube AWS que proporciona las herramientas que necesita para desarrollar, crear e implementar aplicaciones rápidamente. AWS Con él AWS CodeStar, puede configurar toda su cadena de herramientas de entrega continua en cuestión de minutos, lo que le permitirá empezar a publicar código más rápido. Las plantillas de proyectos de habilidades de Alexa te AWS CodeStar permiten crear una simple habilidad de Alexa de Hello World desde tu AWS cuenta con solo unos pocos 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.

Los principales beneficios de crear habilidades de Alexa AWS CodeStar son que puedes empezar con el desarrollo de habilidades AWS y conectar tu cuenta de desarrollador de Amazon al proyecto para implementar habilidades en la fase de desarrollo directamente desde allí 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

- Para crear una cuenta de desarrollador de Amazon, ve a https://developer.amazon.com. El registro es gratuito. La cuenta tiene sus habilidades de Alexa.
- Si no tiene una AWS cuenta, utilice el siguiente procedimiento para crear una.

  Para registrarse en AWS

  1. Abre https://aws.amazon.com/y, a continuación, selecciona Crear una AWS cuenta.

     > **Note**
     >
     > Si ya has iniciado sesión AWS Management Console con Usuario raíz de la cuenta de AWS las credenciales, selecciona Iniciar sesión en una cuenta diferente. 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 que se le indiquen.

> **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 y AWS Management Console, a continuación, abra la AWS CodeStar consola en https://console.aws.amazon.com/codestar/.
2. Elija la AWS región en la que desee crear el proyecto y sus recursos. El tiempo de ejecución de las habilidades de Alexa está disponible en las siguientes AWS regiones:
   - 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 utilizas un nombre diferente, asegúrate de usarlo a lo largo de este tutorial. AWS CodeStar elige un identificador relacionado para este proyecto como ID del proyecto (por ejemplo, my-alexa-skill). Si ve un ID de proyecto diferente, asegúrese de utilizarlo durante todo el tutorial.
7. Elija AWS CodeCommit para el repositorio en este tutorial y no cambie el valor del 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 tienes varios proveedores IDs asociados a tu cuenta de desarrollador de Amazon, elige el que quieras usar para este proyecto. Asegúrese de utilizar una cuenta que tenga asignada el rol de administrador o desarrollador.

11. Elija Next (Siguiente).
12. (Opcional) Si es la primera vez que la utilizas AWS CodeStar en esta AWS región, introduce el nombre visible y la dirección de correo electrónico que quieres usar AWS CodeStar para tu usuario de IAM. Elija Next (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 creé una habilidad para ti y la implementaste en la etapa de desarrollo de habilidades de Alexa. Ahora va a probar dicha habilidad en el simulador de Alexa.

1. En el proyecto de la AWS CodeStar consola, selecciona 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 AWS creé recursos en tu nombre. Estos recursos incluyen el uso de un repositorio de proyectos CodeCommit, un proceso de implementación CodePipeline y una AWS Lambda función. Puede acceder a estos recursos desde la barra de navegación. Por ejemplo, al elegir un repositorio, se muestran detalles sobre el CodeCommit repositorio. Puede ver el estado de implementación de la canalización en la página Canalización. Para ver una lista completa de AWS los recursos creados como parte de su proyecto,

seleccione Descripción 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 AWS CodeStar y consulte la página Pipeline. 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, realizaste un pequeño cambio en el código fuente directamente desde la CodeCommit consola. 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. Si es necesario AWS CodeStar, dirígete al panel del proyecto.
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 AWS CLI. Para obtener información, consulte Instalación de la interfaz de línea de AWS comandos.
   c. Configure la AWS CLI con la clave de acceso de usuario y la clave secreta de IAM. Para obtener información, consulte Configuración de la AWS CLI.
   d. Clone el CodeCommit repositorio del proyecto en su estación de trabajo local. Para obtener más información, consulte Conectarse a un CodeCommit repositorio.

Siguientes pasos

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

- Descubre los aspectos básicos de una habilidad viendo Cómo funcionan las habilidades de Alexa y otros vídeos en el YouTube canal de desarrolladores de Alexa.
- 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.
- Convierte tu idea en una habilidad consultando la documentación del Alexa Skills Kit y del ASK SDKs.

Tutorial: Crear un proyecto con un repositorio GitHub de fuentes

Con AWS CodeStar ella, puedes configurar tu repositorio para crear, revisar y fusionar solicitudes de extracción con tu equipo de proyecto.

En este tutorial, crearás un proyecto con un ejemplo de código fuente de una aplicación web en un GitHub repositorio, una canalización en la que se implementarán tus cambios e EC2 instancias en las que tu aplicación esté alojada en la nube. Una vez creado el proyecto, en este tutorial se muestra cómo crear y combinar una solicitud de GitHub incorporación de cambios que modifique la página de inicio de la aplicación web.

Temas

Paso 1: Crea el proyecto y crea tu GitHub repositorio

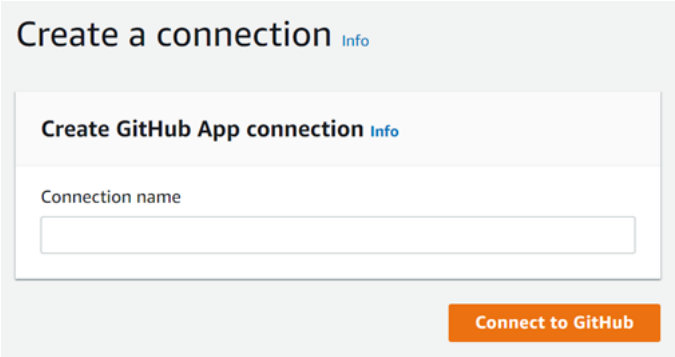
En este paso, usa la consola para crear tu proyecto y crear una conexión con tu nuevo GitHub repositorio. Para acceder a tu GitHub repositorio, debes crear un recurso de conexión que se AWS CodeStar utilice para gestionar la autorización GitHub. Al crear el proyecto, se aprovisionan sus recursos adicionales para el usuario.

1. Inicie sesión en y AWS Management Console, a continuación, abra la AWS CodeStar consola en https://console.aws.amazon.com/codestar/.
2. Elija la AWS región en la que 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, active las casillas de EC2 verificación Aplicación web, Node.js y Amazon. A continuación, elija entre las plantillas disponibles para ese conjunto de opciones.

   Para obtener más información, consulte AWS CodeStar Plantillas de proyectos.
5. Elija Next (Siguiente).
6. En 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, selecciona GitHub.
8. Si lo desea 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. Selecciona Conectar a GitHub.

   Se mostrará la página Crear una conexión.

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



a. En Crear conexión de GitHub aplicación, en Nombre de conexión, introduce un nombre para tu conexión. Selecciona Conectar a GitHub.

   Aparece la GitHub página Conectar a y muestra el campo GitHub Aplicaciones.

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

> **Note**
>
> Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya ha instalado el AWS conector para la GitHub aplicación, elíjalo y omita este paso.

c. En la GitHub página Instalar el AWS conector para, elige la cuenta en la que quieres 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.

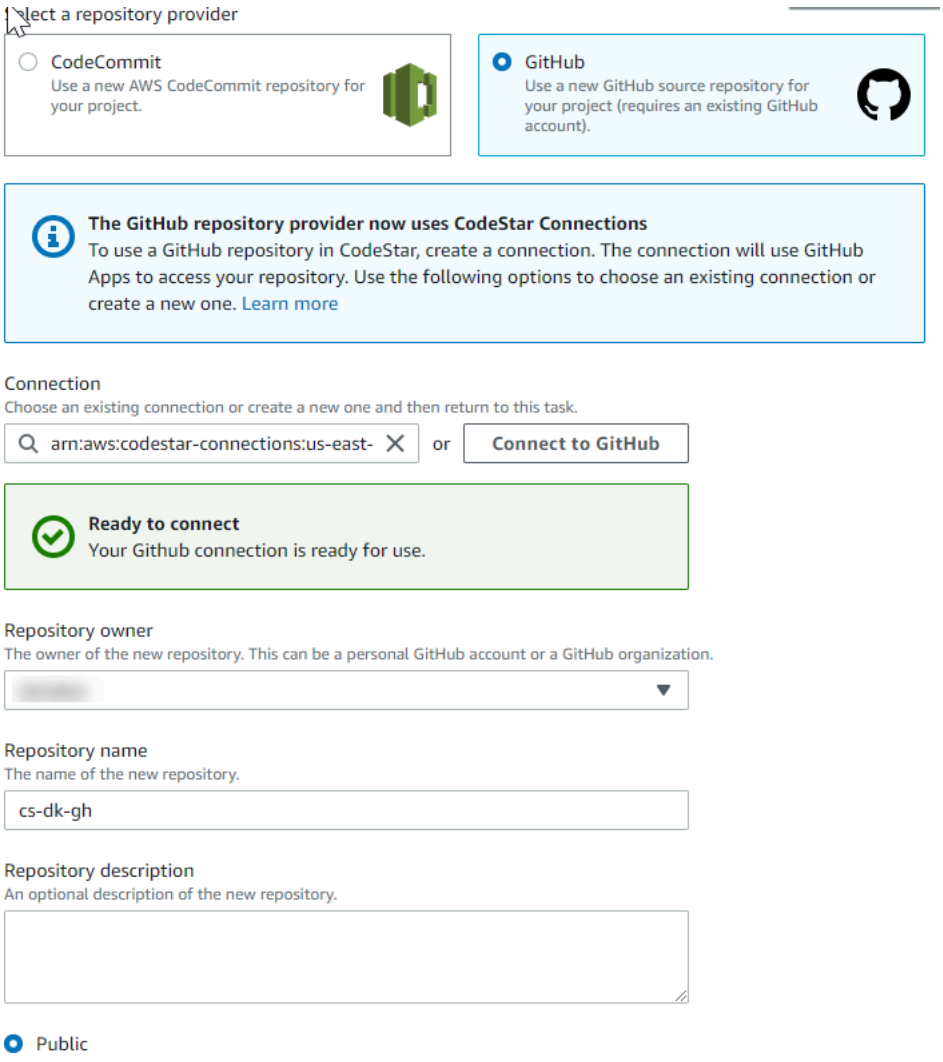
d. Si aparece la página Confirmar la contraseña para continuar, introduzca la GitHub contraseña y, a continuación, seleccione Iniciar sesión.

e. En la GitHub página Instalar el AWS conector para, deje los valores predeterminados y elija Instalar.

f. En la GitHub página Conectar a, el ID de instalación de la nueva instalación aparece en GitHubAplicaciones.

Cuando la conexión se haya creado correctamente, en la página de CodeStar creación del proyecto, 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.



g. Como propietario del repositorio, elige la GitHub organización o tu GitHub cuenta personal.

h. En Nombre del repositorio, acepte el nombre del GitHub repositorio predeterminado o introduzca uno diferente.

i. Elija Público o Privado.

> **Note**
>
> Si quiere usarlo AWS Cloud9 como entorno de desarrollo, debe elegir un repositorio público.

j. (Opcional) En la descripción del repositorio, introduce una descripción para el GitHub repositorio.

9. Configura tus EC2 instancias de Amazon en Amazon EC2 Configuration si tu proyecto está desplegado en EC2 instancias de Amazon y deseas realizar cambios. Por ejemplo, puede elegir entre los tipos de instancia disponibles para el proyecto.

   En Par de claves, elige el par de EC2 claves de Amazon en el que creastePaso 4: Crear un par de EC2 claves de Amazon para AWS CodeStar proyectos. Seleccione Confirmo 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 GitHub, selecciona Ver confirmaciones. Esto abre tu historial de confirmaciones en GitHub.

   Para ver los problemas, seleccione la pestaña Problemas del proyecto. Para crear una nueva emisión en GitHub, selecciona Crear GitHub incidencia. Esto abre el formulario de problemas de tu 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: Crea una solicitud de GitHub extracción

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

1. En GitHub, crea una nueva rama de funciones en tu 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, realiza un cambio en la `feature-branch` rama. Abra la carpeta pública y, a continuación, abra el archivo `index.html`.
3. En la AWS CodeStar consola, en Solicitudes de extracción, para crear una solicitud de extracción GitHub, selecciona Crear solicitud de extracción. Esto abre el formulario de solicitud de extracción de tu repositorio GitHub. En GitHub, elige 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 AWS CodeStar consola, elige tu proyecto. Seleccione la pestaña Repositorio. En Solicitudes de extracción, seleccione Crear la solicitud de extracción.

   El formulario se abre en GitHub. Deje la ramificación principal en la ramificación base. En Comparar con, elija la ramificación de características. Observe las diferencias.
5. En GitHub, selecciona Crear solicitud de extracción. Se creará una solicitud de extracción denominada Update index.html.
6. En la AWS CodeStar consola, 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. Regresa al proyecto AWS CodeStar y consulta la página Pipeline. Debería ver la canalización implementándose.
8. Una vez creado el proyecto, seleccione Ver la aplicación para ver la aplicación web.

AWS CodeStar Plantillas de proyectos

AWS CodeStar Las plantillas de proyecto le permiten comenzar con una aplicación de muestra e implementarla utilizando AWS los recursos creados para respaldar su proyecto de desarrollo. Al elegir una plantilla de AWS CodeStar proyecto, se le proporcionan automáticamente el tipo de aplicación, el lenguaje de programación y la plataforma de cómputo. 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 AWS CodeStar vez creado el proyecto, puede modificar los AWS recursos que respaldan la entrega de la aplicación. AWS CodeStar funciona AWS CloudFormation para permitirle usar el código para crear servicios de soporte y servidores/plataformas sin servidor en la nube. AWS CloudFormation le permite modelar toda su infraestructura en un archivo de texto.

Temas

AWS CodeStar Archivos y recursos del proyecto

Un AWS CodeStar proyecto es una combinación del código fuente y 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. En el momento de la creación del proyecto, una AWS CloudFormation plantilla aprovisiona los recursos de la cadena de herramientas (de forma continuaintegration/continuous deployment (CI/CD).

Puedes utilizarla AWS CodeStar para crear proyectos de dos maneras, según tu nivel de experiencia en la creación de AWS recursos:

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

para poder crear su proyecto AWS CodeStar a partir de la plantilla y, a continuación, rellenar el repositorio con el código de muestra.

Un AWS CodeStar proyecto proporciona un único 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 son recursos de la cadena de herramientas, como AWS los recursos que se utilizan para las notificaciones, el soporte de bases de datos, la supervisión y el seguimiento.
- AWS servicios y recursos creados en relación con su canalización, como el depósito de artefactos de Amazon S3, Amazon CloudWatch Events y funciones de servicio relacionadas.
- Una aplicación de muestra funcional con código fuente completo y un punto de conexión de HTTP pública.
- Un recurso AWS informático, basado en el tipo de plantilla AWS CodeStar del proyecto:
  - Una función Lambda.
  - Una EC2 instancia de Amazon.
  - Un AWS Elastic Beanstalk entorno.
- 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.

  - Una función de AWS CloudFormation IAM para crear los recursos del proyecto mediante la cual AWS CloudFormation se incluyen los permisos para todos los recursos AWS CloudFormation compatibles, incluidas las funciones 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 AWS CloudFormation IAM para crear recursos del proyecto con soporte para un conjunto limitado de recursos. AWS CloudFormation
  - Un rol de IAM para crear un CodePipeline recurso.
  - Un rol de IAM para crear un CodeBuild recurso.
  - Un rol de IAM para crear un CodeDeploy recurso, si corresponde a su tipo de proyecto.
  - Un rol de IAM para crear la aplicación EC2 web de Amazon, si corresponde a tu tipo de proyecto.
  - Un rol de IAM para crear un recurso de CloudWatch eventos.
  - Un rol de ejecución para Lambda que se modifica de forma dinámica para incluir un conjunto parcial de recursos.

El proyecto incluye páginas de detalles que muestran el estado y contienen enlaces a la gestión del equipo, enlaces a las instrucciones IDEs de configuración del repositorio y 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 eliges un AWS CodeStar proyecto en la consola, eliges entre un conjunto de opciones preconfiguradas con ejemplos de código y recursos para empezar rápidamente. Estas opciones se denominan plantillas de proyecto. Cada plantilla de AWS CodeStar proyecto consta de un lenguaje de programación, un tipo de aplicación y una plataforma informática. 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:

- Cuando eliges un AWS Elastic Beanstalk proyecto, lo despliegas en un AWS Elastic Beanstalk entorno de instancias de Amazon Elastic Compute Cloud en la nube.
- Cuando eliges un EC2 proyecto de Amazon, AWS CodeStar crea EC2 instancias de Linux para alojar tu aplicación en la nube. Los miembros de tu equipo de proyecto pueden acceder a las instancias, y tu equipo utilizará el key pair que proporciones a SSH en tus EC2 instancias de Amazon. AWS CodeStar también tiene un SSH administrado que usa los permisos de los miembros del equipo para administrar las conexiones de key pair.
- Cuando lo desee AWS Lambda, AWS CodeStar crea un entorno sin servidores al que se accede a través de Amazon API Gateway, sin instancias ni 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 las tareas que se ejecutan en segundo plano, como las llamadas APIs. Una AWS CodeStar vez creado el proyecto de servicio web de muestra, puede elegir la URL del punto final para ver el resultado de Hello World, pero el uso principal de este tipo de aplicación no es como interfaz de usuario (UI). Las plantillas de AWS CodeStar proyectos de esta categoría admiten el desarrollo en Ruby, Java, ASP.NET, PHP, Node.js y más.

- Aplicación web

  Una aplicación web incluye una IU. Una vez AWS CodeStar creado el proyecto de aplicación web de muestra, puede elegir la URL del punto de conexión para ver una aplicación web interactiva. Las plantillas de AWS CodeStar proyecto de esta categoría admiten el desarrollo en Ruby, Java, ASP.NET, PHP, Node.js y más.

- Página web estática

  Elija esta plantilla si desea un proyecto para un sitio web HTML. Las plantillas de AWS CodeStar proyecto 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 habilidades, AWS CodeStar devuelve un nombre de recurso de Amazon

(ARN) que puede utilizar como punto de enlace del servicio. Para obtener más información, consulte Hospedar una habilidad personalizada como una AWS función 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 un proyecto para una AWS Config regla que le permita automatizar las reglas en todos AWS los recursos de 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 realizar cambios en tu AWS CodeStar proyecto

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 estás trabajando con un AWS CodeStar proyecto de Alexa Skills, no puedes realizar cambios en la habilidad fuera del repositorio de AWS CodeStar origen (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:

- Usar el modo de edición en CodeCommit o GitHub.
- Abrir el proyecto en un IDE, como 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 un recurso de infraestructura, utilícelo AWS CloudFormation para modelar e implementar los recursos de 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 AWS Lambda función, consulteAñadir un recurso a un proyecto.

Modifique el `template.yml` archivo en el repositorio de su proyecto para añadir el tipo de AWS CloudFormation recursos que son recursos de aplicación. Cuando agrega un recurso de aplicación a la `Resources` sección del `template.yml` archivo AWS CloudFormation y AWS CodeStar crea el recurso automáticamente. Para obtener una lista de AWS CloudFormation los recursos y las propiedades necesarias, consulte la Referencia AWS de tipos de recursos. Para obtener más información, consulte este ejemplo en Paso 1: edite el rol del CloudFormation trabajador en IAM.

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

Cómo administrar permisos para cambiar los recursos de aplicaciones

Cuando se utilizan AWS CloudFormation para agregar recursos de aplicaciones en tiempo de ejecución, como una función Lambda, el rol de AWS CloudFormation trabajador puede usar 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 de cómo cambiar los permisos del rol de AWS CloudFormation trabajador, consultePaso 5: Añadir permisos a nivel de recursos con una política insertada.

AWS CodeStar Mejores prácticas

AWS CodeStar está integrado con una serie de productos y servicios. En las siguientes secciones se describen las mejores prácticas AWS CodeStar y estos productos y servicios relacionados.

Temas

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 AWS CodeStar plantilla 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 ejemplo de tu AWS CodeStar proyecto utiliza las dependencias que se enumeran en el `package.json` archivo de tu repositorio de código fuente. 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

Puedes usar las funciones de registro AWS para determinar las acciones que los usuarios han realizado en tu 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 usar para registrar las llamadas a la AWS API y los eventos relacionados realizados por una AWS cuenta o en su nombre. Para obtener más información, consulte Registrar llamadas a la AWS CodeStar API con AWS CloudTrail.

Trabajar con proyectos en AWS CodeStar

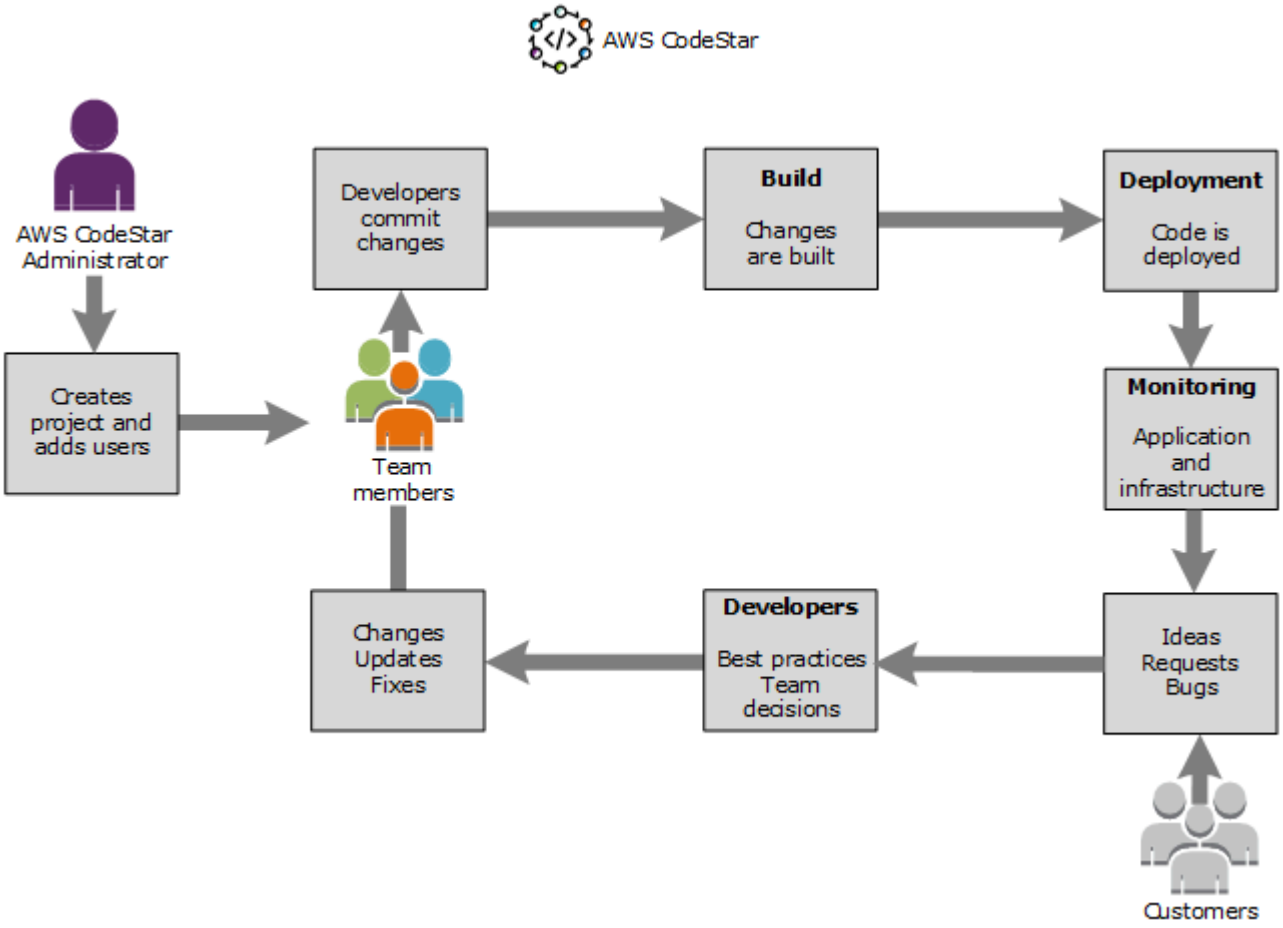
Cuando utilizas una plantilla de AWS CodeStar proyecto, puedes crear rápidamente un proyecto que ya esté configurado con los recursos que necesitas, entre los que se incluyen:

- 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.

El siguiente diagrama muestra un flujo de trabajo básico de un AWS CodeStar proyecto.



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

La AWS CodeStar consola se utiliza 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 AWS Management Console con un usuario de IAM que tenga la `AWSCodeStarFullAccess` política o 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)
- Crea un proyecto en AWS CodeStar (AWS CLI)

Crear un proyecto en AWS CodeStar (consola)

Utilice la AWS CodeStar consola para crear un proyecto.

Para crear un proyecto en AWS CodeStar

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

   Asegúrese de haber iniciado sesión en la AWS región en la que desea crear el proyecto y sus recursos. Por ejemplo, para crear un proyecto en EE. UU. Este (Ohio), asegúrese de haber seleccionado esa AWS región. Para obtener información sobre AWS las regiones en las que AWS CodeStar está disponible, consulte Regiones y puntos finales en la Referencia AWS general.
2. En la página AWS CodeStar, seleccione Crear proyecto.
3. En la página Elija una plantilla de proyecto, elija el tipo de proyecto de la lista de plantillas de AWS CodeStar proyectos. Puede utilizar la barra de filtros para restringir las opciones. Por

ejemplo, para implementar un proyecto de aplicación web escrito en Node.js en EC2 instancias de Amazon, active las casillas de EC2 verificación Aplicación web, Node.js y Amazon. A continuación, elija entre las plantillas disponibles para ese conjunto de opciones.

Para obtener más información, consulte AWS CodeStar Plantillas de proyectos.

4. Elija Next (Siguiente).
5. En el campo de entrada de texto del nombre del proyecto, introduzca un nombre para el proyecto, como*`My First Project`*. 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 denominado *`My First Project`* es *`my-first-projec`*. 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.

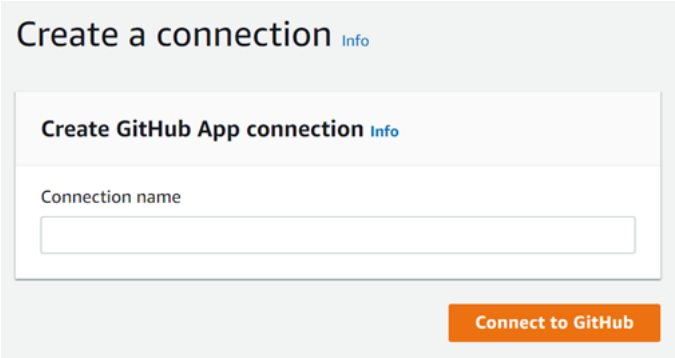
   Para obtener información sobre los límites de los nombres de los proyectos y los proyectos IDs, consulteLímites en AWS CodeStar.

   > **Note**
   >
   > El proyecto IDs debe ser único para su AWS cuenta en una AWS región.

6. Elige el proveedor del repositorio, AWS CodeCommito GitHub.
7. Si lo elige AWS CodeCommit, en Nombre del repositorio, acepte el nombre de AWS CodeCommit repositorio predeterminado o introduzca uno diferente. A continuación, vaya al paso 9.
8. Si lo desea 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. Selecciona Conectar a GitHub.

   Se mostrará la página Crear una conexión.

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



a. En Crear conexión a una GitHub aplicación, en el campo de texto de entrada del nombre de la conexión, introduzca un nombre para la conexión. Selecciona Conectar a GitHub.

   Aparece la GitHub página Conectar a y muestra el campo GitHub Aplicaciones.

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

   > **Note**
   >
   > Se instala una aplicación para todas las conexiones a un proveedor en particular. Si ya ha instalado el AWS conector para la GitHub aplicación, elíjalo y omita este paso.

c. En la GitHub página Instalar el AWS conector para, elige la cuenta en la que quieres 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.

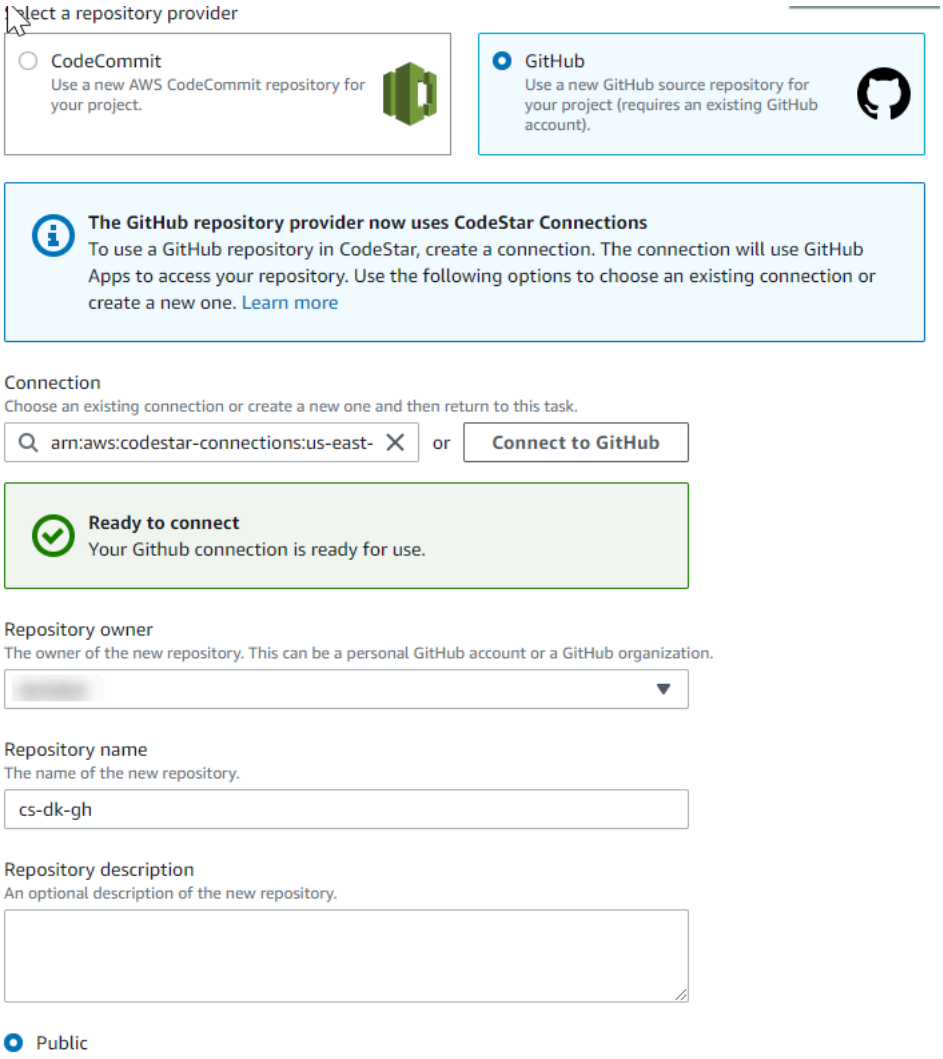
d. Si aparece la página Confirmar la contraseña para continuar, introduzca la GitHub contraseña y, a continuación, seleccione Iniciar sesión.

e. En la GitHub página Instalar el AWS conector para, mantenga los valores predeterminados y seleccione Instalar.

f. En la GitHub página Conectar a, el identificador de instalación de la nueva instalación aparece en el campo de entrada de texto GitHub Aplicaciones.

Una vez creada la conexión, en la página de CodeStar creación del proyecto, 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.



g. Como propietario del repositorio, elige la GitHub organización o tu GitHub cuenta personal.

h. En Nombre del repositorio, acepte el nombre del GitHub repositorio predeterminado o introduzca uno diferente.

i. Elija Público o Privado.

> **Note**
>
> Para usarlo AWS Cloud9 como entorno de desarrollo, debe elegir Público.

j. (Opcional) En la descripción del repositorio, introduzca una descripción para el GitHub repositorio.

> **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: Crea un proyecto de habilidades de Alexa en AWS CodeStar.

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

   > **Note**
   >
   > Los distintos tipos de EC2 instancias de Amazon proporcionan distintos niveles de potencia informática y pueden tener costes asociados diferentes. Para obtener más información, consulte Tipos de EC2 instancias de Amazon y EC2 precios de Amazon.
   > 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 eliges un tipo de EC2 instancia de Amazon que no sea compatible con las instancias dedicadas, no podrás elegir una VPC cuya tenencia de instancias esté configurada como Dedicada.
   > Para obtener más información, consulte ¿Qué es Amazon VPC? y Conceptos básicos de las instancias dedicadas.

   En Par de claves, elige el par de EC2 claves de Amazon en el que creastePaso 4: Crear un par de EC2 claves de Amazon para AWS CodeStar proyectos. Seleccione Confirmo 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).

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

13. 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.

Crea un proyecto en AWS CodeStar (AWS CLI)

Un AWS CodeStar proyecto es una combinación del código fuente y 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. En el momento de la creación del proyecto, una AWS CloudFormation plantilla aprovisiona los recursos de la cadena de herramientas (de forma continuaintegration/continuous deployment (CI/CD).

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

Una cadena de herramientas completa requiere los siguientes recursos recomendados:

1. Un GitHub repositorio CodeCommit o repositorio que contiene su código fuente.
2. Una CodePipeline canalización configurada para escuchar los cambios en tu repositorio.
   a. Cuando utilices CodeBuild pruebas unitarias o de integración, te recomendamos que añadas una etapa de compilación a tu canalización para crear artefactos de compilación.
   b. Te recomendamos que añadas una etapa de despliegue a tu canalización que utilice CodeDeploy o AWS CloudFormation despliegue el artefacto de compilación y el código fuente en tu infraestructura de tiempo de ejecución.

> **Note**
>
> Como CodePipeline requiere al menos dos etapas en una canalización y la primera debe ser la etapa de origen, agrega una etapa de compilación o implementación como segunda etapa.

AWS CodeStar Las cadenas de herramientas se definen como una CloudFormationplantilla.

Para ver un tutorial en el que se explica esta tarea y se configuran los recursos de muestra, consulte Tutorial: Cree un proyecto AWS CodeStar con AWS CLI.

Requisitos previos:

Al crear un proyecto, debe proporcionar los siguientes parámetros en un archivo de entrada. Si no se proporciona lo siguiente, AWS CodeStar crea 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 todos los archivos de configuración necesarios, como un buildspec.yml para un CodeBuild proyecto o un appspec.yml para una implementación. CodeDeploy
  - Puedes incluir elementos opcionales en tu código fuente, como un archivo README o un template.yml para recursos ajenos a la cadena de herramientas. AWS
- Plantilla de la cadena de herramientas. La plantilla de la cadena de herramientas proporciona los AWS recursos y las funciones de IAM que se van a gestionar 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 AWS región preferida en. AWS CLI Su proyecto se crea en la AWS región configurada en 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 a un archivo (por ejemplo,*`input.json`*) en una ubicación de su equipo local o instancia donde AWS CLI esté instalado. 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. El nombre descriptivo de este AWS CodeStar proyecto.
- *project-id*: obligatorio. El identificador de proyecto de este AWS CodeStar proyecto.

> **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. La descripción de este AWS CodeStar proyecto.
- *sourceCode*: opcional. Información de configuración para el código fuente proporcionado para el proyecto. Actualmente, solo se admite un único objeto `sourceCode`. Cada `sourceCode` objeto contiene información sobre la ubicación en la que se recupera el código fuente AWS CodeStar y el destino en el que se rellena el código fuente.
  - *source*: obligatorio. Define la ubicación donde se ha cargado el código fuente. La única fuente compatible es Amazon S3. AWS CodeStar recupera el código fuente y lo incluye en el repositorio una vez creado el proyecto.
    - *S3*: opcional. La ubicación de Amazon S3 del código fuente.
      - *bucket-name*: El depósito que contiene tu código fuente.
      - *bucket-key*: el prefijo del bucket y la clave de objeto que apuntan 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 CodeCommit repositorio que debe contener el código fuente. Este repositorio debe estar en la plantilla de la cadena de herramientas.

> **Note**
>
> Para CodeCommit ello, debe proporcionar el nombre del repositorio que definió en la pila de su cadena de herramientas. AWS CodeStar inicializa este repositorio con el código fuente que proporcionó en Amazon S3.

- *gitHub*: Este objeto representa la información necesaria para crear el GitHub repositorio e iniciarlo con el código fuente. Si elige un GitHub repositorio, se requieren los siguientes valores.

> **Note**
>
> Para GitHub, no puede especificar un GitHub repositorio existente. AWS CodeStar crea uno para usted y rellena este repositorio con el código fuente que cargó en Amazon S3. AWS CodeStar utiliza la siguiente información para crear su repositorio en GitHub.

  - *name*: obligatorio. El nombre de tu GitHub repositorio.
  - *description*: obligatorio. La descripción de tu GitHub repositorio.
  - *type*: obligatorio. El tipo de GitHub repositorio. Los valores válidos son User (usuario) u Organization (organización).
  - *owner*: obligatorio. El nombre de GitHub usuario del propietario del repositorio. Si el repositorio debe ser propiedad de una GitHub organización, proporciona 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 deseas habilitar las incidencias en GitHub este repositorio. Los valores válidos son true (verdadero) o false (falso).
  - *token*: opcional. Se trata de un token de acceso personal que se AWS CodeStar utiliza para acceder a tu GitHub cuenta. Este token deben contener los siguientes ámbitos: repo, user y admin:repo_hook. Para recuperar un token de acceso personal GitHub, consulte Creación de un token de acceso personal para la línea de comandos en el GitHub sitio web.

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

- *`toolchain`*: Información sobre la cadena de herramientas de CI/CD que se configurará cuando se cree 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 cualquier modificación de parámetros a la AWS CloudFormation que hacer referencia y la función que se utilizará para crear la pila. AWS CodeStar recupera la plantilla y la 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`*: el prefijo del bucket y la clave de objeto que apuntan al archivo.yml o .json que contiene la plantilla de la cadena de herramientas (por ejemplo,). `files/toolchain.yml`
  - *`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 usa el rol de servicio predeterminado creado para su cuenta si la cadena de herramientas es una plantilla de inicio rápido. 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 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. Las etiquetas adjuntas a su proyecto. 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,
```

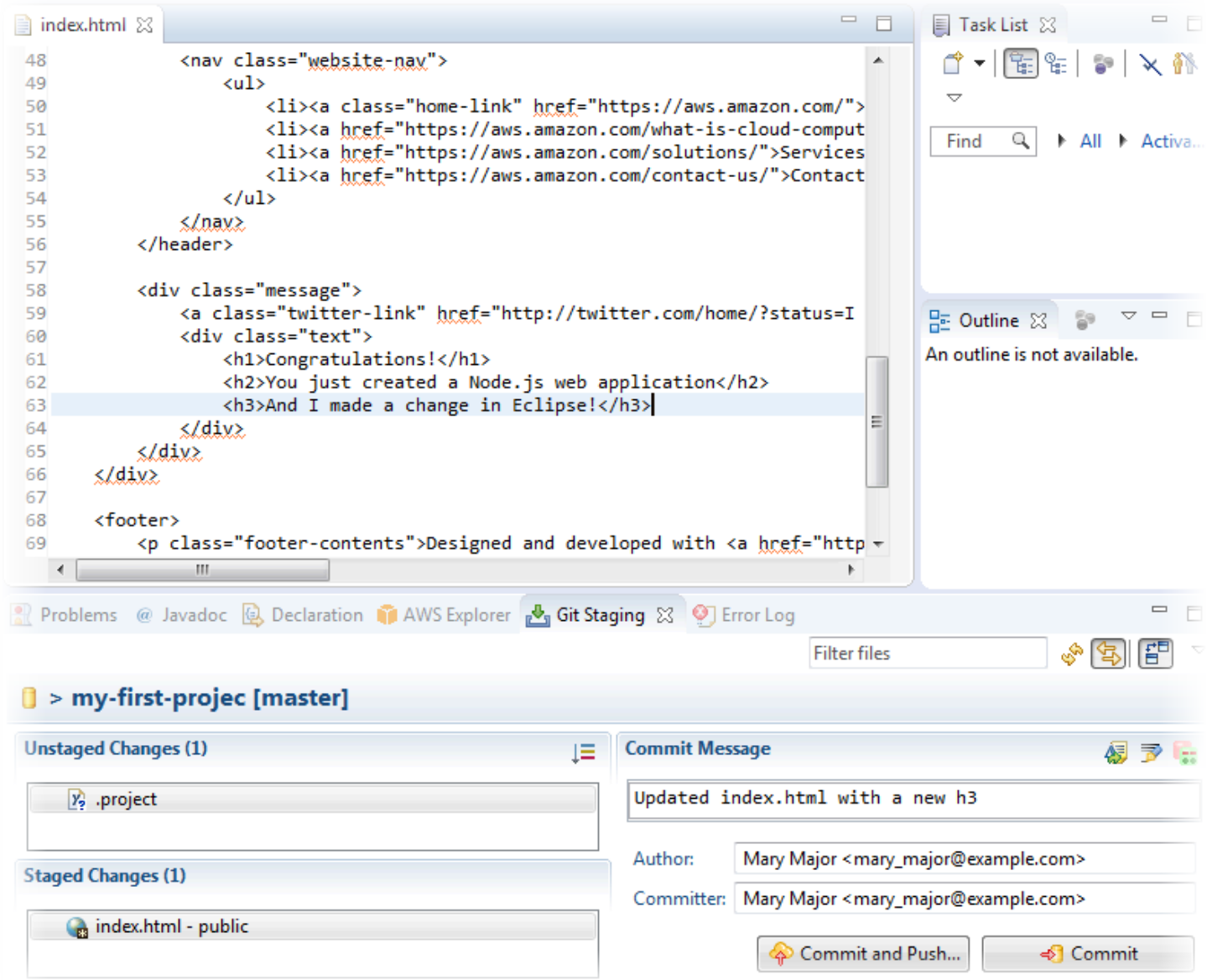
```
    "stackId": "arn:aws:cloudformation:us-east-1:account_ID:stack/awscodestar-
myproject/stack-ID",
    "status": {
        "state": "CreateInProgress"
    }
}
```

- 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.

Utilice un IDE con AWS CodeStar

Al integrar un IDE con él AWS CodeStar, puede seguir escribiendo y desarrollando código en el entorno que prefiera. Los cambios que realices se incluyen en el AWS CodeStar proyecto cada vez que confirmas e insertas el código.



Temas

- Úselo AWS Cloud9 con AWS CodeStar
- Usa Eclipse con AWS CodeStar
- Utilice Visual Studio con AWS CodeStar

Úselo AWS Cloud9 con AWS CodeStar

Se puede utilizar AWS Cloud9 para realizar cambios en el código y desarrollar software en un AWS CodeStar proyecto. AWS Cloud9 es un IDE en línea al que puede acceder a través de su 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 EC2 instancia de Amazon aloja un entorno de AWS Cloud9 desarrollo. Este entorno proporciona el AWS Cloud9 IDE y el acceso a los archivos de código del AWS CodeStar proyecto. Para obtener más información, consulte la Guía del usuario de AWS Cloud9.

Puede usar la AWS CodeStar consola o la AWS Cloud9 consola para crear entornos de AWS Cloud9 desarrollo para proyectos en los que se almacene su código CodeCommit. Para AWS CodeStar los proyectos que almacenan su código GitHub, solo puedes usar la AWS Cloud9 consola. En este tema se describe cómo utilizar ambas consolas.

Para usarla AWS Cloud9, necesitas:

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

Temas

Cree un AWS Cloud9 entorno para un proyecto

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

1. Siga los pasos que se indican en Creación de un proyecto si desea crear un proyecto nuevo.
2. Abre el proyecto en la AWS CodeStar consola. 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 AWS región que AWS Cloud9 no es compatible, no verás AWS Cloud9 las opciones en la pestaña IDE de la barra de navegación. Sin

embargo, puedes usar la AWS Cloud9 consola para crear un entorno de desarrollo, abrir el nuevo entorno y, después, conectarlo al AWS CodeCommit repositorio 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 ver la lista de AWS regiones compatibles, consulte AWS Cloud9la Referencia general de Amazon Web Services.

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

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

¿La cuenta tiene una VPC con al menos una subred en esa VPC?	¿La VPC que desea AWS Cloud9 usar es la VPC predeterminada de la cuenta?	¿La VPC tiene una única subred?	Haga lo siguiente
No	—	—	Si no existe una VPC, créela. Expanda 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 . 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

¿La cuenta tiene una VPC con al menos una subred en esa VPC?	¿La VPC que desea AWS Cloud9 usar es la VPC predeterminada de la cuenta?	¿La VPC tiene una única subred?	Haga lo siguiente
			sobre cómo crear una subred para AWS Cloud9 en la Guía del usuario de AWS Cloud9 .
Sí	Sí	Sí	Continúe con 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	Para Red (VPC), elija la VPC que desee usar. AWS Cloud9 En Subred, elige la subred que quieres AWS Cloud9 usar en esa VPC.

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

3. 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 período de tiempo predeterminado tras el cual se AWS Cloud9 apaga el entorno cuando no se ha utilizado, amplíe la configuración de ahorro de costes y, a continuación, cambie la configuración.
5. Seleccione Crear entorno.

Para abrir el entorno, consulte Abra un AWS Cloud9 entorno 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.

Abra un AWS Cloud9 entorno para un proyecto

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

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

   > **Important**
   >
   > Si el código fuente del proyecto está almacenado GitHub, IDE no aparecerá en la barra de navegación. Sin embargo, puede usar la AWS Cloud9 consola 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 Úselo GitHub con AWS Cloud9.

2. Para sus AWS Cloud9 entornos o AWS Cloud9 entornos compartidos, elija Open IDE para el entorno que desee abrir.

Puede utilizar el AWS Cloud9 IDE para empezar a trabajar con el código del AWS CodeCommit repositorio del proyecto de forma inmediata. 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 .

Comparta un AWS Cloud9 entorno con un miembro del equipo del proyecto

Después de crear un entorno de AWS Cloud9 desarrollo para un AWS CodeStar proyecto, puedes invitar a otros usuarios de tu AWS cuenta, incluidos los miembros del equipo del proyecto, a acceder 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 pueden usar el AWS Cloud9 IDE compartido para ver los cambios de código de cada miembro resaltados en el editor de código y para chatear por texto con otros miembros mientras programan.

Añadir a un miembro del equipo a un proyecto no le permite participar automáticamente en ningún entorno de AWS Cloud9 desarrollo relacionado con el proyecto. Para invitar a un miembro del equipo del proyecto a acceder al entorno de un proyecto, debe determinar el rol de acceso correcto del miembro del entorno, aplicar políticas AWS administradas al usuario e invitarlo 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 de entornos compartidos de la pestaña IDE de la AWS CodeStar consola 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 está almacenado en GitHub, el IDE no aparecerá en la barra de navegación. Sin embargo, puedes usar la AWS Cloud9 consola para invitar a otros usuarios de tu AWS cuenta, incluidos los miembros del equipo del proyecto, a acceder a un entorno. Para ello, consulte Úselo 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 AWS Cloud9 entorno de un proyecto

Al eliminar un proyecto y todos sus AWS recursos AWS CodeStar, también se eliminan todos los entornos de AWS Cloud9 desarrollo relacionados que se crearon con la AWS CodeStar consola 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 AWS CodeStar consola, en la barra de navegación, elija IDE.

   > **Important**
   >
   > Si el código fuente del proyecto está almacenado GitHub, IDE no aparecerá en la barra de navegación. Sin embargo, puedes usar la AWS Cloud9 consola 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.

Úselo GitHub con AWS Cloud9

En el caso de los AWS CodeStar proyectos que tienen su código fuente almacenado GitHub, la AWS CodeStar consola no permite trabajar directamente con entornos de AWS Cloud9 desarrollo. Sin embargo, puedes usar la AWS Cloud9 consola para trabajar con el código fuente de los GitHub repositorios.

1. Usa la AWS Cloud9 consola para crear un entorno de AWS Cloud9 desarrollo. Para obtener más información, consulte Creación de un entorno en la Guía del usuario de AWS Cloud9 .
2. Utilice la AWS Cloud9 consola para abrir el 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 GitHub repositorio (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 ver los comandos que se utilizan para clonar el GitHub repositorio, consulte Clonación de un repositorio en el sitio web de GitHub ayuda.

   Para ir a la página principal del GitHub repositorio, con el proyecto abierto en la AWS CodeStar consola, en la barra de navegación lateral, selecciona 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 Enviar a un repositorio remoto y Obtener un repositorio remoto en el GitHub sitio web de ayuda. Para ver los comandos de Git, consulta la hoja de trucos de Git en el sitio web de GitHub ayuda.

> **Note**
>
> Para evitar que Git te pida tus credenciales de GitHub inicio de sesión cada vez que insertes o extraigas código del repositorio, puedes usar un asistente de credenciales. Para obtener más información, consulta Cómo almacenar en caché tu GitHub contraseña en Git en el sitio web de GitHub ayuda.

Recursos adicionales

Para obtener más información sobre su uso AWS Cloud9, consulta lo siguiente en la Guía del AWS Cloud9 usuario:

- Tutorial
- Trabajo con entornos
- Uso del IDE
- Ejemplos

Usa Eclipse con AWS CodeStar

Puedes usar Eclipse para realizar cambios en el código y desarrollar software en un AWS CodeStar proyecto. Puedes editar el código AWS CodeStar del proyecto con Eclipse y, a continuación, confirmar e insertar los cambios en el repositorio fuente del AWS CodeStar proyecto.

> **Note**
>
> La información de este tema solo se aplica a AWS CodeStar los proyectos que almacenan su código fuente en él CodeCommit. Si tu AWS CodeStar proyecto almacena su código

fuente GitHub, puedes usar una herramienta como EGit la de Eclipse. Para obtener más información, consulta la EGit documentación del sitio EGit web.

Si el AWS CodeStar proyecto almacena su código fuente CodeCommit, debe instalar una versión del AWS Toolkit for Eclipse que sea compatible AWS CodeStar. También debes ser miembro del equipo del AWS CodeStar proyecto con el rol de propietario o colaborador.

Para utilizar Eclipse, también necesita:

- Un usuario de IAM que se ha añadido a un AWS CodeStar proyecto como miembro del equipo.
- Si el AWS CodeStar proyecto 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 el AWS Toolkit for Eclipse en tu ordenador local.

Temas

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 AWS CodeStar kit de herramientas se incluye como parte del Toolkit for Eclipse.

Para instalar el Toolkit for Eclipse con AWS CodeStar el módulo

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: Importa tu AWS CodeStar proyecto a Eclipse

Una vez instalado el Toolkit for Eclipse, puede AWS CodeStar importar proyectos y editar, confirmar y enviar código desde el IDE.

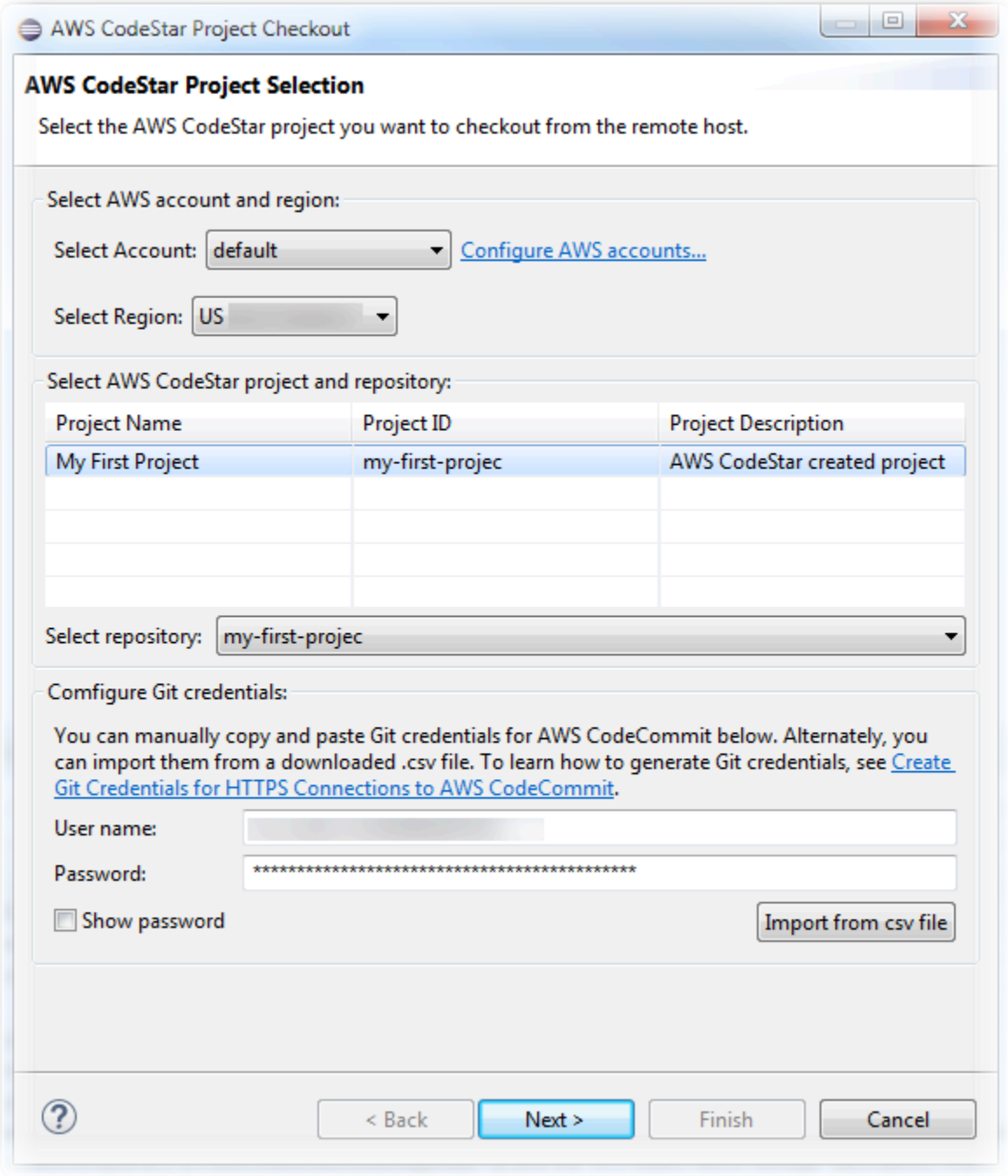
> **Note**
>
> Puedes añadir varios AWS CodeStar proyectos a un único espacio de trabajo en Eclipse, pero debes actualizar las credenciales del proyecto cuando cambies de un proyecto a otro.

Para importar un AWS CodeStar proyecto

1. En el AWS menú, selecciona Importar AWS CodeStar proyecto. También puede elegir File (Archivo) y luego elegir Import (Importar). En Seleccionar, expanda AWS y, a continuación, elija AWS CodeStar Proyecto.

   Elija Next (Siguiente).
2. En Selección de AWS CodeStar proyectos, elija su AWS perfil y la AWS región en la que se aloja el AWS CodeStar proyecto. Si no tiene un AWS perfil configurado con una clave de acceso y una clave secreta en su ordenador, elija Configurar AWS cuentas y siga las instrucciones.

   En Seleccionar AWS CodeStar proyecto y repositorio, elige tu AWS CodeStar proyecto. 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 Next (Siguiente).



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 el Explorador de proyectos, expande el árbol del proyecto para buscar los archivos del AWS CodeStar proyecto.

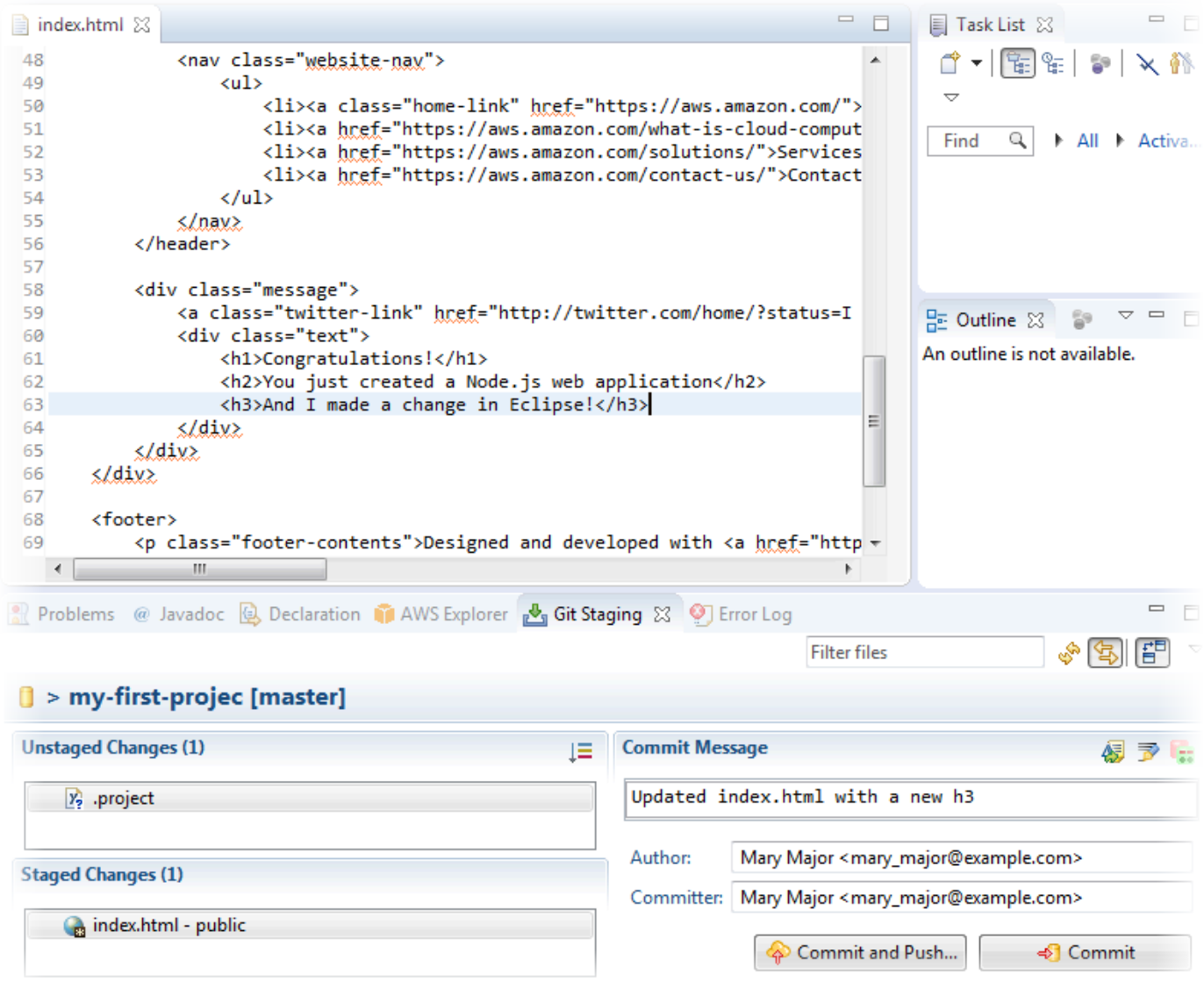
Paso 3: Edita el código AWS CodeStar del proyecto en Eclipse

Después de importar un AWS CodeStar proyecto a un espacio de trabajo de Eclipse, puede editar el código del proyecto, guardar los cambios y confirmar e insertar el código en el repositorio fuente del proyecto. Este es el mismo proceso que se sigue para cualquier repositorio de Git que utilice el EGit complemento para Eclipse. Para obtener más información, consulta la Guía del EGit usuario en el sitio web de Eclipse.

Para editar el código del proyecto y realizar tu primera confirmación con el repositorio fuente de un AWS CodeStar proyecto

1. En el Explorador de proyectos, expande el árbol del proyecto para buscar los archivos del AWS CodeStar proyecto.
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).



Para ver los cambios de código implementados, vuelva al panel de su proyecto. Para obtener más información, consulte Paso 3: ver el proyecto.

Utilice Visual Studio con AWS CodeStar

Puede usar Visual Studio para realizar cambios en el código y desarrollar software en un AWS CodeStar proyecto.

> **Note**
>
> Visual Studio para Mac no es compatible con el AWS kit de herramientas, por lo que no se puede utilizar con AWS CodeStarél.

La información de este tema se aplica únicamente a los AWS CodeStar proyectos en CodeCommit los que se almacena su código fuente. Si el AWS CodeStar proyecto almacena el código fuente GitHub, puede utilizar una herramienta como la GitHub extensión para Visual Studio. Para obtener más información, consulte la página de información general en el sitio web de la GitHub extensión para Visual Studio y la sección GitHub Introducción a Visual Studio en el GitHub sitio web.

Para usar Visual Studio para editar el código del repositorio de código fuente de un AWS CodeStar proyecto, debe instalar una versión del AWS Toolkit for Visual Studio que sea compatible 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 ha añadido a un AWS CodeStar proyecto como miembro del equipo.
- AWS las credenciales de su usuario de IAM (por ejemplo, la clave de acceso y la clave secreta).
- Permisos suficientes para instalar Visual Studio y el 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 Toolkit for Visual Studio con AWS CodeStar el módulo y configurar el acceso al repositorio de proyectos

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 Primeros pasos con la AWS Toolkit for Visual Studio página, introduzca o importe sus AWS credenciales y, a continuación, seleccione Guardar y cerrar.
3. En Visual Studio, abra Team Explorer. En Hosted Service Providers (Proveedores de servicios alojados), busque CodeCommit y elija Connect (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, estará listo para empezar a editar el código en Visual Studio y a registrar los cambios e insertarlos en CodeCommit el repositorio del proyecto.

Cambiar AWS los recursos de un AWS CodeStar proyecto

Tras crear un proyecto en AWS CodeStar, puede cambiar el conjunto predeterminado de AWS recursos que se AWS CodeStar añaden al proyecto.

Cambios de recursos admitidos

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

Cambio	Notas
Añada una etapa a AWS CodePipeline.	Consulte Añadir un escenario a AWS CodePipeline.
Cambiar la configuración del entorno de Elastic Beanstalk.	Consulte Cambiar la configuración del AWS Elastic Beanstalk entorno.
Cambie el código o la configuración de una AWS Lambda función, su función de IAM o su API en Amazon API Gateway.	Consulte Cambiar una AWS Lambda función en el código fuente.
Añada un recurso a un AWS Lambda proyecto y amplíe los permisos para crear el nuevo recurso y acceder a él.	Consulte Añadir un recurso a un proyecto.
Añada el cambio de tráfico con CodeDeploy para una AWS Lambda función.	Consulte Desviar el tráfico para un proyecto de AWS Lambda.

Cambio	Notas
Añadir AWS X-Ray soporte	Consulte Habilitar el seguimiento para un proyecto.
Edita el archivo buildspec.yml de tu proyecto para añadir una fase de compilación de pruebas unitarias para su ejecución. AWS CodeBuild	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 las funciones definidas en la cadena de herramientas ni en las pilas. AWS CloudFormation
Modifique su proyecto de Lambda para añadir un punto de conexión.	
Modifique su EC2 proyecto para añadir un punto final.	
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 los parámetros SSM de forma segura en un AWS CodeStar proyecto.	Consulte the section called “Utilice de forma segura los parámetros de SSM en un proyecto AWS CodeStar”.

No se admiten los cambios siguientes.

- Cambie a un objetivo de despliegue diferente (por ejemplo, despliegue AWS Elastic Beanstalk en lugar de AWS CodeDeploy).
- Añadir un nombre de punto de conexión web sencillo.

- Cambie el nombre del CodeCommit repositorio (para un AWS CodeStar proyecto al que esté conectado CodeCommit).
- En el caso de un AWS CodeStar proyecto conectado a GitHub, desconecte el GitHub repositorio y, a continuación, vuelva a conectar el repositorio a ese proyecto o conecte cualquier otro repositorio a ese proyecto. Puedes usar la CodePipeline consola (no la AWS CodeStar consola) para desconectarte y volver a conectarte a ella GitHub en la etapa Source de una canalización. Sin embargo, si vuelves a conectar la etapa Source a un GitHub repositorio diferente, en el AWS CodeStar panel de control del proyecto, es posible que la información de los mosaicos Repositorio e Issues sea incorrecta o esté desactualizada. Al desconectar el GitHub repositorio, no se elimina la información del repositorio del historial de confirmaciones y se generan GitHub mosaicos en el panel del AWS CodeStar proyecto. Para eliminar esta información, usa el GitHub sitio web para deshabilitar el acceso GitHub desde el AWS CodeStar proyecto. Para revocar el acceso, en el GitHub sitio web, utiliza la sección OAuth Aplicaciones autorizadas de la página de configuración del perfil de tu GitHub cuenta.
- Desconecte el CodeCommit repositorio (en el caso de un AWS CodeStar proyecto al que esté conectado CodeCommit) y, a continuación, vuelva a conectar el repositorio a ese proyecto o conecte cualquier otro repositorio a ese proyecto.

Añadir un escenario a AWS CodePipeline

Puedes añadir una nueva etapa a una canalización que se AWS CodeStar cree 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 AWS recurso que AWS CodeStar no se haya creado, es posible que la canalización se interrumpa. Esto se debe a que es posible que la función de IAM que se AWS CodeStar creó no AWS CodePipeline tenga acceso a esos recursos de forma predeterminada.
>
> Para intentar dar AWS CodePipeline acceso a AWS los recursos que AWS CodeStar no se crearon, es posible que desee cambiar el rol de IAM que AWS CodeStar los creó. Esto no se admite porque AWS CodeStar podría eliminar los cambios en el rol de IAM al realizar comprobaciones de actualización periódicas en el proyecto.

Cambiar la configuración del AWS Elastic Beanstalk entorno

Puede cambiar la configuración de un AWS CodeStar entorno de Elastic Beanstalk que se crea en un proyecto. Por ejemplo, es posible que desee cambiar el entorno de Elastic Beanstalk predeterminado de su proyecto de instancia AWS CodeStar única a carga equilibrada. 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 impulsar el cambio de plantilla AWS CodeStar y AWS CloudFormation aprovisionar 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 AWS Lambda función en el código fuente

Puede cambiar el código o la configuración de una función de Lambda, o su rol de IAM o API API Gateway, que se AWS CodeStar crea en un proyecto. Para ello, le recomendamos que utilice el modelo de aplicaciones AWS sin servidor (AWS SAM) junto con el `template.yaml` archivo del repositorio de su proyecto. CodeCommit 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 Cómo crear aplicaciones sin servidor mediante AWS SAM en el GitHub sitio web.

Habilitar el seguimiento para un proyecto

AWS X-Ray ofrece rastreo, que puede utilizar para analizar el comportamiento del rendimiento de las aplicaciones distribuidas (por ejemplo, las latencias en los tiempos de respuesta). Tras añadir los seguimientos al AWS CodeStar proyecto, puede utilizar la AWS X-Ray consola para ver las vistas de las aplicaciones y los 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 los proyectos de Amazon EC2 o Elastic Beanstalk creados después del 3 de agosto de AWS CodeStar 2018, `/template.yml` aprovisionó un archivo en el repositorio del proyecto.

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

Puede modificar este archivo para añadir el seguimiento añadiendo el AWS X-Ray recurso a la sección. `Resources` A continuación, modifique los permisos de IAM de su proyecto para AWS CloudFormation permitir la creación del 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 aprovisionado con una política de límite de permisos.
> En el caso de los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar dotó al proyecto de una política de límites de permisos.

1. Inicie sesión en AWS Management Console y abra la AWS CodeStar consola 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 Recursos del proyecto, busque el rol de IAM creado para el rol 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 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 en AWS CodeStar . 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

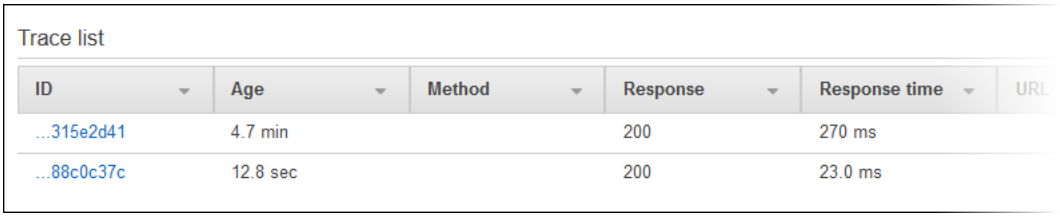
- Confirme y envíe los cambios realizados en el archivo `template.yml`.

> **Note**
>
> Esto iniciará la canalización. Si realizas los cambios antes de actualizar los permisos de IAM, la canalización se iniciará, la actualización de la AWS CloudFormation pila detectará 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 AWS CloudFormation pila comienza cuando la canalización de tu proyecto comienza la etapa de implementación. Para ver el estado de la actualización de la pila, en tu AWS CodeStar panel de control, selecciona la AWS CloudFormation etapa de la canalización.

   Si la actualización de la pila AWS CloudFormation arroja errores, consulta las pautas de solución de problemas enAWS 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 .



Añadir un recurso a un proyecto

Cada AWS CodeStar plantilla de todos los proyectos incluye un AWS CloudFormation archivo que modela las dependencias en AWS tiempo de ejecución de la aplicación, como las tablas de bases de datos y las funciones 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 los proyectos de Amazon EC2 o Elastic Beanstalk creados después del 3 de agosto de AWS CodeStar 2018, `/template.yml` aprovisionó un archivo en el repositorio del proyecto.

Puede modificar este archivo añadiendo AWS CloudFormation recursos a la sección. `Resources` La modificación del `template.yml` archivo permite AWS CodeStar AWS CloudFormation añadir el nuevo recurso a su proyecto. Algunos recursos requieren que añadas otros permisos a la política para el rol de CloudFormation trabajador de tu 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 ver una lista de AWS CloudFormation los recursos y sus propiedades obligatorias, consulta la Referencia AWS de tipos de recursos.

1. Paso 1: edite el rol del CloudFormation trabajador 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 de esta sección para modificar la plantilla AWS CodeStar del proyecto a fin de añadir un recurso y, a continuación, ampliar los permisos del rol de CloudFormation trabajador del proyecto

en IAM. En este ejemplo, el AWS::SQS::Queuerecurso se añade al `template.yml` archivo. El cambio inicia una respuesta automática AWS CloudFormation que añade una cola de Amazon Simple Queue Service a su proyecto.

Paso 1: edite el rol del CloudFormation trabajador 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 aprovisionado con una política de límite de permisos.
> En el caso de los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar dota a tu proyecto de una política de límites de permisos.

1. Inicie sesión en la AWS CodeStar consola AWS Management Console y ábrala 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 Recursos del proyecto, busque el rol de IAM creado para el AWS CloudFormation rol CodeStarWorker/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, yelija 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 AWS CodeStar consola 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 realizas los cambios antes de actualizar los permisos de IAM, la canalización se iniciará y la actualización de la AWS CloudFormation pila detectará errores, lo que provocará que se anule la actualización de la pila. Si esto ocurre, corrija los permisos y, a continuación, reinicie la canalización.

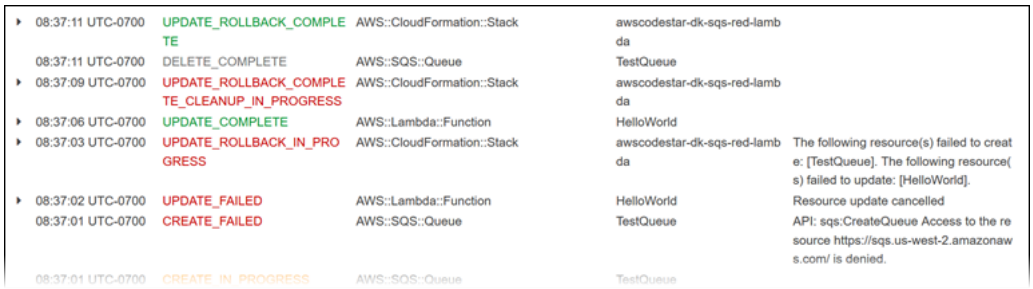
Paso 4: Monitorizar la actualización de la pila de AWS CloudFormation

1. Cuando la canalización de tu proyecto comience la fase de implementación, se iniciará la actualización de la AWS CloudFormation pila. Puedes elegir la AWS CloudFormation etapa de tu canalización en tu AWS CodeStar panel de control para ver la actualización del stack.

   Solución de problemas:

   La actualización de la pila falla si faltan los permisos a nivel de recursos necesarios. Consulta el estado del fallo en la vista del AWS CodeStar panel de control de la cartera de tu proyecto.

Selecciona el CloudFormationenlace en la fase de implementación de tu proceso para solucionar el error en la AWS CloudFormation consola. 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.

Añade los permisos que falten editando la política asociada al rol de AWS CloudFormation trabajador de tu proyecto. Consulte Paso 1: edite el rol del CloudFormation trabajador en IAM.

2. Después de ejecutar correctamente la canalización, los recursos se crean en la pila de AWS CloudFormation . En la lista de recursos de AWS CloudFormation, consulta el recurso creado para tu proyecto. En este ejemplo, la TestQueue cola aparece en la sección 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.

Utilización del editor de política de JSON para la creación de una política

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

   Si es la primera vez que elige Políticas, aparecerá la página Welcome to Managed Policies (Bienvenido a políticas administradas). Elija 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 Next (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, introduzca 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 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:
>
> - En el caso de los proyectos creados después del 3 de agosto de 2018, AWS CodeStar aprovisionó su proyecto de Amazon EC2, Elastic Beanstalk o Lambda `/template.yml` con un archivo en el repositorio del proyecto.
> - En el caso de los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar aprovisionó su proyecto con una política de límites de permisos.

Todos los AWS CodeStar proyectos utilizan un archivo de AWS CloudFormation plantilla que modela las dependencias del AWS tiempo de ejecución de la aplicación, como las instancias de Linux y las funciones 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

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

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

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

Solo para proyectos de Elastic Beanstalk EC2 o Amazon:

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

Temas

Paso 1: Crear un nuevo grupo de implementación en CodeDeploy (solo Amazon EC2 Projects)

Elija su CodeDeploy aplicación y, a continuación, añada un nuevo grupo de despliegue asociado a la nueva instancia.

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

1. Abra la CodeDeploy consola en https://console.aws.amazon.com/codedeploy.
2. Elija la CodeDeploy aplicación 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 Nombre de grupo de implementación, escriba ***<project-id>-prod-Env***.
5. En Función de servicio, elija la función de trabajador de la cadena de herramientas para su AWS CodeStar proyecto.
6. En Deployment type (Tipo de implementación), elija In-place (In situ).
7. En Configuración del entorno, seleccione la pestaña Amazon EC2 Instances.
8. En el grupo de etiquetas, en Key (Clave), elija `aws:cloudformation:stack-name`. En Valor, elija `awscodestar-<projectid>-infrastructure-prod` (la pila que se va a crear para la GenerateChangeSetacción).
9. En Deployment settings (Configuración de implementación), elija `CodeDeployDefault.AllAtOnce`.
10. Borre Choose a load balancer (Elegir un balanceador de carga).
11. Elija Crear grupo de implementación.

    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 de producción de un EC2 proyecto de Amazon debe tener las mismas acciones que la etapa de implementación creada para el proyecto.

Para copiar parámetros y campos desde la etapa de implementación

1. En el panel de control AWS CodeStar del proyecto, selecciona Detalles de la canalización para abrir la canalización en la CodePipeline consola.

2. Elija Editar.
3. En la etapa de implementación, elija Editar etapa.
4. Selecciona el icono de edición de la GenerateChangeSetacción. 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"
   }
   ```

   - EC2 Proyectos de Amazon:

   ```
   {

     "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 GenerateChangeSet acción en tu nueva etapa de producción

> **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 tu canalización aún no aparece, en el panel de control del AWS CodeStar proyecto, selecciona Detalles de la 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, seleccione AWS CloudFormation.
7. En Action mode (Modo acción), elija Create or replace a change set (Crear o reemplazar un conjunto de cambios).
8. En Nombre de pila, introduce un nombre nuevo para la AWS CloudFormation pila que se va a crear mediante 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`

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

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 `Stage` parámetro, que se muestra aquí en formato JSON, para un EC2 proyecto de Amazon:

```
{

  "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 AWS CodePipeline panel, selecciona Guardar cambio de canalización y, a continuación, selecciona 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 ExecuteChangeSet acción en tu nueva etapa de Prod

1. Si aún no estás viendo tu canalización, en el panel de control del AWS CodeStar proyecto, selecciona Detalles de la canalización para abrir la canalización en la consola.
2. Elija Editar.
3. En la nueva etapa de Prod, después de la nueva GenerateChangeSetacción, selecciona + Añadir grupo de acciones.
4. En Nombre de la acción, escriba un nombre (por ejemplo, **`ExecuteChangeSet`**).
5. En Proveedor de acción, seleccione AWS CloudFormation.
6. En Action mode (Modo de acción), elija Execute a change set (Ejecutar un conjunto de cambios).
7. En Nombre de pila, introduce el nuevo nombre de la AWS CloudFormation pila que has introducido en la GenerateChangeSet acción (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 AWS CodePipeline panel, selecciona Guardar cambio de canalización y, a continuación, selecciona 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 CodeDeploy implementación en tu nueva etapa de producción (solo EC2 proyectos de Amazon)

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, seleccione AWS CodeDeploy.
4. En Nombre de la aplicación, elija el nombre de la CodeDeploy aplicación para su proyecto.
5. En Deployment group (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 AWS CodePipeline panel, selecciona Guardar cambio de canalización y, a continuación, selecciona 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 AWS CodePipeline panel, selecciona Guardar cambio de canalización y, a continuación, selecciona Guardar cambio. Consulte la canalización actualizada.
9. Para enviar los cambios y comenzar una compilación de canalización, seleccione Publicar modificación y, a continuación, Publicar.

Paso 4: Introduce un cambio y supervisa la actualización de la AWS CloudFormation pila

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 inicia la etapa de implementación, comienza la actualización de la AWS CloudFormation pila. Puedes elegir la AWS CloudFormation etapa de tu canalización en tu AWS CodeStar panel de control 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. Tras completar correctamente tu canalización, los recursos se crean en tu AWS CloudFormation pila. En la AWS CloudFormation consola, elige la pila de infraestructura para tu 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 de recursos de la AWS CloudFormation consola, consulte el recurso creado para su proyecto. En este ejemplo, la nueva EC2 instancia de Amazon 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 AWS CloudFormation en la consola y expanda 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 AWS CloudFormation consola y expanda 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 EC2 proyecto de Amazon, elige la nueva EC2 instancia de Amazon en la lista de recursos del proyecto en la AWS CodeStar consola. El enlace se abre en la página de instancias de la EC2 consola de Amazon. Selecciona la pestaña Descripción, copia la URL en el DNS público (IPv4) y abre 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 la implementación de versiones de AWS Lambda funciones para las funciones de sus proyectos AWS CodeStar sin servidor. Una AWS Lambda implementación cambia el tráfico entrante de una función Lambda existente a una versión actualizada de la función Lambda. Le recomendamos que prueba 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 de esta sección para modificar la plantilla AWS CodeStar del proyecto y actualizar los permisos de IAM de sus CodeStarWorker funciones. Esta tarea inicia una respuesta automática AWS CloudFormation que crea AWS Lambda funciones con alias y, a continuación, indica que se traslade el tráfico AWS CodeDeploy a un entorno actualizado.

> **Note**
>
> Complete estos pasos solo si creó su CodeStar proyecto de AWS antes del 12 de diciembre de 2018.

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

- Valor controlado: 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.
- Rll-at-once: Todo el tráfico pasa de la función Lambda original a la versión actualizada de la función Lambda de una sola vez.

Tipo de preferencia de implementación
Canary10Percent30Minutes
Canary10Percent5Minutes
Canary10Percent10Minutes
Canary10Percent15Minutes
Lineal: 10 (10 minutos) PercentEvery
Lineal PercentEvery 10:1 minuto
Lineal: 10PercentEvery, 2 minutos
Lineal: 10PercentEvery, 3 minutos
AllAtOnce

Para obtener más información sobre AWS CodeDeploy las implementaciones en una plataforma AWS Lambda informática, consulte Implementaciones en una plataforma informática AWS Lambda.

Para obtener más información sobre AWS SAM, consulte el Modelo de aplicaciones AWS sin servidor (AWSSAM) en. GitHub

Requisitos previos:

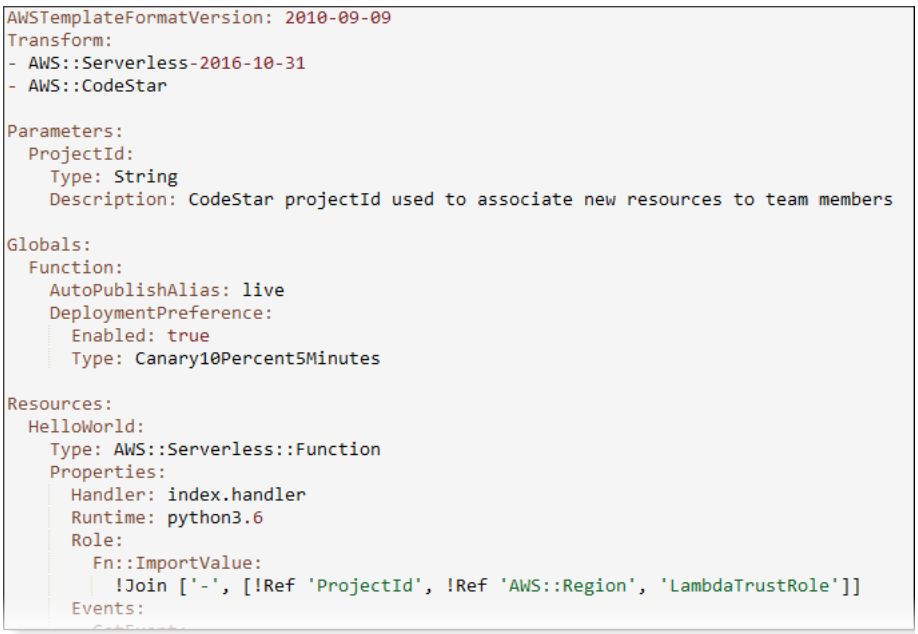
Al crear un proyecto sin servidor, seleccione cualquier plantilla con la plataforma de computación Lambda. Debe haber iniciado sesión como administrador para llevar a cabo los pasos 4 a 6.

Paso 1: Modifique la plantilla SAM para añadir los parámetros de despliegue de AWS Lambda la versión

1. Abra la AWS CodeStar consola 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: edita el AWS CloudFormation rol para añadir permisos

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

   > **Note**
   >
   > Debe iniciar sesión con las AWS Management Console credenciales asociadas al usuario de IAM que creó o con el que se identificó. Configuración AWS CodeStar Este usuario debe tener el nombre de política AWS gestionada **`AWSCodeStarFullAccess`**adjunto.

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 AWS CloudFormation rol CodeStarWorker/. El rol se abrirá en la consola de IAM.
4. En la pestaña Permissions, en Inline Policies, 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 *`id`* marcadores *`region`* y 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"
},
{
  "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: Confirma y presiona el cambio de plantilla para iniciar el cambio de AWS Lambda versión

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 confirmas los cambios antes de actualizar los permisos de IAM, la canalización se iniciará y la actualización de la AWS CloudFormation pila detectará errores que anularán la actualización de la pila. Si esto ocurre, corrija los permisos y, a continuación, reinicie la canalización.

2. La actualización de la AWS CloudFormation pila comienza cuando la canalización de tu proyecto comienza la fase de implementación. Para ver la notificación de actualización de la pila cuando comience la implementación, en tu AWS CodeStar panel de control, selecciona la AWS CloudFormation etapa de tu canalización.

   Durante la actualización de la pila, actualiza AWS CloudFormation automáticamente los recursos del proyecto de la siguiente manera:

   - AWS CloudFormation procesa el `template.yml` archivo mediante la creación de funciones Lambda con alias, enlaces de eventos y recursos.
   - AWS CloudFormation llama a Lambda para crear la nueva versión de la función.
   - AWS CloudFormation crea un AppSpec archivo y hace una llamada AWS CodeDeploy para cambiar 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 sobre los enlaces de eventos y los recursos del AWS CodeDeploy AppSpec archivo, consulte la sección AppSpec «recursos» (solo implementaciones de AWS Lambda) y la sección AppSpec «ganchos» 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 del proyecto, en la lista de recursos del proyecto, consulte los recursos de la AWS CodeDeploy aplicación, el grupo de AWS CodeDeploy implementación y la función de AWS CodeDeploy servicio 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 AWS CodeDeploy implementación.
6. En su directorio de trabajo local, puede realizar cambios en su AWS Lambda función y archivar el cambio en el repositorio de su proyecto. AWS CloudFormation permite gestionar AWS CodeDeploy la próxima revisión de la misma manera. Para obtener más información sobre cómo volver a implementar, detener o revertir una implementación de Lambda, consulte Implementaciones en una AWS plataforma informática Lambda.

Haga la transición de su CodeStar proyecto de AWS a producción

Una vez que haya creado la aplicación con un CodeStar proyecto de AWS y haya visto lo que CodeStar ofrece AWS, es posible que desee hacer la transición de su proyecto a un uso de producción. Una forma de hacerlo es replicar los AWS recursos de la aplicación fuera de AWS CodeStar. Seguirá necesitando un repositorio, un proyecto de compilación, una canalización y una implementación, pero en lugar de dejar que AWS los CodeStar cree por usted, los recreará utilizando AWS CloudFormation.

> **Note**
>
> Puede resultar útil crear o ver un proyecto similar utilizando primero uno de los inicios CodeStar rápidos de AWS y usarlo como plantilla para su propio proyecto a fin de asegurarse de incluir los recursos y las políticas que necesita.

Un CodeStar proyecto de AWS es una combinación del código fuente y 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. En el momento de la creación del

proyecto, una AWS CloudFormation plantilla aprovisiona los recursos de su cadena de herramientas (de forma continuaintegration/continuous deployment (CI/CD).

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

Una cadena de herramientas completa requiere los siguientes recursos recomendados:

1. Un GitHub repositorio CodeCommit o repositorio que contiene su código fuente.
2. Una CodePipeline canalización que está configurada para escuchar los cambios en tu repositorio.
   a. Cuando utilice AWS CodeBuild para ejecutar pruebas unitarias o de integración, le recomendamos que añada una etapa de compilación a su proceso para crear artefactos de compilación.
   b. Le recomendamos que añada una etapa de despliegue a la canalización que utilice CodeDeploy o AWS CloudFormation despliegue el artefacto de compilación y el código fuente en su infraestructura de tiempo de ejecución.

> **Note**
>
> Como CodePipeline requiere al menos dos etapas en una canalización y la primera debe ser la etapa de origen, agrega una etapa de compilación o implementación como segunda etapa.

Temas

Cree un GitHub repositorio

Para crear un GitHub repositorio, debe definirlo en la plantilla de su 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 ya un token de acceso personal GitHub para AWS poder conectarse GitHub en su nombre. Además del token de acceso personal GitHub, también debes tener `s3.GetObject` permiso para usar el `Code` objeto que ingreses.

Para especificar un GitHub repositorio público, añada un código como el siguiente a la plantilla de su cadena de herramientas. 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 las incidencias en el GitHub repositorio.
- Si el GitHub repositorio es privado.
- El token de acceso GitHub personal que creaste.
- Descripción, nombre y propietario del repositorio que va a crear.

Para obtener detalles completos sobre la información que se debe especificar, consulte el AWS::CodeStar::GitHubRepositorio en la Guía del AWS CloudFormation usuario.

Trabajar con etiquetas de proyecto 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 AWS CodeStar consola, en el panel de navegación lateral, selecciona 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 AWS CodeStar consola, en el panel de navegación lateral, selecciona 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 el comando AWS CLI para ejecutar el AWS CodeStar list-tags-for-project comando, 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 AWS CodeStar proyecto

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. Sus funciones de proyecto se eliminan de sus usuarios de IAM, pero sus perfiles de usuario no AWS CodeStar se modifican. Puede utilizar la AWS CodeStar consola o AWS CLI eliminar un proyecto. La eliminación de un proyecto requiere el rol de AWS CodeStar servicio`aws-codestar-service-role`, que no debe modificarse y ser asumible por él. AWS CodeStar

Important

La eliminación de un proyecto AWS CodeStar no se puede deshacer. De forma predeterminada, todos AWS los recursos del proyecto se eliminan de tu AWS cuenta, incluidos:

- El CodeCommit repositorio del proyecto junto con todo lo almacenado en ese repositorio.
- Las funciones AWS CodeStar del proyecto y las políticas de IAM asociadas configuradas para el proyecto y sus recursos.
- Cualquier EC2 instancia de Amazon creada para el proyecto.
- La aplicación de implementación y los recursos asociados, como por ejemplo:
  - Una CodeDeploy aplicación y los grupos de despliegue asociados.
  - Una AWS Lambda función y API Gateway asociada APIs.
  - Una AWS Elastic Beanstalk aplicación y un entorno asociado.
- La canalización de despliegue continuo del proyecto en CodePipeline.
- Las AWS CloudFormation pilas asociadas al proyecto.
- Cualquier entorno AWS Cloud9 de desarrollo creado con la AWS CodeStar consola. 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 desactiva esta opción, el proyecto se elimina en AWS CodeStar IAM y las funciones del proyecto que permitían el acceso a esos recursos se eliminan de IAM, pero se conservan todos los demás recursos. Es posible que siga incurriendo en cargos por estos 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 proyecto: se ha eliminado un AWS CodeStar 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)

Puedes usar la AWS CodeStar consola para eliminar un proyecto.

Para eliminar un proyecto en AWS CodeStar

1. Abre la AWS CodeStar consola 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 de la AWS CodeStar consola.

   Important

   Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos recursos no se eliminan, aunque selecciones la casilla de verificación.
   Tu proyecto no se puede eliminar si alguna política AWS CodeStar gestionada se ha asociado manualmente a funciones 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 usar el 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
```

Trabajando con AWS CodeStar equipos

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 AWS CodeStar proyectos y tener diferentes AWS CodeStar roles (y, por lo tanto, diferentes permisos) en cada uno de ellos. En la AWS CodeStar consola, los usuarios ven todos los proyectos asociados a tu AWS cuenta, pero solo pueden ver los proyectos en los que forman parte del equipo y trabajar en ellos.

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 AWS CodeStar tiene tres funciones:

Funciones y permisos en un AWS CodeStar proyecto

Nombre de función	Ver el panel y el estado del proyecto	Add/Remove/ AccessRecursos del proyecto	Añadir/Eliminar miembros del equipo	Eliminar proyecto
Propietario	x	x	x	x
Colaborador	x	x		
Lector	x			

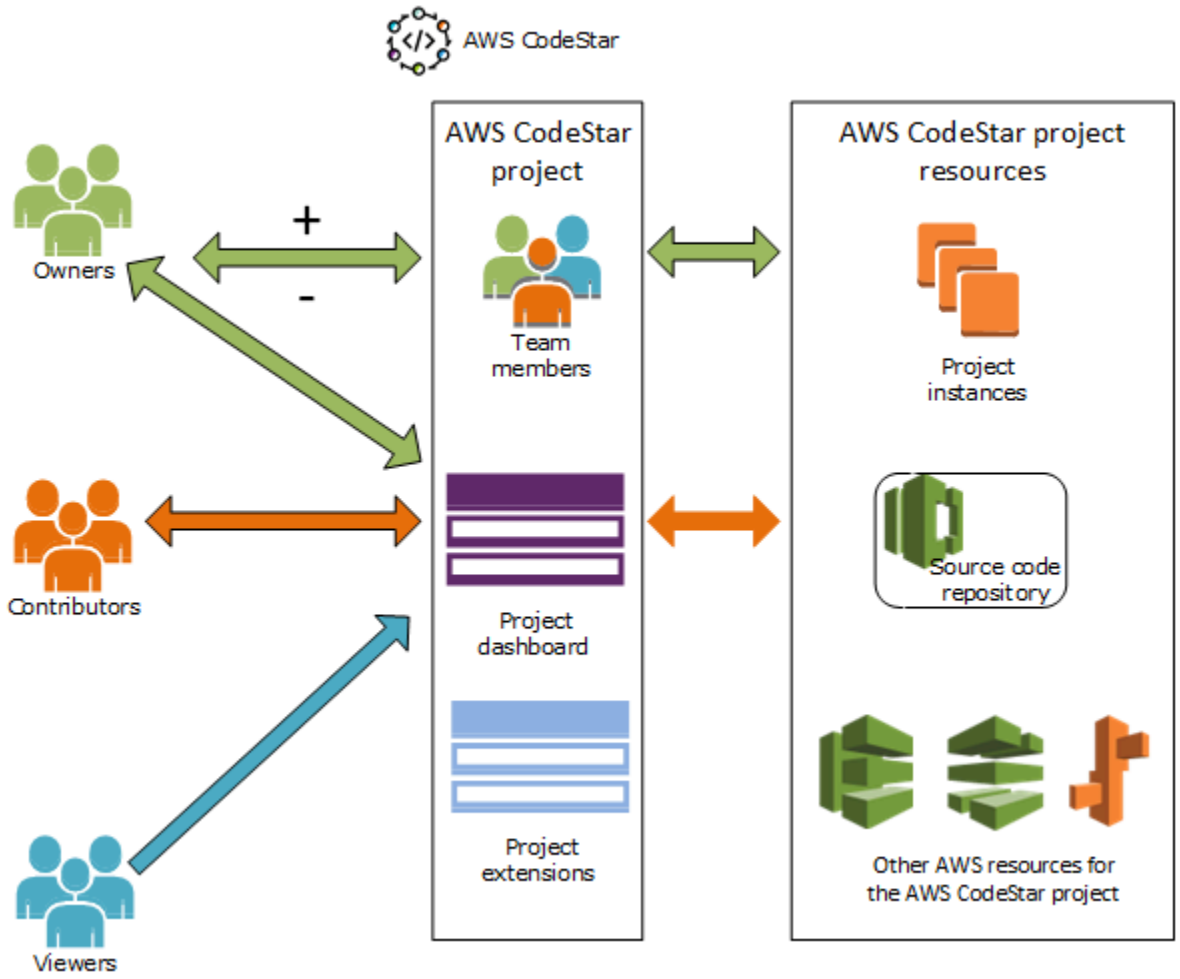
- Propietario: puede añadir y eliminar a otros miembros del equipo, contribuir con código a un repositorio del proyecto si el código está almacenado en él CodeCommit, conceder o denegar a otros miembros del equipo el acceso remoto a cualquier EC2 instancia de Amazon que ejecute Linux asociada al proyecto, configurar el panel del proyecto y eliminar el proyecto.
- Colaborador: puede añadir y eliminar recursos del panel, como un mosaico de JIRA, contribuir con código al repositorio del proyecto si el código está almacenado e interactuar plenamente con el panel. CodeCommit 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.

- Visor: puede ver el panel del proyecto, el código en el que está almacenado y CodeCommit, en los mosaicos del panel, el estado del proyecto y sus recursos.

> **Important**
>
> Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), el acceso a esos 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 AWS CodeStar proyecto puede utilizar la AWS CodeStar consola para acceder a los recursos ajenos al proyecto AWS pero relacionados con él.
>
> AWS CodeStar no permite que los miembros del equipo del proyecto participen automáticamente en ningún entorno de AWS Cloud9 desarrollo relacionado con un proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte Comparta un AWS Cloud9 entorno 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 Ejemplos de políticas CodeStar basadas en la identidad de AWS.

En el siguiente diagrama se muestra la relación entre cada rol y un proyecto de AWS CodeStar .



Temas

- Añadir miembros del equipo a un AWS CodeStar proyecto
- Administrar los permisos de los miembros AWS CodeStar del equipo
- Eliminar miembros del equipo de un AWS CodeStar proyecto

Añadir miembros del equipo a un AWS CodeStar proyecto

Si tienes el rol de propietario en un AWS CodeStar proyecto o tienes la `AWSCodeStarFullAccess` política aplicada a tu usuario de IAM, puedes añadir otros usuarios de IAM al equipo del proyecto. Se trata de un proceso sencillo que asigna un AWS CodeStar rol (propietario, colaborador o espectador) 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.

Los beneficios de AWS CodeStar los roles y la pertenencia al equipo 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 AWS CodeStar consola solo si son miembros del equipo.
- El acceso del usuario a un proyecto viene definido por el rol.

Para obtener más información sobre los equipos y AWS CodeStar las funciones, consulte Trabajando con AWS CodeStar equipos yCómo trabajar con su perfil AWS CodeStar de usuario .

Para añadir un miembro del equipo a un proyecto, debe tener el rol de AWS CodeStar propietario del proyecto o la `AWSCodeStarFullAccess` política.

> **Important**
>
> Añadir a un miembro del equipo no afecta al acceso de ese miembro a los recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA). Esos 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 AWS CodeStar proyecto puede usar la AWS CodeStar consola para acceder a recursos ajenos a ese proyecto AWS pero relacionados con él.
>
> Añadir a un miembro del equipo a un proyecto no le permite participar automáticamente en ningún entorno de AWS Cloud9 desarrollo relacionado con el proyecto. Para permitir a un miembro del equipo participar en un entorno compartido, consulte Comparta un AWS Cloud9 entorno 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)

Puedes usar la AWS CodeStar consola para añadir un miembro del equipo a tu 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 AWS CodeStar proyecto (consola)

1. Abre la AWS CodeStar consola 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 han agregado a otro AWS CodeStar proyecto aparecen en la lista de AWS CodeStar usuarios existentes.

     En el rol del proyecto, elija el AWS CodeStar rol (propietario, colaborador o espectador) para 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 acceder a los recursos AWS CodeStar del proyecto. Aplica las políticas necesarias para crear y administrar las credenciales de Git para el código almacenado CodeCommit en IAM o para cargar las claves EC2 SSH de Amazon 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 Administre la información de visualización de su perfil de AWS CodeStar usuario .

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 el usuario de IAM, vuelve a la AWS CodeStar consola, actualiza la lista de usuarios y elige el usuario de IAM que creaste en la lista desplegable. Introduce el nombre AWS CodeStar para mostrar, la dirección de correo electrónico y el rol del proyecto que deseas aplicar a este nuevo usuario y, a continuación, selecciona Añadir 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 tu AWS CodeStar proyecto.
   - Si el código fuente está almacenado en CodeCommit, instrucciones para configurar el acceso con las credenciales de Git al CodeCommit repositorio desde sus ordenadores locales.
   - Información sobre cómo el usuario puede gestionar su nombre visible, dirección de correo electrónico y clave EC2 SSH pública de Amazon, tal y como se describe enCómo trabajar con su perfil AWS CodeStar de usuario .
   - 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 utilizarla 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 los permisos de los miembros AWS CodeStar del equipo

Los permisos de los miembros del equipo se modifican cambiando su AWS CodeStar rol. A cada miembro del equipo se le puede asignar solo un rol en un AWS CodeStar proyecto, pero se pueden asignar muchos usuarios al mismo rol. Puedes usar la AWS CodeStar consola o AWS CLI administrar los permisos.

> **Important**
>
> Para cambiar el rol de un miembro del equipo, debes tener el rol de AWS CodeStar propietario de ese proyecto o aplicar la `AWSCodeStarFullAccess` política.
> Cambiar los permisos de un miembro del equipo no afecta al acceso de ese miembro a ningún recurso externo AWS (por ejemplo, un GitHub repositorio 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 AWS CodeStar proyecto puede utilizar la AWS CodeStar consola para acceder a recursos ajenos a ese proyecto AWS pero relacionados con él.
> Cambiar el rol de un miembro del equipo en un proyecto no permite ni impide automáticamente que ese miembro participe en ningún entorno de AWS Cloud9 desarrollo del proyecto. Para permitir o evitar que un miembro del equipo participe en un entorno compartido, consulte Comparta un AWS Cloud9 entorno con un miembro del equipo del proyecto.

También puede conceder permisos a los usuarios para que accedan de forma remota a cualquier instancia de Amazon EC2 Linux asociada al proyecto. Tras conceder este permiso, el usuario debe cargar una clave pública de SSH que esté asociada a su perfil de AWS CodeStar usuario 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)

Puedes usar la AWS CodeStar consola para gestionar las funciones de los miembros del equipo. También puedes gestionar si los miembros del equipo tienen acceso remoto a las EC2 instancias de Amazon asociadas a tu proyecto.

Para cambiar el rol de un miembro del equipo

1. Abre la AWS CodeStar consola 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, elige el AWS CodeStar rol (propietario, colaborador o espectador) que quieres conceder a este usuario.

   Para obtener más información sobre AWS CodeStar los roles y sus permisos, consulteTrabajando con AWS CodeStar equipos.

   Seleccione Editar el miembro del equipo.

Para conceder a un miembro del equipo permisos de acceso remoto a las EC2 instancias de Amazon

1. Abre la AWS CodeStar consola 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 AWS CodeStar usuarios, 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)

Puedes utilizarla AWS CLI para gestionar la función de proyecto asignada a un miembro del equipo. Puedes usar los mismos AWS CLI comandos para administrar si ese miembro del equipo tiene acceso remoto a las EC2 instancias de Amazon asociadas a tu 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 la función de proyecto de un usuario de IAM llamado John_Doe y cambiar sus permisos de visor sin acceso remoto a las instancias de Amazon del proyecto: EC2

```
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"
}
```

Eliminar miembros del equipo de un AWS CodeStar proyecto

Tras eliminar a un usuario de un AWS CodeStar proyecto, el usuario sigue apareciendo en el historial de confirmaciones del repositorio del proyecto, pero ya no tiene acceso al CodeCommit repositorio ni a ningún otro recurso del proyecto, como la cartera de proyectos. (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 puede acceder al panel del proyecto y el proyecto ya no aparece en la lista de proyectos que el usuario ve en el AWS CodeStar panel. Puedes usar la AWS CodeStar consola o AWS CLI eliminar miembros del equipo de tu proyecto.

> **Important**
>
> Si bien eliminar a un miembro del equipo de un proyecto deniega el acceso remoto a EC2 las instancias de Amazon del proyecto, no cierra ninguna de las sesiones SSH activas del usuario.
>
> Eliminar a un miembro del equipo no afecta al acceso de ese miembro a ningún recurso externo AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA). Esos 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 un proyecto no elimina automáticamente los entornos de AWS Cloud9 desarrollo relacionados con ese miembro del equipo ni impide que ese miembro participe en cualquier entorno de AWS Cloud9 desarrollo relacionado al que haya sido invitado. Para eliminar un entorno de desarrollo, consulte Eliminar un AWS Cloud9 entorno de un proyecto. Para evitar que un miembro del equipo participe en un entorno compartido, consulte Comparta un AWS Cloud9 entorno con un miembro del equipo del proyecto.

Para eliminar a un miembro del equipo de un proyecto, debes tener el rol de AWS CodeStar propietario de ese proyecto o tener la `AWSCodeStarFullAccess` política aplicada a tu cuenta.

Temas

- Eliminar miembros del equipo (consola)
- Eliminar miembros del equipo (AWS CLI)

Eliminar miembros del equipo (consola)

Puedes usar la AWS CodeStar consola para eliminar miembros del equipo de tu proyecto.

Para eliminar un miembro del equipo de un proyecto

1. Abre la AWS CodeStar consola 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)

Puedes usar el AWS CLI para eliminar miembros del equipo de tu 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"
}
```

Cómo trabajar con su perfil AWS CodeStar de usuario

Su perfil de AWS CodeStar usuario está asociado a su usuario de IAM. Este perfil contiene un nombre para mostrar y una dirección de correo electrónico que se utilizan en todos los AWS CodeStar proyectos 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 público-privadas de SSH que utilizas cuando te conectas a las EC2 instancias de Amazon asociadas a los AWS CodeStar proyectos a los que perteneces.

> **Note**
>
> La información de estos temas cubre únicamente su perfil de AWS CodeStar usuario. Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos proveedores de recursos pueden usar sus propios perfiles de usuario, que pueden tener una configuración diferente. Para obtener más información, consulte la documentación del proveedor de recursos.

Temas

Administre la información de visualización de su perfil de AWS CodeStar usuario

Puede utilizar la AWS CodeStar consola o AWS CLI cambiar el nombre visible y la dirección de correo electrónico de su perfil de usuario. Un perfil de usuario no es específico del proyecto. Está asociada a tu usuario de IAM y se aplica a todos los AWS CodeStar proyectos a los que perteneces en una AWS región. Si pertenece a proyectos en más de una AWS región, tiene perfiles de usuario independientes.

Solo puede administrar su propio perfil de usuario en la AWS CodeStar consola. Si dispone de la `AWSCodeStarFullAccess` política, puede utilizarla AWS CLI para ver y gestionar otros perfiles.

> **Note**
>
> La información de este tema se refiere únicamente a su perfil de AWS CodeStar usuario. Si tu proyecto utiliza recursos externos AWS (por ejemplo, un GitHub repositorio o problemas en Atlassian JIRA), esos proveedores de recursos pueden usar sus propios perfiles de usuario, que pueden tener una configuración diferente. Para obtener más información, consulte la documentación del proveedor de recursos.

Temas

Administrar el perfil de usuario (consola)

Puedes gestionar tu perfil de usuario en la AWS CodeStar consola accediendo a cualquier proyecto del que seas miembro del equipo y cambiando la información de tu perfil. Como los perfiles de usuario son específicos de cada usuario, no de un proyecto, los cambios en tu perfil de usuario aparecen en todos los proyectos de una AWS región en la que seas 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 aquellos con el rol de AWS CodeStar propietario de un proyecto o con la `AWSCodeStarFullAccess` política aplicada, puede cambiar la información que se muestra.

Para cambiar la información de visualización en todos los proyectos de una AWS región

1. Abra la AWS CodeStar consola 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 usar el AWS CLI para crear y administrar su perfil de usuario en AWS CodeStar. También puede utilizarla AWS CLI para ver la información de su perfil de usuario y para ver todos los perfiles de usuario configurados para su AWS cuenta en una AWS región.

Asegúrese de que su AWS perfil esté 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 AWS región de su AWS cuenta

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. Utiliza este par de claves público-privadas de SSH para acceder a las EC2 instancias de Amazon que ejecutan Linux. Si un propietario del proyecto le ha concedido permiso de acceso remoto, solo podrá acceder a las instancias asociadas al proyecto. Puedes usar la AWS CodeStar consola o AWS CLI administrar tu clave pública.

> **Important**
>
> AWS CodeStar El propietario del proyecto puede conceder a los propietarios, colaboradores y espectadores acceso SSH a EC2 las instancias de Amazon 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 para los entornos. AWS Cloud9

Temas

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. Puedes generar una clave en cualquier formato permitido por Amazon EC2. Para obtener información sobre los formatos aceptables, consulta Cómo importar tu 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 la clave pública, se elimina el acceso a las EC2 instancias de Amazon que ejecutan Linux en todos los proyectos en los que se te concedió el 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 gestionar 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. Puedes generar una clave en cualquier formato permitido por Amazon EC2. Para obtener información sobre los formatos aceptables, consulta Cómo importar tu 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"
}
```

Conéctese a Amazon EC2 Instance con su clave privada

Asegúrese de haber creado un par de EC2 claves de Amazon. 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 EC2 claves de Amazon para AWS CodeStar proyectos. Para añadir la clave pública a su perfil de usuario, consulte las instrucciones indicadas anteriormente en este tema.

Para conectarse a una instancia de Amazon EC2 Linux mediante su clave privada

1. Con el proyecto abierto en la AWS CodeStar consola, en el panel de navegación, selecciona Proyecto.
2. En Recursos del proyecto, elige el enlace ARN en la fila donde Type es Amazon EC2 y Name comienza con instance.
3. En la EC2 consola de Amazon, selecciona Connect.
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, consulta los siguientes recursos en la Guía del EC2 usuario de Amazon.

- 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 AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted 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 AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los AWS programas de de . Para obtener más información sobre los programas de conformidad aplicables AWS CodeStar, consulte Servicios de AWS en el ámbito del programa de conformidad AWS .
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. También eres responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza AWS CodeStar. Los siguientes temas muestran cómo configurarlo AWS CodeStar para cumplir sus objetivos de seguridad y conformidad. También aprenderá a utilizar otros AWS servicios que le ayudan a supervisar y proteger sus AWS CodeStar recursos.

Cuando crees políticas personalizadas y utilices los límites de permisos AWS CodeStar, asegúrate de que el acceso sea con los privilegios mínimos concediendo solo los permisos necesarios para realizar una tarea y limitando los permisos a los recursos específicos. Para evitar que los miembros de otros proyectos accedan a los recursos de tu proyecto, otorga a los miembros de la organización permisos independientes para cada proyecto. 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, puedes usar un servicio como AWS Control Tower with AWS Organizations para aprovisionar cuentas para cada rol de desarrollador de un DevOps grupo. 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 gestión del acceso con privilegios mínimos a AWS los recursos mediante una estrategia de cuentas múltiples, consulte la estrategia de cuentas múltiples de AWS para su landing zone en la Guía del usuario de Control Tower AWS .

Temas

Protección de datos en AWS CodeStar

El AWS modelo de se aplica a protección de datos en AWS CodeStar. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global en la que se ejecutan todos los Nube de AWS. Eres responsable de mantener el control sobre el contenido alojado en esta infraestructura. También eres 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, consulta las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulta la publicación de blog sobre el Modelo de responsabilidad compartida de AWS y GDPR en el Blog de seguridad de AWS .

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:

- Utiliza la autenticación multifactor (MFA) en cada cuenta.
- Utilice SSL/TLS para comunicarse con los recursos. AWS Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure la API y el registro de actividad de los usuarios con. AWS CloudTrail
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.

- Utiliza servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados por FIPS 140-3 para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un punto final FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulta Estándar de procesamiento de la información federal (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, la API o AWS CLI AWS SDKs Cualquier dato que ingrese en etiquetas o campos de texto de formato libre utilizados para nombres se puede emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para 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 proyecto IDs. 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 herramienta Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a los AWS recursos. Los administradores de IAM controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar los recursos de AWS CodeStar . La IAM es una Servicio de AWS herramienta que puede utilizar sin coste adicional.

Temas

- AWS CodeStar Políticas y permisos a nivel de proyecto
- Ejemplos de políticas CodeStar basadas en la identidad de AWS
- Solución de problemas de AWS CodeStar Identity and Access

Público

La forma de usar AWS Identity and Access Management (IAM) varía según el trabajo que realice en AWS CodeStar.

Usuario del servicio: si utiliza el CodeStar servicio de AWS para realizar su trabajo, el administrador le proporcionará las credenciales y los permisos que necesita. A medida que vaya utilizando más CodeStar funciones de AWS para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarle a solicitar los permisos correctos al administrador. Si no puede acceder a una función de AWS CodeStar, consulteSolución de problemas de AWS CodeStar Identity and Access.

Administrador de servicios: si está a cargo de CodeStar los recursos de AWS en su empresa, probablemente tenga acceso total a AWS CodeStar. Es su trabajo determinar a qué CodeStar funciones y recursos de AWS deben acceder los usuarios del servicio. Luego, debe enviar solicitudes a su gestionador de IAM para cambiar los permisos de los usuarios de su servicio. Revise la información de esta página para conocer los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar IAM con AWS CodeStar, consulteCómo CodeStar funciona AWS con IAM.

Administrador de IAM: si es administrador de IAM, puede que le interese obtener más información sobre cómo redactar políticas para administrar el acceso a AWS. CodeStar Para ver ejemplos de políticas de AWS CodeStar basadas en la identidad que puede utilizar en IAM, consulte. Ejemplos de políticas CodeStar basadas en la identidad de AWS

Autenticación con identidades

La autenticación es la forma de iniciar sesión para AWS usar sus credenciales de identidad. Debe estar autenticado (con quien haya iniciado sesión AWS) como usuario de IAM o asumiendo una función de IAM. 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 (Centro de identidades

de IAM), 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, su gestionador habrá configurado previamente la federación de identidades mediante roles de IAM. Cuando accedes AWS mediante la federación, estás asumiendo un rol de forma indirecta.

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 proporciona 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 las solicitudes de la AWS API en la Guía del usuario de IAM.

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 multifactor (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte Autenticación multifactor en la Guía del usuario de AWS IAM Identity Center y Uso de la autenticación multifactor (MFA) en AWSen la Guía del usuario de IAM.

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 utiliza 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 las tareas que requieren que inicie sesión como usuario raíz, consulte Tareas que requieren credenciales de usuario raíz en la Guía del usuario de IAM.

Usuarios y grupos de IAM

Un usuario de IAM es una identidad propia Cuenta de AWS que tiene permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos emplear credenciales temporales, en lugar de crear usuarios de IAM que tengan credenciales de larga duración como contraseñas y claves de acceso. No obstante, si tiene casos de uso específicos que requieran credenciales de larga duración con usuarios de IAM, recomendamos rotar las claves de acceso.

Para más información, consulta Rotar las claves de acceso periódicamente para casos de uso que requieran credenciales de larga duración en la Guía del usuario de IAM.

Un grupo de IAM es una identidad que especifica un conjunto de usuarios de IAM. No puedes iniciar sesión como grupo. Puedes 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 IAMAdminsy concederle permisos para administrar los recursos de IAM.

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 permanentes a largo plazo y los roles proporcionan credenciales temporales. Para más información, consulte Cuándo crear un usuario de IAM (en lugar de un rol) en la Guía del usuario de IAM.

Roles de IAM

Un rol de IAM es una identidad dentro de usted Cuenta de AWS que tiene permisos específicos. Es similar a un usuario de IAM, pero no está asociado a una persona determinada. Puede asumir temporalmente una función de IAM en el AWS Management Console cambiando de función. Puede asumir un rol llamando a una operación de AWS API AWS CLI o utilizando una URL personalizada. Para más información sobre los métodos para el uso de roles, consulte Uso de roles de IAM en la Guía del usuario de IAM.

Los roles de IAM con credenciales temporales son útiles en las siguientes situaciones:

- Acceso de usuario federado: para asignar permisos a una identidad federada, puedes 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 acerca de roles para federación, consulta Creación de un rol para un proveedor de identidades de terceros en la Guía del usuario de IAM. Si utiliza el IAM Identity Center, debe configurar un conjunto de permisos. IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM para controlar a qué puedes acceder las identidades después de autenticarse. Para obtener información acerca de los conjuntos de permisos, consulta Conjuntos de permisos en la Guía del usuario de AWS IAM Identity Center .
- Permisos de usuario de IAM temporales: un usuario de IAM puedes asumir un rol de IAM para recibir temporalmente permisos distintos que le permitan realizar una tarea concreta.
- Acceso entre cuentas: puedes utilizar un rol de IAM para permitir que alguien (una entidad principal de confianza) de otra cuenta acceda a los recursos de la cuenta. Los roles son la forma

principal de conceder acceso entre cuentas. Sin embargo, con algunas Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para obtener información acerca de la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulta Acceso a recursos entre cuentas en IAM en la Guía del usuario de IAM.

- 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 usuario o un rol de IAM para realizar acciones en AWS ellas, se te considera principal. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. El FAS utiliza los permisos del principal que llama Servicio de AWS y los solicita Servicio de AWS para realizar solicitudes a los servicios descendentes. Las solicitudes de FAS 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 sobre las políticas a la hora de realizar solicitudes de FAS, consulta Reenviar sesiones de acceso.
  - Rol de servicio: un rol de servicio es un rol de IAM que adopta un servicio para realizar acciones en su nombre. Un administrador de IAM puedes crear, modificar y eliminar un rol de servicio desde IAM. Para obtener más información, consulta Creación de un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.
  - Función vinculada al servicio: una función vinculada a un servicio es un tipo de función de servicio que está vinculada a un. Servicio de AWS El servicio puedes 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 administrador de IAM puedes ver, pero no editar, los permisos de los roles vinculados a servicios.
- Aplicaciones que se ejecutan en Amazon EC2: puedes usar un rol de IAM para administrar las credenciales temporales de las aplicaciones que se ejecutan en una EC2 instancia y realizan AWS CLI solicitudes a la AWS API. Esto es preferible a almacenar las claves de acceso en 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 rol de IAM para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon en la Guía del usuario de IAM.

Para obtener información sobre el uso de los roles de IAM, consulte Cuándo crear un rol de IAM (en lugar de un usuario) en la Guía del usuario de IAM.

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 documentos JSON. Para obtener más información sobre la estructura y el contenido de los documentos de política JSON, consulta Información general de políticas JSON en la Guía del usuario de IAM.

Los administradores pueden usar las políticas de AWS JSON 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. Un administrador de IAM puedes crear políticas de IAM para conceder permisos a los usuarios para realizar acciones en los recursos que necesitan. A continuación, el administrador puedes añadir las políticas de IAM a roles y los usuarios puedes asumirlos.

Las políticas de IAM definen permisos para una acción independientemente del método que se utiliza 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 la API AWS Management Console AWS CLI, la o la AWS API.

Políticas basadas en identidad

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puedes asociar a una identidad, como un usuario de IAM, 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 más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de 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 roles de su Cuenta de AWS empresa. Las políticas administradas incluyen políticas AWS administradas y políticas administradas por el cliente. Para más información sobre cómo elegir una política administrada o una política insertada, consulte Elegir entre políticas administradas y políticas insertadas en la Guía del usuario de IAM.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Los ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios puedes utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puedes 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 puedes usar políticas AWS gestionadas de IAM en una política 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 políticas JSON.

Amazon S3 y Amazon VPC son ejemplos de servicios compatibles. AWS WAF 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 puedes establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Límites de permisos: un límite de permisos es una característica avanzada que le permite establecer los permisos máximos que una política basada en identidad puedes conceder a una entidad de IAM (usuario o rol de IAM). Puedes 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, consulta Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.

- Políticas de control de servicios (SCPs): SCPs son políticas de JSON que especifican los permisos máximos para una organización o unidad organizativa (OU). AWS Organizations AWS Organizations es un servicio para agrupar y gestionar 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. El 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 puedes proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para más información, consulta Políticas de sesión en la Guía del usuario de IAM.

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 usuario de IAM.

Cómo CodeStar funciona AWS con IAM

Antes de usar IAM para administrar el acceso a AWS CodeStar, debe comprender qué funciones de IAM están disponibles para su uso con AWS. CodeStar Para obtener una visión general de cómo AWS CodeStar y otros AWS servicios funcionan con IAM, consulte AWS Servicios que funcionan con IAM en la Guía del usuario de IAM.

Temas

- Políticas de AWS CodeStar basadas en la identidad
- Políticas basadas en CodeStar recursos de AWS

- Autorización basada en CodeStar etiquetas de AWS
- Funciones de AWS CodeStar IAM
- Acceso de usuarios de IAM a AWS CodeStar
- Acceso de usuarios federados a AWS CodeStar
- Uso de credenciales temporales con AWS CodeStar
- Roles vinculados a servicios
- Roles de servicio

Políticas de AWS CodeStar basadas en la identidad

Con las políticas de 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 utiliza en una política JSON, consulte Referencia de los elementos de las políticas JSON de IAM en la Guía del usuario de IAM.

Acciones

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puedes realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puedes utilizar para conceder o denegar el acceso en una política. Las acciones políticas suelen tener el mismo nombre que la operación de AWS API asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. 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 en AWS CodeStar utilizan el siguiente prefijo antes de la acción:`codestar:`. Por ejemplo, para permitir que un usuario de IAM 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 CodeStar las acciones de AWS, consulte Acciones definidas por AWS CodeStar en la Guía del usuario de IAM.

Recursos

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puedes realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Las instrucciones deben contener un elemento `Resource` o `NotResource`. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Puedes 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, utiliza un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

El recurso AWS CodeStar del proyecto tiene el siguiente ARN:

```
arn:aws:codestar:region:account:project/resource-specifier
```

Para obtener más información sobre el formato de ARNs, 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 CodeStar acciones de AWS, 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 CodeStar recursos de AWS y sus tipos ARNs, consulte Recursos definidos por AWS CodeStar en la Guía del usuario de IAM. Para saber con qué acciones puede especificar el ARN de cada recurso, consulte Acciones definidas por AWS. CodeStar

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 usuario de IAM.

Ejemplos

Para ver ejemplos de políticas de AWS CodeStar basadas en la identidad, consulte. Ejemplos de políticas CodeStar basadas en la identidad de AWS

Políticas basadas en CodeStar recursos de AWS

AWS CodeStar no admite políticas basadas en recursos.

Autorización basada en CodeStar etiquetas de AWS

Puede adjuntar etiquetas a los CodeStar proyectos de AWS o pasarlas en una solicitud a 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 CodeStar los recursos de AWS, consultethe 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 CodeStar proyectos de AWS basados en etiquetas

Funciones de AWS CodeStar IAM

Un rol de IAM es una entidad de su AWS cuenta que tiene permisos específicos.

Puede utilizarla AWS CodeStar como usuario de IAM, usuario federado, usuario raíz o como función asumida. Todos los tipos de usuarios con los permisos adecuados pueden gestionar los permisos del proyecto para sus AWS recursos, pero los AWS CodeStar gestionan automáticamente para los usuarios de IAM. Las políticas de IAM y los roles otorgan permisos y acceso a ese usuario en función del rol del proyecto. Puede utilizar la consola de IAM para crear otras políticas que asignen AWS CodeStar y otros permisos a un usuario de IAM.

Por ejemplo, puede que quiera permitir a un usuario ver pero no cambiar un proyecto de AWS CodeStar . En este caso, se añade el usuario de IAM a un AWS CodeStar proyecto con la función de espectador. Cada AWS CodeStar proyecto tiene un conjunto de políticas que le 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 los usuarios de IAM y los usuarios federados. Solo los usuarios de IAM se pueden añadir a equipos. Para conceder permisos para proyectos a los usuarios de IAM, añada el usuario al equipo de 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	Usuario de IAM	Usuario federado	Usuario raíz
Administración de claves SSH para acceso remoto a proyectos de Amazon EC2 y Elastic Beanstalk	✓		
AWS CodeCommit Acceso SSH	✓		
Permisos de usuario de IAM gestionados por AWS CodeStar	✓		
Permisos de proyectos administrados manualmente		✓	✓
Los usuarios pueden añadirse al proyecto como miembros del equipo	✓		

Acceso de usuarios de IAM a AWS CodeStar

Al añadir un usuario de IAM a un proyecto y elegir un rol para el usuario, AWS CodeStar aplica la política adecuada automáticamente al usuario de IAM. En el caso de los usuarios de IAM, no es necesario asociar ni administrar políticas o permisos directamente en IAM. Para obtener información sobre cómo añadir un usuario de IAM a un AWS CodeStar proyecto, consulte. Añadir miembros del equipo a un AWS CodeStar proyecto Para obtener información sobre cómo eliminar un usuario de IAM de un AWS CodeStar proyecto, consulte. Eliminar miembros del equipo de un AWS CodeStar proyecto

Asociar una política insertada a un usuario de IAM

Al añadir un usuario a un proyecto, AWS CodeStar se adjunta automáticamente la política gestionada al proyecto que coincide con la función del usuario. No debes adjuntar manualmente una política AWS CodeStar gestionada para un proyecto a un usuario de IAM. Con la excepción de`AWSCodeStarFullAccess`, no recomendamos adjuntar políticas que cambien los permisos de un usuario de IAM en un AWS CodeStar proyecto. Si decide crear y asociar sus propias políticas, consulte Añadir y eliminar permisos de identidad de IAM en la Guía del usuario de IAM.

Acceso de usuarios federados a AWS CodeStar

En lugar de crear un usuario de IAM o utilizar el usuario raíz, puede utilizar las identidades de usuario del directorio de usuarios de su empresa AWS Directory Service, un proveedor de identidades web o los usuarios de IAM que asuman funciones. Esto se conoce como usuarios federados.

Conceda a los usuarios federados el acceso a su AWS CodeStar proyecto adjuntando manualmente las políticas gestionadas descritas en Políticas y permisos a AWS CodeStar nivel de proyecto a la función de IAM del usuario. Debe adjuntar la política de propietario, colaborador o espectador después de AWS CodeStar crear los recursos del proyecto y las funciones de IAM.

Requisitos previos:

- Debe tener configurado un proveedor de identidad. Por ejemplo, puedes configurar un proveedor de identidad SAML y configurar la AWS autenticación a través de ese proveedor. Para obtener más información sobre la configuración de un proveedor de identidad, consulte Creación de proveedores de identidad de IAM. Para obtener más información sobre federación SAML, consulte Acerca de la federación basada en SAML 2.0.
- Tiene que haber creado un rol que asuma un usuario federado cuando se solicita acceso a través de un proveedor de identidad. Debe asociarse una política de confianza de STS al rol que permita a los usuarios federados asumir el rol. Para obtener más información, consulte Usuarios federados y roles en la Guía del usuario de IAM.
- Debes haber creado tu 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 AWSCode StarFullAccess 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 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.

> **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 AWS CodeStar Viewer/Contributor/Owner gestionada de su proyecto a la función del usuario federado.

1. Abra la consola de IAM. 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 AWS CodeStar Viewer/Contributor/Owner gestionada de su proyecto 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 los usuarios de IAM, tiene que asociar y desasociar manualmente las políticas administradas de los usuarios federados. Esto equivale a asignar permisos de proyecto a los miembros del equipo en. AWS CodeStar Para realizar estos 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.

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.
- Tu AWS CodeStar proyecto debe haber sido creado. La política administrada no está disponible en IAM hasta que se cree el proyecto.

1. Abra la consola de IAM. 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 del rol 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 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.

1. Abra la consola de IAM. 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 administrada 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 los usuarios de IAM, tiene que asociar y desasociar manualmente las políticas administradas de los usuarios federados. Para realizar estos 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.

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 de AWS Cloud9 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 consola de IAM. 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 realizar estos 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.

1. Abra la consola de IAM. 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 utilizar credenciales temporales para iniciar sesión con federación, asumir un rol de IAM o asumir un rol de acceso entre cuentas. Para obtener credenciales de seguridad temporales, puede llamar a operaciones de AWS STS API como AssumeRoleo 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 para miembros del equipo solo permite añadir un usuario de IAM 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 a servicios aparecen en la cuenta de IAM 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 a servicios.

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 cuenta de IAM 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 apoya las funciones de servicio. AWS CodeStar utiliza un rol de servicio cuando crea y administra los recursos de su proyecto. aws-codestar-service-role Para obtener más información, consulte Términos y conceptos sobre los roles en la Guía del usuario de IAM.

> **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 para el primer acceso: sus credenciales de usuario raíz y Creación del primer grupo y usuario administrador en la Guía del usuario de IAM.

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 funciones y políticas de IAM que necesita para gestionar los recursos del proyecto. Las políticas se dividen en tres categorías:

- Políticas de IAM para miembros del equipo del proyecto.
- Políticas de IAM para roles de trabajador.
- Políticas de IAM para un rol de ejecución en tiempo de ejecución.

Políticas de IAM para miembros del equipo

Al crear un proyecto, AWS CodeStar crea tres políticas gestionadas por el cliente para que el propietario, el colaborador y el espectador puedan acceder al proyecto. Todos los AWS CodeStar proyectos contienen políticas de IAM para estos tres niveles de acceso. Estos niveles de acceso son específicos del proyecto y se definen mediante una política gestionada de IAM con un nombre estándar, donde *project-id* está 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 los permisos, asocie políticas adicionales al usuario de IAM.

A medida que se añaden miembros del equipo (usuarios de IAM) al proyecto y se eligen sus niveles de acceso, se asocia la política correspondiente al usuario de IAM, otorgando al usuario un conjunto adecuado de permisos para actuar en los recursos del proyecto. En la mayoría de casos, no es necesario asociar ni administrar políticas o permisos directamente en IAM. No se recomienda adjuntar manualmente una política de nivel de AWS CodeStar acceso a un usuario de IAM. Si es absolutamente necesario, como complemento de una política de nivel de AWS CodeStar acceso, puede crear sus propias políticas gestionadas o integradas para aplicar su propio nivel de permisos a un usuario de IAM.

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 usuario de IAM con esta política puede realizar todas AWS CodeStar las acciones del proyecto, pero a diferencia de un usuario de IAM 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.

Políticas de IAM para roles de trabajador

Si crea su AWS CodeStar proyecto después del PDT del 6 de diciembre de 2018, AWS CodeStar crea dos funciones de trabajador `CodeStar-project-id-ToolChain` y`CodeStar-project-id-CloudFormation`. Un rol de trabajador es un rol de IAM específico del proyecto que se AWS CodeStar crea para transferirlo 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 (a partir del 6 de diciembre de 2018, PDT).

El rol de CloudFormation trabajador incluye permisos para los recursos seleccionados que admite AWS CloudFormation, así como permisos para crear usuarios, roles y políticas de IAM 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 entidad de IAM (usuario o rol) creada en el conjunto de infraestructuras. Para ver un ejemplo de la instrucción de política insertada para este rol, consulte AWS CloudFormation Política sobre el rol de los trabajadores.

Para CodeStar los proyectos de AWS creados antes del 6 de diciembre de 2018, PDT AWS CodeStar crea roles de trabajador individuales para los recursos de la cadena de herramientas CodePipeline CodeBuild, como y CloudWatch Events, y también crea un rol 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 2018 PDT)
- AWS CodePipeline Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018 PDT)
- AWS CodeBuild Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018 PDT)
- Política de funciones de CloudWatch los trabajadores de Amazon Events (antes del 6 de diciembre de 2018 PDT)

Política de IAM para el rol de ejecución

Para los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar crea una función de ejecución genérica para el proyecto de muestra en su 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 muestra, podrá crear funciones de IAM adicionales, y la AWS CloudFormation política de 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 2018, PDT crea AWS CodeStar 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 plantilla SAM, 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 PDT del 6 de diciembre de 2018, al crear un proyecto, AWS CodeStar crea una política administrada por el cliente y la asigna como límite de permisos de IAM a las funciones de IAM en el proyecto. AWS CodeStar exige que todas las entidades de IAM 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 permisos, consulte Lógica de evaluación de políticas en la Guía del usuario de IAM.

AWS CodeStar utiliza un límite de permisos específico del proyecto para evitar la escalada de privilegios a recursos ajenos al proyecto. El límite de CodeStar permisos de AWS incluye ARNs los recursos del proyecto. Para ver un ejemplo de esta instrucción de política, consulte Política de límites de CodeStar permisos de AWS.

La CodeStar transformación de AWS actualiza esta política al añadir o eliminar un recurso compatible del proyecto mediante la pila de aplicaciones (`template.yml`).

Adición de un límite de permisos de IAM a proyectos existentes

Si tiene un CodeStar proyecto de AWS que se creó antes del 6 de diciembre de 2018 PDT, debe añadir manualmente un límite de permisos a las funciones de IAM 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 CodeStar administrados por AWS 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 necesite permisos de IAM adicionales para actualizar la pila desde la consola de AWS CloudFormation .
4. (Opcional) Si desea crear roles de IAM específicos de aplicaciones, complete este paso. Desde la consola de IAM, actualiza la política integrada adjunta al AWS CloudFormation rol de tu proyecto para incluir el siguiente fragmento. Es posible que necesite recursos de IAM 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. Impulse un cambio en la cartera de proyectos para que AWS CodeStar actualice el límite de los permisos con los permisos adecuados.

Para obtener más información, consulte Añadir un rol de IAM a un proyecto.

Ejemplos de políticas CodeStar basadas en la identidad de AWS

De forma predeterminada, los usuarios y roles de IAM no tienen permiso para crear o modificar CodeStar los recursos de AWS. Tampoco pueden realizar tareas con la AWS API AWS Management Console AWS CLI, o. Un administrador debe crear políticas de IAM que concedan permisos a los usuarios y a los roles para realizar operaciones de la API concretas en los recursos especificados que necesiten. El administrador debe adjuntar esas políticas a los usuarios o grupos de IAM que necesiten esos permisos.

Para obtener información acerca de cómo crear una política basada en identidad de IAM con estos documentos de políticas JSON de ejemplo, consulte Creación de políticas en la pestaña JSON en la Guía del usuario de IAM.

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 (a partir del 6 de diciembre de 2018, PDT)
- AWS CloudFormation Política sobre el rol de los trabajadores
- AWS CloudFormation Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018 PDT)
- AWS CodePipeline Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018 PDT)
- AWS CodeBuild Política sobre las funciones de los trabajadores (antes del 6 de diciembre de 2018 PDT)
- Política de funciones de CloudWatch los trabajadores de Amazon Events (antes del 6 de diciembre de 2018 PDT)
- Política de límites de CodeStar permisos de AWS

- Listado de recursos para un proyecto
- Uso de la CodeStar consola de AWS
- Permitir a los usuarios ver sus propios permisos
- Actualización de un proyecto de AWS CodeStar
- Añadir un miembro de equipo a un proyecto
- Mostrar los perfiles de usuario asociados a una cuenta AWS
- Visualización de CodeStar proyectos de AWS basados en etiquetas
- AWS CodeStar actualizaciones de las políticas AWS administradas

Prácticas recomendadas relativas a políticas

Las políticas basadas en la identidad determinan si alguien puede crear, acceder o eliminar CodeStar los recursos de AWS de su cuenta. 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. Con el fin de obtener más información, consulta las políticas administradas por AWS o las políticas administradas por AWS para funciones de tarea en la Guía de usuario de IAM.
- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se puedes llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulta Políticas y permisos en IAM en la Guía del usuario de IAM.
- Utiliza condiciones en las políticas de IAM para restringir aún más el acceso: puedes agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puedes escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando 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, consulta Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.

- Utiliza el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte Política de validación de Analizador de acceso de IAM en la Guía de usuario de IAM.
- Requerir autenticación multifactor (MFA): si tiene un escenario que requiere usuarios de IAM o un usuario raíz en Cuenta de AWS su cuenta, active la MFA para mayor seguridad. Para solicitar la MFA cuando se invocan las operaciones de la API, agregue las condiciones de la MFA a sus políticas. Para más información, consulte Configuración del acceso a una API protegido por MFA en la Guía de usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte las Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

AWSCodeStarServiceRole 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 instrucciones de Configuración AWS CodeStar, asoció una política denominada `AWSCodeStarFullAccess` a su usuario de IAM. 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 según 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`—Recupere 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 CodeStar propietario de AWS permite a los usuarios realizar todas las acciones de un CodeStar proyecto de AWS sin restricciones. AWS CodeStar aplica la `CodeStar_`*`project-id`*`_Owner` política a los miembros del equipo del proyecto con el nivel de acceso del 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 CodeStar colaborador de AWS permite a un usuario contribuir al proyecto y cambiar el panel 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 CodeStar espectadores de AWS permite a los usuarios 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 los espectadores. Los usuarios con acceso como espectadores pueden ver un proyecto en AWS CodeStar, pero no pueden cambiar sus recursos 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 (a partir del 6 de diciembre de 2018, PDT)

Para AWS CodeStar los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar crea una política en línea para un rol 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 Políticas de IAM para roles de trabajador.

```
{
  "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 sobre el rol de los trabajadores

Para AWS CodeStar los proyectos creados después del PDT del 6 de diciembre de 2018, AWS CodeStar crea una política en línea para un rol de trabajador que crea AWS CloudFormation recursos para su proyecto de AWS CodeStar . 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 Políticas de IAM para roles de trabajador.

```
{
{
```

```
    "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 2018 PDT)

Si su CodeStar proyecto de AWS se creó antes del 6 de diciembre de 2018 PDT, AWS CodeStar creó una política en línea para un puesto 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 2018 PDT)

Si su CodeStar proyecto de AWS se creó antes del 6 de diciembre de 2018 PDT, AWS CodeStar creó una política en línea para un puesto 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 2018 PDT)

Si su CodeStar proyecto de AWS se creó antes del 6 de diciembre de 2018 PDT, AWS CodeStar creó una política en línea para un puesto 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 2018 PDT)

Si su CodeStar proyecto de AWS se creó antes del 6 de diciembre de 2018 PDT, AWS CodeStar creó 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"
        }
    ]
}
```

Política de límites de CodeStar permisos de AWS

Si crea un CodeStar proyecto de AWS después del 6 de diciembre de 2018 PDT, 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 dinámico 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, quiere conceder acceso a un usuario de IAM específico de su AWS cuenta 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 CodeStar consola de AWS

No se requieren permisos específicos para acceder a la CodeStar consola de AWS, pero no puede hacer nada útil a menos que tenga la `AWSCodeStarFullAccess` política o uno de los roles 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 Políticas de IAM para miembros del equipo.

No necesita conceder permisos mínimos de consola a los usuarios que solo realizan llamadas a la API AWS CLI o a la AWS misma. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

Permitir a los usuarios ver sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas gestionadas e insertadas que se asocian a la identidad de sus usuarios. Esta política incluye permisos para completar esta acción en la consola o mediante programación mediante la API AWS CLI o AWS .

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

```
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Actualización de un proyecto de AWS CodeStar

En este ejemplo, desea conceder a un usuario de IAM específico de su 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 usuario de IAM específico la posibilidad de añadir miembros del equipo a un AWS CodeStar proyecto con el ID del proyecto*my-first-projec*, pero negarle explícitamente 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"
    }
      ]

  ]
}
```

Mostrar los perfiles de usuario asociados a una cuenta AWS

En este ejemplo, permite que un usuario de IAM 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 CodeStar proyectos de AWS basados en etiquetas

Puede utilizar las condiciones de su política basada en la identidad para controlar el acceso a los CodeStar proyectos de AWS 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}"}
            }
        }
    ]
}
```

También puede asociar esta política al usuario de IAM en su cuenta. Si un usuario llamado `richard-roe` intenta ver un CodeStar proyecto de AWS, 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 de la política de JSON de IAM: Condición en la Guía del usuario de IAM.

AWS CodeStar actualizaciones de las políticas AWS administradas

Consulte los detalles sobre las actualizaciones de las políticas AWS administradas de AWS CodeStar desde que este servicio comenzó a realizar el seguimiento de estos cambios. Para recibir alertas automáticas sobre los cambios en esta página, suscríbase a la fuente RSS de la página del historial de CodeStar documentos de AWS.

Cambio	Descripción	Fecha
AWSCodeStarFullAccess Política: actualice la	Se ha actualizado la política de roles de AWS CodeStar	24 de marzo de 2023

Cambio	Descripción	Fecha
AWSCode StarFullAccess política	acceso. El resultado de la política es el mismo, pero la formación de nubes requiere ListStacks algo adicional DescribeStacks, algo que ya es obligatorio.	
AWSCodeStarService Role Política: actualice la AWSCode StarServiceRole política	La política del rol de CodeStar servicio de AWS se actualizó para corregir las acciones redundantes de la declaración de política. La política de roles de servicio permite al CodeStar servicio de AWS 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 en sus políticas AWS administradas.	23 de septiembre de 2021

Solución de problemas de AWS CodeStar Identity and Access

Utilice la siguiente información como ayuda para diagnosticar y solucionar problemas comunes que puedan surgir al trabajar con AWS CodeStar e IAM.

Temas

- No estoy autorizado a realizar ninguna acción en AWS CodeStar
- No estoy autorizado a realizar lo siguiente: PassRole
- Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis CodeStar recursos de AWS

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 gestionador es la persona que le proporcionó las credenciales de inicio de sesión.

En el siguiente ejemplo, el error se produce cuando el usuario de IAM, `mateojackson`, intenta utilizar la consola para ver detalles sobre una *`widget`*, pero no tiene permisos `codestar:`*`GetWidget`*.

```
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 lo siguiente: PassRole

Si recibe un error que indica que no está autorizado a realizar la `iam:PassRole` acción, sus políticas deben actualizarse para que pueda transferir una función a AWS CodeStar.

Algunas Servicios de AWS le 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 usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en AWS CodeStar. 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 su AWS administrador. El gestionador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis CodeStar recursos de AWS

Puedes crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puedes 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), puede utilizar esas políticas para permitir que las personas accedan a sus recursos.

Para obtener más información, consulte lo siguiente:

- Para saber si AWS CodeStar admite estas funciones, consulteCómo CodeStar funciona AWS con IAM.
- Para obtener información sobre cómo proporcionar acceso a los recursos de su Cuentas de AWS propiedad, consulte Proporcionar acceso a un usuario de IAM en otro usuario de su propiedad Cuenta de AWS en la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso a tus recursos a terceros Cuentas de AWS, consulta Cómo proporcionar acceso a recursos que Cuentas de AWS son propiedad de terceros en la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso mediante una federación de identidades, consulta Proporcionar acceso a usuarios autenticados externamente (identidad federada) en la Guía del usuario de IAM.
- Para conocer sobre la diferencia entre las políticas basadas en roles y en recursos para el acceso entre cuentas, consulte Acceso a recursos entre cuentas en IAM en la Guía del usuario de IAM.

Registrar llamadas a la AWS CodeStar API con AWS CloudTrail

AWS CodeStar está integrado con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un AWS servicio en AWS CodeStar. CloudTrail captura todas las llamadas a la API AWS CodeStar como eventos. Las llamadas capturadas incluyen las llamadas desde la AWS CodeStar consola y las llamadas en código a las operaciones de la AWS CodeStar API. Si crea un registro, puede habilitar la entrega continua de CloudTrail eventos a un bucket de S3, incluidos los eventos para ellos AWS CodeStar. Si no configuras una ruta, podrás ver los eventos más recientes en la CloudTrail consola, en el historial de eventos. Con la información recopilada por usted CloudTrail, puede determinar el destinatario de la solicitud AWS CodeStar, la dirección IP desde la que se realizó la solicitud, quién la realizó, cuándo se realizó y otros detalles.

Para obtener más información CloudTrail, consulta la Guía AWS CloudTrail del usuario.

AWS CodeStar Información en CloudTrail

CloudTrail está habilitada en su AWS cuenta al crear la cuenta. Cuando se produce una actividad en AWS CodeStar, esa actividad se registra en un CloudTrail evento junto con otros eventos de AWS servicio en el historial de eventos. Puedes ver, buscar y descargar los eventos recientes en tu AWS cuenta. Para obtener más información, consulte Visualización de eventos con el historial de CloudTrail eventos.

Para tener un registro continuo de los eventos de tu AWS cuenta, incluidos los eventos de tu cuenta AWS CodeStar, crea una ruta. De forma predeterminada, cuando creas una ruta en la consola, la ruta se aplica a todas AWS las regiones. La ruta registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al depósito de S3 que especifique. Puede configurar otros AWS servicios para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail Servicios e integraciones compatibles
- Configuración de las notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de CloudTrail registro de varias regiones y recibir archivos de CloudTrail registro de varias cuentas

Todas AWS CodeStar las acciones se registran CloudTrail y se documentan en la referencia de la AWS CodeStar API. Por ejemplo, las llamadas a las `DescribeProject` `AssociateTeamMember` acciones y las llamadas generan entradas en los archivos de CloudTrail registro. `UpdateProject`

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario lo ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales raíz o del usuario de IAM.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro AWS servicio.

Para obtener más información, consulte el Elemento userIdentity de CloudTrail .

Descripción de las entradas de los archivos de AWS CodeStar registro

CloudTrail Los archivos de registro contienen una o más entradas de registro. Un evento representa una solicitud única de cualquier fuente e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. CloudTrail Los archivos de registro no son un registro ordenado de las llamadas a la API pública, por lo que no aparecen en ningún orden específico.

En el siguiente ejemplo, se muestra una entrada de CloudTrail registro que muestra la llamada a una `CreateProject` operación AWS CodeStar:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAJLIN2OF3UBEXAMPLE: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": "AROAJLIN2OF3UBEXAMPLE",
        "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 conformidad para AWS CodeStar

AWS CodeStar no está incluido en el ámbito de ningún programa de AWS cumplimiento.

Para obtener una lista de AWS los servicios incluidos en el ámbito de los programas de cumplimiento específicos, consulte AWS los servicios incluidos en el ámbito de aplicación por programa de cumplimiento. Para obtener información general, consulte Programas de conformidad de AWS.

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulta Descarga de informes en AWS Artifact.

Resiliencia en AWS CodeStar

La infraestructura AWS global se basa en AWS regiones y zonas de disponibilidad. AWS Las regiones proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia. Con las zonas de disponibilidad, puedes 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 AWS las regiones y las zonas de disponibilidad, consulte Infraestructura global.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 AWS 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 llamadas a la API AWS publicadas para acceder a CodeStar través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Exigimos TLS 1.2 y recomendamos TLS 1.3.
- Conjuntos de cifrado con confidencialidad directa total (PFS) como DHE (Ephemeral Diffie-Hellman) o 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 estar firmadas mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad principal de IAM. También puedes 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 mediante Internet AWS CodeStar están abiertos a la Internet pública, a menos que modifique manualmente la configuración de acceso a través de Amazon EC2, API Gateway o Elastic Beanstalk. Esto se hace de forma intencionada. Puede modificar la configuración de acceso en Amazon EC2, 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 puntos finales de la VPC (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 AWS servicios 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	Un máximo de 333 proyectos en una AWS cuenta. El límite real varía en función del nivel de dependencia de otros servicios (por ejemplo, el número máximo de canalizaciones CodePipeline permitido para tu AWS cuenta).
Número de AWS CodeStar proyectos a los que puede pertenecer un usuario de IAM	Máximo de 10 por cada usuario de IAM individual.
Proyecto IDs	El proyecto IDs debe ser único en una AWS cuenta. El proyecto IDs debe tener al menos 2 caracteres y no puede superar los 15 caractere s. Los caracteres permitidos son: Letras de la a a la z inclusive. Número del 0 al 9 inclusive. El carácter especial - (signo menos). Cualquier otro carácter como, por ejemplo, mayúsculas, espacios, . (punto), @ (signo de la arroba) o _ (guion bajo) no están permitidos.
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 de un AWS CodeStar proyecto	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 su uso 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

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 EC2 configuración de Amazon al crear un proyecto

Problema: cuando editas las opciones de EC2 configuración de Amazon durante la creación del proyecto, ves un mensaje de error o una opción atenuada y no puedes 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 EC2 configuración de Amazon) 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 VPCs. Puede que haya eliminado la VPC predeterminada y no haya creado otras. Abre la consola de Amazon VPC en https://console.aws.amazon.com/vpc/, selecciona Your VPCs y asegúrate 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 proyecto: se ha eliminado un AWS CodeStar 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: abre 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. Abre 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 fuente 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/.

- Aplicación y grupos de implementación asociados 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 consola en AWS Lambda . https://console.aws.amazon.com/lambda/
- Uno o más APIs 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.

  Abre la EC2 consola de Amazon 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 del equipo a un AWS CodeStar proyecto .

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 AWS CodeStar Viewer/Contributor/Owner gestionada de su proyecto a la función del usuario federado.

Agregue usuarios federados a su AWS Cloud9 entorno adjuntando políticas manualmente. Consulte Adjunte una política AWS Cloud9 administrada 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 administrada 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 administradas 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 que 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/

   (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* aparece 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 AWSElasticBeanstalkEnhancedHealthy AWSElasticBeanstalkService. (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).

Posibles soluciones:

- 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 del fallo en la vista del AWS CodeStar panel de control de la cartera de proyectos.

A continuación, selecciona el CloudFormationenlace 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 Adición de un límite de permisos de IAM a proyectos existentes

No se pudo 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.

Correcciones posibles: Para obtener información sobre los niveles de permisos de un GitHub repositorio, consulta https://docs.github.com/en/free-pro-team@ latest/github/setting-up-and-managing-organizations-and-teams/permission - levels-for-an-organization.

AWS CodeStar Notas de publicación de la guía del usuario

En la siguiente tabla se describen los cambios importantes de cada versión de la Guía del AWS CodeStar usuario. Para recibir notificaciones sobre los cambios en 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 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. Para hacer referencia a la política actualizada, consulte la AWSCodeStarFullAccess Política.	24 de marzo de 2023
Actualizaciones de la política de roles de servicio	Se ha actualizado la política de funciones de AWS CodeStar servicio. Para hacer referencia a la política actualizada, consulte la AWSCodeStarServiceRole Política.	23 de septiembre de 2021
Utilice un recurso de conexión para los proyectos con un repositorio GitHub de origen	Cuando utilizas la consola para crear un proyecto en AWS CodeStar un GitHub repositorio, se utiliza un recurso de conexión para gestionar tus GitHub acciones. Las conexiones utilizan GitHub aplicaciones, mientras que la	27 de abril de 2021

	GitHub autorización anterior se utilizaba OAuth. Para ver un tutorial que muestra cómo crear un proyecto que utilice una conexión a GitHub, consulte el Tutorial: Crear un proyecto con un repositorio de GitHub código fuente. En el tutorial también se explica cómo crear, revisar y combinar una solicitud de extracción para el repositorio de fuentes del proyecto.	
AWS CodeStar apoya AWS Cloud9 en la región EE.UU. Oeste (Norte de California)	AWS CodeStar ahora admite su uso AWS Cloud9 en la región EE.UU. Oeste (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 AWS CodeStar servicio pasó a una nueva experiencia de usuario en la AWS consola. La guía del usuario se ha actualizado a la nueva experiencia de la consola.	12 de agosto de 2020

AWS CodeStar los proyectos se pueden crear con la AWS CodeStar CLI	AWS CodeStar los proyectos se pueden crear con el comando CLI. AWS CodeStar crea su proyecto e infraestr uctura utilizando el código fuente y una plantilla de cadena de herramientas que usted proporcione. Consulte Crear un proyecto en AWS CodeStar (AWS CLI).	24 de octubre de 2018
Todas las plantillas de AWS CodeStar proyectos ahora incluyen AWS CloudFormation un archivo para las actualiza ciones de la infraestructura	AWS CodeStar funciona AWS CloudFormation para permitirle usar el código para crear servicios y servidore s de soporte o plataform as sin servidor en la nube. El AWS CloudFormation archivo ahora está disponibl e para todos los tipos de plantillas de AWS CodeStar proyectos (plantillas con la plataforma de procesamiento Lambda o Elastic Beanstalk). EC2 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

AWS CodeStar Las notificaciones de actualización de la Guía del usuario ahora están disponibles a través de RSS	La versión HTML de la Guía del AWS CodeStar usuario ahora admite una fuente RSS de las actualizaciones que se documentan en la página de notas de publicación de la actualización 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 introducidos en cada versión de la Guía del AWS CodeStar usuario antes del 30 de junio de 2018.

Cambio	Descripción	Fecha de modificación
La guía AWS CodeStar del usuario ya está disponible en GitHub	Esta guía ya está disponible en GitHub. También puede utilizarla GitHub para enviar comentarios y solicitudes de cambio en relación con el contenido de esta guía. Para obtener más información, pulse el GitHub icono Editar en la barra de navegación de la guía o consulte el aws-codestar-user-guide repositorio awsdocs/ en el sitio web. GitHub	22 de febrero de 2018
AWS CodeStar ya está disponible en Asia Pacífico (Seúl)	AWS CodeStar ya está disponible en la región de Asia Pacífico (Seúl). Para obtener más información, consulte	14 de febrero de 2018

Cambio	Descripción	Fecha de modificación
	AWS CodeStar en la Referencia general de Amazon Web Services.	
AWS CodeStar ya está disponible en Asia Pacífico (Tokio) y Canadá (Central)	AWS CodeStar ya está disponible en las regiones de Asia Pacífico (Tokio) y Canadá (Central). Para obtener más información, consulte AWS CodeStar en la Referencia general de Amazon Web Services.	20 de diciembre de 2017
AWS CodeStar ahora es compatible AWS Cloud9	AWS CodeStar ahora admite el uso AWS Cloud9 de un IDE en línea basado en un navegador web para trabajar con el código del proyecto. Para obtener más información, consulte Úselo AWS Cloud9 con AWS CodeStar. Para obtener una lista de AWS las regiones compatibles, consulte AWS Cloud9la. Referencia general de Amazon Web Services	30 de noviembre de 2017
AWS CodeStar ahora es compatible GitHub	AWS CodeStar ahora admite almacenar el código del proyecto en GitHub. Para obtener más información, consulte la sección sobre crear un proyecto.	12 de octubre de 2017
AWS CodeStar ahora disponible en EE. UU. Oeste (Norte de California) y Europa (Londres)	AWS CodeStar ya está disponible en las regiones de EE. UU. Oeste (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
AWS CodeStar ahora disponible en Asia Pacífico (Sídney), Asia Pacífico (Singapur) y Europa (Fráncfort)	AWS CodeStar ya está disponible en las regiones de 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

Cambio	Descripción	Fecha de modificación
AWS CloudTrail ahora es compatible AWS CodeStar	AWS CodeStar ahora está integrado con CloudTrail un servicio que captura las llamadas a la API realizadas por o en su nombre AWS CodeStar en su AWS cuenta y entrega los archivos de registro a un bucket de Amazon S3 que especifique. Para obtener más información, consulte Registrar llamadas a la AWS CodeStar API 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

AWS Glosario

Para obtener la AWS terminología más reciente, consulte el AWS glosario de la Glosario de AWS Referencia.