
AWS CodeBuild

Guía del usuario

Versión de API 2016-10-06



AWS CodeBuild: Guía del usuario

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

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

Table of Contents

¿Qué es AWS CodeBuild?	1
.....	1
¿Cómo se ejecuta CodeBuild?	1
Precios para CodeBuild	2
¿Cómo puedo comenzar a usar CodeBuild?	2
Conceptos	3
Funcionamiento de CodeBuild	3
Pasos siguientes	4
Introducción	5
Primeros pasos con la consola	5
Steps	5
Paso 1: Crear el código fuente	6
Paso 2: Cree el archivo buildspec	8
Paso 3: Cree dos buckets de S3	9
Paso 4: Cargar el código fuente y el archivo buildspec	10
Paso 5: Cree el proyecto de compilación	11
Paso 6: Ejecute la compilación	12
Paso 7: Ver información resumida sobre la compilación	13
Paso 8: Ver información detallada sobre la compilación	13
Paso 9: Obtener el artefacto de salida de la compilación	14
Paso 10: Eliminación del bucket de S3	15
Encapsulación	15
Primeros pasos con AWS CLI	16
Steps	16
Paso 1: Crear el código fuente	16
Paso 2: Cree el archivo buildspec	18
Paso 3: Cree dos buckets de S3	20
Paso 4: Cargar el código fuente y el archivo buildspec	21
Paso 5: Cree el proyecto de compilación	21
Paso 6: Ejecute la compilación	24
Paso 7: Ver información resumida sobre la compilación	26
Paso 8: Ver información detallada sobre la compilación	27
Paso 9: Obtener el artefacto de salida de la compilación	29
Paso 10: Eliminación del bucket de S3	30
Encapsulación	30
Muestras	31
Ejemplos de Windows	31
Ejecución de los ejemplos	31
Estructura de directorios	32
Files	33
Ejemplos basados en casos de uso	44
Ejemplo de Amazon ECR	46
Ejemplo de Amazon EFS	50
Ejemplo de AWS CodeDeploy	54
AWS CodePipelineejemplo de integración con compilaciones por lotes	57
Ejemplo de integración de AWS CodePipeline con varios orígenes de entrada y varios artefactos de salida	62
Ejemplo de AWS Config	65
Ejemplo de AWS Elastic Beanstalk	66
Ejemplo de filtros de webhook y solicitudes de extracción de Bitbucket	74
Ejemplo de insignias de compilación	78
Ejemplo de notificaciones de compilación	81
Crear un informe de pruebas con el ejemplo de AWS CLI	94
Ejemplo de Docker en una imagen personalizada	99

Ejemplo de Docker	101
Ejemplo de GitHub Enterprise Server	107
Ejemplo de filtros de webhooks y solicitudes de extracción de GitHub	113
Alojar la salida de la compilación en un bucket de S3	116
Versiones del entorno de ejecución en el ejemplo del archivo buildspec	118
Ejemplo de versión de origen	125
Ejemplo de un registro privado conAWS Secrets ManagerEjemplo de	128
Ejemplo de varios orígenes de entrada y varios artefactos de salida	129
Ejemplo para utilizar el control de versiones semánticas para asignar nombre a artefactos de compilación	132
Planificación de una compilación	135
Referencia de la especificación de compilación	136
Nombre de archivo y ubicación de almacenamiento de buildspec	137
Sintaxis de buildspec	137
Ejemplo de un archivo buildspec	152
Versiones de buildspec	154
Referencia de la especificación de compilación de Batch	154
Referencia de entornos de compilación	160
Imágenes de Docker proporcionadas por CodeBuild	161
Tipos de computación del entorno de compilación	169
Shells y comandos de los entornos de compilación	171
Variables de entorno en los entornos de compilación	172
Tareas en segundo plano en entornos de compilación	176
Crear localmente	176
Prerequisites	176
Configurar la imagen de compilación	176
Ejecutar el agente de CodeBuild	177
Recepción de notificaciones sobre nuevas versiones del agente de CodeBuild	178
Compatibilidad con VPC	180
Casos de uso	180
Permitir el acceso a Amazon VPC en sus proyectos de CodeBuild	180
Prácticas recomendadas para las VPC	181
Solución de problemas con la configuración de la VPC	182
Uso de puntos de enlace de la VPC	182
Antes de crear los puntos de enlace de la VPC	183
Creación de puntos de enlace de la VPC para CodeBuild	183
Creación de una política de puntos de enlace de la VPC para CodeBuild	184
Plantilla de VPC de AWS CloudFormation	184
Uso de un servidor proxy	188
Componentes necesarios para ejecutar CodeBuild en un servidor proxy	189
Ejecución de CodeBuild en un servidor proxy explícito	191
Ejecución de CodeBuild en un servidor proxy transparente	194
Ejecución de un administrador de paquetes y otras herramientas en un servidor proxy	195
Trabajar con proyectos de compilación y compilaciones	197
Trabajar con proyectos de compilación	197
Creación de un proyecto de compilación	197
Creación de una regla de notificación	223
Ver una lista de nombres de proyectos de compilación	225
Ver los detalles de un proyecto de compilación	227
Almacenamiento en caché de compilaciones	229
Crear desencadenadores de compilación	233
Editar desencadenadores de compilación	236
Webhooks	237
Cambiar la configuración de un proyecto de compilación	254
Eliminar un proyecto de compilación	268
Uso de proyectos compartidos	269
Etiquetado de un proyecto	273

Compilaciones Batch lotes	276
Proyectos de compilación pública	279
Trabajo con compilaciones	280
Ejecutar una compilación	281
Ver detalles de las compilaciones	289
Ver una lista de identificadores de compilación	292
Ver una lista de identificadores de compilación de un proyecto de compilación	294
Detener una compilación	297
Detener una compilación por lotes	298
Reintentar una compilación	299
Administrador de sesiones	300
Eliminación de compilaciones	304
Uso de informes de pruebas	305
Crear un informe de pruebas	306
Uso de grupos de informes	306
Crear un grupo de informes	307
Actualizar un grupo de informes	311
Especificar archivos de prueba	313
Especificar comandos de prueba	313
Nomenclatura de grupos de informes	314
Etiquetar un grupo de informes	314
Usar grupos de informes compartidos	319
Uso de informes	323
Uso de permisos de informes de pruebas	323
Crear un rol para los informes de pruebas	324
Permisos para las operaciones de informes de pruebas	325
Ejemplos de permisos de informes de pruebas	325
Consultar los informes de pruebas	326
Consultar los informes de pruebas de una compilación	326
Consultar los informes de pruebas de un grupo de informes	326
Consultar los informes de pruebas de una cuenta de AWS	327
Generación de informes de pruebas con marcos de pruebas	327
Generación de informes con Jasmine	327
Generación de informes con Jest	329
Generación de informes con pytest	330
Generación de informes con RSpec	330
Informes de cobertura de códigos	331
Crear un informe de cobertura de código	332
Registro y monitorización	333
Registrar llamadas a la API de AWS CodeBuild con AWS CloudTrail	333
AWS CodeBuildInformación en CloudTrail	333
Descripción de las entradas de los archivos de registro de AWS CodeBuild	334
Monitorización de AWS CodeBuild	335
Métricas de CloudWatch	336
Métricas de utilización de recursos de CloudWatch	338
Dimensiones de CloudWatch	339
Alarmas de CloudWatch	339
Métricas de CodeBuild	340
Métricas de utilización de recursos CodeBuild	345
Alarmas CodeBuild	350
Seguridad	351
Protección de datos	351
Cifrado de datos	352
Administración de claves	353
Privacidad de tráfico	353
Identity and Access Management	353

Authentication	354
Control de acceso	355
Información general sobre la administración del acceso	355
Uso de políticas basadas en identidades	358
Referencia de permisos de AWS CodeBuild	376
Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild	381
Visualización de recursos en la consola	384
Validación de conformidad	384
Resiliencia	385
Seguridad de la infraestructura	385
Acceso al proveedor de origen	385
Token de acceso a GitHub Enterprise Server	386
La contraseña de la aplicación Bitbucket	388
Temas avanzados	391
Configuración avanzada	391
Agregar permisos de acceso de CodeBuild a un grupo de IAM o usuario de IAM	391
Cree un rol de servicio de CodeBuild	396
Creación de una clave administrada por el cliente	400
Instalación y configuración de AWS CLI	402
Referencia de la línea de comandos	402
Referencia de los SDK y las herramientas de AWS	403
SDK y herramientas de AWS compatibles con AWS CodeBuild	403
Especificación del punto de enlace de	404
Especificación del punto de enlace de AWS CodeBuild (AWS CLI)	404
Especificación del punto de enlace de AWS CodeBuild (SDK de AWS)	405
Ejecutar CodeBuild directamente	406
Prerequisites	407
Ejecutar AWS CodeBuild directamente	407
Usar CodePipeline con CodeBuild	407
Prerequisites	408
Crear canalización (consola)	409
Cree una canalización (AWS CLI)	412
Agregar una acción de compilación	415
Agregar acción de prueba	418
Uso de CodeBuild con Jenkins	420
Configuración de Jenkins	420
Instalación del complemento	420
Uso del complemento	420
Uso de CodeBuild v	422
Integrar Codecov en un proyecto de compilación	422
aplicaciones sin servidor	425
Recursos relacionados	50
Solución de problemas	427
Apache Maven crea artefactos de referencia en un repositorio incorrecto	428
De forma predeterminada, los comandos de compilación se ejecutan como usuario raíz	429
Las compilaciones podrían registrar errores si los nombres de archivo tienen caracteres que no pertenecen al inglés (Estados Unidos)	429
Las compilaciones pueden registrar errores al obtener parámetros del almacén de parámetros de Amazon EC2	430
No se puede obtener acceso al filtro de ramificaciones en la consola CodeBuild	431
No se puede ver si la compilación se ha realizado correctamente o no	431
Estado de compilación no notificado al proveedor de origen	431
No se puede encontrar ni seleccionar la imagen base de la plataforma Windows Server Core 2019	432
Comandos anteriores de los archivos buildspec no reconocidos por comandos más recientes	432
Error: «Access denied (Acceso denegado)» al intentar descargar la caché	432
Error: «BUILD_CONTAINER_UNABLE_TO_PULL_IMAGE» al usar una imagen de compilación personalizada	433

Error: Error: «Build container found dead before complete the build. build container died by it was out of memory, or the Docker image is not supported (Se ha detectado que el contenedor de compilaciones está inactivo antes ErrorCode: 500"	434
Error: «No se puede conectar al demonio de Docker» al ejecutar una compilación	434
Error: «CodeBuild is not authorized to perform: sts:AssumeRole» al crear o actualizar un proyecto de compilación	435
Error: «Error al llamar a GetBucketAcl: El propietario del bucket ha cambiado o el rol de servicio ya no tiene permiso para llamar a s3:GetBucketAcl»	435
Error: «Error al cargar artefactos: Arn» no válido al ejecutar una compilación	436
Error: «El clon de Git falló: No se puede acceder 'your-repository-URL': Problema de certificado SSL: Certificado autofirmado»	436
Error: «The bucket you are are try to access must be address using the specified endpoint» al ejecutar una compilación	436
Error: «The policy's default version was not created by enhanced zero click role creation or not the most recent version created by enhanced zero click role creation»	437
Error: «Esta imagen de compilación requiere seleccionar al menos una versión de tiempo de ejecución»	437
Error: «EN COLA: INSUFFICIENT_SUSNET» cuando se produce un error en una compilación de una cola de compilaciones	438
Error: «Unable to download cache: RequestError: Error de envío de solicitud causado por: x509: Error al cargar las raíces del sistema y no se han proporcionado raíces»	439
Error: «Unable to download certificate from S3. AccessDenied"	439
Error: «Unable to locate credentials»	439
Error de tiempo de espera agotado de RequestError al ejecutar CodeBuild en un servidor proxy	440
El shell de Bourne (sh) debe existir en las imágenes de compilación	441
Advertencia: Al ejecutar una compilación, «Skipping install of runtimes. runtime version selection is not supported by this build image» (Si se omite la instalación de los entornos de ejecución	442
Error: «No se puede verificar la identidad de JobWorker»	442
No se ha podido iniciar la compilación	442
Acceder a los metadatos de GitHub en compilaciones almacenadas en caché localmente	442
AccessDenied: El propietario del bucket del grupo de informes no coincide con el propietario del bucket de S3...	443
Cuotas	444
Cuotas de servicio	444
Otros límites	445
Proyectos de compilación	445
Builds	445
Reports	445
Tags	446
Avisos de terceros de AWS CodeBuild para Windows	447
1) imagen base Docker: windowsservercore	447
2) Imagen de Docker base de Windows - choco	448
3) Imagen de Docker base de Windows - git —versión 2.16.2	448
4) Imagen de Docker base de Windows - microsoft-build-tools —versión 15.0.26320.2	449
5) Imagen de Docker base de Windows - nuget.commandline —version 4.5.1	451
7) Imagen de Docker base de Windows - netfx-4.6.2-devpack	452
8) Imagen de Docker base de Windows - visualfsharpools, v 4.0	453
9) Imagen de Docker base de Windows - netfx-pcl-reference-assemblies-4.6	453
10) Imagen de Docker base de Windows - visualcppbuildtools, v 14.0.25420.1	456
11) Imagen de Docker base de Windows - microsoft-windows-netfx3-ondemand-package.cab	458
12) Imagen de Docker base de Windows - dotnet-sdk	459
Historial de documentos	460
Actualizaciones anteriores	468
Glosario de AWS	477
.....	cdlxxviii

¿Qué es AWS CodeBuild?

AWS CodeBuild es un servicio de compilación completamente administrado en la nube. CodeBuild compila su código fuente, ejecuta pruebas unitarias y produce artefactos listos para implementarse. CodeBuild elimina la necesidad de aprovisionar, administrar y escalar sus propios servidores de compilación. Este servicio proporciona entornos de compilación preconfigurados para lenguajes de programación y herramientas de compilación populares como Apache Maven, Gradle, etc. También puede personalizar entornos de compilación en CodeBuild para utilizar sus propias herramientas de compilación. CodeBuild se amplía automáticamente para satisfacer el aumento de solicitudes de compilación.

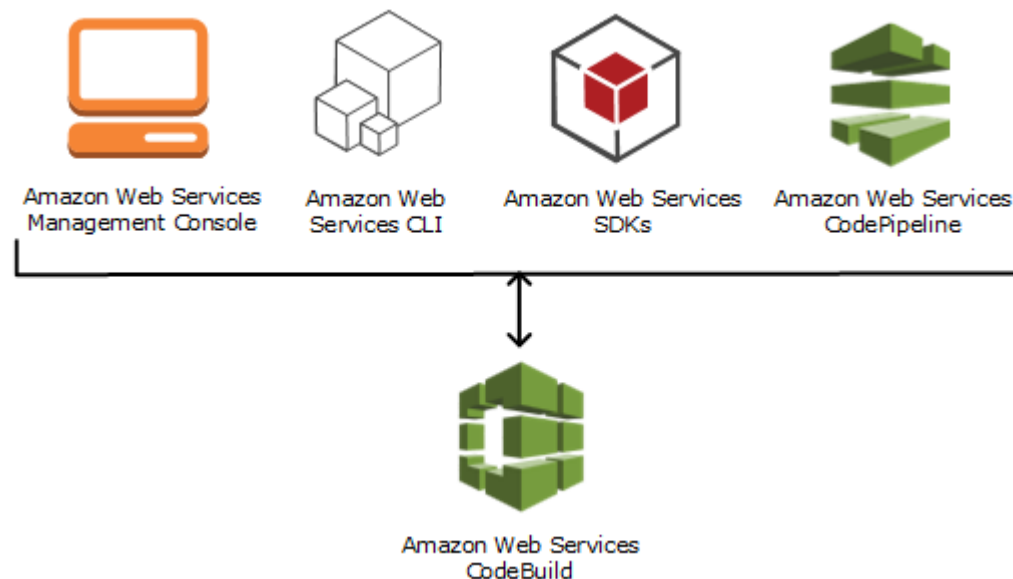
CodeBuild ofrece los siguientes beneficios:

- Totalmente administrado: CodeBuild elimina la necesidad de configurar, aplicar parches, actualizar y administrar sus propios servidores de compilación.
- Bajo demanda: CodeBuild se amplía a petición para satisfacer sus necesidades de compilación. Solo paga por el número de minutos de compilación que consuma.
- Fuera de la caja— CodeBuild proporciona entornos de compilación preconfigurados para los lenguajes de programación más populares. Lo único que tiene que hacer es apuntar a su script de compilación para iniciar su primera compilación.

Para obtener más información, consulte [AWS CodeBuild](#).

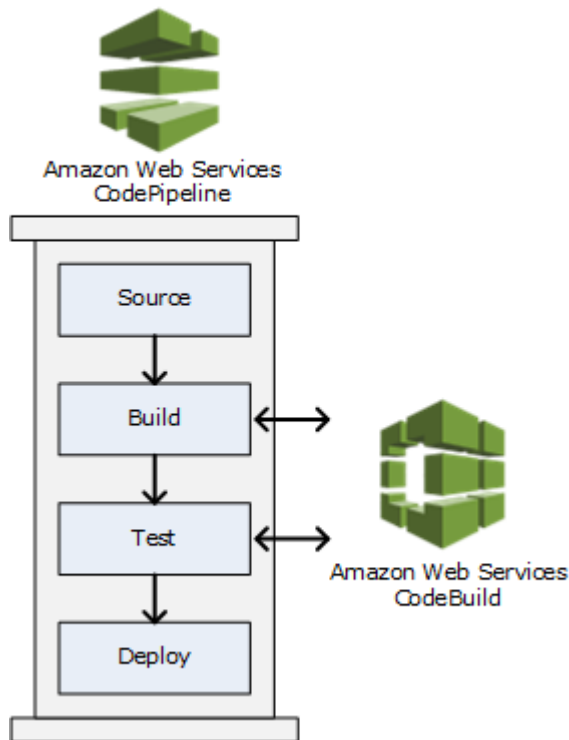
¿Cómo se ejecuta CodeBuild?

Puede utilizar el [AWS CodeBuild](#) o [AWS CodePipeline](#) para ejecutar CodeBuild. También puede automatizar la ejecución de CodeBuild utilizando la [AWS Command Line Interface \(AWS CLI\)](#) o el [AWS SDK](#) para los SDK.



Para ejecutar CodeBuild mediante la consola de CodeBuild, AWS CLI, o bien AWS Para los SDK, consulte [Ejecutar AWS CodeBuild directamente \(p. 406\)](#).

Como se muestra en el siguiente diagrama, puede añadir CodeBuild como acción de compilación o prueba a la fase de compilación o prueba de una canalización en AWS CodePipeline. AWS CodePipeline es un servicio de entrega continua que puede utilizar para modelar, visualizar y automatizar los pasos necesarios para lanzar su código. Esto incluye la compilación de su código. Una canalización es una construcción de flujo de trabajo que describe la forma en la que se someten los cambios en el código a un proceso de lanzamiento.



Para utilizar CodePipeline para crear una canalización y después añadir una acción de compilación o prueba de CodeBuild, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#). Para obtener más información acerca de CodePipeline, consulte la [AWS CodePipeline Guía del usuario de](#).

La consola CodeBuild también proporciona una manera rápida de buscar sus recursos, como repositorios, proyectos de compilación, aplicaciones de implementación y canalizaciones. Elija Go to resource (Ir al recurso) o pulse la tecla / y escriba el nombre del recurso. Se muestran todas las coincidencias en la lista. En las búsquedas, no se distingue entre mayúsculas y minúsculas. Solo puede ver los recursos para los que tiene permiso. Para obtener más información, consulte [Visualización de recursos en la consola \(p. 384\)](#).

Precios para CodeBuild

Para obtener información, consulte [Precios de CodeBuild](#).

¿Cómo puedo comenzar a usar CodeBuild?

Le recomendamos que siga los pasos que se describen a continuación:

1. Descripción Para obtener más información sobre CodeBuild leyendo la información de [Conceptos](#) (p. 3).
2. Experimento Con CodeBuild en un escenario de ejemplo siguiendo las instrucciones de [Primeros pasos con la consola](#) (p. 5).
3. Usar CodeBuild en sus propios escenarios siguiendo las instrucciones de [Planificación de una compilación](#) (p. 135).

Conceptos de AWS CodeBuild

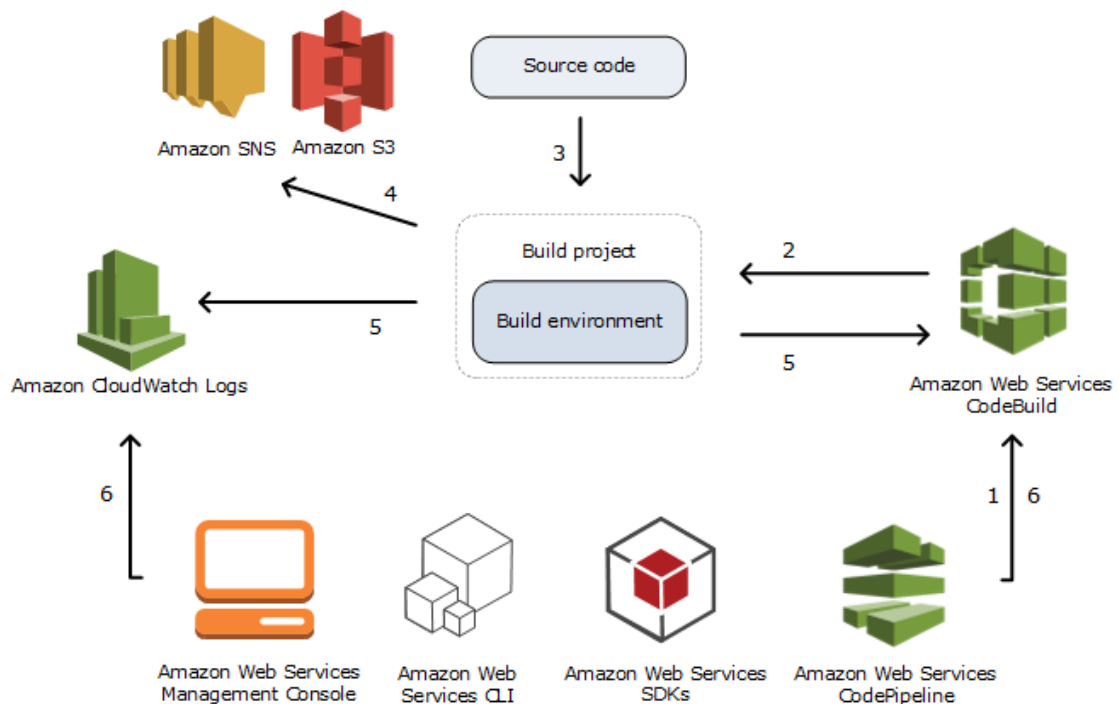
Los siguientes conceptos son importantes para comprender cómo funciona CodeBuild.

Temas

- [Funcionamiento de CodeBuild](#) (p. 3)
- [Pasos siguientes](#) (p. 4)

Funcionamiento de CodeBuild

En el siguiente diagrama se muestra lo que ocurre al ejecutar una compilación con CodeBuild:



1. Como entrada, debe proporcionar a CodeBuild un proyecto de compilación. A Proyecto de compilación incluye información sobre cómo ejecutar una compilación, incluido dónde se va a obtener el código fuente, qué entorno de compilación se va a usar, qué comandos de compilación se van a ejecutar y dónde se va a almacenar la salida de la compilación. A Entorno de compilación representa una combinación de sistema operativo, el tiempo de ejecución de un lenguaje de programación y herramientas que CodeBuild utiliza para ejecutar una compilación. Para obtener más información, consulte:
 - [Creación de un proyecto de compilación](#) (p. 197)

- [Referencia de entornos de compilación \(p. 160\)](#)
- 2. CodeBuild utiliza el proyecto de compilación para crear el entorno de compilación.
- 3. CodeBuild descarga el código fuente en el entorno de compilación y, a continuación, utiliza la especificación de compilación, tal y como se define en el proyecto de compilación, o incluida directamente en el código fuente. ASpeces una colección de comandos de compilación y opciones relacionadas, en formato YAML, que CodeBuild utiliza para ejecutar una compilación. Para obtener más información, consulte [Referencia de la especificación de compilación \(p. 136\)](#).
- 4. Si existe alguna salida de la compilación, el entorno de compilación la carga en un bucket de S3. El entorno de compilación también puede realizar las tareas que indique en la especificación de compilación (por ejemplo, enviar notificaciones de compilación a un tema de Amazon SNS). Para ver un ejemplo, consulte [Ejemplo de notificaciones de compilación \(p. 81\)](#).
- 5. Mientras se ejecuta la compilación, el entorno de compilación envía información a CodeBuild y Amazon CloudWatch Logs.
- 6. Mientras se ejecuta la compilación, puede usar elAWS CodeBuildconsola de,AWS CLI, o bienAWSdescarga los SDK para obtener información resumida sobre la compilación de CodeBuild e información detallada sobre la compilación de Amazon CloudWatch Logs. Si usaAWS CodePipelinePara ejecutar compilaciones, puede obtener información limitada sobre la compilación de CodePipeline.

Pasos siguientes

Ahora que ya sabe más sobre AWS CodeBuild, le recomendamos que siga estos pasos:

1. Experimentocon CodeBuild en un escenario de ejemplo siguiendo las instrucciones de[Primeros pasos con la consola \(p. 5\)](#).
2. UsarCodeBuild en sus propios escenarios siguiendo las instrucciones de[Planificación de una compilación \(p. 135\)](#).

Getting started with CodeBuild

En los siguientes tutoriales, utilizará AWS CodeBuild para crear una colección de archivos de entrada de código fuente de muestra en una versión implementable del código fuente.

Ambos tutoriales tienen la misma entrada y los mismos resultados, pero uno usa la consola de AWS CodeBuild y el otro, la AWS CLI.

Important

No es conveniente que utilice la cuenta raíz de AWS para completar este tutorial.

Empezar a trabajar con AWS CodeBuild utilizando la consola

En este tutorial, utilizará AWS CodeBuild para compilar una colección de archivos de entrada de código fuente de muestra (artefactos de entrada de compilación o entrada de compilación) en una versión desplegable del código fuente (artefacto de salida de compilación o salida de compilación). En concreto, indicará a CodeBuild que utilice Apache Maven, una herramienta de compilación común, para compilar un conjunto de archivos de clases Java en un archivo Java Archive (JAR). No necesita estar familiarizado con Apache Maven o Java para completar este tutorial.

Puede trabajar con CodeBuild a través de la consola de CodeBuild, AWS CodePipeline, el AWS CLI, o el AWS SDK. En este tutorial se muestra cómo usar la consola de CodeBuild. Para obtener más información sobre el uso de CodePipeline, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#). Para obtener más información sobre el uso de los SDK de AWS, consulte [Ejecutar CodeBuild directamente \(p. 406\)](#).

Important

Los pasos de este tutorial requieren que cree recursos (por ejemplo, un bucket de S3) que podrían generar cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon S3, AWS KMS y CloudWatch Logs. Para obtener más información, consulte [AWS CodeBuild Precios de](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), y [Precios de Amazon CloudWatch](#).

Steps

- [Paso 1: Crear el código fuente \(p. 6\)](#)
- [Paso 2: Cree el archivo buildspec \(p. 8\)](#)
- [Paso 3: Cree dos buckets de S3 \(p. 9\)](#)
- [Paso 4: Cargar el código fuente y el archivo buildspec \(p. 10\)](#)
- [Paso 5: Cree el proyecto de compilación \(p. 11\)](#)
- [Paso 6: Ejecute la compilación \(p. 12\)](#)
- [Paso 7: Ver información resumida sobre la compilación \(p. 13\)](#)
- [Paso 8: Ver información detallada sobre la compilación \(p. 13\)](#)

- [Paso 9: Obtener el artefacto de salida de la compil \(p. 14\)](#)
- [Paso 10: Eliminación del bucket de S3 \(p. 15\)](#)
- [Encapsulación \(p. 15\)](#)

Paso 1: Crear el código fuente

(Parte de: [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#))

En este paso, creará el código fuente que desea que CodeBuild compile en el bucket de salida. Este código fuente se compone de dos archivos de clases Java y un archivo Apache Maven Object Model (POM).

1. En un directorio vacío del equipo o la instancia local, cree esta estructura de directorios.

```
(root directory name)
  |-- src
    |-- main
    |   |-- java
    |-- test
    |   |-- java
```

2. Con el editor de texto que desee, cree este archivo, asígnele el nombre `MessageUtil.java` y guárdelo en el directorio `src/main/java`.

```
public class MessageUtil {
    private String message;

    public MessageUtil(String message) {
        this.message = message;
    }

    public String printMessage() {
        System.out.println(message);
        return message;
    }

    public String salutationMessage() {
        message = "Hi!" + message;
        System.out.println(message);
        return message;
    }
}
```

Este archivo de clases crea como resultado la cadena de caracteres que se la ha pasado. El constructor `MessageUtil` establece la cadena de caracteres. El método `printMessage` crea la salida. El método `salutationMessage` muestra `Hi!` seguido de la cadena de caracteres.

3. Cree este archivo, asígnele el nombre `TestMessageUtil.java` y guárdelo en el directorio `/src/test/java`.

```
import org.junit.Test;
import org.junit.Ignore;
import static org.junit.Assert.assertEquals;

public class TestMessageUtil {

    String message = "Robert";
    MessageUtil messageUtil = new MessageUtil(message);
```

```
@Test
public void testPrintMessage() {
    System.out.println("Inside testPrintMessage()");
    assertEquals(message,messageUtil.printMessage());
}

@Test
public void testSalutationMessage() {
    System.out.println("Inside testSalutationMessage()");
    message = "Hi!" + "Robert";
    assertEquals(message,messageUtil.salutationMessage());
}
}
```

Este archivo de clases establece la variable `message` de la clase `MessageUtil` en `Robert`. A continuación, comprueba si la variable `message` se ha establecido correctamente comprobando si las cadenas `Robert` y `Hi!Robert` aparecen en la salida.

4. Cree este archivo, asígnele el nombre `pom.xml` y guárdelo en el directorio raíz (nivel superior).

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/
maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.example</groupId>
  <artifactId>messageUtil</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging>
  <name>Message Utility Java Sample App</name>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.0</version>
      </plugin>
    </plugins>
  </build>
</project>
```

Apache Maven utiliza las instrucciones de este archivo para convertir los archivos `MessageUtil.java` y `TestMessageUtil.java` en un archivo denominado `messageUtil-1.0.jar` y, a continuación, ejecuta las pruebas especificadas.

En este punto, la estructura de directorios debería ser similar a la siguiente.

```
(root directory name)
|-- pom.xml
`-- src
    |-- main
    |   |-- java
    |       |-- MessageUtil.java
    `-- test
```

```
^-- java
^-- TestMessageUtil.java
```

Paso siguiente

[Paso 2: Cree el archivo buildspec \(p. 8\)](#)

Paso 2: Cree el archivo buildspec

(Paso anterior: [Paso 1: Crear el código fuente \(p. 6\)](#))

En este paso, creará un archivo de especificación de compilación. Abuildspeces un conjunto de comandos de compilación y opciones relacionadas, en formato YAML, que CodeBuild utiliza para ejecutar una compilación. Sin una especificación de compilación, CodeBuild no puede convertir correctamente la entrada de la compilación ni puede encontrar el artefacto de salida de la compilación en el entorno de compilación para cargarlo en el bucket de salida.

Cree este archivo, asígnele el nombre `buildspec.yml` y guárdelo en el directorio raíz (nivel superior).

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto11
  pre_build:
    commands:
      - echo Nothing to do in the pre_build phase...
  build:
    commands:
      - echo Build started on `date`
      - mvn install
  post_build:
    commands:
      - echo Build completed on `date`
artifacts:
  files:
    - target/messageUtil-1.0.jar
```

Important

Como una declaración de especificación de compilación debe ser una declaración YAML válida, los espacios de la declaración son importantes. Si el número de espacios de la declaración de especificación de compilación no coincide con esta declaración, la compilación podría producir un error inmediatamente. Puede utilizar un validador de YAML para comprobar si la declaración de especificación de compilación es una declaración YAML válida.

Note

En lugar de incluir un archivo de especificación de compilación en el código fuente, puede declarar los comandos de la compilación por separado cuando cree un proyecto de compilación. Esto resulta útil si desea compilar el código fuente con diferentes comandos de compilación sin actualizar el código fuente del repositorio cada vez. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

En esta declaración de especificación de compilación:

- `version` representa la versión del estándar de especificación de compilación que se va a usar. Esta declaración de especificación de compilación usa la última versión, `0.2`.

- `phases` representa las fases de compilación en las que puede indicar a CodeBuild que ejecute comandos. Estas fases de compilación se muestran aquí como `install`, `pre_build`, `build` y `post_build`. No puede cambiar los nombres de estas fases de compilación ni puede crear nombres de fases de compilación adicionales.

En este ejemplo, durante el `build` fase, CodeBuild ejecuta el `mvn install` comando. Este comando indica a Apache Maven que compile, pruebe y empaquete los archivos de clases Java compilados en un artefacto de salida de la compilación. En aras de una mayor exhaustividad, se incluyen algunos comandos `echo` en cada fase de compilación de este ejemplo. Cuando vea información de compilación detallada más adelante en este tutorial, el resultado de estos `echo` pueden ayudarle a comprender mejor cómo CodeBuild ejecuta los comandos y en qué orden. (Aunque en este ejemplo se incluyen todas las fases de compilación, no es necesario que incluya una fase de compilación si no piensa ejecutar ningún comando durante esa fase). Para cada fase de compilación, CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

- `artifacts` representa el conjunto de artefactos de salida de la compilación que CodeBuild carga en el bucket de salida. `files` representa los archivos que se incluyen en la salida de la compilación. CodeBuild carga el único `messageUtil-1.0.jar` archivo encontrado en el `target` relative directory en el entorno de compilación. El nombre del archivo `messageUtil-1.0.jar` y el nombre del directorio `target` se basan en la forma en que Apache Maven crea y almacena los artefactos de salida de la compilación para este ejemplo únicamente. En sus propias compilaciones, estos nombres de archivos y directorios son diferentes.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

En este punto, la estructura de directorios debería ser similar a la siguiente.

```
(root directory name)
|-- pom.xml
|-- buildspec.yml
`-- src
    |-- main
    |   |-- java
    |   |-- MessageUtil.java
    |-- test
    |   |-- java
    |   |-- TestMessageUtil.java
```

Paso siguiente

[Paso 3: Cree dos buckets de S3 \(p. 9\)](#)

Paso 3: Cree dos buckets de S3

(Paso anterior: [Paso 2: Cree el archivo buildspec \(p. 8\)](#))

Aunque puede usar un solo bucket para este tutorial, dos buckets hacen que sea más fácil ver de dónde proviene la entrada de compilación y hacia dónde va la salida de compilación.

- Uno de estos buckets (bucket de entrada) almacena la entrada de compilación. En este tutorial, el nombre de este depósito de entrada es `codebuild-region-ID-account-ID-input-bucket`, donde *region-ID* es la región de AWS del bucket y *account-ID* es su ID de cuenta de AWS.
- El otro bucket (bucket de salida) almacena la salida de la compilación. En este tutorial, el nombre de este bucket de salida es `codebuild-region-ID-account-ID-output-bucket`.

Si elige nombres diferentes para estos buckets, asegúrese de usarlos en este tutorial.

Estos dos buckets están en la misma región de AWS que sus compilaciones. Por ejemplo, si indica a CodeBuild que ejecute una compilación en la región EE.UU. Este (Ohio), estos depósitos también deben estar en la región EE.UU. Este (Ohio).

Para obtener más información, consulte [Creación de un bucket](#) en la Guía del usuario de Amazon Simple Storage Service.

Note

Aunque CodeBuild también es compatible con la entrada de compilación almacenada en repositorios de CodeCommit, GitHub y Bitbucket, este tutorial no muestra cómo usarlos. Para obtener más información, consulte [Planificación de una compilación](#) (p. 135).

Paso siguiente

[Paso 4: Cargar el código fuente y el archivo buildspec](#) (p. 10)

Paso 4: Cargar el código fuente y el archivo buildspec

(Paso anterior: [Paso 3: Cree dos buckets de S3](#) (p. 9))

En este paso, añadirá el código fuente y el archivo de especificación de compilación al bucket de entrada.

Con la utilidad zip del sistema operativo, cree un archivo llamado `MessageUtil.zip` que incluya `MessageUtil.java`, `TestMessageUtil.java`, `pom.xml` y `buildspec.yml`.

La estructura de directorios del archivo `MessageUtil.zip` debe ser similar a la siguiente.

```
MessageUtil.zip
|-- pom.xml
|-- buildspec.yml
`-- src
    |-- main
    |   |-- java
    |   |-- MessageUtil.java
    |-- test
    |   |-- java
    |   |-- TestMessageUtil.java
```

Important

No incluya el directorio (*root directory name*), solo los directorios y archivos del directorio (*root directory name*).

Cargue el archivo `MessageUtil.zip` identificado en el bucket de entrada denominado `codebuild-region-ID-account-ID-input-bucket`.

Important

Para los repositorios de CodeCommit, GitHub y Bitbucket, por convención, debe almacenar un archivo de especificación de compilación llamado `buildspec.yml` en la raíz (nivel superior) de cada repositorio o incluir la declaración de especificación de compilación como parte de la definición del proyecto de compilación. No cree un archivo ZIP que contenga el código fuente del repositorio y el archivo de especificación de compilación. Solo para entradas de compilación almacenadas en buckets de S3, debe crear un archivo ZIP que contenga el código fuente y convencionalmente, un archivo de especificación de

compilación denominado `buildspec.yml` en la raíz (nivel superior) o incluya la declaración de especificaciones de compilación como parte de la definición del proyecto de compilación. Si desea utilizar otro nombre para el archivo de especificación de compilación, o si quiere hacer referencia a una especificación de compilación en una ubicación distinta de la raíz, puede especificar una invalidación de la especificación de compilación como parte de la definición del proyecto de compilación. Para obtener más información, consulte [Nombre de archivo y ubicación de almacenamiento de buildspec](#) (p. 137).

Paso siguiente

[Paso 5: Cree el proyecto de compilación](#) (p. 11)

Paso 5: Cree el proyecto de compilación

(Paso anterior: [Paso 4: Cargar el código fuente y el archivo buildspec](#) (p. 10))

En este paso, creará un proyecto de compilación que usará AWS CodeBuild para ejecutar la compilación. Un proyecto de compilación incluye información sobre cómo ejecutar una compilación, incluido dónde obtener el código fuente, qué entorno de compilación usar, qué comandos de compilación se van a ejecutar y dónde almacenar la salida de la compilación. Un entorno de compilación representa una combinación de sistema operativo, tiempo de ejecución de un lenguaje de programación y herramientas que CodeBuild utiliza para ejecutar una compilación. El entorno de compilación se expresa como una imagen de Docker. Para obtener más información, consulte [Descripción general de Docker](#) en la página de documentos de Docker.

Para este entorno de compilación, indicará a CodeBuild que utilice una imagen de Docker que contenga una versión del Kit de desarrollo de Java (JDK) y Apache Maven.

Para crear el proyecto de compilación

1. Inicie sesión en AWS Management Console y abra el icono AWS CodeBuild Consola de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Usar AWS selector de regiones para elegir un AWS Región en la que se admite CodeBuild. Para obtener más información, consulte [AWS CodeBuild Cuotas y puntos de enlace de](#) en la Referencia general de Amazon Web Services.
3. Si aparece una página de información de CodeBuild, elija Crear un proyecto de compilación. En caso contrario, en el panel de navegación, expanda Build, elija Proyectos de compilación, a continuación, elija Crear un proyecto de compilación.
4. En la página Create build project (Crear proyecto de compilación), en Project configuration (Configuración del proyecto), en Project name (Nombre de proyecto), escriba un nombre para este proyecto de compilación (en este ejemplo, `codebuild-demo-project`). Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. Si elige otro nombre, asegúrese de utilizarlo durante todo el tutorial.

Note

En la página Crear un proyecto de compilación, es posible que vea un mensaje de error similar al siguiente: No está autorizado para realizar esta operación.. Lo más probable es que se deba a que inició sesión en el AWS Management Console como usuario de IAM que no tiene permiso para crear proyectos de compilación. Para solucionarlo, cierre la sesión en AWS Management Console, a continuación, inicie sesión de nuevo con credenciales que pertenezcan a alguna de las siguientes entidades de IAM:

- Un usuario de IAM administrador en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la IAM User Guide.
- Un usuario de IAM en su AWS cuenta con el `AWSCodeBuildAdminAccess`, `AmazonS3ReadOnlyAccess`, y `IAMFullAccess` políticas

administradas asociadas a ese usuario de IAM o a un grupo de IAM al que pertenece el usuario de IAM. Si no tiene un usuario o grupo de IAM en su AWS con estos permisos y no puede agregar estos permisos a su usuario o grupo de IAM, póngase en contacto con su AWS administrador de cuentas para obtener ayuda. Para obtener más información, consulte [Políticas administradas \(predefinidas\) por AWS para AWS CodeBuild \(p. 360\)](#).

Ambas opciones incluyen permisos de administrador que le permiten crear un proyecto de compilación y completar este tutorial. Le recomendamos que utilice siempre los permisos mínimos necesarios para realizar la tarea. Para obtener más información, consulte [Referencia de permisos de AWS CodeBuild \(p. 376\)](#).

5. En Fuente, para Proveedor de origen, elige Amazon S3.
6. Para Bucket, elige Codebuild-**ID de región-account-id**-cubo de entrada.
7. En S3 object key (Clave de objeto de S3), escriba **MessageUtil.zip**.
8. En Environment (Entorno), para Environment image (Imagen de entorno), deje Managed image (Imagen administrada) seleccionado.
9. En Operating system (Sistema operativo), elija Amazon Linux 2.
10. En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
11. Para Image (Imagen), elige aws/codebuild/amazonlinux2-x86_64-standard:3.0.
12. En Service role (Rol de servicio), deje la opción New service role (Nuevo rol de servicio) seleccionada y no haga ningún cambio en Role name (Nombre de rol).
13. En Buildspec, deje Use a buildspec file (Usar un archivo buildspec) seleccionado.
14. En Artifacts, para Tipo, elige Amazon S3.
15. Para Nombre del bucket, elige Codebuild-**ID de región-account-id**-cubo de salida.
16. Deje Name (Nombre) y Path (Ruta) en blanco.
17. Elija Create build project (Crear proyecto de compilación).

Paso siguiente

[Paso 6: Ejecute la compilación \(p. 12\)](#)

Paso 6: Ejecute la compilación

(Paso anterior: [Paso 5: Cree el proyecto de compilación \(p. 11\)](#))

En este paso, indicará a AWS CodeBuild que ejecute la compilación con la configuración del proyecto de compilación.

Para ejecutar la compilación

1. Abra el icono AWS CodeBuild Consola de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. En la lista de proyectos de compilación, elija codebuild-demo-project, a continuación, elija Iniciar compilación. La compilación comienza inmediatamente.

Paso siguiente

[Paso 7: Ver información resumida sobre la compilación \(p. 13\)](#)

Paso 7: Ver información resumida sobre la compilación

(Paso anterior: [Paso 6: Ejecute la compilación \(p. 12\)](#))

En este paso, verá información resumida sobre el estado de la compilación.

Para ver la información resumida de la compilación

1. Si el archivo `decodebuild-demo-project:<build-ID>` no se muestra la, en la barra de navegación, elija **Historial de compilación**. A continuación, en la lista de proyectos de compilación, para **Proyecto de**, elige **Ejecutar compilación** `link forcodebuild-demo-project`. Debe haber un solo enlace coincidente. (Si ya ha completado este tutorial antes, elija el vínculo con el valor más reciente en la columna **Completed (Completado)**).
2. En la página **Estado de la compilación** `page`, en **Detalles de la fase**, deben mostrarse las siguientes fases de compilación, con **Succeeded** en la **Estado** `column`:
 - SUBMITTED
 - QUEUED
 - PROVISIONING
 - DOWNLOAD_SOURCE
 - INSTALL
 - PRE_BUILD
 - BUILD
 - POST_BUILD
 - UPLOAD_ARTIFACTS
 - FINALIZING
 - COMPLETED

En **Build Status (Estado de la compilación)**, debería mostrarse **Succeeded (Realizado correctamente)**.

Si en su lugar aparece **In Progress (En curso)** elija el botón de actualizar.

3. Junto a cada fase de compilación, el valor **Duration (Duración)** indica cuánto tiempo ha tardado la fase de compilación. El valor **End time (Hora de finalización)** indica que esa fase de compilación ha terminado.

Paso siguiente

[Paso 8: Ver información detallada sobre la compilación \(p. 13\)](#)

Paso 8: Ver información detallada sobre la compilación

(Paso anterior: [Paso 7: Ver información resumida sobre la compilación \(p. 13\)](#))

En este paso, verá información detallada acerca de su compilación en CloudWatch Logs.

Note

Para proteger la información confidencial, lo siguiente está oculto en los registros de CodeBuild:

- ID de clave de acceso de AWS. Para obtener más información, consulte [Administración de las claves de acceso de los usuarios de IAM](#) en la AWS Identity and Access Management Guía del usuario de.
- Cadenas especificadas mediante el almacén de parámetros. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial sobre la consola del almacén de parámetros de Systems Manager](#) en la Guía del usuario de Amazon EC2 Systems Manager.
- Cadenas especificadas mediante AWS Secrets Manager. Para obtener más información, consulte [Administración de claves \(p. 353\)](#).

Para ver información detallada sobre la compilación

1. Con la página de detalles de la compilación mostrada del paso anterior, se muestran las últimas 10,000 líneas del log de compilación en Build logs. Para ver el log de compilación completo en CloudWatch Logs, elija la [Ver el registro completo](#) link.
2. En la secuencia de logs de CloudWatch Logs, puede examinar los eventos de log. De forma predeterminada, solo se muestra el último conjunto de eventos de log. Para ver eventos de log anteriores, desplácese hasta el principio de la lista.
3. En este tutorial, la mayoría de los eventos de log contienen información detallada acerca de la descarga e instalación de CodeBuild de archivos de dependencias de la compilación en su entorno de compilación, que posiblemente no le interese. Puede usar el cuadro Filter events para reducir la información que se muestra. Por ejemplo, si escribe "[INFO]" en el cuadro Filter events (Filtrar eventos), solo se muestran los eventos que contienen [INFO]. Para obtener más información, consulte [Sintaxis de patrones y filtros](#) en la Guía del usuario de Amazon CloudWatch.

Paso siguiente

[Paso 9: Obtener el artefacto de salida de la compil \(p. 14\)](#)

Paso 9: Obtener el artefacto de salida de la compil

(Paso anterior: [Paso 8: Ver información detallada sobre la compilación \(p. 13\)](#))

En este paso, obtienes el `messageUtil-1.0.jar` archivo que CodeBuild compiló y cargó en el bucket de salida.

Puede usar la consola de CodeBuild o la consola de Amazon S3 para realizar este paso.

Para obtener el artefacto de salida de la compilación (consola de AWS CodeBuild)

1. Con la consola de CodeBuild todavía abierta y la página de detalles de la compilación mostrada del paso anterior, elija la [Detalles del compilación](#) y desplácese hacia abajo hasta la [Artifacts](#) sección.

Note

Si no se muestra la página de detalles de la compilación, en la barra de navegación, elija [Historial de compilación](#), a continuación, elija [Ejecutar compilación](#) link.

2. El enlace a la carpeta Amazon S3 se encuentra en la [Ubicación de carga de artefactos](#). Este enlace abre la carpeta en Amazon S3 donde se encuentra el `messageUtil-1.0.jar` archivo artefacto de salida de compilación.

Para obtener el artefacto de salida de la compilación (consola de Amazon S3)

1. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.

2. Abrir `codebuild-region-ID-account-ID-output-bucket`.
3. Abra la carpeta `codebuild-demo-project`.
4. Abra la carpeta `target`, donde se encuentra el archivo de artefactos de salida de compilación `messageUtil-1.0.jar`.

Paso siguiente

[Paso 10: Eliminación del bucket de S3 \(p. 15\)](#)

Paso 10: Eliminación del bucket de S3

(Paso anterior: [Paso 9: Obtener el artefacto de salida de la compil \(p. 14\)](#))

Para evitar cargos continuos a suAWS, puede eliminar los depósitos de entrada y salida utilizados en este tutorial. Para obtener instrucciones, consulte[Eliminación o vaciado de un bucket](#) en laGuía del usuario de Amazon Simple Storage Service.

Si utiliza el usuario de IAM o un usuario de IAM administrador de para eliminar estos bucket, el usuario debe tener más permisos de acceso. Añada la siguiente instrucción entre los marcadores (**### BEGIN ADDING STATEMENT HERE ###** y **### END ADDING STATEMENTS HERE ###**) a una política de acceso existente para el usuario.

Las elipsis (...) de esta declaración se usan para abreviar. No elimine las instrucciones de la política de acceso existente. No escriba estos puntos suspensivos en la política.

```
{
  "Version": "2012-10-17",
  "Id": "...",
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Effect": "Allow",
      "Action": [
        "s3:DeleteBucket",
        "s3:DeleteObject"
      ],
      "Resource": "*"
    }
    ### END ADDING STATEMENT HERE ###
  ]
}
```

Paso siguiente

[Encapsulación \(p. 15\)](#)

Encapsulación

En este tutorial, se ha usado AWS CodeBuild para construir un conjunto de archivos de clase Java en un archivo JAR. A continuación, ha consultado los resultados de la compilación.

Ahora puede intentar usar CodeBuild en sus propios escenarios. Siga las instrucciones en [Planificación de una compilación \(p. 135\)](#). Si piensa que aún no está preparado, tal vez desee intentar compilar algunos de los ejemplos. Para obtener más información, consulte [Muestras \(p. 31\)](#).

Introducción a AWS CodeBuild utilizando la AWS CLI

En este tutorial, usará AWS CodeBuild para compilar una selección de archivos de entrada de código fuente de muestra (llamados artefactos de entrada de la compilación o entrada de la compilación) en una versión implementable del código fuente (llamada artefacto de salida de compilación o salida de compilación). En concreto, indicará a CodeBuild que use Apache Maven, una herramienta de compilación común, para compilar un conjunto de archivos de clases Java en un archivo Java Archive (JAR). No necesita estar familiarizado con Apache Maven o Java para completar este tutorial.

Puede trabajar con CodeBuild a través de la consola de CodeBuild, AWS CodePipeline, el AWS CLI, o el AWS SDK. En este tutorial se muestra cómo usar CodeBuild con la AWS CLI. Para obtener más información sobre el uso de CodePipeline, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#). Para obtener más información sobre el uso de los SDK de AWS, consulte [Ejecutar CodeBuild directamente \(p. 406\)](#).

Important

Los pasos de este tutorial requieren que cree recursos (por ejemplo, un bucket de S3) que podrían generar cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon S3, AWS KMS y CloudWatch Logs. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), y [Precios de Amazon CloudWatch](#).

Steps

- [Paso 1: Crear el código fuente \(p. 16\)](#)
- [Paso 2: Cree el archivo buildspec \(p. 18\)](#)
- [Paso 3: Cree dos buckets de S3 \(p. 20\)](#)
- [Paso 4: Cargar el código fuente y el archivo buildspec \(p. 21\)](#)
- [Paso 5: Cree el proyecto de compilación \(p. 21\)](#)
- [Paso 6: Ejecute la compilación \(p. 24\)](#)
- [Paso 7: Ver información resumida sobre la compilación \(p. 26\)](#)
- [Paso 8: Ver información detallada sobre la compilación \(p. 27\)](#)
- [Paso 9: Obtener el artefacto de salida de la compilación \(p. 29\)](#)
- [Paso 10: Eliminación del bucket de S3 \(p. 30\)](#)
- [Encapsulación \(p. 30\)](#)

Paso 1: Crear el código fuente

(Parte de: [Introducción a AWS CodeBuild utilizando la AWS CLI \(p. 16\)](#))

En este paso, creará el código fuente que desea que CodeBuild compile en el bucket de salida. Este código fuente se compone de dos archivos de clases Java y un archivo Apache Maven Object Model (POM).

1. En un directorio vacío del equipo o la instancia local, cree esta estructura de directorios.

`(root directory name)`

```
  |-- src
    |-- main
    |   |-- java
    |-- test
    |   |-- java
```

2. Con el editor de texto que desee, cree este archivo, asígnele el nombre `MessageUtil.java` y guárdelo en el directorio `src/main/java`.

```
public class MessageUtil {
    private String message;

    public MessageUtil(String message) {
        this.message = message;
    }

    public String printMessage() {
        System.out.println(message);
        return message;
    }

    public String salutationMessage() {
        message = "Hi!" + message;
        System.out.println(message);
        return message;
    }
}
```

Este archivo de clases crea como resultado la cadena de caracteres que se la ha pasado. El constructor `MessageUtil` establece la cadena de caracteres. El método `printMessage` crea la salida. El método `salutationMessage` muestra `Hi!` seguido de la cadena de caracteres.

3. Cree este archivo, asígnele el nombre `TestMessageUtil.java` y guárdelo en el directorio `/src/test/java`.

```
import org.junit.Test;
import org.junit.Ignore;
import static org.junit.Assert.assertEquals;

public class TestMessageUtil {

    String message = "Robert";
    MessageUtil messageUtil = new MessageUtil(message);

    @Test
    public void testPrintMessage() {
        System.out.println("Inside testPrintMessage()");
        assertEquals(message, messageUtil.printMessage());
    }

    @Test
    public void testSalutationMessage() {
        System.out.println("Inside testSalutationMessage()");
        message = "Hi!" + "Robert";
        assertEquals(message, messageUtil.salutationMessage());
    }
}
```

Este archivo de clases establece la variable `message` de la clase `MessageUtil` en `Robert`. A continuación, comprueba si la variable `message` se ha establecido correctamente comprobando si las cadenas `Robert` y `Hi!Robert` aparecen en la salida.

4. Cree este archivo, asígnele el nombre `pom.xml` y guárdelo en el directorio raíz (nivel superior).


```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/
maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.example</groupId>
  <artifactId>messageUtil</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging>
  <name>Message Utility Java Sample App</name>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.0</version>
      </plugin>
    </plugins>
  </build>
</project>
```

Apache Maven utiliza las instrucciones de este archivo para convertir los archivos `MessageUtil.java` y `TestMessageUtil.java` en un archivo denominado `messageUtil-1.0.jar` y, a continuación, ejecuta las pruebas especificadas.

En este punto, la estructura de directorios debería ser similar a la siguiente.

```
(root directory name)
|-- pom.xml
`-- src
    |-- main
    |   |-- java
    |   |   |-- MessageUtil.java
    |-- test
    |   |-- java
    |   |   |-- TestMessageUtil.java
```

Paso siguiente

[Paso 2: Cree el archivo buildspec \(p. 18\)](#)

Paso 2: Cree el archivo buildspec

(Paso anterior: [Paso 1: Crear el código fuente \(p. 16\)](#))

En este paso, creará un archivo de especificación de compilación. Abuildspeces un conjunto de comandos de compilación y opciones de configuración relacionadas, en formato YAML, que CodeBuild utiliza para ejecutar una compilación. Sin una especificación de compilación, CodeBuild no puede convertir correctamente la entrada de la compilación en la salida de la compilación ni puede encontrar el artefacto de salida de la compilación en el entorno de compilación que desea cargarlo en el bucket

Cree este archivo, asígnele el nombre `buildspec.yml` y guárdelo en el directorio raíz (nivel superior).

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto11
  pre_build:
    commands:
      - echo Nothing to do in the pre_build phase...
  build:
    commands:
      - echo Build started on `date`
      - mvn install
  post_build:
    commands:
      - echo Build completed on `date`
artifacts:
  files:
    - target/messageUtil-1.0.jar
```

Important

Como una declaración de especificación de compilación debe ser una declaración YAML válida, los espacios de la declaración son importantes. Si el número de espacios de la declaración de especificación de compilación no coincide con esta declaración, la compilación podría producir un error inmediatamente. Puede utilizar un validador de YAML para comprobar si la declaración de especificación de compilación es una declaración YAML válida.

Note

En lugar de incluir un archivo de especificación de compilación en el código fuente, puede declarar los comandos de la compilación por separado cuando cree un proyecto de compilación. Esto resulta útil si desea compilar el código fuente con diferentes comandos de compilación sin actualizar el código fuente del repositorio cada vez. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

En esta declaración de especificación de compilación:

- `version` representa la versión del estándar de especificación de compilación que se va a usar. Esta declaración de especificación de compilación usa la última versión, `0.2`.
- `phases` representa las fases de compilación en las que puede indicar a CodeBuild que ejecute comandos. Estas fases de compilación se muestran aquí como `install`, `pre_build`, `build` y `post_build`. No puede cambiar los nombres de estas fases de compilación ni puede crear nombres de fases de compilación adicionales.

En este ejemplo, durante la `build` fase, CodeBuild ejecuta el `mvn install` comando. Este comando indica a Apache Maven que compile, pruebe y empaquete los archivos de clases Java compilados en un artefacto de salida de la compilación. En aras de una mayor exhaustividad, se incluyen algunos comandos `echo` en cada fase de compilación de este ejemplo. Cuando vea información detallada sobre la compilación más adelante en este tutorial, el resultado de estos `echo` pueden ayudarle a comprender mejor cómo CodeBuild ejecuta los comandos y en qué orden. (Aunque en este ejemplo se incluyen todas las fases de compilación, no es necesario que incluya una fase de compilación si no piensa ejecutar ningún comando durante esa fase). Para cada fase de compilación, CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

- `artifacts` representa el conjunto de artefactos de salida de la compilación que CodeBuild carga en el bucket de salida. `files` representa los archivos que se incluyen en la salida de la compilación. CodeBuild carga el único `messageUtil-1.0.jar` archivo encontrado en el `target` relativo en el entorno

de compilación. El nombre del archivo `messageUtil-1.0.jar` y el nombre del directorio `target` se basan en la forma en que Apache Maven crea y almacena los artefactos de salida de la compilación para este ejemplo únicamente. En sus propias compilaciones, estos nombres de archivos y directorios son diferentes.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

En este punto, la estructura de directorios debería ser similar a la siguiente.

```
(root directory name)
|-- pom.xml
|-- buildspec.yml
`-- src
    |-- main
    |   |-- java
    |   |-- MessageUtil.java
    |-- test
    |   |-- java
    |   |-- TestMessageUtil.java
```

Paso siguiente

[Paso 3: Cree dos buckets de S3 \(p. 20\)](#)

Paso 3: Cree dos buckets de S3

(Paso anterior: [Paso 2: Cree el archivo buildspec \(p. 18\)](#))

Aunque puede usar un solo bucket para este tutorial, dos buckets hacen que sea más fácil ver de dónde proviene la entrada de compilación y hacia dónde va la salida de compilación.

- Uno de estos buckets (bucket de entrada) almacena la entrada de compilación. En este tutorial, el nombre de este depósito de entrada es `codebuild-region-ID-account-ID-input-bucket`, donde *region-ID* es la región de AWS del bucket y *account-ID* es su ID de cuenta de AWS.
- El otro bucket (bucket de salida) almacena la salida de la compilación. En este tutorial, el nombre de este bucket de salida es `codebuild-region-ID-account-ID-output-bucket`.

Si elige nombres diferentes para estos buckets, asegúrese de usarlos en este tutorial.

Estos dos buckets están en la misma región de AWS que sus compilaciones. Por ejemplo, si indica a CodeBuild que ejecute una compilación en la región EE.UU. Este (Ohio), estos bucket también deben estar en la región EE.UU. Este (Ohio).

Para obtener más información, consulte [Creación de un bucket](#) en la Amazon Simple Storage Service User Guide.

Note

Aunque CodeBuild permite también crear una entrada de compilación almacenada en los repositorios de CodeCommit, GitHub y Bitbucket, este tutorial no muestra cómo usarlos. Para obtener más información, consulte [Planificación de una compilación \(p. 135\)](#).

Paso siguiente

[Paso 4: Cargar el código fuente y el archivo buildspec \(p. 21\)](#)

Paso 4: Cargar el código fuente y el archivo buildspec

(Paso anterior: [Paso 3: Cree dos buckets de S3 \(p. 20\)](#))

En este paso, añadirá el código fuente y el archivo de especificación de compilación al bucket de entrada.

Con la utilidad zip del sistema operativo, cree un archivo llamado `MessageUtil.zip` que incluya `MessageUtil.java`, `TestMessageUtil.java`, `pom.xml` y `buildspec.yml`.

La estructura de directorios del archivo `MessageUtil.zip` debe ser similar a la siguiente.

```
MessageUtil.zip
|-- pom.xml
|-- buildspec.yml
|-- src
|   |-- main
|   |   |-- java
|   |   |   |-- MessageUtil.java
|   |-- test
|   |   |-- java
|   |   |   |-- TestMessageUtil.java
```

Important

No incluya el directorio (*root directory name*), solo los directorios y archivos del directorio (*root directory name*).

Cargue el archivo `MessageUtil.zip` identificado en el bucket de entrada denominado `codebuild-region-ID-account-ID-input-bucket`.

Important

Para los repositorios de CodeCommit, GitHub y Bitbucket, por convención, debe almacenar un archivo de especificación de compilación llamado `buildspec.yml` en la raíz (nivel superior) de cada repositorio o incluir la declaración de especificación de compilación como parte de la definición del proyecto de compilación. No cree un archivo ZIP que contenga el código fuente del repositorio y el archivo de especificación de compilación.

Solo para entradas de compilación almacenadas en buckets de S3, debe crear un archivo ZIP que contenga el código fuente y convencionalmente, un archivo de especificación de compilación denominado `buildspec.yml` en la raíz (nivel superior) o incluya la declaración de especificaciones de compilación como parte de la definición del proyecto de compilación.

Si desea utilizar otro nombre para el archivo de especificación de compilación, o si quiere hacer referencia a una especificación de compilación en una ubicación distinta de la raíz, puede especificar una invalidación de la especificación de compilación como parte de la definición del proyecto de compilación. Para obtener más información, consulte [Nombre de archivo y ubicación de almacenamiento de buildspec \(p. 137\)](#).

Paso siguiente

[Paso 5: Cree el proyecto de compilación \(p. 21\)](#)

Paso 5: Cree el proyecto de compilación

(Paso anterior: [Paso 4: Cargar el código fuente y el archivo buildspec \(p. 21\)](#))

En este paso, creará un proyecto de compilación que usará AWS CodeBuild para ejecutar la compilación. Aproximadamente incluye información sobre cómo ejecutar una compilación, incluido dónde obtener

el código fuente, qué entorno de compilación usar, qué comandos de compilación se van a ejecutar y dónde almacenar la salida de la compilación. Un entorno de compilación representa una combinación de sistema operativo, tiempo de ejecución de un lenguaje de programación y herramientas que CodeBuild utiliza para ejecutar una compilación. El entorno de compilación se expresa como una imagen de Docker. Para obtener más información, consulte [Descripción general de Docker](#) en la página de documentos de Docker.

En este entorno de compilación, indicará a CodeBuild que use una imagen de Docker que contenga una versión del Kit de desarrollo de Java (JDK) y Apache Maven.

Para crear el proyecto de compilación

1. Utilice la AWS CLI para ejecutar el comando `create-project`:

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

En el resultado se muestran datos con formato JSON. Copie los datos en un archivo denominado `create-project.json` en la ubicación del equipo o instancia local en la que haya instalado la AWS CLI. Si ha elegido usar un nombre de archivo diferente, asegúrese de usarlo en este tutorial.

Modifique los datos copiados de modo que sigan este formato y, a continuación, guarde los resultados:

```
{
  "name": "codebuild-demo-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
  },
  "artifacts": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-output-bucket"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:4.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "serviceIAMRole"
}
```

Reemplazar *serviceIAMRole* con el nombre de recurso de Amazon (ARN) de un rol de servicio de CodeBuild (por ejemplo, `arn:aws:iam::account-ID:role/role-name`). Para crear uno, consulte [Cree un rol de servicio de CodeBuild](#) (p. 396).

En estos datos:

- `name` representa un identificador obligatorio para este proyecto de compilación (en este ejemplo, `codebuild-demo-project`). Los nombres de los proyectos de compilación deben ser únicos en todos los proyectos de compilación de su cuenta.
- Para `source`, `type` es un valor obligatorio que representa el tipo de repositorio del código fuente (en este ejemplo, `S3` para un bucket de Amazon S3).
- En `source`, `location` representa la ruta del código fuente (en este ejemplo, el nombre del bucket de entrada seguido del nombre del archivo ZIP).
- Para `artifacts`, `type` es un valor obligatorio que representa el tipo de repositorio del artefacto de salida de la compilación (en este ejemplo, `S3` para un bucket de Amazon S3).
- En `artifacts`, `location` representa el nombre del bucket de salida que ha creado o identificado anteriormente (en este ejemplo, `codebuild-region-ID-account-ID-output-bucket`).

- `Paraenvironment, type` es un valor obligatorio que representa el tipo de entorno de compilación (en este ejemplo, `LINUX_CONTAINER`).
- `Paraenvironment, image` es un valor obligatorio que representa el nombre de la imagen de Docker y la combinación de etiquetas que usa este proyecto de compilación, tal y como se especifica en el tipo de repositorio de imágenes de Docker (en este ejemplo, `aws/codebuild/standard:4.0` para una imagen Docker en el repositorio de imágenes de CodeBuild Docker). `aws/codebuild/standard` es el nombre de la imagen de Docker. `4.0` es la etiqueta de la imagen de Docker.

Para encontrar más imágenes de Docker que pueda utilizar en sus escenarios, consulte [Referencia de entornos de compilación](#) (p. 160).

- `Paraenvironment, computeType` es un valor obligatorio que representa los recursos informáticos que usa CodeBuild (en este ejemplo, `BUILD_GENERAL1_SMALL`).

Note

Otros valores disponibles en los datos originales con formato JSON, como `description`, `buildspec`, `auth` (incluido `type` y `resource`), `path`, `namespaceType`, `name` (para `artifacts`), `packaging`, `environmentVariables` (incluido `name` y `value`), `timeoutInMinutes`, `encryptionKey` y `tags` (incluido `key` y `value`) son opcionales. No se utilizan en este tutorial, por lo que no se muestran aquí. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

2. Cambie al directorio que contiene el archivo que acaba de guardar y, a continuación, vuelva a ejecutar el comando `create-project`.

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

Si el comando se ejecuta correctamente, aparecen datos similares a los siguientes en el resultado.

```
{
  "project": {
    "name": "codebuild-demo-project",
    "serviceRole": "serviceIAMRole",
    "tags": [],
    "artifacts": {
      "packaging": "NONE",
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-output-bucket",
      "name": "message-util.zip"
    },
    "lastModified": 1472661575.244,
    "timeoutInMinutes": 60,
    "created": 1472661575.244,
    "environment": {
      "computeType": "BUILD_GENERAL1_SMALL",
      "image": "aws/codebuild/standard:4.0",
      "type": "LINUX_CONTAINER",
      "environmentVariables": []
    },
    "source": {
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
    },
    "encryptionKey": "arn:aws:kms:region-ID:account-ID:alias/aws/s3",
    "arn": "arn:aws:codebuild:region-ID:account-ID:project/codebuild-demo-project"
  }
}
```

- `project` representa información acerca de este proyecto de compilación.

- `tags` representa todas las etiquetas que se han declarado.
- `packaging` representa la forma en que el artefacto de salida de la compilación se almacena en el bucket de salida. `NONE` indica que se crea una carpeta en el bucket de salida. El artefacto de salida de la compilación se almacena en esa carpeta.
- `lastModified` representa la fecha, en formato de fecha Unix, en la que se cambió por última vez la información sobre el proyecto de compilación.
- `timeoutInMinutes` representa el número de minutos después de los cuales CodeBuild detiene la compilación si esta no se ha completado. (El valor predeterminado es 60 minutos.)
- `created` representa la fecha, en formato de fecha Unix, en la que se creó el proyecto de compilación.
- `environmentVariables` representa cualquier variable de entorno que se haya declarado y esté disponible para que CodeBuild la use CodeBuild durante la compilación.
- `encryptionKey` representa el ARN de la clave administrada por el cliente que CodeBuild utilizó para cifrar el artefacto de salida de la compilación.
- `arn` representa el ARN del proyecto de compilación.

Note

Después de ejecutar `elcreate-project`, podría aparecer un mensaje de error similar al siguiente: Usuario: **ARN de usuario de** no está autorizado a realizar la operación: CodeBuild:CreateProject. Lo más probable es que se deba a que configuró el AWS CLI con las credenciales de un usuario de IAM que no tiene permisos suficientes para utilizar CodeBuild para crear proyectos de compilación. Para solucionarlo, configure el AWS CLI con credenciales que pertenezcan a alguna de las siguientes entidades de IAM:

- Un usuario de IAM administrador en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la IAM User Guide.
- Un usuario de IAM en su AWS cuenta con el `AWSCodeBuildAdminAccess`, `AmazonS3ReadOnlyAccess`, y `IAMFullAccess` políticas administradas asociadas a ese usuario de IAM o a un grupo de IAM al que pertenece el usuario de IAM. Si no tiene un usuario o grupo de IAM en su AWS con estos permisos y no puede agregar estos permisos a su usuario o grupo de IAM, póngase en contacto con su AWS administrador de cuentas para obtener ayuda. Para obtener más información, consulte [Políticas administradas \(predefinidas\) por AWS para AWS CodeBuild](#) (p. 360).

Paso siguiente

[Paso 6: Ejecute la compilación](#) (p. 24)

Paso 6: Ejecute la compilación

(Paso anterior: [Paso 5: Cree el proyecto de compilación](#) (p. 21))

En este paso, indicará a AWS CodeBuild que ejecute la compilación con la configuración del proyecto de compilación.

Para ejecutar la compilación

1. Utilice la AWS CLI para ejecutar el comando `start-build`:

```
aws codebuild start-build --project-name project-name
```

Sustituya *nombre-proyecto* por el nombre de su proyecto de compilación del paso anterior (por ejemplo, codebuild-demo-project).

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

```
{
  "build": {
    "buildComplete": false,
    "initiator": "user-name",
    "artifacts": {
      "location": "arn:aws:s3:::codebuild-region-ID-account-ID-output-bucket/message-util.zip"
    },
    "projectName": "codebuild-demo-project",
    "timeoutInMinutes": 60,
    "buildStatus": "IN_PROGRESS",
    "environment": {
      "computeType": "BUILD_GENERAL1_SMALL",
      "image": "aws/codebuild/standard:4.0",
      "type": "LINUX_CONTAINER",
      "environmentVariables": [ ]
    },
    "source": {
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
    },
    "currentPhase": "SUBMITTED",
    "startTime": 1472848787.882,
    "id": "codebuild-demo-project:0cfbb6ec-3db9-4e8c-992b-1ab28EXAMPLE",
    "arn": "arn:aws:codebuild:region-ID:account-ID:build/codebuild-demo-project:0cfbb6ec-3db9-4e8c-992b-1ab28EXAMPLE"
  }
}
```

- `build` representa información acerca de esta compilación.
 - `buildComplete` indica si la compilación se ha completado (`true`). De lo contrario, `false`.
 - `initiator` representa la entidad que inició la compilación.
 - `artifacts` representa información sobre la salida de la compilación, incluida su ubicación.
 - `projectName` representa el nombre del proyecto de compilación.
 - `buildStatus` representa el estado actual de la compilación cuando se ejecutó el comando `start-build`.
 - `currentPhase` representa la fase actual de la compilación cuando se ejecutó el comando `start-build`.
 - `startTime` representa la hora, en formato de hora Unix, en la que se inició el proceso de compilación.
 - `id` representa el ID de la compilación.
 - `arn` representa el ARN de la compilación.

Anote el valor de `id`. Lo necesitará en el siguiente paso.

Paso siguiente

[Paso 7: Ver información resumida sobre la compilación \(p. 26\)](#)

Paso 7: Ver información resumida sobre la compilación

(Paso anterior: [Paso 6: Ejecute la compilación](#) (p. 24))

En este paso, verá información resumida sobre el estado de la compilación.

Para ver la información resumida de la compilación

Utilice la AWS CLI para ejecutar el comando `batch-get-builds`.

```
aws codebuild batch-get-builds --ids id
```

Sustituya *id* por el valor de `id` que apareció en el resultado del paso anterior.

Si el comando se ejecuta correctamente, aparecen datos similares a los siguientes en el resultado.

```
{
  "buildsNotFound": [],
  "builds": [
    {
      "buildComplete": true,
      "phases": [
        {
          "phaseStatus": "SUCCEEDED",
          "endTime": 1472848788.525,
          "phaseType": "SUBMITTED",
          "durationInSeconds": 0,
          "startTime": 1472848787.882
        },
        ... The full list of build phases has been omitted for brevity ...
        {
          "phaseType": "COMPLETED",
          "startTime": 1472848878.079
        }
      ],
      "logs": {
        "groupName": "/aws/codebuild/codebuild-demo-project",
        "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=region-ID#logEvent:group=/aws/codebuild/codebuild-demo-project;stream=38ca1c4a-e9ca-4dbc-bef1-d52bfEXAMPLE",
        "streamName": "38ca1c4a-e9ca-4dbc-bef1-d52bfEXAMPLE"
      },
      "artifacts": {
        "md5sum": "MD5-hash",
        "location": "arn:aws:s3:::codebuild-region-ID-account-ID-output-bucket/message-util.zip",
        "sha256sum": "SHA-256-hash"
      },
      "projectName": "codebuild-demo-project",
      "timeoutInMinutes": 60,
      "initiator": "user-name",
      "buildStatus": "SUCCEEDED",
      "environment": {
        "computeType": "BUILD_GENERAL1_SMALL",
        "image": "aws/codebuild/standard:4.0",
        "type": "LINUX_CONTAINER",
        "environmentVariables": []
      },
      "source": {
        "type": "S3",
```

```
    "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
  },
  "currentPhase": "COMPLETED",
  "startTime": 1472848787.882,
  "endTime": 1472848878.079,
  "id": "codebuild-demo-project:38ca1c4a-e9ca-4dbc-bef1-d52bfEXAMPLE",
  "arn": "arn:aws:codebuild:region-ID:account-ID:build/codebuild-demo-project:38ca1c4a-
e9ca-4dbc-bef1-d52bfEXAMPLE"
}
]
}
```

- `buildsNotFound` representa los identificadores de compilación de todas las compilaciones sin información disponible. En este ejemplo, debería estar vacío.
- `builds` representa información de cada compilación con información disponible. En este ejemplo, en el resultado aparece información sobre una única compilación.
- `phases` representa el conjunto de fases de compilación que CodeBuild ejecuta durante el proceso de compilación. La información sobre cada fase de compilación se muestra por separado como `startTime`, `endTime` y `durationInSeconds` (cuándo se inició y terminó la fase de compilación, en formato de hora Unix, y cuánto duró, en segundos), así como `phaseType` como (SUBMITTED, PROVISIONING, DOWNLOAD_SOURCE, INSTALL, PRE_BUILD, BUILD, POST_BUILD, UPLOAD_ARTIFACTS, FINALIZING o COMPLETED) y `phaseStatus` (como SUCCEEDED, FAILED, FAULT, TIMED_OUT, IN_PROGRESS o STOPPED). La primera vez que ejecute el comando `batch-get-builds`, es posible que no haya muchas fases (o que no haya ninguna). Después de ejecuciones posteriores del comando `batch-get-builds` con el mismo ID de compilación, deberían aparecer más fases de compilación en el resultado.
- `logs` representa información en Amazon CloudWatch Logs sobre los logs de compilación.
- `md5sum` y `sha256sum` representan los valores hash MD5 y SHA-256 del artefacto de salida de la compilación. Aparecerán en el resultado solamente si el valor `packaging` del proyecto de compilación se establece en ZIP. (No ha establecido este valor en este tutorial). Puede utilizar estos valores hash junto con una herramienta de suma de comprobación para confirmar la integridad y autenticidad de los archivos.

Note

También puede usar la consola de Amazon S3 para ver estos valores hash. Seleccione la casilla situada junto al artefacto de salida de la compilación, elija **Actions** (Acciones) y luego elija **Properties** (Propiedades). En el panel **Properties**, expanda **Metadata** y consulte los valores de `x-amz-meta-codebuild-content-md5` y `x-amz-meta-codebuild-content-sha256`. (En la consola de Amazon S3, el artefacto de salida de compilación `ETag` el valor de no debe interpretarse como el valor hash MD5 o SHA-256).

Si utiliza los SDK de AWS para obtener estos valores hash, los valores tendrán el nombre `codebuild-content-md5` y `codebuild-content-sha256`.

- `endTime` representa la hora, en formato de hora Unix, en la que terminó el proceso de compilación.

Paso siguiente

[Paso 8: Ver información detallada sobre la compilación \(p. 27\)](#)

Paso 8: Ver información detallada sobre la compilación

(Paso anterior: [Paso 7: Ver información resumida sobre la compilación \(p. 26\)](#))

En este paso, verá información detallada acerca de su compilación en CloudWatch Logs.

Note

Para proteger la información confidencial, lo siguiente está oculto en los logs de CodeBuild:

- ID de clave de acceso de AWS. Para obtener más información, consulte [Administración de las claves de acceso de los usuarios de IAM](#) en la [AWS Identity and Access Management Guía del usuario](#) de.
- Cadenas especificadas mediante el almacén de parámetros. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial de la consola de almacén de parámetros de Systems Manager](#) en la [Amazon EC2 Systems Manager](#).
- Cadenas especificadas mediante AWS Secrets Manager. Para obtener más información, consulte [Administración de claves \(p. 353\)](#).

Para ver información detallada sobre la compilación

1. Utilice un navegador web para ir a la ubicación deepLink que aparecía en el resultado del paso anterior (por ejemplo, <https://console.aws.amazon.com/cloudwatch/home?region=region-ID#logEvent:group=/aws/codebuild/codebuild-demo-project;stream=38ca1c4a-e9ca-4dbc-bef1-d52bfEXAMPLE>).
2. En la secuencia de logs de CloudWatch Logs, puede examinar los eventos de log. De forma predeterminada, solo se muestra el último conjunto de eventos de log. Para ver eventos de log anteriores, desplácese hasta el principio de la lista.
3. En este tutorial, la mayoría de los eventos de log contienen información detallada sobre la descarga e instalación de CodeBuild de los archivos de dependencias de la compilación en su entorno de compilación, que posiblemente no le interese. Puede usar el cuadro Filter events para reducir la información que se muestra. Por ejemplo, si escribe "[INFO]" en el cuadro Filter events (Filtrar eventos), solo se muestran los eventos que contienen [INFO]. Para obtener más información, consulte [Sintaxis de patrones y filtros](#) en la [Guía del usuario de Amazon CloudWatch](#).

Estas partes de una secuencia de logs de CloudWatch Logs pertenecen a este tutorial.

```
...
[Container] 2016/04/15 17:49:42 Entering phase PRE_BUILD
[Container] 2016/04/15 17:49:42 Running command echo Entering pre_build phase...
[Container] 2016/04/15 17:49:42 Entering pre_build phase...
[Container] 2016/04/15 17:49:42 Phase complete: PRE_BUILD Success: true
[Container] 2016/04/15 17:49:42 Entering phase BUILD
[Container] 2016/04/15 17:49:42 Running command echo Entering build phase...
[Container] 2016/04/15 17:49:42 Entering build phase...
[Container] 2016/04/15 17:49:42 Running command mvn install
[Container] 2016/04/15 17:49:44 [INFO] Scanning for projects...
[Container] 2016/04/15 17:49:44 [INFO]
[Container] 2016/04/15 17:49:44 [INFO]
-----
[Container] 2016/04/15 17:49:44 [INFO] Building Message Utility Java Sample App 1.0
[Container] 2016/04/15 17:49:44 [INFO]
-----
...
[Container] 2016/04/15 17:49:55 -----
[Container] 2016/04/15 17:49:55  T E S T S
[Container] 2016/04/15 17:49:55 -----
[Container] 2016/04/15 17:49:55 Running TestMessageUtil
[Container] 2016/04/15 17:49:55 Inside testSalutationMessage()
[Container] 2016/04/15 17:49:55 Hi!Robert
[Container] 2016/04/15 17:49:55 Inside testPrintMessage()
[Container] 2016/04/15 17:49:55 Robert
[Container] 2016/04/15 17:49:55 Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time
elapsed: 0.018 sec
```

```
[Container] 2016/04/15 17:49:55
[Container] 2016/04/15 17:49:55 Results :
[Container] 2016/04/15 17:49:55
[Container] 2016/04/15 17:49:55 Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
...
[Container] 2016/04/15 17:49:56 [INFO]
-----
[Container] 2016/04/15 17:49:56 [INFO] BUILD SUCCESS
[Container] 2016/04/15 17:49:56 [INFO]
-----
[Container] 2016/04/15 17:49:56 [INFO] Total time: 11.845 s
[Container] 2016/04/15 17:49:56 [INFO] Finished at: 2016-04-15T17:49:56+00:00
[Container] 2016/04/15 17:49:56 [INFO] Final Memory: 18M/216M
[Container] 2016/04/15 17:49:56 [INFO]
-----
[Container] 2016/04/15 17:49:56 Phase complete: BUILD Success: true
[Container] 2016/04/15 17:49:56 Entering phase POST_BUILD
[Container] 2016/04/15 17:49:56 Running command echo Entering post_build phase...
[Container] 2016/04/15 17:49:56 Entering post_build phase...
[Container] 2016/04/15 17:49:56 Phase complete: POST_BUILD Success: true
[Container] 2016/04/15 17:49:57 Preparing to copy artifacts
[Container] 2016/04/15 17:49:57 Assembling file list
[Container] 2016/04/15 17:49:57 Expanding target/messageUtil-1.0.jar
[Container] 2016/04/15 17:49:57 Found target/messageUtil-1.0.jar
[Container] 2016/04/15 17:49:57 Creating zip artifact
```

En este ejemplo, CodeBuild ha completado correctamente las fases de precompilación, compilación y postcompilación. Ha ejecutado las pruebas unitarias y ha compilado correctamente el archivo `messageUtil-1.0.jar`.

Paso siguiente

[Paso 9: Obtener el artefacto de salida de la compil \(p. 29\)](#)

Paso 9: Obtener el artefacto de salida de la compil

(Paso anterior: [Paso 8: Ver información detallada sobre la compilación \(p. 27\)](#))

En este paso, obtienes el `messageUtil-1.0.jar` archivo que CodeBuild compiló y cargó en el bucket de salida.

Puede usar la consola de CodeBuild o la consola de Amazon S3 para realizar este paso.

Para obtener el artefacto de salida de la compilación (consola de AWS CodeBuild)

1. Con la consola CodeBuild todavía abierta y la página de detalles de la compilación mostrada del paso anterior, elija la **Detalles de compilación** y desplácese hacia abajo hasta la **Artifacts** sección.

Note

Si la página de detalles de la compilación no se muestra, en la barra de navegación, elija **Historial de compilación**, a continuación, elija la **Compilación** delink.

2. El enlace a la carpeta de Amazon S3 está en la **Ubicación de carga de artefactos**. Este enlace abre la carpeta en Amazon S3 donde se encuentra el `messageUtil-1.0.jar` archivo artefacto de salida de compilación.

Para obtener el artefacto de salida de la compilación (consola de Amazon S3)

1. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.

2. Abrir `codebuild-region-ID-account-ID-output-bucket`.
3. Abra la carpeta `codebuild-demo-project`.
4. Abra la carpeta `target`, donde se encuentra el archivo de artefactos de salida de compilación `messageUtil-1.0.jar`.

Paso siguiente

[Paso 10: Eliminación del bucket de S3 \(p. 30\)](#)

Paso 10: Eliminación del bucket de S3

(Paso anterior: [Paso 9: Obtener el artefacto de salida de la compil \(p. 29\)](#))

Para evitar cargos continuos a suAWS, puede eliminar los buckets de entrada y salida utilizados en este tutorial. Para obtener instrucciones, consulte[Eliminación o vaciado de un bucket](#) en laAmazon Simple Storage Service User Guide.

Si utiliza el usuario de IAM o un usuario administrador de IAM para eliminar estos bucket, el usuario debe tener más permisos de acceso. Añada la siguiente instrucción entre los marcadores (**### BEGIN ADDING STATEMENT HERE ###** y **### END ADDING STATEMENTS HERE ###**) a una política de acceso existente para el usuario.

Las elipsis (...) de esta declaración se usan para abreviar. No elimine las instrucciones de la política de acceso existente. No escriba estos puntos suspensivos en la política.

```
{
  "Version": "2012-10-17",
  "Id": "...",
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Effect": "Allow",
      "Action": [
        "s3:DeleteBucket",
        "s3:DeleteObject"
      ],
      "Resource": "*"
    }
    ### END ADDING STATEMENT HERE ###
  ]
}
```

Paso siguiente

[Encapsulación \(p. 30\)](#)

Encapsulación

En este tutorial, se ha usado AWS CodeBuild para construir un conjunto de archivos de clase Java en un archivo JAR. A continuación, ha consultado los resultados de la compilación.

Ahora puede intentar usar CodeBuild en sus propios escenarios. Siga las instrucciones en [Planificación de una compilación \(p. 135\)](#). Si piensa que aún no está preparado, tal vez desee intentar compilar algunos de los ejemplos. Para obtener más información, consulte [Muestras \(p. 31\)](#).

Ejemplos de CodeBuild

Estos grupos de ejemplos se pueden utilizar para experimentar con AWS CodeBuild:

Temas

- [Ejemplos de Microsoft Windows para CodeBuild \(p. 31\)](#)
- [Ejemplos basados en casos de uso de CodeBuild \(p. 44\)](#)

Ejemplos de Microsoft Windows para CodeBuild

Estas muestras utilizan unAWS CodeBuildentorno de compilación de que ejecuta Microsoft Windows Server 2019, .NET Framework y el SDK de .NET Core para compilar archivos en tiempo de ejecución con código escrito en F# y Visual Basic.

Important

La ejecución de estos ejemplos genera gastos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y porAWSrecursos y acciones relacionados con Amazon S3,AWS KMSy CloudWatch Logs. Para obtener más información, consulte[Precios de CodeBuild](#),[Precios de Amazon S3](#),[AWS Key Management ServicePrecios de](#), y[Precios de Amazon CloudWatch](#).

Ejecución de los ejemplos

Para ejecutar estos ejemplos:

1. Cree los archivos tal y como se describe en las secciones «Estructura de directorios» y «Archivos» de este tema y cárguelos después en un bucket de entrada de S3 o en un repositorio de CodeCommit o GitHub.

Important

No cargue (*root directory name*), solo los archivos incluidos en (*root directory name*).

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga los archivos y cárguelo en el bucket de entrada. No añada (*root directory name*) al archivo ZIP, solo los archivos incluidos en (*root directory name*).

2. Cree un proyecto de compilación. El proyecto de compilación debe utilizar `elmcr.microsoft.com/dotnet/framework/sdk:4.8` imagen para crear proyectos .NET Framework.

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando `create-project` tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{
  "name": "sample-windows-build-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/windows-build-input-artifact.zip"
  }
}
```

```
    },
    "artifacts": {
      "type": "S3",
      "location": "codebuild-region-ID-account-ID-output-bucket",
      "packaging": "ZIP",
      "name": "windows-build-output-artifact.zip"
    },
    "environment": {
      "type": "WINDOWS_SERVER_2019_CONTAINER",
      "image": "mcr.microsoft.com/dotnet/framework/sdk:4.8",
      "computeType": "BUILD_GENERAL1_MEDIUM"
    },
    "serviceRole": "arn:aws:iam::account-ID:role/role-name",
    "encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
  }
}
```

3. Ejecute la compilación y siga los pasos que se indican en [Ejecutar CodeBuild directamente \(p. 406\)](#).
4. Para obtener el artefacto de salida de la compilación, en el bucket de salida de S3, descargue el archivo *windows-build-output-artifact.zip* en el equipo o la instancia local. Extraiga el contenido para obtener el motor de ejecución y otros archivos.
 - El archivo de ejecución del ejemplo de F# que utiliza .NET Framework, *FSharpHelloWorld.exe*, se pueden encontrar en *laFSharpHelloWorld\bin\Debug* directorio.
 - El archivo de tiempo de ejecución del ejemplo de Visual Basic que utiliza .NET Framework, *VBHelloWorld.exe*, se pueden encontrar en *laVBHelloWorld\bin\Debug* directorio.

Estructura de directorios

Estos ejemplos suponen que se utilizan las siguientes estructuras de directorio.

F # y .NET Framework

```
(root directory name)
### buildspec.yml
### FSharpHelloWorld.sln
### FSharpHelloWorld
### App.config
### AssemblyInfo.fs
### FSharpHelloWorld.fsproj
### Program.fs
```

Visual Basic y .NET Framework

```
(root directory name)
### buildspec.yml
### VBHelloWorld.sln
### VBHelloWorld
### App.config
### HelloWorld.vb
### VBHelloWorld.vbproj
### My Project
### Application.Designer.vb
### Application.myapp
### AssemblyInfo.vb
### Resources.Designer.vb
### Resources.resx
### Settings.Designer.vb
### Settings.settings
```

Files

En estos ejemplos, se utilizan los siguientes archivos:

F # y .NET Framework

buildspec.yml (in *(root directory name)*):

```
version: 0.2

env:
  variables:
    SOLUTION: .\FSharpHelloWorld.sln
    PACKAGE_DIRECTORY: .\packages
    DOTNET_FRAMEWORK: 4.8

phases:
  build:
    commands:
      - '& nuget restore $env:SOLUTION -PackagesDirectory $env:PACKAGE_DIRECTORY'
      - '& msbuild -p:FrameworkPathOverride="C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework\.NETFramework\v$env:DOTNET_FRAMEWORK" $env:SOLUTION'
artifacts:
  files:
    - .\FSharpHelloWorld\bin\Debug\*
```

FSharpHelloWorld.sln (in *(root directory name)*):

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 14
VisualStudioVersion = 14.0.25420.1
MinimumVisualStudioVersion = 10.0.40219.1
Project("{F2A71F9B-5D33-465A-A702-920D77279786}") = "FSharpHelloWorld", "FSharpHelloWorld\FSharpHelloWorld.fsproj", "{D60939B6-526D-43F4-9A89-577B2980DF62}"
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {D60939B6-526D-43F4-9A89-577B2980DF62}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
        {D60939B6-526D-43F4-9A89-577B2980DF62}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {D60939B6-526D-43F4-9A89-577B2980DF62}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {D60939B6-526D-43F4-9A89-577B2980DF62}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
EndGlobal
```

App.config (in *(root directory name)*\FSharpHelloWorld):

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.8" />
  </startup>
</configuration>
```


AssemblyInfo.fs (in *(root directory name)*\FSharpHelloWorld):

```
namespace FSharpHelloWorld.AssemblyInfo

open System.Reflection
open System.Runtime.CompilerServices
open System.Runtime.InteropServices

// General Information about an assembly is controlled through the following
// set of attributes. Change these attribute values to modify the information
// associated with an assembly.
[<assembly: AssemblyTitle("FSharpHelloWorld")>]
[<assembly: AssemblyDescription("")>]
[<assembly: AssemblyConfiguration("")>]
[<assembly: AssemblyCompany("")>]
[<assembly: AssemblyProduct("FSharpHelloWorld")>]
[<assembly: AssemblyCopyright("Copyright © 2017")>]
[<assembly: AssemblyTrademark("")>]
[<assembly: AssemblyCulture("")>]

// Setting ComVisible to false makes the types in this assembly not visible
// to COM components. If you need to access a type in this assembly from
// COM, set the ComVisible attribute to true on that type.
[<assembly: ComVisible(false)>]

// The following GUID is for the ID of the typelib if this project is exposed to COM
[<assembly: Guid("d60939b6-526d-43f4-9a89-577b2980df62")>]

// Version information for an assembly consists of the following four values:
//
// Major Version
// Minor Version
// Build Number
// Revision
//
// You can specify all the values or you can default the Build and Revision Numbers
// by using the '*' as shown below:
// [<assembly: AssemblyVersion("1.0.*")>]
[<assembly: AssemblyVersion("1.0.0.0")>]
[<assembly: AssemblyFileVersion("1.0.0.0")>]

do
    ()
```

FSharpHelloWorld.fsproj (in *(root directory name)*\FSharpHelloWorld):

```
<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="14.0" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/
developer/msbuild/2003">
  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props"
    Condition="Exists('$(MSBuildExtensionsPath)\
$(MSBuildToolsVersion)\Microsoft.Common.props')" />
  <PropertyGroup>
    <Configuration Condition=" '$(Configuration)' == '' ">Debug</Configuration>
    <Platform Condition=" '$(Platform)' == '' ">AnyCPU</Platform>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>d60939b6-526d-43f4-9a89-577b2980df62</ProjectGuid>
    <OutputType>Exe</OutputType>
    <RootNamespace>FSharpHelloWorld</RootNamespace>
    <AssemblyName>FSharpHelloWorld</AssemblyName>
    <TargetFrameworkVersion>v4.8</TargetFrameworkVersion>
    <AutoGenerateBindingRedirects>true</AutoGenerateBindingRedirects>
    <TargetFSharpCoreVersion>4.4.0.0</TargetFSharpCoreVersion>
    <Name>FSharpHelloWorld</Name>
```

```

</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
  <DebugSymbols>true</DebugSymbols>
  <DebugType>full</DebugType>
  <Optimize>>false</Optimize>
  <Tailcalls>>false</Tailcalls>
  <OutputPath>bin\Debug\</OutputPath>
  <DefineConstants>DEBUG;TRACE</DefineConstants>
  <WarningLevel>3</WarningLevel>
  <PlatformTarget>AnyCPU</PlatformTarget>
  <DocumentationFile>bin\Debug\FSharpHelloWorld.XML</DocumentationFile>
  <Prefer32Bit>true</Prefer32Bit>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
  <DebugType>pdbonly</DebugType>
  <Optimize>true</Optimize>
  <Tailcalls>true</Tailcalls>
  <OutputPath>bin\Release\</OutputPath>
  <DefineConstants>TRACE</DefineConstants>
  <WarningLevel>3</WarningLevel>
  <PlatformTarget>AnyCPU</PlatformTarget>
  <DocumentationFile>bin\Release\FSharpHelloWorld.XML</DocumentationFile>
  <Prefer32Bit>true</Prefer32Bit>
</PropertyGroup>
<ItemGroup>
  <Reference Include="mscorlib" />
  <Reference Include="FSharp.Core, Version=$(TargetFSharpCoreVersion), Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a">
    <Private>True</Private>
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.Core" />
  <Reference Include="System.Numerics" />
</ItemGroup>
<ItemGroup>
  <Compile Include="AssemblyInfo.fs" />
  <Compile Include="Program.fs" />
  <None Include="App.config" />
</ItemGroup>
<PropertyGroup>
  <MinimumVisualStudioVersion Condition="'$(MinimumVisualStudioVersion)' == ''>11</
MinimumVisualStudioVersion>
</PropertyGroup>
<Choose>
  <When Condition="'$(VisualStudioVersion)' == '11.0'">
    <PropertyGroup Condition="Exists('$(MSBuildExtensionsPath32)\..\Microsoft SDKs\F#
\3.0\Framework\v4.0\Microsoft.FSharp.Targets')">
      <FSharpTargetsPath>$(MSBuildExtensionsPath32)\..\Microsoft SDKs\F#\3.0\Framework
\v4.0\Microsoft.FSharp.Targets</FSharpTargetsPath>
    </PropertyGroup>
  </When>
  <Otherwise>
    <PropertyGroup Condition="Exists('$(MSBuildExtensionsPath32)\Microsoft\VisualStudio\v
$(VisualStudioVersion)\FSharp\Microsoft.FSharp.Targets')">
      <FSharpTargetsPath>$(MSBuildExtensionsPath32)\Microsoft\VisualStudio\v
$(VisualStudioVersion)\FSharp\Microsoft.FSharp.Targets</FSharpTargetsPath>
    </PropertyGroup>
  </Otherwise>
</Choose>
<Import Project="$(FSharpTargetsPath)" />
<!-- To modify your build process, add your task inside one of the targets below and
uncomment it.
    Other similar extension points exist, see Microsoft.Common.targets.
<Target Name="BeforeBuild">
</Target>
<Target Name="AfterBuild">

```

```
</Target>
-->
</Project>
```

Program.fs (in *(root directory name)*\FSharpHelloWorld):

```
// Learn more about F# at http://fsharp.org
// See the 'F# Tutorial' project for more help.

[<EntryPoint>]
let main argv =
    printfn "Hello World"
    0 // return an integer exit code
```

Visual Basic y .NET Framework

buildspec.yml (in *(root directory name)*):

```
version: 0.2

env:
  variables:
    SOLUTION: .\VBHelloWorld.sln
    PACKAGE_DIRECTORY: .\packages
    DOTNET_FRAMEWORK: 4.8

phases:
  build:
    commands:
      - '& "C:\ProgramData\chocolatey\bin\NuGet.exe" restore $env:SOLUTION -
        PackagesDirectory $env:PACKAGE_DIRECTORY'
      - '& "C:\Program Files (x86)\MSBuild\14.0\Bin\MSBuild.exe" -
        p:FrameworkPathOverride="C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework
        \.NETFramework\v$env:DOTNET_FRAMEWORK" $env:SOLUTION'
    artifacts:
      files:
        - .\VBHelloWorld\bin\Debug\*
```

VBHelloWorld.sln (in *(root directory name)*):

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 14
VisualStudioVersion = 14.0.25420.1
MinimumVisualStudioVersion = 10.0.40219.1
Project("{F184B08F-C81C-45F6-A57F-5ABD9991F28F}") = "VBHelloWorld", "VBHelloWorld
\VBHelloWorld.vbproj", "{4DCEC446-7156-4FE6-8CCC-219E34DD409D}"
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {4DCEC446-7156-4FE6-8CCC-219E34DD409D}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
        {4DCEC446-7156-4FE6-8CCC-219E34DD409D}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {4DCEC446-7156-4FE6-8CCC-219E34DD409D}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {4DCEC446-7156-4FE6-8CCC-219E34DD409D}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
```

```
EndGlobal
```

App.config (in *(root directory name)*\VBHelloWorld):

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.8" />
  </startup>
</configuration>
```

HelloWorld.vb (in *(root directory name)*\VBHelloWorld):

```
Module HelloWorld

  Sub Main()
    MsgBox("Hello World")
  End Sub

End Module
```

VBHelloWorld.vbproj (in *(root directory name)*\VBHelloWorld):

```
<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="14.0" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/
developer/msbuild/2003">
  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props"
    Condition="Exists('$(MSBuildExtensionsPath)\
$(MSBuildToolsVersion)\Microsoft.Common.props')" />
  <PropertyGroup>
    <Configuration Condition=" '$(Configuration)' == '' ">Debug</Configuration>
    <Platform Condition=" '$(Platform)' == '' ">AnyCPU</Platform>
    <ProjectGuid>{4DCEC446-7156-4FE6-8CCC-219E34DD409D}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <StartupObject>VBHelloWorld.HelloWorld</StartupObject>
    <RootNamespace>VBHelloWorld</RootNamespace>
    <AssemblyName>VBHelloWorld</AssemblyName>
    <FileAlignment>512</FileAlignment>
    <MyType>Console</MyType>
    <TargetFrameworkVersion>v4.8</TargetFrameworkVersion>
    <AutoGenerateBindingRedirects>true</AutoGenerateBindingRedirects>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugSymbols>true</DebugSymbols>
    <DebugType>full</DebugType>
    <DefineDebug>true</DefineDebug>
    <DefineTrace>true</DefineTrace>
    <OutputPath>bin\Debug\</OutputPath>
    <DocumentationFile>VBHelloWorld.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugType>pdbonly</DebugType>
    <DefineDebug>>false</DefineDebug>
    <DefineTrace>true</DefineTrace>
    <Optimize>true</Optimize>
    <OutputPath>bin\Release\</OutputPath>
    <DocumentationFile>VBHelloWorld.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
  </PropertyGroup>
```

```

<PropertyGroup>
  <OptionExplicit>On</OptionExplicit>
</PropertyGroup>
<PropertyGroup>
  <OptionCompare>Binary</OptionCompare>
</PropertyGroup>
<PropertyGroup>
  <OptionStrict>Off</OptionStrict>
</PropertyGroup>
<PropertyGroup>
  <OptionInfer>On</OptionInfer>
</PropertyGroup>
<ItemGroup>
  <Reference Include="System" />
  <Reference Include="System.Data" />
  <Reference Include="System.Deployment" />
  <Reference Include="System.Xml" />
  <Reference Include="System.Core" />
  <Reference Include="System.Xml.Linq" />
  <Reference Include="System.Data.DataSetExtensions" />
  <Reference Include="System.Net.Http" />
</ItemGroup>
<ItemGroup>
  <Import Include="Microsoft.VisualBasic" />
  <Import Include="System" />
  <Import Include="System.Collections" />
  <Import Include="System.Collections.Generic" />
  <Import Include="System.Data" />
  <Import Include="System.Diagnostics" />
  <Import Include="System.Linq" />
  <Import Include="System.Xml.Linq" />
  <Import Include="System.Threading.Tasks" />
</ItemGroup>
<ItemGroup>
  <Compile Include="HelloWorld.vb" />
  <Compile Include="My Project\AssemblyInfo.vb" />
  <Compile Include="My Project\Application.Designer.vb">
    <AutoGen>True</AutoGen>
    <DependentUpon>Application.myapp</DependentUpon>
  </Compile>
  <Compile Include="My Project\Resources.Designer.vb">
    <AutoGen>True</AutoGen>
    <DesignTime>True</DesignTime>
    <DependentUpon>Resources.resx</DependentUpon>
  </Compile>
  <Compile Include="My Project\Settings.Designer.vb">
    <AutoGen>True</AutoGen>
    <DependentUpon>Settings.settings</DependentUpon>
    <DesignTimeSharedInput>True</DesignTimeSharedInput>
  </Compile>
</ItemGroup>
<ItemGroup>
  <EmbeddedResource Include="My Project\Resources.resx">
    <Generator>VbMyResourcesResXFileCodeGenerator</Generator>
    <LastGenOutput>Resources.Designer.vb</LastGenOutput>
    <CustomToolNamespace>My.Resources</CustomToolNamespace>
    <SubType>Designer</SubType>
  </EmbeddedResource>
</ItemGroup>
<ItemGroup>
  <None Include="My Project\Application.myapp">
    <Generator>MyApplicationCodeGenerator</Generator>
    <LastGenOutput>Application.Designer.vb</LastGenOutput>
  </None>
  <None Include="My Project\Settings.settings">
    <Generator>SettingsSingleFileGenerator</Generator>
  </None>
</ItemGroup>

```

```

    <CustomToolNamespace>My</CustomToolNamespace>
    <LastGenOutput>Settings.Designer.vb</LastGenOutput>
  </None>
  <None Include="App.config" />
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.VisualBasic.targets" />
<!-- To modify your build process, add your task inside one of the targets below and
uncomment it.
    Other similar extension points exist, see Microsoft.Common.targets.
  <Target Name="BeforeBuild">
  </Target>
  <Target Name="AfterBuild">
  </Target>
-->
</Project>

```

Application.Designer.vb (in *(root directory name)*\VBHelloWorld\My Project):

```

'-----
' <auto-generated>
'   This code was generated by a tool.
'   Runtime Version:4.0.30319.42000
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </auto-generated>
'-----

Option Strict On
Option Explicit On

```

Application.myapp (in *(root directory name)*\VBHelloWorld\My Project):

```

<?xml version="1.0" encoding="utf-8"?>
<MyApplicationData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema">
  <MySubMain>false</MySubMain>
  <SingleInstance>false</SingleInstance>
  <ShutdownMode>0</ShutdownMode>
  <EnableVisualStyles>true</EnableVisualStyles>
  <AuthenticationMode>0</AuthenticationMode>
  <ApplicationType>2</ApplicationType>
  <SaveMySettingsOnExit>true</SaveMySettingsOnExit>
</MyApplicationData>

```

AssemblyInfo.vb (in *(root directory name)*\VBHelloWorld\My Project):

```

Imports System
Imports System.Reflection
Imports System.Runtime.InteropServices

' General Information about an assembly is controlled through the following
' set of attributes. Change these attribute values to modify the information
' associated with an assembly.

' Review the values of the assembly attributes

<Assembly: AssemblyTitle("VBHelloWorld")>
<Assembly: AssemblyDescription("")>
<Assembly: AssemblyCompany("")>
<Assembly: AssemblyProduct("VBHelloWorld")>
<Assembly: AssemblyCopyright("Copyright © 2017")>

```

```
<Assembly: AssemblyTrademark("")>

<Assembly: ComVisible(False)>

'The following GUID is for the ID of the typelib if this project is exposed to COM
<Assembly: Guid("137c362b-36ef-4c3e-84ab-f95082487a5a")>

' Version information for an assembly consists of the following four values:
'
' Major Version
' Minor Version
' Build Number
' Revision
'
' You can specify all the values or you can default the Build and Revision Numbers
' by using the '*' as shown below:
' <Assembly: AssemblyVersion("1.0.*")>

<Assembly: AssemblyVersion("1.0.0.0")>
<Assembly: AssemblyFileVersion("1.0.0.0")>
```

Resources.Designer.vb (in *(root directory name)*\VBHelloWorld\My Project):

```
'-----
' <auto-generated>
'   This code was generated by a tool.
'   Runtime Version:4.0.30319.42000
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </auto-generated>
'-----

Option Strict On
Option Explicit On

Namespace My.Resources

    'This class was auto-generated by the StronglyTypedResourceBuilder
    'class via a tool like ResGen or Visual Studio.
    'To add or remove a member, edit your .ResX file then rerun ResGen
    'with the /str option, or rebuild your VS project.
    '''<summary>
    '''   A strongly-typed resource class, for looking up localized strings, etc.
    '''</summary>

    <Global.System.CodeDom.Compiler.GeneratedCodeAttribute("System.Resources.Tools.StronglyTypedResourceBuilder",
    "4.0.0.0"), _
    Global.System.Diagnostics.DebuggerNonUserCodeAttribute(), _
    Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute(), _
    Global.Microsoft.VisualBasic.HideModuleNameAttribute()> _
    Friend Module Resources

        Private resourceMan As Global.System.Resources.ResourceManager

        Private resourceCulture As Global.System.Globalization.CultureInfo

        '''<summary>
        '''   Returns the cached ResourceManager instance used by this class.
        '''</summary>

        <Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableState.Never)>
        Friend ReadOnly Property ResourceManager() As Global.System.Resources.ResourceManager
            Get
```

```

        If Object.ReferenceEquals(resourceMan, Nothing) Then
            Dim temp As Global.System.Resources.ResourceManager = New
Global.System.Resources.ResourceManager("VBHelloWorld.Resources",
GetType(Resources).Assembly)
            resourceMan = temp
        End If
        Return resourceMan
    End Get
End Property

'''<summary>
''' Overrides the current thread's CurrentUICulture property for all
''' resource lookups using this strongly typed resource class.
'''</summary>

<Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableSta
-
Friend Property Culture() As Global.System.Globalization.CultureInfo
    Get
        Return resourceCulture
    End Get
    Set(ByVal value As Global.System.Globalization.CultureInfo)
        resourceCulture = value
    End Set
End Property
End Module
End Namespace

```

Resources.resx (in *(root directory name)*\VBHelloWorld\My Project):

```

<?xml version="1.0" encoding="utf-8"?>
<root>
  <!--
    Microsoft ResX Schema

    Version 2.0

    The primary goals of this format is to allow a simple XML format
    that is mostly human readable. The generation and parsing of the
    various data types are done through the TypeConverter classes
    associated with the data types.

    Example:

    ... ado.net/XML headers & schema ...
    <resheader name="resmimetype">text/microsoft-resx</resheader>
    <resheader name="version">2.0</resheader>
    <resheader name="reader">System.Resources.ResXResourceReader,
System.Windows.Forms, ...</resheader>
    <resheader name="writer">System.Resources.ResXResourceWriter,
System.Windows.Forms, ...</resheader>
    <data name="Name1"><value>this is my long string</value><comment>this is a comment</
comment></data>
    <data name="Color1" type="System.Drawing.Color, System.Drawing">Blue</data>
    <data name="Bitmap1" mimetype="application/x-microsoft.net.object.binary.base64">
    <value>[base64 mime encoded serialized .NET Framework object]</value>
    </data>
    <data name="Icon1" type="System.Drawing.Icon, System.Drawing" mimetype="application/x-
microsoft.net.object.binary.base64">
    <value>[base64 mime encoded string representing a byte array form of the .NET
Framework object]</value>
    <comment>This is a comment</comment>
    </data>

    There are any number of "resheader" rows that contain simple

```


name/value pairs.

Each data row contains a name, and value. The row also contains a type or mimetype. Type corresponds to a .NET class that support text/value conversion through the TypeConverter architecture. Classes that don't support this are serialized and stored with the mimetype set.

The mimetype is used for serialized objects, and tells the ResXResourceReader how to depersist the object. This is currently not extensible. For a given mimetype the value must be set accordingly:

Note - application/x-microsoft.net.object.binary.base64 is the format that the ResXResourceWriter will generate, however the reader can read any of the formats listed below.

```
mimetype: application/x-microsoft.net.object.binary.base64
value      : The object must be serialized with
              : System.Serialization.Formatters.Binary.BinaryFormatter
              : and then encoded with base64 encoding.
```

```
mimetype: application/x-microsoft.net.object.soap.base64
value      : The object must be serialized with
              : System.Runtime.Serialization.Formatters.Soap.SoapFormatter
              : and then encoded with base64 encoding.
```

```
mimetype: application/x-microsoft.net.object.bytearray.base64
value      : The object must be serialized into a byte array
              : using a System.ComponentModel.TypeConverter
              : and then encoded with base64 encoding.
```

```
-->
<xsd:schema id="root" xmlns="" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
  <xsd:element name="root" msdata:IsDataSet="true">
    <xsd:complexType>
      <xsd:choice maxOccurs="unbounded">
        <xsd:element name="metadata">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="value" type="xsd:string" minOccurs="0" />
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" />
            <xsd:attribute name="type" type="xsd:string" />
            <xsd:attribute name="mimetype" type="xsd:string" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="assembly">
          <xsd:complexType>
            <xsd:attribute name="alias" type="xsd:string" />
            <xsd:attribute name="name" type="xsd:string" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="data">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="value" type="xsd:string" minOccurs="0"
msdata:Ordinal="1" />
              <xsd:element name="comment" type="xsd:string" minOccurs="0"
msdata:Ordinal="2" />
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" msdata:Ordinal="1" />
            <xsd:attribute name="type" type="xsd:string" msdata:Ordinal="3" />
            <xsd:attribute name="mimetype" type="xsd:string" msdata:Ordinal="4" />
          </xsd:complexType>
        </xsd:element>
      </xsd:choice>
    </xsd:element>
  </xsd:schema>
```

```

        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="value" type="xsd:string" minOccurs="0"
msdata:Ordinal="1" />
          </xsd:sequence>
          <xsd:attribute name="name" type="xsd:string" use="required" />
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
<resheader name="resmimetype">
  <value>text/microsoft-resx</value>
</resheader>
<resheader name="version">
  <value>2.0</value>
</resheader>
<resheader name="reader">
  <value>System.Resources.ResXResourceReader, System.Windows.Forms, Version=2.0.0.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089</value>
</resheader>
<resheader name="writer">
  <value>System.Resources.ResXResourceWriter, System.Windows.Forms, Version=2.0.0.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089</value>
</resheader>
</root>

```

Settings.Designer.vb (in *(root directory name)*\VBHelloWorld\My Project):

```

'-----
' <auto-generated>
'   This code was generated by a tool.
'   Runtime Version:4.0.30319.42000
'
'   Changes to this file may cause incorrect behavior and will be lost if
'   the code is regenerated.
' </auto-generated>
'-----

Option Strict On
Option Explicit On

Namespace My

    <Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute(), _

Global.System.CodeDom.Compiler.GeneratedCodeAttribute("Microsoft.VisualStudio.Editors.SettingsDesigner
"11.0.0.0"), _

Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableStat
_
Partial Friend NotInheritable Class MySettings
    Inherits Global.System.Configuration.ApplicationSettingsBase

    Private Shared defaultInstance As MySettings =
CType(Global.System.Configuration.ApplicationSettingsBase.Synchronized(New MySettings),
MySettings)

    #Region "My.Settings Auto-Save Functionality"
    #If _MyType = "WindowsForms" Then
        Private Shared addedHandler As Boolean

        Private Shared addedHandlerLockObject As New Object

```

```
<Global.System.Diagnostics.DebuggerNonUserCodeAttribute(),  
Global.System.ComponentModel.EditorBrowsableAttribute(Global.System.ComponentModel.EditorBrowsableStat  
-  
Private Shared Sub AutoSaveSettings(ByVal sender As Global.System.Object, ByVal e  
As Global.System.EventArgs)  
    If My.Application.SaveMySettingsOnExit Then  
        My.Settings.Save()  
    End If  
End Sub  
#End If  
#End Region  
  
Public Shared ReadOnly Property [Default]() As MySettings  
    Get  
  
        #If _MyType = "WindowsForms" Then  
            If Not addedHandler Then  
                SyncLock addedHandlerLockObject  
                    If Not addedHandler Then  
                        AddHandler My.Application.Shutdown, AddressOf AutoSaveSettings  
                        addedHandler = True  
                    End If  
                End SyncLock  
            End If  
            #End If  
            Return defaultInstance  
        End Get  
    End Property  
End Class  
End Namespace  
  
Namespace My  
  
    <Global.Microsoft.VisualBasic.HideModuleNameAttribute(), _  
    Global.System.Diagnostics.DebuggerNonUserCodeAttribute(), _  
    Global.System.Runtime.CompilerServices.CompilerGeneratedAttribute()> _  
    Friend Module MySettingsProperty  
  
        <Global.System.ComponentModel.Design.HelpKeywordAttribute("My.Settings")> _  
        Friend ReadOnly Property Settings() As Global.VBHelloWorld.My.MySettings  
            Get  
                Return Global.VBHelloWorld.My.MySettings.Default  
            End Get  
        End Property  
    End Module  
End Namespace
```

Settings.settings (in (root directory name)\VBHelloWorld\My Project):

```
<?xml version='1.0' encoding='utf-8'?>  
<SettingsFile xmlns="http://schemas.microsoft.com/VisualStudio/2004/01/settings"  
    CurrentProfile="(Default)" UseMySettingsClassName="true">  
    <Profiles>  
        <Profile Name="(Default)" />  
    </Profiles>  
    <Settings />  
</SettingsFile>
```

Ejemplos basados en casos de uso de CodeBuild

Puede utilizar estos ejemplos basados en casos de uso para experimentar con AWS CodeBuild:

[Ejemplo de Amazon ECR \(p. 46\)](#)

Utiliza una imagen de Docker de un repositorio de Amazon ECR para usar Apache Maven para producir un único archivo JAR.

[Ejemplo de Amazon EFS \(p. 50\)](#)

Muestra cómo configurar un archivo buildspec de modo que un proyecto de CodeBuild se monte y se cree en un sistema de archivos de Amazon EFS.

[Ejemplo de AWS CodeDeploy \(p. 54\)](#)

Utiliza Apache Maven para producir un único archivo JAR. Utiliza CodeDeploy para implementar el archivo JAR en una instancia de Amazon Linux. También puede utilizar CodePipeline para compilar e implementar el ejemplo.

[AWS CodePipelineejemplo de integración con compilaciones por lotes \(p. 62\)](#)

Muestra cómo utilizar AWS CodePipeline para crear una compilación con varios orígenes de entrada y varios artefactos de salida.

[Ejemplo de AWS Config \(p. 65\)](#)

Muestra cómo configurar AWS Config. Muestra que se realiza un seguimiento de los recursos de CodeBuild y describe cómo buscar proyectos de CodeBuild enAWS Config.

[Ejemplo de AWS Elastic Beanstalk \(p. 66\)](#)

Utiliza Apache Maven para producir un único archivo WAR. Utiliza Elastic Beanstalk para desplegar el archivo WAR en una instancia de Elastic Beanstalk.

[Ejemplo de filtros de webhook y solicitudes de extracción de Bitbucket \(p. 74\)](#)

Utiliza CodeBuild con Bitbucket como repositorio de origen y la funcionalidad de webhooks habilitada para volver a compilar el código fuente cada vez que se envía un cambio de código al repositorio.

[Ejemplo de insignias de compilación \(p. 78\)](#)

Muestra cómo configurar CodeBuild con insignias de compilación.

[Ejemplo de notificaciones de compilación \(p. 81\)](#)

Utiliza Apache Maven para producir un único archivo JAR. Envía una notificación de compilación a los suscriptores de un tema de Amazon SNS.

[Crear un informe de pruebas con el ejemplo de AWS CLI \(p. 94\)](#)

Utiliza la AWS CLI para crear, ejecutar y visualizar los resultados de un informe de pruebas.

[Ejemplo de Docker en una imagen personalizada \(p. 99\)](#)

Utiliza una imagen de Docker personalizada para producir una imagen de Docker.

[Ejemplo de Docker \(p. 101\)](#)

Utiliza una imagen compilada proporcionada por CodeBuild con compatibilidad con Docker para producir una imagen de Docker con Apache Maven. Inserta la imagen de Docker en un repositorio de Amazon ECR. También puede adaptar este ejemplo para insertar la imagen de Docker en Docker Hub.

[Ejemplo de GitHub Enterprise Server \(p. 107\)](#)

Utiliza CodeBuild con GitHub Enterprise Server como repositorio de origen con certificados instalados y la funcionalidad de webhooks habilitada para volver a compilar automáticamente el código fuente cada vez que se envía un cambio de código al repositorio.

[Ejemplo de filtros de webhooks y solicitudes de extracción de GitHub \(p. 113\)](#)

Utiliza CodeBuild con GitHub como repositorio de origen y la funcionalidad de webhooks habilitada para volver a compilar el código fuente cada vez que se envía un cambio de código al repositorio.

[Alojar la salida de la compilación en un bucket de S3 \(p. 116\)](#)

Muestra cómo se crea un sitio web estático en un bucket de S3 con artefactos de compilación sin cifrar.

[Ejemplo de varios orígenes de entrada y varios artefactos de salida \(p. 129\)](#)

Muestra cómo utilizar varios orígenes de entrada y varios artefactos de salida en un proyecto de compilación.

[Ejemplo de un registro privado con AWS Secrets Manager \(p. 128\)](#)

Muestra cómo utilizar una imagen de Docker en un registro privado como el entorno en tiempo de ejecución al compilar con CodeBuild. Las credenciales del registro privado se almacenan en AWS Secrets Manager.

[Versiones del entorno de ejecución en el ejemplo del archivo buildspec \(p. 118\)](#)

Muestra cómo especificar los tiempos de ejecución y sus versiones en el archivo buildspec. Es un requisito cuando se utiliza la versión 2.0 de la imagen estándar de Ubuntu.

[Ejemplo de versión de origen \(p. 125\)](#)

En esta página se muestra cómo utilizar una versión específica de su origen en un proyecto de compilación de CodeBuild.

[Ejemplo para utilizar el control de versiones semánticas para asignar nombre a artefactos de compilación \(p. 132\)](#)

Muestra cómo utilizar el control de versiones semánticas para crear un nombre de artefacto en el momento de la compilación.

Ejemplo de Amazon ECR para CodeBuild

En este ejemplo, se utiliza una imagen de Docker de un repositorio de imágenes de Amazon Elastic Container Registry (Amazon ECR) para compilar un proyecto Go de muestra.

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por AWS CodeBuild y por recursos y acciones relacionados con Amazon S3, AWS KMS, CloudWatch Logs y Amazon ECR. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), [Precios de Amazon CloudWatch](#), y [Precios de Amazon Elastic Container](#).

Ejecución del ejemplo

Para ejecutar este ejemplo

1. Para crear e insertar la imagen de Docker en el repositorio de imágenes de Amazon ECR, siga los pasos que se indican en la sección «Ejecución del ejemplo» de [la Ejemplo de Docker \(p. 101\)](#).
2. Crear un proyecto de Go:
 - a. Cree los archivos tal y como se describe en las secciones [Estructura de un proyecto de Go \(p. 49\)](#) y [Archivos de un proyecto de Go \(p. 49\)](#) de este tema y cárguelos en un bucket de entrada de S3 o en un repositorio de AWS CodeCommit, GitHub o Bitbucket.

Important

No cargue (*root directory name*), solo los archivos incluidos en (*root directory name*).

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga los archivos y cárguelo en el bucket de entrada. No añada (*root directory name*) al archivo ZIP, solo los archivos incluidos en (*root directory name*).

- b. Cree un proyecto de compilación, ejecute la compilación y consulte información de la compilación relacionada siguiendo los pasos de [Ejecutar AWS CodeBuild directamente](#) (p. 406).

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando `create-project` tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{
  "name": "sample-go-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/GoSample.zip"
  },
  "artifacts": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-output-bucket",
    "packaging": "ZIP",
    "name": "GoOutputArtifact.zip"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:4.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "arn:aws:iam::account-ID:role/role-name",
  "encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

- c. Para obtener el artefacto de salida de la compilación, abra el bucket de salida de S3.
 - d. Descargue el archivo *GoOutputArtifact.zip* en su equipo o instancia local y después extraiga el contenido del archivo. En el contenido extraído, obtenga el archivo `hello`.
3. Si se cumple una de las siguientes condiciones, debe agregar permisos a su repositorio de imágenes en Amazon ECR para que AWS CodeBuild pueda extraer su imagen Docker en el entorno de compilación.
 - El proyecto utiliza credenciales de CodeBuild para extraer imágenes de Amazon ECR. Esto se especifica mediante el valor `CODEBUILD` del atributo `imagePullCredentialsType` de `ProjectEnvironment`.
 - Su proyecto utiliza una imagen de Amazon ECR entre cuentas. En este caso, su proyecto debe utilizar su rol de servicio para extraer imágenes de Amazon ECR. Para habilitar este comportamiento, establezca el atributo `imagePullCredentialsType` de `ProjectEnvironment` en `SERVICE_ROLE`.
1. Abra la consola de Amazon ECR en <https://console.aws.amazon.com/ecr/>.
 2. En la lista de nombres de repositorio, elija el nombre del repositorio que ha creado o seleccionado.
 3. En el panel de navegación, elija Permissions (Permisos), Edit (Editar) y Add statement (Agregar instrucción).
 4. En Statement name (Nombre de instrucción), introduzca un identificador (por ejemplo, **CodeBuildAccess**).
 5. En Effect (Efecto), deje seleccionado Allow (Permitir). Esto indica que desea permitir el acceso a otra cuenta de AWS.
 6. En Principal, realice una de las siguientes acciones:
 - Si el proyecto utiliza credenciales de CodeBuild para extraer una imagen de Amazon ECR, en Principal del servicio, introduzca **codebuild.amazonaws.com**.

- Si su proyecto utiliza una imagen de Amazon ECR entre cuentas, paraAWSID de cuenta, introduzca los ID de laAWS cuentas a las que desea otorgar acceso.
7. Omita la lista All IAM entities.
 8. En Action (Acción), seleccione las acciones de solo extracción ecr:GetDownloadUrlForLayer, ecr:BatchGetImage y ecr:BatchCheckLayerAvailability.
 9. Seleccione Save

Esta política aparece en Permissions (Permisos). La entidad de seguridad es la que especificó en Principal (Entidad de seguridad) en el paso 3 de este procedimiento:

- Si el proyecto utiliza credenciales de CodeBuild para extraer una imagen de Amazon ECR,"codebuild.amazonaws.com" aparece bajo Directores de servicio.
- Si el proyecto utiliza una imagen de Amazon ECR entre cuentas, el ID deAWS la cuenta a la que desea obtener acceso aparece enAWSID de cuenta.

La siguiente política de ejemplo utiliza credenciales de CodeBuild y una imagen de Amazon ECR de entre cuentas.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CodeBuildAccessPrincipal",
      "Effect": "Allow",
      "Principal": {
        "Service": "codebuild.amazonaws.com"
      },
      "Action": [
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "ecr:BatchCheckLayerAvailability"
      ]
    },
    {
      "Sid": "CodeBuildAccessCrossAccount",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<AWS-account-ID>:root"
      },
      "Action": [
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "ecr:BatchCheckLayerAvailability"
      ]
    }
  ]
}
```

4. Cree un proyecto de compilación, ejecute la compilación y consulte información de la compilación siguiendo los pasos de [Ejecutar CodeBuild directamente \(p. 406\)](#).

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando create-project tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{
  "name": "amazon-ecr-sample-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/GoSample.zip"
  },
}
```

```
"artifacts": {
  "type": "S3",
  "location": "codebuild-region-ID-account-ID-output-bucket",
  "packaging": "ZIP",
  "name": "GoOutputArtifact.zip"
},
"environment": {
  "type": "LINUX_CONTAINER",
  "image": "account-ID.dkr.ecr.region-ID.amazonaws.com/your-Amazon-ECR-repo-name:tag",
  "computeType": "BUILD_GENERAL1_SMALL"
},
"serviceRole": "arn:aws:iam::account-ID:role/role-name",
"encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

5. Para obtener el artefacto de salida de la compilación, abra el bucket de salida de S3.
6. Descargue el archivo *GoOutputArtifact*.zip en su equipo o instancia local y después extraiga el contenido del archivo *GoOutputArtifact*.zip. En el contenido extraído, obtenga el archivo hello.

Estructura de un proyecto de Go

En este ejemplo se presupone que existe esta estructura de directorios.

```
(root directory name)
### buildspec.yml
### hello.go
```

Archivos de un proyecto de Go

Este ejemplo usa los siguientes archivos.

buildspec.yml (in (*root directory name*))

```
version: 0.2

phases:
  install:
    runtime-versions:
      golang: 1.13
  build:
    commands:
      - echo Build started on `date`
      - echo Compiling the Go code
      - go build hello.go
  post_build:
    commands:
      - echo Build completed on `date`
artifacts:
  files:
    - hello
```

hello.go (in (*root directory name*))

```
package main
import "fmt"

func main() {
```



```
fmt.Println("hello world")
fmt.Println("1+1 =", 1+1)
fmt.Println("7.0/3.0 =", 7.0/3.0)
fmt.Println(true && false)
fmt.Println(true || false)
fmt.Println(!true)
}
```

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Ejemplo de Amazon Elastic File System para AWS CodeBuild

Es posible que desee crear su AWS CodeBuild en Amazon Elastic File System, un servicio escalable de archivos compartidos para instancias de Amazon EC2. La capacidad de almacenamiento de Amazon EFS es elástica, por lo que aumenta o disminuye a medida que se añaden y eliminan archivos. Tiene una interfaz de servicios web sencilla que puede utilizar para crear y configurar sistemas de archivos. También administra automáticamente toda la infraestructura de almacenamiento de archivos, por lo que no tiene que preocuparse por la implementación, aplicación de parches o el mantenimiento de configuraciones del sistema de archivos. Para obtener más información, consulte [¿Qué es Amazon Elastic File System?](#) en la Guía del usuario de Amazon Elastic File System.

En este ejemplo, se muestra cómo configurar un proyecto de CodeBuild para que monte y cree una aplicación Java en un sistema de archivos de Amazon EFS. Antes de comenzar, debe disponer de una aplicación Java lista para compilar que esté cargada en un bucket de entrada de S3 o en un repositorio de AWS CodeCommit, GitHub, GitHub Enterprise Server o Bitbucket.

Los datos en tránsito para su sistema de archivos están cifrados. Para cifrar los datos en tránsito con una imagen distinta, consulte [Cifrado de datos en tránsito](#).

Pasos generales

En este ejemplo, se explican los tres pasos de alto nivel necesarios para utilizar Amazon EFS con AWS CodeBuild:

1. Cree una nube virtual privada (VPC) en su AWS account.
2. Crear un sistema de archivos que utilice esta VPC.
3. Cree y compile un proyecto de CodeBuild que utilice la VPC. El proyecto CodeBuild utiliza lo siguiente para identificar el sistema de archivos:
 - Un identificador único del sistema de archivos. El identificador se elige al especificar el sistema de archivos en el proyecto de compilación.
 - El ID del sistema de archivos. El ID se muestra cuando ve el sistema de archivos en la consola de Amazon EFS.
 - Un punto de montaje. Se trata de un directorio en el contenedor Docker que monta el sistema de archivos.
 - Opciones de montaje. Incluyen detalles sobre cómo montar el sistema de archivos.

Note

Un sistema de archivos creado en Amazon EFS solo es compatible con las plataformas Linux.

Creación de una VPC con AWS CloudFormation

Cree su VPC con una plantilla de AWS CloudFormation.

1. Siga las instrucciones de [Plantilla de VPC de AWS CloudFormation \(p. 184\)](#) a fin de utilizar AWS CloudFormation para crear una VPC.

Note

La VPC que se crea con esta plantilla de AWS CloudFormation tiene dos subredes privadas y dos subredes públicas. Solo debe usar subredes privadas cuando utilice AWS CodeBuild para montar el sistema de archivos que creó en Amazon EFS. Si utiliza alguna de las subredes públicas, la compilación genera un error.

2. Inicie sesión en la AWS Management Console y abra la consola de Amazon VPC en <https://console.aws.amazon.com/vpc/>.
3. Elija la VPC que ha creado con AWS CloudFormation.
4. En la pestaña Description (Descripción), anote el nombre de la VPC y su ID. Ambos serán necesarios cuando cree su proyecto de AWS CodeBuild más adelante en este ejemplo.

Crear un sistema de archivos de Amazon Elastic File System con su VPC

Cree un sistema de archivos de Amazon EFS para este ejemplo con la VPC que creó anteriormente.

1. Inicie sesión en AWS Management Console y abra la consola de Amazon EFS en <https://console.aws.amazon.com/efs/>.
2. Seleccione Create file system (Crear sistema de archivos).
3. Desde VPC, elija el nombre de VPC que anotó anteriormente en este ejemplo.
4. Deje las zonas de disponibilidad asociadas a las subredes seleccionadas.
5. Elija Next Step (Paso siguiente).
6. En Agregue etiquetas, para el valor predeterminado NombreKey, en Valor, introduzca el nombre del sistema de archivos de Amazon EFS.
7. Mantenga Bursting (Transmisión por ráfagas) y General Purpose (Uso general) seleccionados como sus modos de desempeño y rendimiento predeterminados, y luego elija Next Step (Paso siguiente).
8. En Configure client access (Configurar acceso de cliente), seleccione Next Step (Paso siguiente).
9. Seleccione Create File System (Crear sistema de archivos).

Crear un proyecto de CodeBuild para utilizarlo con Amazon EFS

Cree un proyecto de AWS CodeBuild que utilice la VPC que creó anteriormente en este ejemplo. Al ejecutar la compilación, se monta el sistema de archivos de Amazon EFS creado anteriormente. A continuación, se almacena el archivo .jar que la aplicación Java ha creado en el directorio de puntos de montaje del sistema de archivos.

1. Abra el icono AWS CodeBuild Consola de en <https://console.aws.amazon.com/codesuite/codebuild/home>.

2. En el panel de navegación, elija Build projects (Proyectos de compilación) y después Create build project (Crear proyecto de compilación).
3. En Project name (Nombre del proyecto), introduzca un nombre para el proyecto.
4. En Source provider (Proveedor de código fuente), elija el repositorio que contiene la aplicación Java que desea compilar.
5. Introduzca información como, por ejemplo, una URL del repositorio, que CodeBuild utiliza para localizar la aplicación. Las opciones son diferentes para cada proveedor de código fuente. Para obtener más información, consulte [Choose source provider](#).
6. En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
7. En Operating system (Sistema operativo), elija Amazon Linux 2.
8. En Runtime(s), elija Standard (Estándar).
9. En Imagen, elija aws/codebuild/amazonlinux2-x86_64-standard:3.0.
10. En Environment type (Tipo de entorno), elija Linux.
11. Seleccione Privileged (Con privilegios).

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

12. En Service role (Rol de servicio), elija New service role (Nuevo rol de servicio). En Nombre del rol, introduzca un nombre para el rol que CodeBuild crea para usted.
13. Expanda Additional configuration (Configuración adicional).
14. En VPC, elija el ID de VPC.
15. En Subnets (Subredes), elija una o varias subredes privadas asociadas con su VPC. Debe utilizar subredes privadas en una compilación que monta un sistema de archivos de Amazon EFS. Si utiliza una subred pública, la compilación genera un error.
16. En Security groups (Grupos de seguridad), elija el grupo de seguridad predeterminado.
17. En File systems (Sistemas de archivos), escriba la siguiente información:
 - En Identifier (Identificador), introduzca un identificador único del sistema de archivos. Debe tener menos de 129 caracteres y contener solo caracteres alfanuméricos y guiones bajos. CodeBuild utiliza este identificador para crear una variable de entorno que identifica al sistema de archivos elástico. El formato de la variable de entorno es CODEBUILD_<file_system_identifier>, en mayúsculas. Por ejemplo, si introduce my_efs, la variable de entorno es CODEBUILD_MY_EFS.
 - En ID, elija el ID del sistema de archivos.
 - (Opcional) Introduzca un directorio en el sistema de archivos. CodeBuild monta este directorio. Si te vas Ruta de directorio en blanco, CodeBuild monta todo el sistema de archivos. La ruta es relativa a la raíz del sistema de archivos.
 - Para Punto de montaje, introduzca la ruta absoluta del directorio en el contenedor de compilación en el que se monta el sistema de archivos. Si este directorio no existe, CodeBuild lo crea durante la compilación.
 - (Opcional) Introduzca las opciones de montaje. Si te vas Opciones de montaje en blanco, CodeBuild utiliza sus opciones de montaje predeterminadas:

```
nfsvers=4.1
rsize=1048576
wsize=1048576
hard
timeo=600
retrans=2
```

Para obtener más información, consulte [Opciones recomendadas de montaje NFS](#) en la Guía del usuario de Amazon Elastic File System.

18. En Build specification (Especificación de compilación), elija Insert build commands (Insertar comandos de compilación) y, a continuación, elija Switch to editor (Cambiar a editor).
19. Introduzca los siguientes comandos de buildspec en el editor. Reemplace `<file_system_identifier>` por el identificador que ha especificado en el paso 17. Use letras mayúsculas (por ejemplo, CODEBUILD_MY_EFS).

```
version: 0.2
phases:
  install:
    runtime-versions:
      java: corretto11
  build:
    commands:
      - mvn compile -Dpgg.skip=true -Dmaven.repo.local=
        $CODEBUILD_<file_system_identifier>
```

20. Utilice los valores predeterminados para el resto de opciones y, a continuación, elija Create build project (Crear proyecto de compilación). Cuando la compilación se haya completado, se muestra la página de la consola para su proyecto.
21. Seleccione Start build.

Resumen de ejemplo de CodeBuild y Amazon EFS

Una vez compilado el proyecto de AWS CodeBuild:

- Tiene un archivo .jar creado por su aplicación Java que se integra en el sistema de archivos Amazon EFS en el directorio de puntos de montaje.
- Se crea una variable de entorno que identifica el sistema de archivos con el identificador de sistema de archivos que especificó al crear el proyecto.

Para obtener más información, consulte [Montaje de sistemas de archivos](#) en la Guía del usuario de Amazon Elastic File System.

Troubleshooting

A continuación se indican errores que podría surgir al configurar EFS con CodeBuild.

Temas

- [CLIENT_ERROR: Error al montar '127.0.0.1: /'. Permiso denegado \(p. 53\)](#)
- [CLIENT_ERROR: error al montar '127.0.0.1: /'. Restablecimiento de la conexión por pares \(p. 54\)](#)
- [VPC_CLIENT_ERROR: Error inesperado de EC2: UnauthorizedOperation \(p. 54\)](#)

CLIENT_ERROR: Error al montar '127.0.0.1: /'. Permiso denegado

No se admite la autorización de IAM para montar EFS con CodeBuild. Si utiliza una política de sistema de archivos EFS personalizada, tendrá que conceder acceso de lectura y escritura a todos los directores de IAM. Por ejemplo:

```
"Principal": {
  "AWS": "*"
}
```

```
}
```

CLIENT_ERROR: error al montar '127.0.0.1: /'. Restablecimiento de la conexión por pares

Existen dos causas posibles de este error:

- La subred de VPC de CodeBuild se encuentra en una zona de disponibilidad distinta a la del destino de montaje de EFS. Puede resolverlo agregando una subred de VPC en la misma zona de disponibilidad que el destino de montaje de EFS.
- El grupo de seguridad no cuenta con permisos para comunicarse con EFS. Puede resolver esto agregando una regla de entrada para permitir todo el tráfico de la VPC (agregue el bloque CIDR principal de la VPC) o del propio grupo de seguridad.

VPC_CLIENT_ERROR: Error inesperado de EC2: UnauthorizedOperation

Este error se produce cuando todas las subredes de la configuración de la VPC para el proyecto CodeBuild son subredes públicas. Debe tener al menos una subred privada en la VPC para garantizar la conectividad de red.

Ejemplo de CodeDeploy para CodeBuild

Este ejemplo indica a AWS CodeBuild que use Maven para producir el resultado de la compilación como un único archivo JAR denominado `my-app-1.0-SNAPSHOT.jar`. Este ejemplo utiliza a continuación CodeDeploy para implementar el archivo JAR en una instancia de Amazon Linux. También puede utilizar AWS CodePipeline para automatizar el uso de CodeDeploy para implementar el archivo JAR en una instancia de Amazon Linux. Este ejemplo se basa en el tema [Maven in 5 Minutes](#) ("Maven en 5 minutos") del sitio web de Apache Maven.

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon S3, AWS KMS, CloudWatch Logs y Amazon EC2. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service](#), [Precios de Amazon CloudWatch](#), y [Precios de Amazon EC2](#).

Ejecución del ejemplo

Para ejecutar este ejemplo

1. Descargue e instale Maven. Para obtener más información, consulte [Downloading Apache Maven e Installing Apache Maven](#) en el sitio web de Apache Maven.
2. Cambie a un directorio vacío del equipo o instancia local y, a continuación, ejecute este comando Maven.

```
mvn archetype:generate -DgroupId=com.mycompany.app -DartifactId=my-app -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
```

Si el comando se ejecuta correctamente, se creará esta estructura de directorios y archivos.

```
.
### my-app
### pom.xml
### src
```

```
### main
#   ### java
#       ### com
#           ### mycompany
#               ### app
#                   ### App.java
### test
    ### java
        ### com
            ### mycompany
                ### app
                    ### AppTest.java
```

3. Cree un archivo con este contenido. Asigne un nombre al archivo `buildspec.yml` y añádalo al directorio `my-app`.

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto8
  build:
    commands:
      - echo Build started on `date`
      - mvn test
  post_build:
    commands:
      - echo Build completed on `date`
      - mvn package
artifacts:
  files:
    - target/my-app-1.0-SNAPSHOT.jar
    - appspec.yml
  discard-paths: yes
```

4. Cree un archivo con este contenido. Asigne un nombre al archivo `appspec.yml` y añádalo al directorio `my-app`.

```
version: 0.0
os: linux
files:
  - source: ./my-app-1.0-SNAPSHOT.jar
    destination: /tmp
```

Cuando termine, la estructura de directorios y archivos debería ser similar a la siguiente.

```
.
### my-app
    ### buildspec.yml
    ### appspec.yml
    ### pom.xml
    ### src
        ### main
        #   ### java
        #       ### com
        #           ### mycompany
        #               ### app
        #                   ### App.java
        ### test
            ### java
                ### com
                    ### mycompany
```

```
### app
### AppTest.java
```

5. Cree un archivo ZIP que contenga la estructura de directorios y archivos dentro demy-app, a continuación, cargue el archivo ZIP en un tipo de repositorio de código fuente compatible conAWS CodeBuild CodeDeploy, como un bucket de entrada de S3 o un repositorio de GitHub o Bitbucket.

Important

Si desea utilizar CodePipeline para implementar el artefacto de salida de la compilación resultante, no puede cargar el código fuente en un repositorio de Bitbucket.

No añada my-app al archivo ZIP, solo los directorios y archivos incluidos en my-app. El archivo ZIP debe incluir estos directorios y archivos:

```
.
### CodeDeploySample.zip
### buildspec.yml
### appspec.yml
### pom.xml
### src
### main
#   ### java
#       ### com
#           ### mycompany
#               ### app
#                   ### App.java
### test
### java
### com
### mycompany
### app
### AppTest.java
```

6. Cree un proyecto de compilación siguiendo los pasos de [Creación de un proyecto de compilación \(p. 197\)](#).

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando create-project tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{
  "name": "sample-codedeploy-project",
  "source": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-input-bucket/CodeDeploySample.zip"
  },
  "artifacts": {
    "type": "S3",
    "location": "codebuild-region-ID-account-ID-output-bucket",
    "packaging": "ZIP",
    "name": "CodeDeployOutputArtifact.zip"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:4.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "arn:aws:iam::account-ID:role/role-name",
  "encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

7. Si va a implementar el artefacto de salida de la compilación con CodeDeploy, siga los pasos que se indican en [Ejecutar una compilación \(p. 281\)](#). De lo contrario, omita este paso. (Esto es así porque,

si va a implementar el artefacto de salida de la compilación con CodePipeline, CodePipeline utiliza CodeBuild para ejecutar la compilación de forma automática).

8. Complete los pasos de configuración para utilizar CodeDeploy, incluidos los siguientes:
 - Conceder al usuario de IAM acceso a CodeDeploy y a AWS servicios y acciones de los que depende CodeDeploy. Para obtener más información, consulte [Suministro de un usuario de IAM](#) en la AWS CodeDeploy Guía del usuario de.
 - Cree o identifique un rol de servicio para habilitar CodeDeploy para que identifique las instancias donde se implementará el artefacto de salida de la compilación. Para obtener más información, consulte [Creación de un rol de servicio para CodeDeploy](#) en la AWS CodeDeploy Guía del usuario de.
 - Cree o identifique un perfil de instancia de IAM para permitir que las instancias tengan acceso al bucket de entrada de S3 o al repositorio de GitHub que contiene el artefacto de salida de la compilación. Para obtener más información, consulte [Crear un perfil de instancia de IAM para sus instancias Amazon EC2](#) en la AWS CodeDeploy Guía del usuario de.
9. Cree o identifique una instancia de Amazon Linux compatible con CodeDeploy donde se vaya a implementar el artefacto de salida de la compilación. Para obtener más información, consulte [Trabajo con instancias para CodeDeploy](#) en la AWS CodeDeploy Guía del usuario de.
10. Cree o identifique una aplicación CodeDeploy y un grupo de implementación. Para obtener más información, consulte [Creación de una aplicación con CodeDeploy](#) en la AWS CodeDeploy Guía del usuario de.
11. Implemente el artefacto de salida de la compilación en la instancia.

Para implementarlo con CodeDeploy, consulte [Implementación de una revisión con CodeDeploy](#) en la AWS CodeDeploy Guía del usuario de.

Para implementarlo con CodePipeline, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#).

12. Para encontrar el artefacto de salida de la compilación una vez completada la implementación, inicie sesión en la instancia y busque en el directorio `/tmp` un archivo denominado `my-app-1.0-SNAPSHOT.jar`.

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

AWS CodePipelineintegración con CodeBuild y compilaciones por lotes

AWS CodeBuildahora admite compilaciones por lotes. Este ejemplo muestra cómo usarAWS CodePipelinepara crear un proyecto de compilación que utilice compilaciones por lotes.

Puede utilizar un archivo con formato JSON que defina la estructura de la canalización y, después, usarlo con la AWS CLI para crear la canalización. Para obtener más información, consulte [AWS CodePipelineReferencia de estructura de la canalización](#) en la AWS CodePipelineGuía del usuario de.

Creación Batch con artefactos individuales

Utilice el archivo JSON siguiente como ejemplo de una estructura de canalización que crea una compilación por lotes con artefactos independientes. Para habilitar las compilaciones por lotes en CodePipeline, configure laBatchEnabledparámetro deconfigurationobjeto detrue.

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::account-id:role/my-AWS-CodePipeline-service-role-name",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source1",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "source1"
              }
            ],
            "configuration": {
              "S3Bucket": "my-input-bucket-name",
              "S3ObjectKey": "my-source-code-file-name.zip"
            },
            "runOrder": 1
          },
          {
            "inputArtifacts": [],
            "name": "Source2",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "source2"
              }
            ],
            "configuration": {
              "S3Bucket": "my-other-input-bucket-name",
              "S3ObjectKey": "my-other-source-code-file-name.zip"
            },
            "runOrder": 1
          }
        ]
      },
      {
        "name": "Build",
        "actions": [
          {
            "inputArtifacts": [
              {
                "name": "source1"
              }
            ],
            "name": "Build",
            "actionTypeId": {
              "category": "Build",
              "owner": "AWS",
              "version": "1",
              "provider": "CodeBuild"
            },
            "outputArtifacts": [
              {
                "name": "build"
              }
            ],
            "configuration": {
              "BuildSpec": "buildspec.yml"
            },
            "runOrder": 1
          }
        ]
      }
    ]
  }
}
```

```
        "name": "source2"
      },
    ],
    "name": "Build",
    "actionTypeId": {
      "category": "Build",
      "owner": "AWS",
      "version": "1",
      "provider": "AWS CodeBuild"
    },
    "outputArtifacts": [
      {
        "name": "build1"
      },
      {
        "name": "build1_artifact1"
      },
      {
        "name": "build1_artifact2"
      },
      {
        "name": "build2_artifact1"
      },
      {
        "name": "build2_artifact2"
      }
    ],
    "configuration": {
      "ProjectName": "my-build-project-name",
      "PrimarySource": "source1",
      "BatchEnabled": "true"
    },
    "runOrder": 1
  }
]
}
},
"artifactStore": {
  "type": "S3",
  "location": "AWS-CodePipeline-internal-bucket-name"
},
"name": "my-pipeline-name",
"version": 1
}
}
```

A continuación se muestra un ejemplo de un archivo buildspec de CodeBuild que funcionará con esta configuración de canalización.

```
version: 0.2
batch:
  build-list:
    - identifier: build1
      env:
        compute-type: BUILD_GENERAL1_SMALL
    - identifier: build2
      env:
        compute-type: BUILD_GENERAL1_MEDIUM
phases:
  build:
    commands:
      - echo 'file' > output_file
```

```
artifacts:
  files:
    - output_file
  secondary-artifacts:
    artifact1:
      files:
        - output_file
    artifact2:
      files:
        - output_file
```

Los nombres de los artefactos de salida especificados en el archivo JSON de la canalización deben coincidir con el identificador de las compilaciones y los artefactos definidos en el archivo buildspec. La sintaxis es **Identificador de compilación** para los artefactos principales, y **Identificador de compilación_ArtifactIdentifier** para los artefactos secundarios.

Por ejemplo, para el nombre del artefacto de salida `build1`, CodeBuild cargará el artefacto principal `build1` a la ubicación `build1`. Para nombre de salida `build1_artifact1`, CodeBuild cargará el artefacto secundario `artifact1build1` a la ubicación `build1_artifact1`, y así sucesivamente. Si solo se especifica una ubicación de salida, el nombre debe ser **Identificador de compilación**. Solo la .

Una vez creado el archivo JSON, puede crear la canalización. Utilice la AWS CLI para ejecutar el comando `create-pipeline` y transferir el archivo al parámetro `--cli-input-json`. Para obtener más información, consulte [Crear una canalización \(CLI\)](#) en la [AWS CodePipeline Guía del usuario](#) de.

Creación Batch con artefactos combinados

Utilice el archivo JSON siguiente como ejemplo de una estructura de canalización que crea una compilación por lotes con artefactos combinados. Para habilitar las compilaciones por lotes en CodePipeline, configure el `BatchEnabled` parámetro de `configuration` objeto de `true`. Para combinar los artefactos de compilación en la misma ubicación, configure el `CombineArtifacts` parámetro de `configuration` objeto de `true`.

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::account-id:role/my-AWS-CodePipeline-service-role-name",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source1",
            "actionTypeId": {
              "category": "Source",
              "owner": "AWS",
              "version": "1",
              "provider": "S3"
            },
            "outputArtifacts": [
              {
                "name": "source1"
              }
            ],
            "configuration": {
              "S3Bucket": "my-input-bucket-name",
              "S3ObjectKey": "my-source-code-file-name.zip"
            },
            "runOrder": 1
          },
          {

```

```
        "inputArtifacts": [],
        "name": "Source2",
        "actionTypeId": {
            "category": "Source",
            "owner": "AWS",
            "version": "1",
            "provider": "S3"
        },
        "outputArtifacts": [
            {
                "name": "source2"
            }
        ],
        "configuration": {
            "S3Bucket": "my-other-input-bucket-name",
            "S3ObjectKey": "my-other-source-code-file-name.zip"
        },
        "runOrder": 1
    }
]
},
{
    "name": "Build",
    "actions": [
        {
            "inputArtifacts": [
                {
                    "name": "source1"
                },
                {
                    "name": "source2"
                }
            ],
            "name": "Build",
            "actionTypeId": {
                "category": "Build",
                "owner": "AWS",
                "version": "1",
                "provider": "AWS CodeBuild"
            },
            "outputArtifacts": [
                {
                    "name": "output1 "
                }
            ],
            "configuration": {
                "ProjectName": "my-build-project-name",
                "PrimarySource": "source1",
                "BatchEnabled": "true",
                "CombineArtifacts": "true"
            },
            "runOrder": 1
        }
    ]
}
],
"artifactStore": {
    "type": "S3",
    "location": "AWS-CodePipeline-internal-bucket-name"
},
"name": "my-pipeline-name",
"version": 1
}
}
```

A continuación se muestra un ejemplo de un archivo buildspec de CodeBuild que funcionará con esta configuración de canalización.

```
version: 0.2
batch:
  build-list:
    - identifier: build1
      env:
        compute-type: BUILD_GENERAL1_SMALL
    - identifier: build2
      env:
        compute-type: BUILD_GENERAL1_MEDIUM

phases:
  build:
    commands:
      - echo 'file' > output_file

artifacts:
  files:
    - output_file
```

Si los artefactos combinados están habilitados para la compilación por lotes, solo se permite una salida. CodeBuild combinará los principales artefactos de todas las compilaciones en un solo archivo ZIP.

Una vez creado el archivo JSON, puede crear la canalización. Utilice la AWS CLI para ejecutar el comando `create-pipeline` y transferir el archivo al parámetro `--cli-input-json`. Para obtener más información, consulte [Crear una canalización \(CLI\)](#) en la AWS CodePipeline Guía del usuario de.

AWS CodePipelineEjemplo de integración de integración con CodeBuild y varios orígenes de entrada y varios artefactos

Un proyecto de AWS CodeBuild puede aceptar varios orígenes de entrada. También puede crear más de un artefacto de salida. En este ejemplo, se muestra cómo se utiliza AWS CodePipeline para crear un proyecto de compilación que utiliza varios orígenes de entrada a fin de crear varios artefactos de salida. Para obtener más información, consulte [Ejemplo de varios orígenes de entrada y varios artefactos de salida \(p. 129\)](#).

Puede utilizar un archivo con formato JSON que defina la estructura de la canalización y, después, usarlo con la AWS CLI para crear la canalización. Utilice el archivo JSON siguiente como ejemplo de una estructura de canalización que crea una compilación con más de un origen de entrada y más de un artefacto de salida. Más adelante en este ejemplo podrá ver cómo dicho archivo especifica las distintas entradas y salidas. Para obtener más información, consulte [Referencia de estructura de canalización de CodePipeline](#) en la AWS CodePipeline Guía del usuario de.

```
{
  "pipeline": {
    "roleArn": "arn:aws:iam::account-id:role/my-AWS-CodePipeline-service-role-name",
    "stages": [
      {
        "name": "Source",
        "actions": [
          {
            "inputArtifacts": [],
            "name": "Source1",
            "actionTypeId": {
```

```
        "category": "Source",
        "owner": "AWS",
        "version": "1",
        "provider": "S3"
    },
    "outputArtifacts": [
        {
            "name": "source1"
        }
    ],
    "configuration": {
        "S3Bucket": "my-input-bucket-name",
        "S3ObjectKey": "my-source-code-file-name.zip"
    },
    "runOrder": 1
},
{
    "inputArtifacts": [],
    "name": "Source2",
    "actionTypeId": {
        "category": "Source",
        "owner": "AWS",
        "version": "1",
        "provider": "S3"
    },
    "outputArtifacts": [
        {
            "name": "source2"
        }
    ],
    "configuration": {
        "S3Bucket": "my-other-input-bucket-name",
        "S3ObjectKey": "my-other-source-code-file-name.zip"
    },
    "runOrder": 1
}
]
},
{
    "name": "Build",
    "actions": [
        {
            "inputArtifacts": [
                {
                    "name": "source1"
                },
                {
                    "name": "source2"
                }
            ],
            "name": "Build",
            "actionTypeId": {
                "category": "Build",
                "owner": "AWS",
                "version": "1",
                "provider": "AWS CodeBuild"
            },
            "outputArtifacts": [
                {
                    "name": "artifact1"
                },
                {
                    "name": "artifact2"
                }
            ],
            "configuration": {
```

```
        "ProjectName": "my-build-project-name",
        "PrimarySource": "source1"
    },
    "runOrder": 1
}
]
}
],
"artifactStore": {
    "type": "S3",
    "location": "AWS-CodePipeline-internal-bucket-name"
},
"name": "my-pipeline-name",
"version": 1
}
}
```

En este archivo JSON:

- Debe designar uno de los orígenes de entrada como `PrimarySource`. Este origen es el directorio en el que CodeBuild busca el archivo `buildspec` y lo ejecuta. La palabra clave `PrimarySource` se utiliza para especificar el origen principal en la sección `configuration` de la fase CodeBuild en el archivo JSON.
- Cada origen de entrada se instala en su propio directorio. Este directorio se almacena en la variable de entorno integrado `$CODEBUILD_SRC_DIR` para el origen principal y `$CODEBUILD_SRC_DIR_yourInputArtifactName` para todos los demás orígenes. Para la canalización de este ejemplo, los dos directorios de origen de entrada son `$CODEBUILD_SRC_DIR` y `$CODEBUILD_SRC_DIR_source2`. Para obtener más información, consulte [Variables de entorno en los entornos de compilación \(p. 172\)](#).
- Los nombres de los artefactos de salida especificados en el archivo JSON de la canalización deben coincidir con los de los artefactos secundarios definidos en el archivo `buildspec`. Esta canalización utiliza el archivo `buildspec` siguiente. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

```
version: 0.2

phases:
  build:
    commands:
      - touch source1_file
      - cd $CODEBUILD_SRC_DIR_source2
      - touch source2_file

artifacts:
  files:
    - '**/*'
  secondary-artifacts:
    artifact1:
      base-directory: $CODEBUILD_SRC_DIR
      files:
        - source1_file
    artifact2:
      base-directory: $CODEBUILD_SRC_DIR_source2
      files:
        - source2_file
```

Una vez creado el archivo JSON, puede crear la canalización. Utilice la AWS CLI para ejecutar el comando `create-pipeline` y transferir el archivo al parámetro `--cli-input-json`. Para obtener más información, consulte [Crear una canalización \(CLI\)](#) en la AWS CodePipeline Guía del usuario de.

Usar AWS Config con ejemplo de CodeBuild

AWS Config proporciona un inventario de sus recursos de AWS y un historial de cambios de configuración en estos recursos. AWS Config admite AWS CodeBuild como un recurso de AWS, lo que significa que el servicio puede realizar un seguimiento de sus proyectos de CodeBuild. Para obtener más información sobre AWS Config, consulte [¿Qué es AWS Config?](#) en la Guía para desarrolladores de AWS Config.

Puede consultar la siguiente información sobre los recursos de CodeBuild en el [Inventory de la AWS Config Console](#) de :

- Una escala de tiempo de los cambios de configuración de CodeBuild.
- Detalles de configuración para cada proyecto de CodeBuild.
- Relaciones con otros recursos de AWS.
- Una lista de cambios en sus proyectos de CodeBuild.

Los procedimientos incluidos en este tema muestran cómo configurar AWS Config para que busque y vea los proyectos de CodeBuild.

Temas

- [Prerequisites](#) (p. 65)
- [Configuración de AWS Config](#) (p. 65)
- [Consulta de proyectos de AWS CodeBuild](#) (p. 65)
- [Ver los detalles de configuración de AWS CodeBuild en la consola de AWS Config](#) (p. 66)

Prerequisites

Cree un proyecto de AWS CodeBuild. Para obtener instrucciones, consulte [Creación de un proyecto de compilación](#) (p. 197).

Configuración de AWS Config.

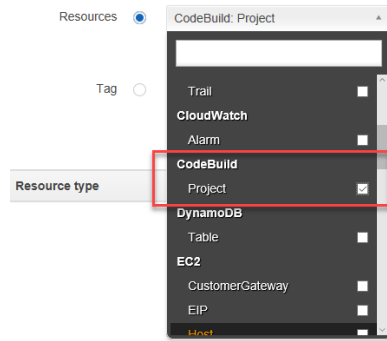
- [Configuración de AWS Config \(consola\)](#)
- [Configuración de AWS Config \(AWS CLI\)](#)

Note

Después de completar la instalación, pueden pasar 10 minutos hasta que los proyectos de AWS CodeBuild aparecen en la consola de AWS Config.

Consulta de proyectos de AWS CodeBuild

1. Inicie sesión en la [AWS Console de administración](#) y abra la [AWS Config Console](#) de en <https://console.aws.amazon.com/config>.
2. En la página [Resource inventory](#), elija [Resources](#). Desplácese hacia abajo y seleccione la casilla [CodeBuild project](#).



3. Elija Look up (Buscar).
4. Una vez añadida la lista de proyectos de CodeBuild, elija el enlace del nombre del proyecto de CodeBuild en Escala de tiempo de configuración column.

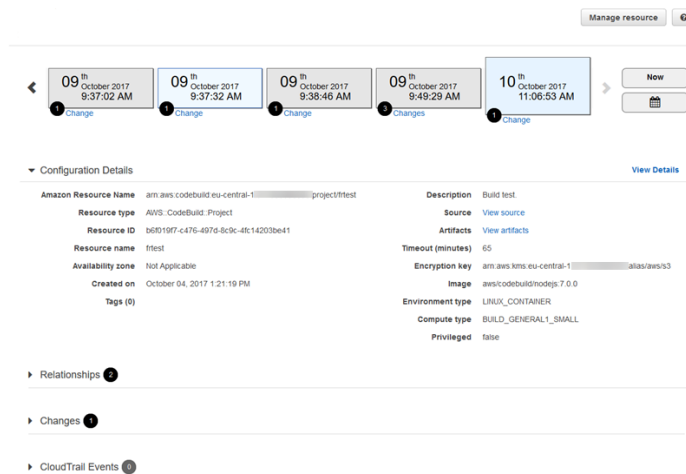
Ver los detalles de configuración de AWS CodeBuild en la consola de AWS Config

Cuando busca recursos en el Inventory de En la página, puede elegir la AWS Config escala de tiempo para ver detalles de su proyecto de CodeBuild. La página de detalles de un recurso proporciona información sobre la configuración, las relaciones y el número de cambios realizados en dicho recurso.

Los bloques de la parte superior de la página se denominan colectivamente escala de tiempo. La escala de tiempo muestra la fecha y la hora a las que se realizó el registro.

Para obtener más información, consulte [Visualización de los detalles de la configuración en la consola de AWS Config](#) en la guía para desarrolladores de AWS Config.

Ejemplo de proyecto CodeBuild en AWS Config:



AWS Elastic Beanstalk ejemplo de CodeBuild

En esta muestra se utiliza AWS CodeBuild con Maven para producir un único archivo WAR denominado `ROOT.war` como resultado de compilación. Después se implementa el archivo WAR en las instancias de un entorno de AWS Elastic Beanstalk.

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon S3, AWS KMS, CloudWatch Logs y Amazon EC2. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), [Precios de Amazon CloudWatch](#), y [Precios de Amazon EC2](#).

Crear el código fuente

En esta sección, usará Maven para producir el código fuente. Posteriormente, usará CodeBuild para crear un archivo WAR basado en este código fuente.

1. Descargue e instale Maven. Para obtener información, consulte [Downloading Apache Maven](#) e [Installing Apache Maven](#) en el sitio web de Apache Maven.
2. Cambie a un directorio vacío del equipo o instancia local y, a continuación, ejecute este comando Maven.

```
mvn archetype:generate "-DgroupId=com.mycompany.app" "-DartifactId=ROOT" "-DarchetypeArtifactId=maven-archetype-webapp" "-DinteractiveMode=false"
```

Si el comando se ejecuta correctamente, se creará esta estructura de directorios y archivos.

```
.
### ROOT
  ### pom.xml
  ### src
    ### main
      ### resources
      ### webapp
        ### WEB-INF
        #   ### web.xml
        ### index.jsp
```

3. Cree un subdirectorio llamado `.ebextensions` en el directorio `ROOT`. En el subdirectorio `.ebextensions`, cree un archivo denominado `fix-path.config` con este contenido.

```
container_commands:
  fix_path:
    command: "unzip ROOT.war 2>&1 > /var/log/my_last_deploy.log"
```

Después de ejecutar Maven, continúe con uno de los siguientes escenarios:

- [Escenario A: Ejecute CodeBuild manualmente e implemente en Elastic Beanstalk manualmente \(p. 67\)](#)
- [Escenario B: Utilice CodePipeline para ejecutar CodeBuild e implementar en Elastic Beanstalk \(p. 70\)](#)
- [Escenario C: Utilice la CLI de Elastic Beanstalk para ejecutar AWS CodeBuild e implementar en un entorno de Elastic Beanstalk \(p. 72\)](#)

Escenario A: Ejecute CodeBuild manualmente e implemente en Elastic Beanstalk manualmente

En este escenario, debe crear y cargar el código fuente. A continuación, utiliza el AWS CodeBuild y AWS Elastic Beanstalk Consolas para compilar el código fuente, crear una aplicación y un entorno de Elastic Beanstalk e implementar el resultado de la compilación en el entorno.

Paso a1: Agregar archivos al código fuente

En este paso, agregará un archivo de configuración de Elastic Beanstalk y creará un archivo buildspec en el código en [Crear el código fuente \(p. 67\)](#). A continuación, cargue el código fuente en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket

1. Cree un archivo denominado `buildspec.yml` con el siguiente contenido. Guarde el archivo en el directorio `ROOT`.

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto11
  post_build:
    commands:
      - mvn package
      - mv target/ROOT.war ROOT.war
artifacts:
  files:
    - ROOT.war
    - .ebextensions/**/*
```

2. La estructura de archivos debe ser como la siguiente.

```
.
### ROOT
### .ebextensions
#   ### fix-path.config
### src
#   ### main
#       ### resources
#           ### webapp
#               ### WEB-INF
#                   #   ### web.xml
#                       ### index.jsp
### buildspec.yml
### pom.xml
```

3. Suba el contenido de la `ROOT` en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket.

Important

No cargar `ROOT`, solo los directorios y archivos incluidos en `ROOT`.

Si utiliza un bucket de entrada de S3, debe tener control de versiones. Asegúrese de crear un archivo ZIP que contenga la estructura de directorios y archivos y, a continuación, cárguelo en el bucket de entrada. No añada `ROOT` al archivo ZIP, solo los directorios y archivos incluidos en `ROOT`. Para obtener más información, consulte [Cómo configurar el control de versiones en un bucket](#) en la Guía para desarrolladores de Amazon S3.

Paso a2: Cree el proyecto de compilación y ejecute la compilación

En este paso, utilizará la consola de AWS CodeBuild para crear un proyecto de compilación y después ejecutará una compilación.

1. Cree o elija un bucket de salida de S3 para almacenar el resultado de la compilación. Si va a almacenar el código fuente en un bucket de entrada de S3, el bucket de salida debe estar en la misma región de AWS que el bucket de entrada.

2. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.

UsarAWSSelector de regiones para elegir unAWSRegión en la que se admite CodeBuild. Debe ser la misma región en la que se encuentra el bucket de salida de S3.

3. Cree un proyecto de compilación y, a continuación, ejecute una compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
 - Para Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - ParaImage (Imagen), eligeaws/codebuild/amazonlinux2-x86_64-standard:3.0.
 - En Artifacts (Artefactos):
 - En Type (Tipo), elija Amazon S3.
 - En Bucket name (Nombre del bucket), escriba el nombre de un bucket de S3.
 - En Name (Nombre), escriba un nombre de archivo de salida de la compilación que sea fácil de recordar. Incluya la extensión .zip.
 - En Artifacts packaging, elija Zip.

Paso a3: Crear la aplicación y el entorno e implementar

En este paso, va a utilizar la consola de AWS Elastic Beanstalk para crear una aplicación y un entorno. Como parte de la creación del entorno, implementará el resultado de la compilación del paso anterior en el entorno.

1. Abra el iconoAWS Elastic BeanstalkConsola de en<https://console.aws.amazon.com/elasticbeanstalk>.

UsarAWSSelector de regiones para elegir elAWSRegión en la que se encuentra el bucket de salida de S3.

2. Cree una aplicación de Elastic Beanstalk. Para obtener más información, consulte [Administración y configuración de aplicaciones de AWS Elastic Beanstalk](#) en la Guía para desarrolladores de AWS Elastic Beanstalk.
3. Cree un entorno de Elastic Beanstalk para esta aplicación. Para obtener más información, consulte [El asistente de creación de nuevo entorno](#) en la Guía para desarrolladores de AWS Elastic Beanstalk. Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
 - En Platform, elija Tomcat.
 - En Application code, seleccione Upload your code y, luego, Upload. En Source code origin (Origen del código fuente), elija Public S3 URL (URL de S3 pública) y escriba la URL completa del archivo ZIP de salida de la compilación en el bucket de salida. Seleccione Upload.
4. Después de que Elastic Beanstalk implemente el resultado de la compilación en el entorno, podrá ver los resultados en un navegador web. Vaya a la URL del entorno de la instancia (por ejemplo, <http://my-environment-name.random-string.region-ID.elasticbeanstalk.com>). El navegador web debe mostrar el texto Hello World!.

Escenario B: Utilice CodePipeline para ejecutar CodeBuild e implementar en Elastic Beanstalk

En este escenario, realizará los pasos para preparar y cargar el código fuente. Cree un proyecto de compilación con CodeBuild y un entorno de y una aplicación de Elastic Beanstalk con AWS Elastic Beanstalk consola de . A continuación, usará la consola de AWS CodePipeline para crear una canalización. Después de crear la canalización, CodePipeline compila el código fuente e implementa el resultado de la compilación en el entorno.

Paso b1: Agregar un archivo de especificación de compilación al código fuente

En este paso, creará un archivo `buildspec.yml` y lo agregará al código que creó en [Crear el código fuente \(p. 67\)](#). A continuación, cargue el código fuente en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket

1. Cree un archivo denominado `buildspec.yml` con el siguiente contenido. Guarde el archivo en el directorio `ROOT`.

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto11
  post_build:
    commands:
      - mvn package
      - mv target/ROOT.war ROOT.war
artifacts:
  files:
    - ROOT.war
    - .ebextensions/**/*
```

2. La estructura de archivos debe ser como la siguiente.

```
.
### ROOT
### .ebextensions
#   ### fix-path.config
### src
#   ### main
#   ### resources
#   ### webapp
#       ### WEB-INF
#       #   ### web.xml
#       ### index.jsp
### buildspec.yml
### pom.xml
```

3. Suba el contenido de la `ROOT` en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket.

Important

No cargar `ROOT`, solo los directorios y archivos incluidos en `ROOT`.

Si utiliza un bucket de entrada de S3, debe tener control de versiones. Asegúrese de crear un archivo ZIP que contenga la estructura de directorios y archivos y, a continuación, cárguelo en el bucket de entrada. No añada `ROOT` al archivo ZIP, solo los directorios y archivos incluidos en `ROOT`. Para obtener más información, consulte [Cómo configurar el control de versiones en un bucket](#) en la Guía para desarrolladores de Amazon S3.

Paso b2: Creación de un proyecto de compilación

En este paso, creará un proyecto de compilación de AWS CodeBuild que usará con la canalización.

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
 - Para Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - ParaImage (Imagen), eligeaws/codebuild/amazonlinux2-x86_64-standard:3.0.
 - En Artifacts (Artefactos):
 - En Type (Tipo), elija Amazon S3.
 - En Bucket name (Nombre del bucket), escriba el nombre de un bucket de S3.
 - En Name (Nombre), escriba un nombre de archivo de salida de la compilación que sea fácil de recordar. Incluya la extensión .zip.
 - En Artifacts packaging, elija Zip.

Paso b3: Cree una aplicación y un entorno de Elastic Beanstalk

En este paso, creará una aplicación y un entorno de Elastic Beanstalk para usarlos con CodePipeline.

1. Abra la consola de Elastic Beanstalk en<https://console.aws.amazon.com/elasticbeanstalk/>.
2. Utilice la consola de AWS Elastic Beanstalk para crear una aplicación. Para obtener más información, consulte [Administración y configuración de aplicaciones de AWS Elastic Beanstalk](#) en la Guía para desarrolladores de AWS Elastic Beanstalk.
3. Utilice la consola de AWS Elastic Beanstalk para crear un entorno. Para obtener más información, consulte [El asistente de creación de nuevo entorno](#) en la Guía para desarrolladores de AWS Elastic Beanstalk. Deje todas las opciones con sus valores predeterminados, excepto Platform (Plataforma). En Platform, elija Tomcat.

Paso b4: Crear la canalización e implementar

En este paso, va a utilizar la consola de AWS CodePipeline para crear una canalización. Después de crear y ejecutar la canalización, CodePipeline utiliza CodeBuild para compilar el código fuente. A continuación, CodePipeline utiliza Elastic Beanstalk para implementar el resultado de la compilación en el entorno.

1. Cree o identifique un rol de servicio que CodePipeline, CodeBuild y Elastic Beanstalk puedan utilizar para obtener acceso a los recursos en su nombre. Para obtener más información, consulte [Prerequisites](#) (p. 408).
2. Abra la consola de CodePipeline en<https://console.aws.amazon.com/codesuite/codepipeline/home>.

UsarAWSSelector de regiones para elegir unAWSRegión en la que se admite CodeBuild. Si va a almacenar el código fuente en un bucket de entrada de S3, el bucket de salida debe estar en la misma región de AWS que el bucket de entrada.

3. Cree una canalización. Para obtener información, consulte [Crear una canalización que use CodeBuild \(consola de CodePipeline\)](#) (p. 409). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.

- En Add build stage (Agregar una fase de compilación), en Build provider (Proveedor de compilación), elija AWS CodeBuild. En Project name (Nombre del proyecto), seleccione el proyecto de compilación que acaba de crear.
 - En Add deploy stage (Añadir fase, de implementación), en Deploy provider (Proveedor de implementación), elija AWS Elastic Beanstalk.
 - Para Nombre de la aplicación, elija la aplicación Elastic Beanstalk que acaba de crear.
 - En Environment name (Nombre del entorno), elija el entorno que acaba de crear.
4. Una vez que se haya ejecutado la canalización correctamente, podrá ver los resultados en un navegador web. Vaya a la URL del entorno de la instancia (por ejemplo, <http://my-environment-name.random-string.region-ID.elasticbeanstalk.com>). El navegador web debe mostrar el texto Hello World!.

Ahora, siempre que realice cambios en el código fuente y cargue estos cambios en el bucket de entrada de S3 original o en el repositorio de CodeCommit, GitHub o Bitbucket, CodePipeline detectará el cambio y ejecutará de nuevo la canalización. Esto hará que CodeBuild vuelva a compilar, lo que a su vez provocará que Elastic Beanstalk implemente el resultado recompilado en el entorno.

Escenario C: Utilice la CLI de Elastic Beanstalk para ejecutarAWS CodeBulde implementar en un entorno de Elastic Beanstalk

En este escenario, realizará los pasos para preparar y cargar el código fuente. A continuación, ejecutará la CLI de Elastic Beanstalk para crear una aplicación y un entorno de Elastic Beanstalk, utilice CodeBuild para compilar el código fuente y, a continuación, implementará el resultado de la compilación en el entorno. Para obtener más información, consulte [Uso de la CLI de EB con CodeBuild](#) en laAWS Elastic BeanstalkGuía para desarrolladores.

Paso c1: Agregar archivos al código fuente

En este paso, agregará un archivo de configuración de Elastic Beanstalk y creará un archivo buildspec en el código que creó en [Crear el código fuente \(p. 67\)](#). Asimismo, creará o identificará un rol de servicio para el archivo buildspec.

1. Cree o identifique un rol de servicio que Elastic Beanstalk y la CLI puedan usar en su nombre. Para obtener información, consulte [Cree un rol de servicio de CodeBuild \(p. 396\)](#).
2. Cree un archivo denominado `buildspec.yml` con el siguiente contenido. Guarde el archivo en el directorio `ROOT`.

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto11
  post_build:
    commands:
      - mvn package
      - mv target/ROOT.war ROOT.war
artifacts:
  files:
    - ROOT.war
    - .ebextensions/**/*
eb_codebuild_settings:
  CodeBuildServiceRole: my-service-role-name
```

```
ComputeType: BUILD_GENERAL1_SMALL
Image: aws/codebuild/standard:4.0
Timeout: 60
```

En el código anterior, sustituya *my-service-role-name* por el nombre del rol de servicio que creó o identificó anteriormente.

3. La estructura de archivos debe ser como la siguiente.

```
.
### ROOT
### .ebextensions
#   ### fix-path.config
### src
#   ### main
#   ### resources
#       ### webapp
#           ### WEB-INF
#               ### web.xml
#               ### index.jsp
### buildspec.yml
### pom.xml
```

Paso c2: Instalación y ejecución de la CLI de EB

1. Si aún no lo ha hecho, instale y configure la CLI de EB en el mismo equipo o instancia en el que creó el código fuente. Para obtener información, consulte [Instale la interfaz de línea de comandos de Elastic Beanstalk \(CLI de EB\)](#) y [Configuración de la CLI de EB](#) en la [AWS Elastic Beanstalk Guía para desarrolladores](#).
2. Desde la línea de comandos o un terminal, ejecute el comando `cd` u otro similar para cambiar al directorio (*root directory name*)/ROOT. Ejecute el comando `eb init` para configurar la CLI de EB.

```
eb init
```

Cuando se le solicite:

- Seleccione una [AWS Región](#) donde [AWS CodeBuild](#) está soportada y en la que desea crear su aplicación y entorno de Elastic Beanstalk.
 - Cree una aplicación de Elastic Beanstalk e introduzca un nombre para la aplicación.
 - Elija la plataforma `Tomcat`.
 - Elija la versión `Tomcat 8 Java 8`.
 - Elija si desea utilizar SSH para configurar el acceso a las instancias de su entorno.
3. Desde el mismo directorio, ejecute `eb create` para crear un entorno de Elastic Beanstalk.

```
eb create
```

Cuando se le solicite:

- Escriba el nombre del nuevo entorno o acepte el nombre recomendado.
 - Escriba el prefijo CNAME de DNS del entorno o acepte el valor recomendado.
 - En este ejemplo, acepte el tipo `Classic load balancer`.
4. Después de ejecutar el comando `eb create`, la CLI de EB hace lo siguiente:
 1. Crea un archivo ZIP a partir del código fuente y lo carga en un bucket de S3 de la cuenta.

2. Crea una versión de aplicación de Elastic Beanstalk y aplicación.
 3. Crea un proyecto de CodeBuild.
 4. Ejecuta una compilación basada en el nuevo proyecto.
 5. Borra el proyecto una vez completada la compilación.
 6. Crea un entorno de Elastic Beanstalk.
 7. Implementa el resultado de la compilación en el entorno.
5. Después de que la CLI de EB implemente el resultado de la compilación para el entorno, podrá ver los resultados en un navegador web. Vaya a la URL del entorno de la instancia (por ejemplo, `http://my-environment-name.random-string.region-ID.elasticbeanstalk.com`). El navegador web debe mostrar el texto `Hello World!`.

Si lo desea, puede realizar cambios en el código fuente y ejecutar el comando `eb deploy` desde el mismo directorio. La CLI de EB realiza los mismos pasos que el comando `eb create`, pero implementa el resultado de la compilación en el entorno existente en lugar de crear un nuevo entorno.

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Ejemplo de filtros de webhook y solicitudes de extracción de Bitbucket

AWS CodeBuild admite webhooks, cuando el repositorio de origen es Bitbucket. Esto significa que para un proyecto de compilación de CodeBuild que tenga su código fuente almacenado en un repositorio de Bitbucket, pueden utilizarse webhooks para volver a compilar el código fuente cada vez que se inserte un cambio de código en el repositorio. Para obtener más información, consulte [??? \(p. 238\)](#).

En este ejemplo se muestra cómo crear una solicitud de extracción mediante un repositorio de Bitbucket. También se muestra cómo utilizar un webhook de Bitbucket para activar CodeBuild para crear una compilación de un proyecto de.

Note

Al utilizar webhooks, es posible que un usuario active una compilación inesperada. Para mitigar este riesgo, consulte [Prácticas recomendadas para el uso de webhooks \(p. 237\)](#).

Temas

- [Prerequisites \(p. 74\)](#)
- [Creación de un proyecto de compilación con Bitbucket como repositorio de origen y habilitación de webhooks \(p. 75\)](#)
- [Activación de una compilación con un webhook de Bitbucket \(p. 77\)](#)

Prerequisites

Para ejecutar este ejemplo, es preciso que conecte el proyecto de AWS CodeBuild con la cuenta de Bitbucket.

Note

CodeBuild ha actualizado sus permisos con Bitbucket. Si ha conectado anteriormente su proyecto a Bitbucket y ahora recibe un error de conexión de Bitbucket, debe volver a conectarse para conceder a CodeBuild permiso para administrar sus webhooks.

Creación de un proyecto de compilación con Bitbucket como repositorio de origen y habilitación de webhooks

Los siguientes pasos describen cómo crear un proyecto de AWS CodeBuild con Bitbucket como repositorio de origen y cómo habilitar webhooks.

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Si aparece una página con información de CodeBuild, seleccioneCrear un proyecto de compilación. En caso contrario, en el panel de navegación expandaBuild, eligeProyectos de compilacióny luegoCrear un proyecto de compilación.
3. Elija Create build project (Crear proyecto de compilación).
4. En Project configuration (Configuración del proyecto):

Project name (Nombre de proyecto)

Introduzca un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. También puede incluir una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.

5. En Source (Código fuente):

Proveedor de origen

SeleccionarBitbucket. Siga las instrucciones para conectarse (o volver a conectarse) a Bitbucket y, a continuación, elijaAutorizar.

Repositorio

SeleccionarRepositorio en mi cuenta de Bitbucket.

Si no te has conectado previamente a tu cuenta de Bitbucket, introduce tu nombre de usuario y contraseña de la aplicación de Bitbucket y seleccionaGuardar credenciales de Bitbucket.

Repositorio de Bitbucket

Introduzca la URL del repositorio de Bitbucket.

6. EnEventos de webhooks de origen principal, seleccione lo siguiente.

Note

LaEventos de webhooks de origen principalla sección solo es visible si eligesRepositorio en mi cuenta de Bitbucketen el paso anterior.

1. Cuando cree el proyecto, seleccione Rebuild every time a code change is pushed to this repository (Volver a compilar cada vez que se inserte un cambio de código en este repositorio).
2. En Event type (Tipo de evento), seleccione uno o varios eventos.
3. Para filtrar en función de cuándo un evento va a desencadenar una compilación, en Start a build under these conditions (Iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.

4. Para filtrar en función de cuándo no se va a desencadenar un evento, en Don't start a build under these conditions (No iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
5. Seleccionar Agregar grupo de filtros para agregar otro grupo de filtros, si es necesario.

Para obtener más información acerca de los tipos de eventos de webhook de Bitbucket y filtros, consulte [Eventos de webhooks de Bitbucket \(p. 238\)](#).

7. En Environment (Entorno):

Imagen de entorno

Elija una de las siguientes opciones:

Para utilizar una imagen Docker administrada por AWS CodeBuild:

Seleccionar Imagen administrada y, a continuación, realice selecciones desde Sistema operativo, Tiempo (s) de ejecución, Imagen (Imagen), y Versión de imagen. Realice una selección en Environment type (Tipo de entorno) si está disponible.

Para usar otra imagen de Docker:

Seleccionar Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige Amazon ECR, use Repositorio de Amazon ECR y Imagen de Amazon ECR para elegir la imagen de Docker en su AWS account.

Para utilizar una imagen privada de Docker:

Seleccionar Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Las credenciales las debe crear Secrets Manager. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Rol de servicio

Elija una de las siguientes opciones:

- Si no tiene un rol de servicio de CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
- Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola para crear o actualizar un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

8. En BuildSpec, realice una de las siguientes operaciones:
 - Seleccionar Utilizar un archivo buildspec para utilizar el archivo buildspec.yml en el directorio raíz del código fuente.
 - Seleccionar Insertar comandos de compilación para utilizar la consola para insertar comandos de compilación.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

9. En Artifacts (Artefactos):

Type

Elija una de las siguientes opciones:

- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts (Sin artefactos).
- Para almacenar la salida de la compilación en un bucket de S3, seleccione Amazon S3y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. De forma predeterminada, el nombre del artefacto es el nombre del proyecto. Si desea usar otro nombre, escríbalo en el cuadro de nombre de artefactos. Si desea producir un archivo ZIP, incluya la extensión zip.
 - En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.
 - Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, appspec.yml, target/my-app.jar). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec \(p. 137\)](#).

Configuración adicional

Expanda Additional configuration (Configuración adicional) y establezca las opciones según sea necesario.

10. Elija Create build project (Crear proyecto de compilación). En la página Review (Revisar), elija Start build (Comenzar compilación) para ejecutar la compilación.

Activación de una compilación con un webhook de Bitbucket

En los proyectos que utilizan webhooks de Bitbucket, AWS CodeBuild crea una compilación cuando el repositorio de Bitbucket detecta un cambio en el código fuente.

1. Abra el icono AWS CodeBuild Consola de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects (Proyectos de compilación) y, a continuación, elija un proyecto asociado a un repositorio de Bitbucket con webhooks. Para obtener información acerca de cómo crear un proyecto de webhook de Bitbucket, consulte [the section called "Creación de un proyecto de compilación con Bitbucket como repositorio de origen y habilitación de webhooks" \(p. 75\)](#).
3. Realice algunos cambios en el código en el repositorio de Bitbucket del proyecto.
4. Cree una solicitud de extracción en su repositorio de Bitbucket. Para obtener más información, consulte el artículo sobre [cómo crear una solicitud de extracción](#).
5. En la página de webhooks de Bitbucket, elija View request (Ver solicitud) para ver una lista de eventos recientes.
6. Seleccione View details (Ver detalles). para ver detalles sobre la respuesta devuelta por CodeBuild. Debe ser similar al siguiente:

```
"response": "Webhook received and build started: https://us-east-1.console.aws.amazon.com/codebuild/home..."
"statusCode": 200
```

7. Vaya a la página de solicitudes de extracción de Bitbucket para ver el estado de la compilación.

Ejemplo de insignias de compilación con Code

AWS CodeBuild ahora admite el uso de insignias de compilación, que proporcionan una imagen incrustable generada dinámicamente (insignia) que muestra el estado de la última compilación de un proyecto. Esta imagen está accesible a través de una URL disponible públicamente generada para su proyecto de CodeBuild. Permite que cualquiera pueda ver el estado de un proyecto CodeBuild. Las insignias de compilación no contienen información de seguridad, por lo que no requieren autenticación.

Creación de un proyecto de compilación con insignias de compilación habilitadas (consola)

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Si aparece una página con información de CodeBuild, seleccioneCrear un proyecto de compilación. En caso contrario, en el panel de navegación expandaBuild, eligeProyectos de compilacióny luegoCrear un proyecto de compilación.
3. En Project name (Nombre de proyecto), escriba un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. También puede incluir una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.
4. En Source (Código fuente), para Source provider (Proveedor de código fuente), elija el tipo de proveedor de código fuente y, a continuación, realice alguna de las siguientes operaciones:

Note

CodeBuild no admite insignias de compilación con el proveedor de origen Amazon S3. Ya queAWS CodePipelineutiliza Amazon S3 para transferir artefactos, las insignias de compilación no son compatibles con proyectos de compilación que forman parte de una canalización creada en CodePipeline.

- Si eligióCodeCommit, luego paraRepositorio, elija el nombre del repositorio. Seleccione Enable build badge (Habilitar insignia de compilación) para convertir el estado de compilación del proyecto en visible e incrustable.
- Si eligió GitHub, siga las instrucciones para conectarse (o volver a conectarse) a GitHub. En la página Authorize application (Autorizar aplicación) de GitHub, para Organization access (Acceso de organización), elija Request access (Solicitar acceso) junto a cada repositorio al que desee que AWS CodeBuild pueda obtener acceso. Después de elegir Authorize application (Autorizar aplicación), de vuelta en la consola de AWS CodeBuild, para Repository (Repositorio), elija el nombre del repositorio que contiene el código fuente. Seleccione Enable build badge (Habilitar insignia de compilación) para convertir el estado de compilación del proyecto en visible e incrustable.
- Si eligió Bitbucket, siga las instrucciones para conectarse (o volver a conectarse) a Bitbucket. En la página Confirm access to your account de Bitbucket, para Organization access, elija Grant access. Después de elegir Grant access (Conceder acceso), de vuelta en la consola de AWS CodeBuild, para Repository (Repositorio), elija el nombre del repositorio que contiene el código fuente. Seleccione Enable build badge (Habilitar insignia de compilación) para convertir el estado de compilación del proyecto en visible e incrustable.

Important

La actualización del código fuente del proyecto podría afectar a la precisión de las insignias de compilación del proyecto.

5. En Environment (Entorno):

En Environment image (Imagen del entorno), realice alguna de las siguientes operaciones:

- Para usar una imagen de Docker administrada por AWS CodeBuild, elija Managed image (Imagen administrada) y después elija una opción en Operating system (Sistema operativo), Runtime (Tiempo de ejecución), Image (Imagen) e Image version (Versión de imagen). Realice una selección en Environment type (Tipo de entorno) si está disponible.
 - Para usar otra imagen de Docker, elija Custom image (Imagen personalizada). Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige Amazon ECR, use Repositorio de Amazon ECR. Imagen de Amazon ECR para elegir la imagen de Docker en su AWS account.
 - Para usar una imagen privada de Docker, elija Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Las credenciales las debe crear Secrets Manager. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.
6. En Service role (Rol de servicio), realice una de las operaciones siguientes:
- Si no tiene un rol de servicio de CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
 - Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola para crear o actualizar un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

7. En BuildSpec, realice una de las siguientes operaciones:
- Seleccionar Utilizar un archivo buildspec para utilizar el archivo buildspec.yml en el directorio raíz del código fuente.
 - Seleccionar Insertar comandos de compilación para utilizar la consola para insertar comandos de compilación.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

8. En Artifacts (Artefactos), en Type (Tipo), realice una de las siguientes operaciones:
- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts (Sin artefactos).
 - Para almacenar la salida de la compilación en un bucket de S3, seleccione Amazon S3 y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. De forma predeterminada, el nombre del artefacto es el nombre del proyecto. Si desea usar otro nombre, escríbalo en el cuadro de nombre de artefactos. Si desea producir un archivo ZIP, incluya la extensión zip.
 - En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.
 - Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación.

Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, `appspec.yml`, `target/my-app.jar`). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec](#) (p. 137).

9. Expanda Additional configuration (Configuración adicional) y elija las opciones que correspondan.
10. Elija Create build project (Crear proyecto de compilación). En la página Review (Revisar), elija Start build (Comenzar compilación) para ejecutar la compilación.

Creación de un proyecto de compilación con insignias de compilación habilitadas (CLI)

Para obtener información acerca de cómo crear un proyecto de compilación, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209). Para incluir insignias de compilación con su proyecto de AWS CodeBuild, debe especificar `badgeEnabled` con un valor de `true`.

Acceso a las insignias de compilación de AWS CodeBuild

Puede utilizar la consola de AWS CodeBuild o la AWS CLI para obtener acceso a las insignias de compilación.

- En la consola CodeBuild, en la lista de proyectos de compilación, en la CLI de Nombre, elija el enlace correspondiente al proyecto de compilación. En la página Build project (Proyecto de compilación): **nombre_proyecto**, en Configuration (Configuración), elija Copy badge URL (Copiar URL de insignia). Para obtener más información, consulte [Ver los detalles de un proyecto de compilación \(consola\)](#) (p. 227).
- En la AWS CLI, ejecute el comando `batch-get-projects`. La URL de insignia de compilación se incluye en la sección de salida de detalles del entorno del proyecto. Para obtener más información, consulte [Ver los detalles de un proyecto de compilación \(AWS CLI\)](#) (p. 227).

La URL de solicitud de insignia de compilación se genera con una ramificación predeterminada común, pero puede especificar cualquier ramificación del repositorio de origen que haya usado para ejecutar una compilación. Por ejemplo:

```
https://codebuild.us-east-1.amazon.com/badges?uuid=...&branch=<branch>
```

También puede especificar una etiqueta del repositorio de origen sustituyendo el `branch` parámetro con el `tag` en la URL de la insignias de. Por ejemplo:

```
https://codebuild.us-east-1.amazon.com/badges?uuid=...&tag=<tag>
```

Publicación de insignias de compilación de Code

Puede incluir la URL de solicitud de insignia de compilación en un archivo de marcado en el repositorio que desee (por ejemplo, GitHub o CodeCommit) para mostrar el estado de la última compilación.

Código de marcado de ejemplo:

```
![Build Status](https://codebuild.us-east-1.amazon.com/badges?uuid=...&branch=main)
```

Estados de la insignias de Code

- **PASSING** La compilación más reciente en la ramificación especificada se ha realizado correctamente.

- **FAILING** Se ha agotado el tiempo de espera, se ha producido un error o se ha detenido la compilación más reciente en la ramificación especificada.
- **IN_PROGRESS** La compilación más reciente en la ramificación especificada está en curso.
- **UNKNOWN** El proyecto aún no ha ejecutado una compilación en la ramificación especificada o en ninguna otra. Además, la característica de insignia de compilación podría estar deshabilitada.

Ejemplo de notificaciones de CodeBuild de

Amazon CloudWatch Events proporciona compatibilidad integrada con AWS CodeBuild. CloudWatch Events es un flujo de eventos del sistema que describen cambios en su AWS de AWS. Con CloudWatch Events, puede escribir reglas declarativas para asociar eventos de interés a las acciones automatizadas que se deben realizar. Este ejemplo utiliza Amazon CloudWatch Events y Amazon Simple Notification Service (Amazon SNS) para enviar notificaciones a los suscriptores siempre que las compilaciones se realizan correctamente, experimentan un error, cambian de fase o cualquier combinación de estos eventos.

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon CloudWatch y Amazon SNS. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon CloudWatch](#), y [Precios de Amazon SNS](#).

Ejecución del ejemplo

Para ejecutar este ejemplo

1. Si ya tiene un tema configurado y suscrito a Amazon SNS que desee utilizar para este ejemplo, vaya directamente al paso 4. De lo contrario, si está utilizando un usuario de IAM en lugar de una AWS cuenta raíz o un usuario de IAM administrador para trabajar con Amazon SNS, agregue la siguiente declaración (entre **### COMIENZA A AGREGAR DECLARACIÓN AQUÍ ###** y **### TERMINA AGREGANDO DECLARACIÓN AQUÍ ###**) al usuario (o grupo de IAM al que está asociado el usuario). No se recomienda usar una cuenta raíz de AWS. Esta instrucción permite visualizar, crear, suscribir y probar el envío de notificaciones a temas de Amazon SNS. Los puntos suspensivos (. . .) se usan por motivos de brevedad y para ayudarle a encontrar el lugar donde debe añadir la instrucción. No elimine ninguna instrucción ni incluya estos puntos suspensivos en la política existente.

```
{
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Action": [
        "sns:CreateTopic",
        "sns:GetTopicAttributes",
        "sns:List*",
        "sns:Publish",
        "sns:SetTopicAttributes",
        "sns:Subscribe"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    ### END ADDING STATEMENT HERE ###
    ...
  ],
  "Version": "2012-10-17"
}
```


Note

La entidad de IAM que modifica esta política debe tener permiso en IAM para modificar políticas.

Para obtener más información, consulte [Edición de políticas administradas por el](#) en la sección «Para editar o eliminar una política insertada de un grupo, un usuario o un rol» en [Trabajo con políticas en línea \(consola\)](#) en la IAM User Guide.

2. Cree o identifique un tema en Amazon SNS. AWS CodeBuild utiliza CloudWatch Events para enviar notificaciones de compilación a este tema a través de Amazon SNS.

Para crear un tema:

1. Abra la consola de Amazon SNS en <https://console.aws.amazon.com/sns>.
2. Elija Create new topic.
3. En el cuadro de diálogo Create new topic (Crear un nuevo tema), en Topic name (Nombre del tema), escriba un nombre para el tema (por ejemplo, **CodeBuildDemoTopic**). Si elige otro nombre, sustitúyalo en todo este ejemplo.
4. Elija Create new topic.
5. En la página Detalles del tema: Tema de demostración de compilación de código, copia el ARN de tema de Valor . Necesitará este valor para el siguiente paso.

Topic details

Publish to topic

Topic ARN

Topic owner

Region

Display name

Para obtener más información, consulte [Crear un tema](#) en la Guía para desarrolladores de Amazon SNS.

3. Suscriba a uno o varios destinatarios al tema para que reciban notificaciones por correo electrónico.

Para suscribir a un destinatario a un tema:

1. Con la consola de Amazon SNS abierta del paso anterior, en el panel de navegación, elija **Suscripciones** y luego **Crear suscripción**.
2. En **Create subscription** (Crear suscripción), en **Topic ARN** (ARN del tema), pegue el ARN del tema que copió en el paso anterior.
3. En **Protocol** (Protocolo), elija **Email** (Correo electrónico).
4. En **Endpoint** (Punto de enlace), escriba la dirección de correo electrónico completa del destinatario.

Create subscrip

Topic A

Proto

Endpo

5. Elija Create Subscription.

6. Amazon SNS envía un correo electrónico de confirmación de suscripción al destinatario. Para empezar a recibir notificaciones por correo electrónico, el destinatario debe elegir el enlace Confirm subscription en el mensaje de confirmación de la suscripción. Cuando el destinatario hace clic en el enlace, si se ha suscrito correctamente, Amazon SNS muestra un mensaje de confirmación en el navegador web.

Para obtener más información, consulte [Suscripción a un tema](#) en la Guía para desarrolladores de Amazon SNS.

4. Si utiliza un usuario de IAM en lugar de una AWS cuenta raíz o un usuario de IAM administrador para trabajar con CloudWatch Events, agregue la siguiente instrucción (entre **### COMIENZA A AGREGAR DECLARACIÓN AQUÍ ###** y **### TERMINA AGREGANDO DECLARACIÓN AQUÍ ###**) al usuario (o grupo de IAM al que está asociado el usuario). No se recomienda usar una cuenta raíz de AWS. Esta declaración se utiliza para permitir que el usuario pueda trabajar con CloudWatch Events. Los puntos suspensivos (. . .) se usan por motivos de brevedad y para ayudarle a encontrar el lugar donde debe añadir la instrucción. No elimine ninguna instrucción ni incluya estos puntos suspensivos en la política existente.

```
{
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Action": [
        "events:*",
        "iam:PassRole"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    ### END ADDING STATEMENT HERE ###
    ...
  ],
  "Version": "2012-10-17"
}
```

Note

La entidad de IAM que modifica esta política debe tener permiso en IAM para modificar políticas.

Para obtener más información, consulte [Edición de políticas administradas por el](#) en la sección «Para editar o eliminar una política insertada de un grupo, un usuario o un rol» en [Trabajo con políticas en línea \(consola\)](#) en la IAM User Guide.

5. Cree una regla en CloudWatch Events. Para ello, abra la consola de CloudWatch, en <https://console.aws.amazon.com/cloudwatch>.
6. En el panel de navegación, bajo Events, elija Rules y después elija Create rule.
7. En la página Paso 1: Página Crear regla, Patrón de eventos y Crear un patrón de eventos para que coincida con eventos por servicio debería estar seleccionada ya.
8. Para Nombre del servicio, elija CodeBuild. En Event Type (Tipo de evento), la opción All Events (Todos los eventos) ya debería estar seleccionada.
9. El código siguiente debería aparecer en Event Pattern Preview (Vista previa del patrón de eventos):

```
{
  "source": [
    "aws.codebuild"
  ]
}
```

```
}
```

10. Elija Edit (Editar) y sustituya el código de Event Pattern Preview (Vista previa de patrón del eventos) por uno de los dos patrones de reglas siguientes.

Este primer patrón de reglas desencadena un evento siempre que se inicia o se completa una compilación en los proyectos de compilación especificados en AWS CodeBuild.

```
{
  "source": [
    "aws.codebuild"
  ],
  "detail-type": [
    "CodeBuild Build State Change"
  ],
  "detail": {
    "build-status": [
      "IN_PROGRESS",
      "SUCCEEDED",
      "FAILED",
      "STOPPED"
    ],
    "project-name": [
      "my-demo-project-1",
      "my-demo-project-2"
    ]
  }
}
```

En la regla anterior, realice los siguientes cambios de código según sea necesario.

- Para desencadenar un evento cuando se inicia o se completa una compilación, deje todos los valores tal y como se muestran en la matriz `build-status` o quite completamente la matriz `build-status`.
- Para activar un evento solo cuando se complete una compilación, elimine `IN_PROGRESS` de la matriz `build-status`.
- Para activar un evento solo cuando se inicie una compilación, elimine todos los valores a excepción de `IN_PROGRESS` de la matriz `build-status`.
- Para activar eventos para todos los proyectos de compilación, elimine toda la matriz `project-name`.
- Para activar eventos solo para proyectos de compilación individuales, especifique el nombre de cada proyecto de compilación en la matriz `project-name`.

Este segundo patrón de reglas desencadena un evento siempre que una compilación pasa de una fase a otra en los proyectos de compilación especificados en AWS CodeBuild.

```
{
  "source": [
    "aws.codebuild"
  ],
  "detail-type": [
    "CodeBuild Build Phase Change"
  ],
  "detail": {
    "completed-phase": [
      "SUBMITTED",
      "PROVISIONING",
      "DOWNLOAD_SOURCE",
      "INSTALL",

```

```
    "PRE_BUILD",
    "BUILD",
    "POST_BUILD",
    "UPLOAD_ARTIFACTS",
    "FINALIZING"
  ],
  "completed-phase-status": [
    "TIMED_OUT",
    "STOPPED",
    "FAILED",
    "SUCCEEDED",
    "FAULT",
    "CLIENT_ERROR"
  ],
  "project-name": [
    "my-demo-project-1",
    "my-demo-project-2"
  ]
}
}
```

En la regla anterior, realice los siguientes cambios de código según sea necesario.

- Para desencadenar un evento con cada cambio de fase de compilación (lo que podría enviar hasta nueve notificaciones por cada compilación), deje todos los valores tal y como se muestran en la matriz `completed-phase` o elimine completamente la matriz `completed-phase`.
- Para activar eventos solo para cambios de fase de compilación individuales, elimine el nombre de cada fase de compilación en la matriz `completed-phase` para el que no desee que se active un evento.
- Para activar un evento para cada cambio de estado de fase de compilación, deje todos los valores tal y como se muestran en la matriz `completed-phase-status` o elimine completamente la matriz `completed-phase-status`.
- Para activar eventos solo para cambios de estado de fase de compilación individuales, elimine el nombre de cada estado de fase de compilación en la matriz `completed-phase-status` para el que no desee que se active un evento.
- Para activar eventos para todos los proyectos de compilación, elimine la matriz `project-name`.
- Para activar eventos para proyectos de compilación individuales, especifique el nombre de cada proyecto de compilación en la matriz `project-name`.

Para obtener más información sobre los patrones de eventos, consulte [Patrones de eventos](#) en la Guía del usuario de Amazon EventBridge.

Para obtener más información sobre cómo filtrar con patrones de eventos, consulte [Filtrar en función del contenido con patrones de eventos](#) en la Guía del usuario de Amazon EventBridge.

Note

Si desea desencadenar eventos con los cambios de estado de compilación y los cambios de fase de compilación, debe crear dos reglas distintas: una para los cambios de estado y otra para los cambios de fase. Si intenta combinar las dos reglas en una sola, la regla combinada podría producir resultados inesperados o dejar de funcionar.

Cuando haya terminado de reemplazar el código, seleccione Save.

11. En Targets, seleccione Add target.
12. En la lista de destinos, elija SNS topic.
13. En Topic, elija el tema que identificó o creó anteriormente.
14. ~~Expanda Configure input, a continuación, elija Input Transformer.~~

Version de API 2016-10-06

15. En el cuadro Input Path (Ruta de entrada), escriba una de las siguientes rutas de entrada.

Si en una regla el valor de detail-type es CodeBuild Build State Change, escriba lo siguiente.

```
{"build-id": "${detail.build-id}", "project-name": "${detail.project-name}", "build-status": "${detail.build-status"}
```

Si en una regla el valor de detail-type es CodeBuild Build Phase Change, escriba lo siguiente.

```
{"build-id": "${detail.build-id}", "project-name": "${detail.project-name}", "completed-phase": "${detail.completed-phase}", "completed-phase-status": "${detail.completed-phase-status"}
```

Para obtener otros tipos de información, consulte la [Referencia del formato de entrada de las notificaciones de compilación \(p. 90\)](#).

16. En el cuadro Input Template (Plantilla de entrada), escriba una de las siguientes plantillas de entrada.

Si en una regla el valor de detail-type es CodeBuild Build State Change, escriba lo siguiente.

```
"Build '<build-id>' for build project '<project-name>' has reached the build status of '<build-status>'. "
```

Si en una regla el valor de detail-type es CodeBuild Build Phase Change, escriba lo siguiente.

```
"Build '<build-id>' for build project '<project-name>' has completed the build phase of '<completed-phase>' with a status of '<completed-phase-status>'. "
```

17. Seleccione Configure details.
18. En la página Paso 2: Configurar detalles de la regla, escriba un nombre y una descripción opcional. En State (Estado), deje seleccionado el valor Enabled (Habilitado).
19. Elija Create rule.
20. Para crear proyectos de compilación, ejecutar las compilaciones y consultar información sobre la compilación, siga los pasos que se indican en [Ejecutar CodeBuild directamente \(p. 406\)](#).
21. Confirme que CodeBuild está enviando correctamente notificaciones de compilación. Por ejemplo, compruebe si hay correos electrónicos de notificación de compilación en su bandeja de entrada.

Para cambiar el comportamiento de una regla, elija la regla que desee cambiar en la consola de CloudWatch y elija Actions y luego Editar. Realice los cambios que correspondan en la regla y elija Configure details (Configurar detalles) y Update rule (Actualizar regla).

Para dejar de utilizar una regla para enviar notificaciones de compilación, en la consola de CloudWatch, elija la regla que desea dejar de usar y elija Actions y luego Deshabilitar.

Para eliminar completamente una regla, elija la regla que desee eliminar en la consola de CloudWatch, elija la regla que desee eliminar en la consola de CloudWatch Actions y luego Eliminar.

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).

- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Referencia del formato de entrada de las notificaciones de compilación

CloudWatch envía las notificaciones en formato JSON.

Las notificaciones de cambio de estado de compilación usan el siguiente formato:

```
{
  "version": "0",
  "id": "c030038d-8c4d-6141-9545-00ff7b7153EX",
  "detail-type": "CodeBuild Build State Change",
  "source": "aws.codebuild",
  "account": "123456789012",
  "time": "2017-09-01T16:14:28Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:codebuild:us-west-2:123456789012:build/my-sample-project:8745a7a9-c340-456a-9166-edf953571bEX"
  ],
  "detail": {
    "build-status": "SUCCEEDED",
    "project-name": "my-sample-project",
    "build-id": "arn:aws:codebuild:us-west-2:123456789012:build/my-sample-project:8745a7a9-c340-456a-9166-edf953571bEX",
    "additional-information": {
      "artifact": {
        "md5sum": "da9c44c8a9a3cd4b443126e823168fEX",
        "sha256sum": "6ccc2ae1df9d155ba83c597051611c42d60e09c6329dcb14a312cecc0a8e39EX",
        "location": "arn:aws:s3:::codebuild-123456789012-output-bucket/my-output-artifact.zip"
      },
      "environment": {
        "image": "aws/codebuild/standard:4.0",
        "privileged-mode": false,
        "compute-type": "BUILD_GENERAL1_SMALL",
        "type": "LINUX_CONTAINER",
        "environment-variables": []
      },
      "timeout-in-minutes": 60,
      "build-complete": true,
      "initiator": "MyCodeBuildDemoUser",
      "build-start-time": "Sep 1, 2017 4:12:29 PM",
      "source": {
        "location": "codebuild-123456789012-input-bucket/my-input-artifact.zip",
        "type": "S3"
      },
      "logs": {
        "group-name": "/aws/codebuild/my-sample-project",
        "stream-name": "8745a7a9-c340-456a-9166-edf953571bEX",
        "deep-link": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=/aws/codebuild/my-sample-project;stream=8745a7a9-c340-456a-9166-edf953571bEX"
      },
      "phases": [
        {
          "phase-context": [],
          "start-time": "Sep 1, 2017 4:12:29 PM",
          "end-time": "Sep 1, 2017 4:12:29 PM",

```

```
"duration-in-seconds": 0,
"phase-type": "SUBMITTED",
"phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:12:29 PM",
  "end-time": "Sep 1, 2017 4:13:05 PM",
  "duration-in-seconds": 36,
  "phase-type": "PROVISIONING",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:05 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 4,
  "phase-type": "DOWNLOAD_SOURCE",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 0,
  "phase-type": "INSTALL",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 0,
  "phase-type": "PRE_BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 70,
  "phase-type": "BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 0,
  "phase-type": "POST_BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 0,
  "phase-type": "UPLOAD_ARTIFACTS",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
  "end-time": "Sep 1, 2017 4:14:26 PM",
  "duration-in-seconds": 4,
  "phase-type": "FINALIZING",
```

```
        "phase-status": "SUCCEEDED"
      },
      {
        "start-time": "Sep 1, 2017 4:14:26 PM",
        "phase-type": "COMPLETED"
      }
    ]
  },
  "current-phase": "COMPLETED",
  "current-phase-context": "[]",
  "version": "1"
}
```

Las notificaciones de cambio de fase de compilación usan el siguiente formato:

```
{
  "version": "0",
  "id": "43ddc2bd-af76-9ca5-2dc7-b695e15adeEX",
  "detail-type": "CodeBuild Build Phase Change",
  "source": "aws.codebuild",
  "account": "123456789012",
  "time": "2017-09-01T16:14:21Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:codebuild:us-west-2:123456789012:build/my-sample-project:8745a7a9-c340-456a-9166-edf953571bEX"
  ],
  "detail": {
    "completed-phase": "COMPLETED",
    "project-name": "my-sample-project",
    "build-id": "arn:aws:codebuild:us-west-2:123456789012:build/my-sample-project:8745a7a9-c340-456a-9166-edf953571bEX",
    "completed-phase-context": "[]",
    "additional-information": {
      "artifact": {
        "md5sum": "da9c44c8a9a3cd4b443126e823168fEX",
        "sha256sum": "6ccc2ae1df9d155ba83c597051611c42d60e09c6329dcb14a312cecc0a8e39EX",
        "location": "arn:aws:s3:::codebuild-123456789012-output-bucket/my-output-artifact.zip"
      },
      "environment": {
        "image": "aws/codebuild/standard:4.0",
        "privileged-mode": false,
        "compute-type": "BUILD_GENERAL1_SMALL",
        "type": "LINUX_CONTAINER",
        "environment-variables": []
      },
      "timeout-in-minutes": 60,
      "build-complete": true,
      "initiator": "MyCodeBuildDemoUser",
      "build-start-time": "Sep 1, 2017 4:12:29 PM",
      "source": {
        "location": "codebuild-123456789012-input-bucket/my-input-artifact.zip",
        "type": "S3"
      },
      "logs": {
        "group-name": "/aws/codebuild/my-sample-project",
        "stream-name": "8745a7a9-c340-456a-9166-edf953571bEX",
        "deep-link": "https://console.aws.amazon.com/cloudwatch/home?region=us-west-2#logEvent:group=/aws/codebuild/my-sample-project;stream=8745a7a9-c340-456a-9166-edf953571bEX"
      },
      "phases": [
        {
```

```
"phase-context": [],
"start-time": "Sep 1, 2017 4:12:29 PM",
"end-time": "Sep 1, 2017 4:12:29 PM",
"duration-in-seconds": 0,
"phase-type": "SUBMITTED",
"phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:12:29 PM",
  "end-time": "Sep 1, 2017 4:13:05 PM",
  "duration-in-seconds": 36,
  "phase-type": "PROVISIONING",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:05 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 4,
  "phase-type": "DOWNLOAD_SOURCE",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 0,
  "phase-type": "INSTALL",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:13:10 PM",
  "duration-in-seconds": 0,
  "phase-type": "PRE_BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:13:10 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 70,
  "phase-type": "BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 0,
  "phase-type": "POST_BUILD",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
  "end-time": "Sep 1, 2017 4:14:21 PM",
  "duration-in-seconds": 0,
  "phase-type": "UPLOAD_ARTIFACTS",
  "phase-status": "SUCCEEDED"
},
{
  "phase-context": [],
  "start-time": "Sep 1, 2017 4:14:21 PM",
```

```
        "end-time": "Sep 1, 2017 4:14:26 PM",
        "duration-in-seconds": 4,
        "phase-type": "FINALIZING",
        "phase-status": "SUCCEEDED"
      },
      {
        "start-time": "Sep 1, 2017 4:14:26 PM",
        "phase-type": "COMPLETED"
      }
    ]
  },
  "completed-phase-status": "SUCCEEDED",
  "completed-phase-duration-seconds": 4,
  "version": "1",
  "completed-phase-start": "Sep 1, 2017 4:14:21 PM",
  "completed-phase-end": "Sep 1, 2017 4:14:26 PM"
}
```

Crear un informe de pruebas en CodeBuild conAWS CLIEjemplo de

Las pruebas que especifique en su archivo buildspec se ejecutan durante la compilación. En este ejemplo se muestra cómo se utiliza laAWS CLIp para incorporar pruebas en las compilaciones de CodeBuild. Puede usar JUnit para crear pruebas unitarias o puede usar otra herramienta para crear pruebas de configuración. Después, puede evaluar los resultados de las pruebas para solucionar problemas u optimizar la aplicación.

Puede usar la API de CodeBuild o laAWS CodeBuildconsola para obtener acceso a los resultados de la prueba. En este ejemplo se muestra cómo configurar el informe para que los resultados de las pruebas se exporten a un bucket de S3.

Temas

- [Prerequisites \(p. 94\)](#)
- [Crear un grupo de informes \(p. 95\)](#)
- [Configurar un proyecto con un grupo de informes \(p. 96\)](#)
- [Ejecutar y ver los resultados de un informe \(p. 97\)](#)

Prerequisites

- Cree los casos de prueba. Esta muestra se escribe bajo el supuesto de que tiene casos de prueba para incluirlos en su informe de pruebas de muestra. Especifique la ubicación de los archivos de prueba en el archivo buildspec.

Se admiten los siguientes formatos de archivo de informe de prueba:

- Pepino JSON (.json)
- XML JUnit (.xml)
- XML de unidad (.xml)
- XML NUnit3 (.xml)
- XML TestNG (.xml)
- Visual Studio TRX (.trx)

Cree sus casos de prueba con cualquier marco de pruebas que pueda crear archivos de informes en uno de esos formatos (por ejemplo, el complemento Surefire JUnit, TestNG o Cucumber).

- Cree un bucket de S3 y anote su nombre. Para obtener más información, consulte [¿Cómo se puede crear un bucket de S3?](#) en la Guía del usuario de Amazon S3.
- Cree un rol de IAM y anote su ARN. Necesita el ARN cuando cree su proyecto de compilación.
- Si el rol no tiene los permisos siguientes, añádalos.

```
{
  "Effect": "Allow",
  "Resource": [
    "*"
  ],
  "Action": [
    "codebuild:CreateReportGroup",
    "codebuild:CreateReport",
    "codebuild:UpdateReport",
    "codebuild:BatchPutTestCases"
  ]
}
```

Para obtener más información, consulte [Permisos para las operaciones de informes de pruebas \(p. 325\)](#).

Crear un grupo de informes

1. Cree un archivo denominado `CreateReportGroupInput.json`.
2. Cree una carpeta en el bucket de S3 donde se exportarán los resultados de las pruebas.
3. Copie lo siguiente en `CreateReportGroupInput.json`. Para `<bucket-name>`, use el nombre del bucket de S3. Para `<path-to-folder>`, escriba la ruta de acceso a la carpeta en el bucket de S3.

```
{
  "name": "<report-name>",
  "type": "TEST",
  "exportConfig": {
    "exportConfigType": "S3",
    "s3Destination": {
      "bucket": "<bucket-name>",
      "path": "<path-to-folder>",
      "packaging": "NONE"
    }
  }
}
```

4. Ejecute el siguiente comando en el directorio que contiene `CreateReportGroupInput.json`:

```
aws codebuild create-report-group --cli-input-json file://CreateReportGroupInput.json
```

El resultado es similar al siguiente. Anote el ARN de `reportGroup`. Se utiliza al crear un proyecto que utiliza este grupo de informes.

```
{
  "reportGroup": {
    "arn": "arn:aws:codebuild:us-west-2:123456789012:report-group/<report-name>",
    "name": "<report-name>",
    "type": "TEST",
    "exportConfig": {
      "exportConfigType": "S3",
      "s3Destination": {
        "bucket": "<s3-bucket-name>",
```

```
    "path": "<folder-path>",
    "packaging": "NONE",
    "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3"
  },
  "created": 1570837165.885,
  "lastModified": 1570837165.885
}
```

Configurar un proyecto con un grupo de informes

Para ejecutar un informe, primero debe crear un proyecto de compilación de CodeBuild configurado con el grupo de informes. Los casos de prueba especificados para su grupo de informes se ejecutan cuando se ejecuta una compilación.

1. Cree un archivo buildspec denominado `buildspec.yml`.
2. Utilice el siguiente código YAML como plantilla para su archivo `buildspec.yml`. Asegúrese de incluir los comandos que ejecutan las pruebas. En la sección `reports`, especifique los archivos que contienen los resultados de los casos de prueba. Estos archivos almacenan los resultados de las pruebas a los que se puede acceder con CodeBuild. Caducan 30 días después de su creación. Estos archivos son distintos de los archivos de resultados de casos de prueba sin procesar que se exportan a un bucket de S3.

```
version: 0.2
phases:
  install:
    runtime-versions:
      java: openjdk8
  build:
    commands:
      - echo Running tests
      - <enter commands to run your tests>

reports:
  <report-name-or-arn>: #test file information
  files:
    - '<test-result-files>'
  base-directory: '<optional-base-directory>'
  discard-paths: false #do not remove file paths from test result files
```

Note

En lugar del ARN de un grupo de informes existente, también puede especificar un nombre para un grupo de informes que no se haya creado. Si especifica un nombre en lugar de un ARN, CodeBuild crea un grupo de informes cuando ejecuta una compilación. Su nombre contiene el nombre de proyecto y el nombre especificado en el archivo `buildspec` en este formato: `project-name-report-group-name`. Para obtener más información, consulte [Crear un informe de pruebas \(p. 306\)](#) y [Nomenclatura de grupos de informes \(p. 314\)](#).

3. Cree un archivo denominado `project.json`. Este archivo contiene la entrada para el comando `create-project`.
4. Copie el siguiente JSON en `project.json`. Para `source`, introduzca el tipo y la ubicación del repositorio que contiene los archivos de origen. Para `serviceRole`, especifique el ARN del rol que está utilizando.

```
{
```

```
"name": "test-report-project",
"description": "sample-test-report-project",
"source": {
  "type": "CODECOMMIT|CODEPIPELINE|GITHUB|S3|BITBUCKET|GITHUB_ENTERPRISE|NO_SOURCE",
  "location": "<your-source-url>"
},
"artifacts": {
  "type": "NO_ARTIFACTS"
},
"cache": {
  "type": "NO_CACHE"
},
"environment": {
  "type": "LINUX_CONTAINER",
  "image": "aws/codebuild/standard:4.0",
  "computeType": "small"
},
"serviceRole": "arn:aws:iam::<your-aws-account-id>:role/service-role/<your-role-name>"
}
```

5. Ejecute el siguiente comando en el directorio que contiene `project.json`: De este modo se crea un proyecto llamado `test-project`.

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

Ejecutar y ver los resultados de un informe

En esta sección, se ejecuta una compilación del proyecto creado anteriormente. Durante el proceso de compilación, CodeBuild crea un informe con los resultados de los casos de prueba. El informe se incluye en el grupo de informes especificado.

1. Para iniciar una compilación, ejecute el siguiente comando. `test-report-project` es el nombre del proyecto de compilación creado anteriormente. Anote el ID de compilación que aparece en la salida.

```
aws codebuild start-build --project-name test-report-project
```

2. Ejecute el siguiente comando para obtener información sobre su compilación, incluido el ARN del informe. Para `<build-id>`, especifique su ID de compilación. Anote el ARN del informe en la propiedad `reportArns` de la salida.

```
aws codebuild batch-get-builds --ids <build-id>
```

3. Ejecute el comando siguiente para obtener detalles sobre el informe. Para `<report-arn>`, especifique el ARN del informe.

```
aws codebuild batch-get-reports --report-arns <report-arn>
```

El resultado es similar al siguiente. Esta salida de muestra presenta cuántas pruebas fueron correctas, tuvieron errores, se omitieron, produjeron un error o devolvieron un estado desconocido.

```
{
  "reports": [
    {
      "status": "FAILED",
      "reportGroupArn": "<report-group-arn>",
      "name": "<report-group-name>",
      "created": 1573324770.154,
```



```
"exportConfig": {
  "exportConfigType": "S3",
  "s3Destination": {
    "bucket": "<your-S3-bucket>",
    "path": "<path-to-your-report-results>",
    "packaging": "NONE",
    "encryptionKey": "<encryption-key>"
  }
},
"expired": 1575916770.0,
"truncated": false,
"executionId": "arn:aws:codebuild:us-west-2:123456789012:build/<name-of-build-
project>:2c254862-ddf6-4831-a53f-6839a73829c1",
"type": "TEST",
"arn": "<report-arn>",
"testSummary": {
  "durationInNanoSeconds": 6657770,
  "total": 11,
  "statusCounts": {
    "FAILED": 3,
    "SKIPPED": 7,
    "ERROR": 0,
    "SUCCEEDED": 1,
    "UNKNOWN": 0
  }
}
},
"reportsNotFound": []
}
```

4. Ejecute el comando siguiente para enumerar la información sobre los casos de prueba para su informe. Para **<report-arn>**, especifique el ARN del informe. Para el parámetro **--filter** opcional, puede especificar un resultado de estado (SUCCEEDED, FAILED, SKIPPED, ERROR o UNKNOWN).

```
aws codebuild describe-test-cases \
  --report-arn <report-arn> \
  --filter status=SUCCEEDED|FAILED|SKIPPED|ERROR|UNKNOWN
```

El resultado es similar al siguiente.

```
{
  "testCases": [
    {
      "status": "FAILED",
      "name": "Test case 1",
      "expired": 1575916770.0,
      "reportArn": "<report-arn>",
      "prefix": "Cucumber tests for agent",
      "message": "A test message",
      "durationInNanoSeconds": 1540540,
      "testRawDataPath": "<path-to-output-report-files>"
    },
    {
      "status": "SUCCEEDED",
      "name": "Test case 2",
      "expired": 1575916770.0,
      "reportArn": "<report-arn>",
      "prefix": "Cucumber tests for agent",
      "message": "A test message",
      "durationInNanoSeconds": 1540540,
      "testRawDataPath": "<path-to-output-report-files>"
    }
  ]
}
```

```
}  
]  
}
```

Ejemplo de Docker en una imagen personalizada para CodeBuild

Este ejemplo compila y ejecuta una imagen de Docker utilizando AWS CodeBuild y una imagen de compilación de Docker personalizada (`docker:dind` en Docker Hub).

Para obtener información sobre cómo crear una imagen de Docker mediante el uso de una imagen de compilación proporcionada por CodeBuild con compatibilidad con Docker en su lugar, consulte nuestro [Ejemplo de Docker](#) (p. 101).

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por CodeBuild y por AWS recursos y acciones relacionados con Amazon S3, AWS KMS y CloudWatch Logs. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), y [Precios de Amazon CloudWatch](#).

Temas

- [Ejecución del ejemplo](#) (p. 99)
- [Estructura de directorios](#) (p. 100)
- [Files](#) (p. 100)
- [Recursos relacionados](#) (p. 50)

Ejecución del ejemplo

Para ejecutar este ejemplo

1. Cree los archivos tal y como se describe en las secciones "Estructura de directorios" y "Archivos" de este tema y cárguelos en un bucket de entrada de S3 o en un repositorio de AWS CodeCommit, GitHub o Bitbucket.

Important

No cargue (*root directory name*), solo los archivos incluidos en (*root directory name*).

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga los archivos y cárguelo en el bucket de entrada. No añada (*root directory name*) al archivo ZIP, solo los archivos incluidos en (*root directory name*).

2. Cree un proyecto de compilación, ejecute la compilación y consulte información de la compilación relacionada siguiendo los pasos de [Ejecutar AWS CodeBuild directamente](#) (p. 406).

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando `create-project` tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{  
  "name": "sample-docker-custom-image-project",  
  "source": {
```

```
"type": "S3",
"location": "codebuild-region-ID-account-ID-input-
bucket/DockerCustomImageSample.zip"
},
"artifacts": {
  "type": "NO_ARTIFACTS"
},
"environment": {
  "type": "LINUX_CONTAINER",
  "image": "docker:dind",
  "computeType": "BUILD_GENERAL1_SMALL",
  "privilegedMode": true
},
"serviceRole": "arn:aws:iam::account-ID:role/role-name",
"encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

3. Para ver los resultados de compilación, busque en el log de la compilación la cadena Hello, world! Para obtener más información, consulte [Ver detalles de las compilaciones](#) (p. 289).

Estructura de directorios

En este ejemplo se presupone que existe esta estructura de directorios.

```
(root directory name)
### buildspec.yml
### Dockerfile
```

Files

La imagen base del sistema operativo utilizado en este ejemplo es Ubuntu. El ejemplo usa los siguientes archivos. Para obtener más información acerca del controlador de almacenamiento de OverlayFS al que se hace referencia en el archivo buildspec, consulte [Use the OverlayFS storage driver](#) en el sitio web de Docker.

buildspec.yml (in (*root directory name*))

```
version: 0.2

phases:
  install:
    commands:
      - nohup /usr/local/bin/dockerd --host=unix:///var/run/docker.sock --
host=tcp://127.0.0.1:2375 --storage-driver=overlay2 &
      - timeout 15 sh -c "until docker info; do echo .; sleep 1; done"
  pre_build:
    commands:
      - docker build -t helloworld .
  build:
    commands:
      - docker images
      - docker run helloworld echo "Hello, World!"
```

Note

Si el sistema operativo base es Alpine Linux, en el `buildspec.yml` añada el argumento `-t a timeout:`

```
- timeout -t 15 sh -c "until docker info; do echo .; sleep 1; done"
```

Dockerfile (in *(root directory name)*)

```
FROM maven:3.3.9-jdk-8  
  
RUN echo "Hello World"
```

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Ejemplo de Docker para CodeBuild

Este ejemplo produce como salida de la compilación una imagen de Docker y la inserta después en un repositorio de imágenes de Amazon Elastic Container Registry (Amazon ECR). Puede adaptar este ejemplo para insertar la imagen de Docker en Docker Hub. Para obtener más información, consulte [Adaptación del ejemplo para insertar la imagen en Docker Hub \(p. 105\)](#).

Para obtener información acerca de cómo crear una imagen de Docker utilizando una imagen de compilación de Docker personalizada (`docker:dind` en Docker Hub), consulte nuestro [Ejemplo de Docker en una imagen personalizada \(p. 99\)](#).

Este ejemplo se ha probado con `golang:1.12`.

Utiliza la nueva característica de fases múltiples de Docker, que produce una imagen de Docker como salida de la compilación. A continuación, envía la imagen de Docker a un repositorio de imágenes de Amazon ECR. Las compilaciones de imágenes de fases múltiples de Docker ayudan a reducir el tamaño de la imagen de Docker final. Para obtener más información, consulte [Use multi-stage builds with Docker](#).

Important

La ejecución de este ejemplo puede producir cargos en su cuenta de AWS. Entre ellos, se incluyen posibles cargos por AWS CodeBuild y por recursos y acciones relacionados con Amazon S3, AWS KMS, CloudWatch Logs y Amazon ECR. Para obtener más información, consulte [Precios de CodeBuild](#), [Precios de Amazon S3](#), [AWS Key Management Service Precios de](#), [Precios de Amazon CloudWatch](#), y [Precios de Amazon Elastic Container](#).

Temas

- [Ejecución del ejemplo \(p. 102\)](#)
- [Estructura de directorios \(p. 104\)](#)
- [Files \(p. 104\)](#)
- [Adaptación del ejemplo para insertar la imagen en Docker Hub \(p. 105\)](#)

- [Recursos relacionados \(p. 50\)](#)

Ejecución del ejemplo

Para ejecutar este ejemplo

1. Si ya tiene un repositorio de imágenes en Amazon ECR que desee utilizar, vaya al paso 3. De lo contrario, si está utilizando un usuario de IAM en lugar de un AWS cuenta raíz o un usuario administrador de IAM para trabajar con Amazon ECR, agregue esta declaración (entre **### COMIENZA A AGREGAR DECLARACIÓN AQUÍ ###** y **### TERMINA AGREGANDO DECLARACIÓN AQUÍ ###**) al usuario (o grupo de IAM al que está asociado el usuario). **Uso de AWS** No se recomienda la cuenta raíz de. Esta instrucción permite crear repositorios de Amazon ECR para almacenar imágenes de Docker. Los puntos suspensivos (. . .) se usan por motivos de brevedad y para ayudarle a encontrar el lugar donde debe añadir la instrucción. No elimine ninguna instrucción ni incluya estos puntos suspensivos en la política. Para obtener más información, consulte [Uso de políticas insertadas mediante la AWS Management Console](#) en la IAM User Guide.

```
{
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Action": [
        "ecr:CreateRepository"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    ### END ADDING STATEMENT HERE ###
    ...
  ],
  "Version": "2012-10-17"
}
```

Note

La entidad de IAM que modifica esta política debe tener permiso en IAM para modificar políticas.

2. Cree un repositorio de imágenes en Amazon ECR. No olvide crear el repositorio en la misma región de AWS en la que va a crear el entorno de compilación y a ejecutar la compilación. Para obtener más información, consulte [Creación de un repositorio](#) en la Guía del usuario de Amazon ECR. El nombre del repositorio debe coincidir con el que especificará más adelante en este procedimiento, representado por la variable de entorno IMAGE_REPO_NAME.
3. Añada esta instrucción (entre **### BEGIN ADDING STATEMENT HERE ###** y **### END ADDING STATEMENT HERE ###**) a la política asociada a al rol de servicio de AWS CodeBuild. Esta instrucción permite a CodeBuild cargar imágenes de Docker en repositorios de Amazon ECR. Los puntos suspensivos (. . .) se usan por motivos de brevedad y para ayudarle a encontrar el lugar donde debe añadir la instrucción. No elimine ninguna instrucción ni incluya estos puntos suspensivos en la política.

```
{
  "Statement": [
    ### BEGIN ADDING STATEMENT HERE ###
    {
      "Action": [
        "ecr:BatchCheckLayerAvailability",
        "ecr:CompleteLayerUpload",
        "ecr:GetAuthorizationToken",
        "ecr:InitiateLayerUpload",
        "ecr:PutImage",

```

```
        "ecr:UploadLayerPart"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    ### END ADDING STATEMENT HERE ###
    ...
  ],
  "Version": "2012-10-17"
}
```

Note

La entidad de IAM que modifica esta política debe tener permiso en IAM para modificar políticas.

4. Cree los archivos tal y como se describe en las secciones "Estructura de directorios" y "Archivos" de este tema y cárguelos en un bucket de entrada de S3 o en un repositorio de AWS CodeCommit, GitHub o Bitbucket.

Important

No cargue (*root directory name*), solo los archivos incluidos en (*root directory name*).

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga los archivos y cárguelo en el bucket de entrada. No añada (*root directory name*) al archivo ZIP, solo los archivos incluidos en (*root directory name*).

5. Siga los pasos de [Ejecutar CodeBuild directamente \(p. 406\)](#) para crear un proyecto de compilación, ejecutar la compilación y consultar información de la compilación.

Si utiliza la consola para crear un proyecto:

- a. En Operating system (Sistema operativo), elija Ubuntu.
- b. En Runtime, elija Standard (Estándar).
- c. En Imagen, elija aws/codebuild/standard:4.0.
- d. Dado que utilizar este proyecto de compilación para crear una imagen de Docker, seleccione Privileged (Con privilegios).

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

- e. Añada las siguientes variables de entorno:
 - AWS_DEFAULT_REGION con un valor de *region-ID*
 - AWS_ACCOUNT_ID con un valor de ID de *account-ID*
 - IMAGE_TAG con un valor de Latest
 - IMAGE_REPO_NAME con un valor de *Amazon-ECR-repo-name*

Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando `create-project` tenga un aspecto similar al siguiente. (Sustituya los marcadores de posición por sus propios valores).

```
{
  "name": "sample-docker-project",
```

```
"source": {
  "type": "S3",
  "location": "codebuild-region-ID-account-ID-input-bucket/DockerSample.zip"
},
"artifacts": {
  "type": "NO_ARTIFACTS"
},
"environment": {
  "type": "LINUX_CONTAINER",
  "image": "aws/codebuild/standard:4.0",
  "computeType": "BUILD_GENERAL1_SMALL",
  "environmentVariables": [
    {
      "name": "AWS_DEFAULT_REGION",
      "value": "region-ID"
    },
    {
      "name": "AWS_ACCOUNT_ID",
      "value": "account-ID"
    },
    {
      "name": "IMAGE_REPO_NAME",
      "value": "Amazon-ECR-repo-name"
    },
    {
      "name": "IMAGE_TAG",
      "value": "latest"
    }
  ],
  "privilegedMode": true
},
"serviceRole": "arn:aws:iam::account-ID:role/role-name",
"encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

6. Confirme que CodeBuild ha insertado correctamente la imagen de Docker en el repositorio:

1. Abra la consola de Amazon ECR en <https://console.aws.amazon.com/ecr/>.
2. Elija el nombre del repositorio. La imagen debe mostrarse en la columna Image tag (Etiqueta de imagen).

Estructura de directorios

En este ejemplo se presupone que existe esta estructura de directorios.

```
(root directory name)
### buildspec.yml
### Dockerfile
```

Files

Este ejemplo usa los siguientes archivos.

buildspec.yml (in (*root directory name*))

```
version: 0.2

phases:
  pre_build:
    commands:
      - echo Logging in to Amazon ECR...
```

```
- aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --username
AWS --password-stdin $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com
build:
  commands:
    - echo Build started on `date`
    - echo Building the Docker image...
    - docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
    - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $AWS_ACCOUNT_ID.dkr.ecr.
$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG
  post_build:
    commands:
      - echo Build completed on `date`
      - echo Pushing the Docker image...
      - docker push $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/
$IMAGE_REPO_NAME:$IMAGE_TAG
```

Dockerfile (in *(root directory name)*)

```
FROM golang:1.12-alpine AS build
#Install git
RUN apk add --no-cache git
#Get the hello world package from a GitHub repository
RUN go get github.com/golang/example/hello
WORKDIR /go/src/github.com/golang/example/hello
# Build the project and send the output to /bin/HelloWorld
RUN go build -o /bin/HelloWorld

FROM golang:1.12-alpine
#Copy the build's output binary from the previous build container
COPY --from=build /bin/HelloWorld /bin/HelloWorld
ENTRYPOINT ["/bin/HelloWorld"]
```

Note

CodeBuild anula el `ENTRYPOINT` para imágenes de Docker personalizadas.

Adaptación del ejemplo para insertar la imagen en Docker Hub

Para insertar la imagen de Docker en Docker Hub en lugar de en Amazon ECR, modifique el código de este ejemplo.

Note

Si está utilizando una versión de Docker anterior a la 17.06, quite la opción `--no-include-email`.

1. Sustituya estas líneas de código específicas de Amazon ECR en `labuildspec.yml` file:

```
...
pre_build:
  commands:
    - echo Logging in to Amazon ECR...
    - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --
username AWS --password-stdin $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com
  build:
    commands:
      - echo Build started on `date`
      - echo Building the Docker image...
      - docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
      - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $AWS_ACCOUNT_ID.dkr.ecr.
$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG
```



```
post_build:
  commands:
    - echo Build completed on `date`
    - echo Pushing the Docker image...
    - docker push $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/
      $IMAGE_REPO_NAME:$IMAGE_TAG
  ...
```

Por estas líneas de código específicas de Docker Hub:

```
...
pre_build:
  commands:
    - echo Logging in to Docker Hub...
    # Type the command to log in to your Docker Hub account here.
build:
  commands:
    - echo Build started on `date`
    - echo Building the Docker image...
    - docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
    - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $IMAGE_REPO_NAME:$IMAGE_TAG
post_build:
  commands:
    - echo Build completed on `date`
    - echo Pushing the Docker image...
    - docker push $IMAGE_REPO_NAME:$IMAGE_TAG
  ...
```

2. Cargue el código modificado en un bucket de S3 o en un repositorio de AWS CodeCommit, GitHub o Bitbucket

Important

No cargue (*root directory name*), solo los archivos incluidos en (*root directory name*).

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga los archivos y cárguelo en el bucket de entrada. No añada (*root directory name*) al archivo ZIP, solo los archivos incluidos en (*root directory name*).

3. Sustituya estas líneas de código de la entrada con formato JSON del comando `create-project`:

```
...
"environmentVariables": [
  {
    "name": "AWS_DEFAULT_REGION",
    "value": "region-ID"
  },
  {
    "name": "AWS_ACCOUNT_ID",
    "value": "account-ID"
  },
  {
    "name": "IMAGE_REPO_NAME",
    "value": "Amazon-ECR-repo-name"
  },
  {
    "name": "IMAGE_TAG",
    "value": "latest"
  }
]
...
```

Por estas líneas de código:

```
...
  "environmentVariables": [
    {
      "name": "IMAGE_REPO_NAME",
      "value": "your-Docker-Hub-repo-name"
    },
    {
      "name": "IMAGE_TAG",
      "value": "latest"
    }
  ]
}
```

4. Siga los pasos de [Ejecutar CodeBuild directamente \(p. 406\)](#) para crear un entorno de compilación, ejecutar la compilación y consultar información de la compilación relacionada.
5. Confirme que AWS CodeBuild ha insertado correctamente la imagen de Docker en el repositorio. Inicie sesión en Docker Hub, vaya al repositorio y elija la pestaña Tags. La etiqueta `latest` debe contener un valor de Last Updated muy reciente.

Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información sobre cuotas de CodeBuild, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Ejemplo de GitHub Enterprise Server para CodeBuild

AWS CodeBuild admite GitHub Enterprise Server como repositorio de código fuente. Este ejemplo muestra cómo configurar proyectos de CodeBuild cuando el repositorio de GitHub Enterprise Server tiene instalado un certificado. También muestra cómo habilitar webhooks para que CodeBuild vuelva a compilar el código fuente cada vez que se envíe un cambio de código al repositorio de GitHub Enterprise Server.

Prerequisites

1. Genere un token de acceso personal para su proyecto CodeBuild. Le recomendamos que cree un usuario de GitHub Enterprise y que genere un token de acceso personal para dicho usuario. Cópielo en el portapapeles para que pueda utilizarlo al crear un proyecto de CodeBuild. Para obtener más información, consulte [Creating a personal access token for the command line](#) en el sitio web GitHub Help.

Cuando cree el token de acceso personal, incluya el ámbito del repositorio en la definición.

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories

2. Descargue el certificado desde GitHub Enterprise Server. CodeBuild utiliza el certificado para establecer una conexión SSL de confianza con el repositorio.

Clientes Linux/macOS:

En una ventana de terminal de , ejecute el siguiente comando:

```
echo -n | openssl s_client -connect HOST:PORTNUMBER \  
| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > /folder/filename.pem
```

Sustituya los marcadores de posición del comando por los siguientes valores:

HOST. La dirección IP del repositorio de GitHub Enterprise Server.

PORTNUMBER. El número de puerto que se utiliza para conectarse (por ejemplo, 443).

folder. La carpeta en la que se ha descargado el certificado.

nombre de archivo. El nombre de archivo del archivo de certificado.

Important

Guarde el certificado como un archivo .pem.

Clientes de Windows:

Use el navegador para descargar el certificado desde GitHub Enterprise Server. Para ver los detalles del certificado del sitio, seleccione el icono del candado. Para obtener información sobre cómo exportar el certificado, consulte la documentación del navegador.

Important

Guarde el certificado como un archivo .pem.

3. Cargue el archivo de certificado en un bucket de S3. Para obtener información acerca de cómo crear un bucket de S3, consulte [¿Cómo puedo crear un bucket de S3?](#) Para obtener información acerca de cómo cargar objetos en un bucket de S3, consulte [¿Cómo puedo cargar archivos y carpetas en un bucket?](#)

Note

Este cubo debe estar en el mismoAWS región como compilaciones. Por ejemplo, si indica a CodeBuild que ejecute una compilación en la región EE.UU. Este (Ohio), el bucket debe estar en la región EE.UU. Este (Ohio).

Crear un proyecto de compilación con GitHub Enterprise Server como repositorio de código fuente y habilitar los webhooks (consola)

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Si aparece una página con información de CodeBuild, seleccioneCrear un proyecto de compilación. En caso contrario, en el panel de navegación expandaBuild, eligeProyectos de compilacióny luegoCrear un proyecto de compilación.
3. En Project name (Nombre de proyecto), escriba un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. También puede incluir una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.
4. En Source (Código fuente), en Source provider (Proveedor de código fuente), elija GitHub Enterprise.

- En Personal Access Token, pegue el token que ha copiado en el portapapeles y elija Save Token. En Repository URL, introduzca la URL del repositorio de GitHub Enterprise Server.

Note

Solo tiene que introducir y guardar el token de acceso personal una vez. Todos los proyectos futuros de AWS CodeBuild utilizarán este token.

- En Repository URL (URL del repositorio), escriba la ruta al repositorio, incluido el nombre del repositorio.
- Expanda Additional configuration (Configuración adicional).
- Seleccione Rebuild every time a code change is pushed to this repository (Volver a compilar cada vez que un cambio de código se inserte en el repositorio) para recompilar cada vez que se inserte un cambio de código en este repositorio.
- Seleccione Enable insecure SSL (Habilitar SSL no seguro) para omitir las advertencias de SSL cuando se conecte al repositorio del proyecto de GitHub Enterprise Server.

Note

Le recomendamos que solamente utilice Enable insecure SSL (Habilitar SSL no seguro) para realizar pruebas. No debe utilizarse en un entorno de producción.

Source

Source 1 - Primary

Source provider

GitHub Enterprise ▼

Repository URL

https://<host-name>/<user-name>/<repository-name>

Disconnect GitHub Enterprise account

▼ Additional configuration

Git clone depth, Insecure SSL

Git clone depth - *optional*

1 ▼

Webhook - *optional*

☒ Rebuild every time a code change is pushed to this repository

Branch filter - *optional*

Enter a regular expression

Insecure SSL - *optional*

Enable this flag to ignore SSL warnings while connecting to project source.

☐ Enable insecure SSL

5. En Environment (Entorno):

En Environment image (Imagen del entorno), realice alguna de las siguientes operaciones:

- Para usar una imagen de Docker administrada por AWS CodeBuild, elija Managed image (Imagen administrada) y después elija una opción en Operating system (Sistema operativo), Runtime (Tiempo de ejecución), Image (Imagen) e Image version (Versión de imagen). Realice una selección en Environment type (Tipo de entorno) si está disponible.
 - Para usar otra imagen de Docker, elija Custom image (Imagen personalizada). Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige Amazon ECR, use Repositorio de Amazon ECR. Imagen de Amazon ECR para elegir la imagen de Docker en su AWS account.
 - Para usar una imagen privada de Docker, elija Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Las credenciales las debe crear Secrets Manager. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.
6. En Service role (Rol de servicio), realice una de las operaciones siguientes:
- Si no tiene un rol de servicio de CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
 - Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola para crear o actualizar un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

7. Expanda Additional configuration (Configuración adicional).

Si desea que CodeBuild funcione con su VPC:

- Para VPC:, elija el ID de VPC que utiliza CodeBuild.
- Para Subredes de la VPC, elija las subredes que incluyen recursos utilizados por CodeBuild.
- Para Grupos de seguridad de la VPC, elija los grupos de seguridad que utiliza CodeBuild para permitir el acceso a los recursos de las VPC.

Para obtener más información, consulte [Usar AWS CodeBuild con Amazon Virtual Private Cloud \(p. 180\)](#).

8. En BuildSpec, realice una de las siguientes operaciones:

- Seleccionar Utilizar un archivo buildspec para utilizar el archivo buildspec.yml en el directorio raíz del código fuente.
- Seleccionar Insertar comandos de compilación para utilizar la consola para insertar comandos de compilación.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

9. En Artifacts (Artefactos), en Type (Tipo), realice una de las siguientes operaciones:

- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts (Sin artefactos).

- Para almacenar la salida de la compilación en un bucket de S3, elija Amazon S3y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. De forma predeterminada, el nombre del artefacto es el nombre del proyecto. Si desea usar otro nombre, escríbalo en el cuadro de nombre de artefactos. Si desea producir un archivo ZIP, incluya la extensión zip.
 - En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.
 - Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, `appspec.yml, target/my-app.jar`). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec \(p. 137\)](#).
10. En Cache type (Tipo de caché), seleccione una de las opciones siguientes:
- Si no desea usar una memoria caché, elija No cache.
 - Si desea utilizar una caché de Amazon S3, elija Amazon S3y, a continuación, haga lo siguiente:
 - En Bucket, elija el nombre del bucket de S3 donde se almacena la caché.
 - (Opcional) Para Prefijo de ruta de caché, introduzca un prefijo de ruta de Amazon S3. El valor Cache path prefix (Prefijo de ruta de caché) es similar a un nombre de directorio. Le permite almacenar la caché en el mismo directorio en un bucket.

Important

No añada una barra (/) al final del prefijo de ruta.

- Si desea utilizar una caché local, seleccione Local y elija uno o varios modos de caché local.

Note

El modo Docker layer cache (Caché de capas de Docker) solo está disponible para Linux. Si lo selecciona, el proyecto deberá ejecutarse en modo con privilegios. La ARM_CONTAINERY LINUX_GPU_CONTAINER tipos de entorno y BUILD_GENERAL1_2XLARGE el tipo de cómputo no admite el uso de una caché local.

El uso de memoria caché ahorra mucho tiempo de compilación, ya que algunas partes reutilizables del entorno de compilación se almacenan en ella y se usan en las distintas compilaciones. Para obtener información acerca de cómo especificar una caché en el archivo de especificación de compilación, consulte [Sintaxis de buildspec \(p. 137\)](#). Para obtener más información acerca del almacenamiento en caché, consulte [Almacenamiento en caché de compilaciones de AWS CodeBuild \(p. 229\)](#).

11. Elija Create build project (Crear proyecto de compilación). En la página del proyecto de compilación, elija Start build (Iniciar compilación).
12. Si ha habilitado los webhooks en Source (Código fuente), aparecerá el cuadro de diálogo Create webhook (Crear webhook) con los valores para Payload URL (URL de carga) y Secret (Secreto).

Important

El cuadro de diálogo Create webhook solo aparecerá una vez. Copie la URL de carga y la clave secreta. Las necesitará al añadir un webhook en GitHub Enterprise Server. Si necesita generar de nuevo una URL de carga y una clave secreta, primero debe eliminar el webhook del repositorio de GitHub Enterprise Server. En tu proyecto de CodeBuild, borra la Webhook casilla de verificación y, a continuación, seleccione Save (Guardar). A continuación, puede crear o actualizar un proyecto CodeBuild con la CLI de Webhook casilla de verificación seleccionada. El cuadro de diálogo Create webhook aparecerá de nuevo.

13. En GitHub Enterprise Server, elija el repositorio en el que está almacenado su proyecto de CodeBuild está almacenado.

14. Elija Settings (Configuración), elija Hooks & services (Enlaces y servicios) y, a continuación, elija Add webhook (Añadir webhook).
15. Introduzca la URL de carga y la clave secreta, acepte los valores predeterminados para el resto de los campos y elija Add webhook.

requests 0 Projects 0 Wiki Pulse Graphs Settings

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL *

Content type

application/json

Secret

By default, we verify SSL certificates when delivering payloads. [Disable SSL verification](#)

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me everything.

☐ Let me select individual events.

☒ Active
We will deliver event details when this hook is triggered.

Add webhook

16. Vuelva al proyecto de CodeBuild. Cierre el cuadro de diálogo Create webhook y elija Start build.

Ejemplo de filtros de webhooks y solicitudes de extracción de GitHub

AWS CodeBuild admite webhooks, cuando el repositorio de origen es GitHub. Esto significa que para un proyecto de compilación de CodeBuild que tenga su código fuente almacenado en un repositorio de GitHub, pueden utilizarse webhooks para volver a compilar el código fuente cada vez que se inserte un cambio de código en el repositorio.

Note

Al utilizar webhooks, es posible que un usuario active una compilación inesperada. Para mitigar este riesgo, consulte [Prácticas recomendadas para el uso de webhooks](#) (p. 237).

Creación de un proyecto de compilación con GitHub como repositorio de origen y habilitación de webhooks (consola)

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Si aparece una página con información de CodeBuild, seleccioneCrear un proyecto de compilación. En caso contrario, en el panel de navegación expandaBuild, eligeProyectos de compilacióny luegoCrear un proyecto de compilación.
3. Elija Create build project (Crear proyecto de compilación).
4. En Project configuration (Configuración del proyecto):

Project name (Nombre de proyecto)

Introduzca un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. También puede incluir una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.

5. En Source (Código fuente):

Proveedor de origen

SeleccionarGitHub. Siga las instrucciones para conectarse (o volver a conectarse) a GitHub y, a continuación, elija Authorize (Autorizar).

Repositorio

Seleccione Repository in my GitHub account (Repositorio en mi cuenta de GitHub).

Repositorio GitHub

Introduce la URL de tu repositorio de GitHub.

6. EnEventos de webhooks de origen principal, seleccione lo siguiente.

Note

LaEventos de webhooks de origen principalla sección solo es visible si eligesRepositorio en mi cuenta de GitHuben el paso anterior.

1. Cuando cree el proyecto, seleccione Rebuild every time a code change is pushed to this repository (Volver a compilar cada vez que se inserte un cambio de código en este repositorio).
2. En Event type (Tipo de evento), seleccione uno o varios eventos.
3. Para filtrar en función de cuándo un evento va a desencadenar una compilación, en Start a build under these conditions (Iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
4. Para filtrar en función de cuándo no se va a desencadenar un evento, en Don't start a build under these conditions (No iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
5. SeleccionarAgregar grupo de filtrospara agregar otro grupo de filtros, si es necesario.

Para obtener más información acerca de los tipos de eventos y filtros de GitHub webhook, consulte[Eventos de webhooks de GitHub \(p. 246\)](#).

7. En Environment (Entorno):

Imagen de entorno

Elija una de las siguientes opciones:

Versión de API 2016-10-06

Para utilizar una imagen Docker administrada por AWS CodeBuild:

Seleccionar Imagen administrada y, a continuación, realice selecciones desde Sistema operativo, Tiempo (s) de ejecución, Image (Imagen), y Versión de imagen. Realice una selección en Environment type (Tipo de entorno) si está disponible.

Para usar otra imagen de Docker:

Seleccionar Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige Amazon ECR, use Repositorio de Amazon ECR y Imagen de Amazon ECR para elegir la imagen de Docker en su AWS account.

Para utilizar una imagen privada de Docker:

Seleccionar Imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Las credenciales las debe crear Secrets Manager. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Rol de servicio

Elija una de las siguientes opciones:

- Si no tiene un rol de servicio de CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
- Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola para crear o actualizar un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

8. En BuildSpec, realice una de las siguientes operaciones:

- Seleccionar Utilizar un archivo buildspec para utilizar el archivo buildspec.yml en el directorio raíz del código fuente.
- Seleccionar Insertar comandos de compilación para utilizar la consola para insertar comandos de compilación.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

9. En Artifacts (Artefactos):

Type

Elija una de las siguientes opciones:

- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts (Sin artefactos).
- Para almacenar la salida de la compilación en un bucket de S3, elija Amazon S3 y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. De forma predeterminada, el nombre del artefacto es el nombre del proyecto. Si desea usar otro

nombre, escríbalo en el cuadro de nombre de artefactos. Si desea producir un archivo ZIP, incluya la extensión zip.

- En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.
- Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, `appspec.yml`, `target/my-app.jar`). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec \(p. 137\)](#).

Configuración adicional

Expanda Additional configuration (Configuración adicional) y establezca las opciones según sea necesario.

10. Elija Create build project (Crear proyecto de compilación). En la página Review (Revisar), elija Start build (Comenzar compilación) para ejecutar la compilación.

Comprobaciones de verificación

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Aplique alguna de las siguientes acciones:
 - Elija el enlace del proyecto de compilación con webhooks que desea verificar y, a continuación, seleccione Build details (Detalles de compilación).
 - Seleccione el botón de opción situado junto al proyecto de compilación con webhooks que desea verificar, elijaView details (Ver detalles).y luego enDetalles de compilaciónPestaña.
4. EnEventos de webhooks de origen principal, elige elWebhookEnlace URL.
5. En el repositorio de GitHub, en la página Settings (Configuración), bajo Webhooks, compruebe que las opciones Pull Requests (Solicitudes de extracción) y Pushes (Inserciones) están seleccionadas.
6. En la configuración del perfil de GitHub, en Personal settings (Configuración personal), Applications (Aplicaciones), Authorized OAuth Apps (Aplicaciones de OAuth autorizadas), debería ver que la aplicación tiene autorización para acceder a la región de AWS seleccionada.

Crear un sitio web estático con la salida de compilación alojada en un bucket de S3

Puede deshabilitar el cifrado de los artefactos en una compilación, por ejemplo, para publicar artefactos en una ubicación configurada para alojar un sitio web. (Los artefactos cifrados no se pueden publicar). En este ejemplo, se muestra cómo utilizar los webhooks para desencadenar una compilación y publicar sus artefactos en un bucket de S3 configurado como un sitio web.

1. Siga las instrucciones de [configuración para sitio web estático](#) si quiere configurar un bucket de S3 para que funcione como un sitio web.
2. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
3. Si aparece una página con información de CodeBuild, seleccioneCrear un proyecto de compilación. En caso contrario, en el panel de navegación expandaBuild, eligeProyectos de compilacióny luegoCrear un proyecto de compilación.
4. En Project name (Nombre de proyecto), escriba un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS. También puede

incluir una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.

5. En Source (Código fuente), para Source provider (Proveedor de código fuente), elija GitHub. Siga las instrucciones para conectarse (o volver a conectarse) a GitHub y, a continuación, elija Authorize (Autorizar).

En Webhook, seleccione Rebuild every time a code change is pushed to this repository (Volver a compilar cada vez que un cambio de código se inserte en el repositorio). Solo puede seleccionar esta casilla de verificación si eligió Use a repository in my account (Usar un repositorio de mi cuenta).

Source Add source

Source 1 - Primary

Source provider

GitHub

Repository

☐ Public repository ☒ Repository in my GitHub account

GitHub repository

↕ ↻

Disconnect GitHub account

▼ Additional configuration

Git clone depth

Git clone depth - optional

1

Build Status - optional

☐ Report build statuses to source provider when your builds start and finish

Webhook - optional

☒ Rebuild every time a code change is pushed to this repository

Branch filter - optional

Enter a regular expression

6. En Environment (Entorno):

En Environment image (Imagen del entorno), realice alguna de las siguientes operaciones:

- Para usar una imagen de Docker administrada por AWS CodeBuild, elija Managed image (Imagen administrada) y después elija una opción en Operating system (Sistema operativo), Runtime (Tiempo de ejecución), Image (Imagen) e Image version (Versión de imagen). Realice una selección en Environment type (Tipo de entorno) si está disponible.
- Para usar otra imagen de Docker, elija Custom image (Imagen personalizada). Para Tipo de entorno, elige ARM, Linux, GPU Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si

eligeAmazon ECR, useRepositorio de Amazon ECRylimagen de Amazon ECRpara elegir la imagen de Docker en suAWSaccount.

- Para usar una imagen privada de Docker, elijaimagen personalizada. ParaTipo de entorno, eligeARM, Linux, GPU Linux, o bienWindows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Las credenciales las debe crear Secrets Manager. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.
7. En Service role (Rol de servicio), realice una de las operaciones siguientes:
- Si no tiene un rol de servicio de CodeBuild, elijaNuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
 - Si dispone de un rol de servicio CodeBuild, elijaRol de servicio existente. EnARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola para crear o actualizar un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

8. EnBuildSpec, realice una de las siguientes operaciones:
- SeleccionarUtilizar un archivo buildspecpara utilizar el archivo buildspec.yml en el directorio raíz del código fuente.
 - SeleccionarInsertar comandos de compilaciónpara utilizar la consola para insertar comandos de compilación.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

9. EnArtifacts, paraTipo, eligeAmazon S3para almacenar la salida de la compilación en un bucket de S3.
10. En Bucket name (Nombre del bucket), elija el nombre del bucket de S3 que ha configurado para que funcione como un sitio web en el paso 1.
11. Si eligió Insert build commands (Insertar comandos de compilación) en Environment: How to build (Entorno: cómo compilar), escriba las ubicaciones de los archivos de la compilación que desea incluir en el bucket de salida en Output files (Archivos de salida). Si tiene varias ubicaciones, utilice una coma para separarlas (por ejemplo, **appspect.yml, target/my-app.jar**). Para obtener más información, consulte [Artifacts reference-key in the buildspec file](#).
12. Seleccione Disable artifacts encryption (Deshabilitar cifrado de artefactos).
13. Expanda Additional configuration (Configuración adicional) y elija las opciones que correspondan.
14. Elija Create build project (Crear proyecto de compilación). En la página del proyecto de compilación, en Build history (Historial de compilaciones), elija Start build (Comenzar compilación) para ejecutar la compilación.
15. (Opcional) Siga las instrucciones en[Ejemplo: Acelere su sitio web con Amazon CloudFront](#)en laGuía para desarrolladores de Amazon S3.

Versiones del entorno de ejecución en el ejemplo del archivo buildspec para CodeBuild

Si utiliza la versión 1.0 o posterior de la imagen estándar de Amazon Linux 2 (AL2) o la versión 2.0 o posterior de la imagen estándar de Ubuntu, puede especificar uno o varios entornos de ejecución en

el `runtime-versions` sección del archivo buildspec. Este ejemplo muestra cómo puede cambiar el runtime de su proyecto, especificar más de un runtime y especificar un runtime que depende de otro. Para obtener información sobre los runtimes compatibles, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).

Note

Si utiliza Docker en el contenedor de compilación, la compilación debe ejecutarse en un modo con privilegios. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#) y [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#).

Actualización de la versión del entorno de ejecución

Puede modificar el entorno de ejecución utilizado por su proyecto a una nueva versión actualizando el `runtime-versions` sección del archivo buildspec. Los siguientes ejemplos muestran cómo se especifican las versiones 8 y 11 de Java.

- Una sección `runtime-versions` que especifica la versión 8 de Java:

```
phases:
  install:
    runtime-versions:
      java: corretto8
```

- Una sección `runtime-versions` que especifica la versión 11 de Java:

```
phases:
  install:
    runtime-versions:
      java: corretto11
```

Los siguientes ejemplos muestran cómo especificar diferentes versiones de Python utilizando la imagen estándar 5.0 de Ubuntu o la imagen estándar 3.0 de:

- A `runtime-versions` que especifica Python versión 3.7:

```
phases:
  install:
    runtime-versions:
      python: 3.7
```

- A `runtime-versions` que especifica Python versión 3.8:

```
phases:
  install:
    runtime-versions:
      python: 3.8
```

Esta muestra presenta un proyecto que comienza con el tiempo de ejecución de la versión 8 de Java y, a continuación, se actualiza al tiempo de ejecución de la versión 10 de Java.

1. Siga los pasos 1 y 2 de [Crear el código fuente \(p. 67\)](#) para generar el código fuente. Si se realiza correctamente, se crea un directorio denominado `my-web-app` con sus archivos de código fuente.
2. Cree un archivo denominado `buildspec.yml` con el siguiente contenido. Guarde el archivo en el directorio `(root directory)/my-web-app`.

```
version: 0.2
```

```
phases:
  install:
    runtime-versions:
      java: corretto8
  build:
    commands:
      - java -version
      - mvn package
artifacts:
  files:
    - '**/*'
base-directory: 'target/my-web-app'
```

En el archivo buildspec:

- La sección `runtime-versions` especifica que el proyecto utiliza la versión 8 del tiempo de ejecución de Java.
- El comando `- java -version` muestra la versión de Java que utiliza su proyecto cuando se compila.

La estructura de archivos debe ser como la siguiente.

```
(root directory name)
### my-web-app
### src
#   ### main
#   ### resources
#   ### webapp
#       ### WEB-INF
#           ### web.xml
#               ### index.jsp
### buildspec.yml
### pom.xml
```

3. Suba el contenido de `my-web-app` en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket.

Important

No cargue `(root directory name)` o `(root directory name)/my-web-app`, solo los directorios y archivos incluidos en `(root directory name)/my-web-app`.

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga la estructura de directorios y los archivos para cargarlo después en el bucket de entrada. No añada `(root directory name)` o `(root directory name)/my-web-app` al archivo ZIP, solo los directorios y archivos incluidos en `(root directory name)/my-web-app`.

4. Abra el icono **AWS CodeBuild Consola** de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
5. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
 - Para Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - Para Image (Imagen), elija `aws/codebuild/amazonlinux2-x86_64-standard:3.0`.
6. Seleccione **Start build**.

7. En Build configuration (Configuración de la compilación), acepte los valores predeterminados y elija Start build (Comenzar compilación).
8. Una vez finalizada la compilación, consulte la salida de compilación en la pestaña Build logs (Registros de compilación). Debería ver un resultado similar a este:

```
[Container] Date Time Phase is DOWNLOAD_SOURCE
[Container] Date Time CODEBUILD_SRC_DIR=/codebuild/output/src460614277/src
[Container] Date Time YAML location is /codebuild/output/src460614277/src/buildspec.yml
[Container] Date Time Processing environment variables
[Container] Date Time Selecting 'java' runtime version 'corretto8' based on manual
selections...
[Container] Date Time Running command echo "Installing Java version 8 ..."
Installing Java version 8 ...

[Container] Date Time Running command export JAVA_HOME="$JAVA_8_HOME"

[Container] Date Time Running command export JRE_HOME="$JRE_8_HOME"

[Container] Date Time Running command export JDK_HOME="$JDK_8_HOME"

[Container] Date Time Running command for tool_path in "$JAVA_8_HOME"/bin/*
"$JRE_8_HOME"/bin/*;
```

9. Actualice la sección runtime-versions con la versión 11 de Java:

```
install:
  runtime-versions:
    java: corretto11
```

10. Después de guardar el cambio, ejecute la compilación de nuevo y consulte la salida de compilación. Debería ver que la versión instalada de Java es la 11. Debería ver un resultado similar a este:

```
[Container] Date Time Phase is DOWNLOAD_SOURCE
[Container] Date Time CODEBUILD_SRC_DIR=/codebuild/output/src460614277/src
[Container] Date Time YAML location is /codebuild/output/src460614277/src/buildspec.yml
[Container] Date Time Processing environment variables
[Container] Date Time Selecting 'java' runtime version 'corretto11' based on manual
selections...
Installing Java version 11 ...

[Container] Date Time Running command export JAVA_HOME="$JAVA_11_HOME"

[Container] Date Time Running command export JRE_HOME="$JRE_11_HOME"

[Container] Date Time Running command export JDK_HOME="$JDK_11_HOME"

[Container] Date Time Running command for tool_path in "$JAVA_11_HOME"/bin/*
"$JRE_11_HOME"/bin/*;
```

Especificación de una dependencia del entorno de ejecución

En este ejemplo se muestra cómo especificar un runtime y un runtime de dependencia. Por ejemplo, las versiones del runtime de Android compatibles dependen de la versión 8 del runtime de Java. Por ejemplo, si especifica la versión 29 de Android y utiliza Amazon Linux 2 o Ubuntu, también puede especificar la versión 8 de Java. Si no especifica el entorno de ejecución dependiente, CodeBuild intentará elegirlo por usted.

El proyecto de compilación de este ejemplo utiliza código fuente en GitHub [AWSEjemplos de](#). El código fuente utiliza el runtime de Android versión 28 y el proyecto de compilación utiliza Amazon Linux 2, por lo que buildspec también especifica la versión 8 de Java.

1. Abra el iconoAWS CodeBuildConsola de en<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
 - ParaFuente:
 - En Source provider (Proveedor de código fuente), elija GitHub.

Si no has conectado con anterioridad tu cuenta de GitHub, eligeConnect mediante OAuthorConnect con un token de acceso personal de GitHuby siga las instrucciones para conectarse (o volver a conectarse) a GitHub y autorizar el acceso aAWS CodeBuild.
 - En Repository (Repositorio), elija Public repository (Repositorio público).
 - En URL de repositorio, especifique **`https://github.com/aws-samples/aws-mobile-android-notes-tutorial`**
 - Para Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - ParaImage (Imagen), eligeaws/codebuild/amazonlinux2-x86_64-standard:3.0.
3. En Build specifications (Especificaciones de compilación), elija Insert build commands (Insertar comandos de compilación) y, a continuación, elija Switch to editor (Cambiar a editor).
4. En Build commands (Comandos de compilación), reemplace el texto del marcador de posición por lo siguiente:

```
version: 0.2

phases:
  install:
    runtime-versions:
      android: 29
      java: corretto8
    build:
      commands:
        - ./gradlew assembleDebug
  artifacts:
    files:
      - app/build/outputs/apk/app-debug.apk
```

La sección runtime-versions especifica los runtimes de la versión 29 de Android y de la versión 8 de Java.

5. Elija Create build project (Crear proyecto de compilación).
6. Seleccione Start build.
7. En Build configuration (Configuración de la compilación), acepte los valores predeterminados y elija Start build (Comenzar compilación).
8. Una vez finalizada la compilación, consulte la salida de compilación en la pestaña Build logs (Registros de compilación). Debería ver un resultado similar a este. Indica que se ha instalado la versión 29 de Android y la versión 8 de Java:

```
[Container] 2019/05/14 23:21:42 Entering phase DOWNLOAD_SOURCES
[Container] Date Time Running command echo "Installing Android version 29 ..."
Installing Android version 29 ...
```

```
[Container] Date Time Running command echo "Installing Java version 8 ..."  
Installing Java version 8 ...
```

Especificación de dos entornos de ejecución

Puede especificar más de un tiempo de ejecución en el mismo proyecto de compilación de CodeBuild. Este proyecto de muestra utiliza dos archivos de código fuente: uno que utiliza el tiempo de ejecución de Go y otro que utiliza el tiempo de ejecución de Node.js.

1. Cree un directorio llamado `my-source`.
2. En el directorio `my-source`, cree un directorio llamado `golang-app`.
3. Cree un archivo denominado `hello.go` con el siguiente contenido. Guarde el archivo en el directorio `golang-app`.

```
package main  
import "fmt"  
  
func main() {  
    fmt.Println("hello world from golang")  
    fmt.Println("1+1 =", 1+1)  
    fmt.Println("7.0/3.0 =", 7.0/3.0)  
    fmt.Println(true && false)  
    fmt.Println(true || false)  
    fmt.Println(!true)  
    fmt.Println("good bye from golang")  
}
```

4. En el directorio `my-source`, cree un directorio llamado `nodejs-app`. Debe estar al mismo nivel que el directorio `golang-app`.
5. Cree un archivo denominado `index.js` con el siguiente contenido. Guarde el archivo en el directorio `nodejs-app`.

```
console.log("hello world from nodejs");  
console.log("1+1 =" + (1+1));  
console.log("7.0/3.0 =" + 7.0/3.0);  
console.log(true && false);  
console.log(true || false);  
console.log(!true);  
console.log("good bye from nodejs");
```

6. Cree un archivo denominado `package.json` con el siguiente contenido. Guarde el archivo en el directorio `nodejs-app`.

```
{  
  "name": "mycompany-app",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"run some tests here\""  
  },  
  "author": "",  
  "license": "ISC"  
}
```

7. Cree un archivo denominado `buildspec.yml` con el siguiente contenido. Guarde el archivo en el directorio `my-source`, al mismo nivel que los directorios `nodejs-app` y `golang-app`. `Laruntime-versions` especifica los tiempos de ejecución de la versión 12 de Node.js y la versión 1.13 de Go.

```
version: 0.2

phases:
  install:
    runtime-versions:
      golang: 1.13
      nodejs: 12
  build:
    commands:
      - echo Building the Go code...
      - cd $CODEBUILD_SRC_DIR/golang-app
      - go build hello.go
      - echo Building the Node code...
      - cd $CODEBUILD_SRC_DIR/nodejs-app
      - npm run test
artifacts:
  secondary-artifacts:
    golang_artifacts:
      base-directory: golang-app
      files:
        - hello
    nodejs_artifacts:
      base-directory: nodejs-app
      files:
        - index.js
        - package.json
```

8. La estructura de archivos debe ser como la siguiente.

```
my-source
### golang-app
#   ### hello.go
### nodejs.app
#   ### index.js
#   ### package.json
### buildspec.yml
```

9. Suba el contenido de `my-source` en un bucket de entrada de S3 o en un repositorio de CodeCommit, GitHub o Bitbucket.

Important

Si utiliza un bucket de entrada de S3, no olvide crear un archivo ZIP que contenga la estructura de directorios y los archivos para cargarlo después en el bucket de entrada. No añada `my-source` al archivo ZIP, solo los directorios y archivos incluidos en `my-source`.

10. Abra el icono **AWS CodeBuild Consola** de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
11. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones.
- Para Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - Para Image (Imagen), elija `aws/codebuild/amazonlinux2-x86_64-standard:3.0`.
12. Elija Create build project (Crear proyecto de compilación).
13. Seleccione Start build.

14. En Build configuration (Configuración de la compilación), acepte los valores predeterminados y elija Start build (Comenzar compilación).
15. Una vez finalizada la compilación, consulte la salida de compilación en la pestaña Build logs (Registros de compilación). Debería ver un resultado similar a este. Muestra la salida de los tiempos de ejecución de Go y Node.js. También muestra la salida de las aplicaciones de Go y Node.js.

```
[Container] Date Time Processing environment variables
[Container] Date Time Selecting 'golang' runtime version '1.13' based on manual
  selections...
[Container] Date Time Selecting 'nodejs' runtime version '12' based on manual
  selections...
[Container] Date Time Running command echo "Installing Go version 1.13 ..."
Installing Go version 1.13 ...

[Container] Date Time Running command echo "Installing Node.js version 12 ..."
Installing Node.js version 12 ...

[Container] Date Time Running command n $NODE_12_VERSION
  installed : v12.20.1 (with npm 6.14.10)

[Container] Date Time Moving to directory /codebuild/output/src819694850/src
[Container] Date Time Registering with agent
[Container] Date Time Phases found in YAML: 2
[Container] Date Time  INSTALL: 0 commands
[Container] Date Time  BUILD: 1 commands
[Container] Date Time Phase complete: DOWNLOAD_SOURCE State: SUCCEEDED
[Container] Date Time Phase context status code:  Message:
[Container] Date Time Entering phase INSTALL
[Container] Date Time Phase complete: INSTALL State: SUCCEEDED
[Container] Date Time Phase context status code:  Message:
[Container] Date Time Entering phase PRE_BUILD
[Container] Date Time Phase complete: PRE_BUILD State: SUCCEEDED
[Container] Date Time Phase context status code:  Message:
[Container] Date Time Entering phase BUILD
[Container] Date Time Running command echo Building the Go code...
Building the Go code...

[Container] Date Time Running command cd $CODEBUILD_SRC_DIR/golang-app

[Container] Date Time Running command go build hello.go

[Container] Date Time Running command echo Building the Node code...
Building the Node code...

[Container] Date Time Running command cd $CODEBUILD_SRC_DIR/nodejs-app

[Container] Date Time Running command npm run test

> mycompany-app@1.0.0 test /codebuild/output/src924084119/src/nodejs-app
> echo "run some tests here"

run some tests here
```

Ejemplo de versión de origen con AWS CodeBuild

En este ejemplo se muestra cómo especificar una versión de su origen con un formato que no sea un ID de confirmación (también conocido como SHA de confirmación). Puede especificar la versión del origen de las siguientes formas:

- En el caso de un proveedor de origen de Amazon S3, utilice el ID de versión del objeto que representa el archivo ZIP de entrada de compilación.

- En CodeCommit, Bitbucket, GitHub y GitHub Enterprise Server, utilice una de las siguientes opciones:
 - Solicitud de extracción como referencia de solicitud de extracción (por ejemplo, `refs/pull/1/head`).
 - Ramificación como nombre de ramificación.
 - ID de confirmación.
 - Etiqueta.
- Referencia e ID de confirmación. La referencia puede ser uno de los siguientes elementos:
 - Una etiqueta (por ejemplo, `refs/tags/mytagv1.0^{full-commit-SHA}`).
 - Una ramificación (por ejemplo, `refs/heads/mydevbranch^{full-commit-SHA}`).
 - Una solicitud de extracción (por ejemplo, `refs/pull/1/head^{full-commit-SHA}`).

Note

Puede especificar la versión de un origen de solicitud de extracción solo si el repositorio es GitHub o GitHub Enterprise Server.

Si utiliza una referencia y un ID de confirmación para especificar una versión, la fase `DOWNLOAD_SOURCE` de la compilación será más rápida que si proporciona solo la versión. Esto se debe a que al añadir una referencia, CodeBuild no necesita descargar todo el repositorio para encontrar la confirmación.

- Puede especificar una versión de origen con solo un ID de confirmación, como por ejemplo `12345678901234567890123467890123456789`. Si lo hace, CodeBuild debe descargar todo el repositorio para encontrar la versión.
- Puede especificar una versión de origen con una referencia y un ID de confirmación en este formato: `refs/heads/branchname^{full-commit-SHA}` (por ejemplo, `refs/heads/main^{12345678901234567890123467890123456789}`). Si lo hace, CodeBuild solo descarga la ramificación especificada para encontrar la versión.

Note

Para acelerar el `DOWNLOAD_SOURCE` fase de tu compilación, también puedes configurar la profundidad de clon de Git a un número bajo. CodeBuild descarga menos versiones de tu repositorio.

Para especificar una versión de repositorio de GitHub con un ID de confirmación, realice el siguiente procedimiento:

1. Abra el icono AWS CodeBuild Consola de en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y [Ejecutar una compilación \(consola\)](#) (p. 282). Deje todas las opciones con sus valores predeterminados, excepto las siguientes opciones:
 - En Source (Código fuente):
 - En Source provider (Proveedor de código fuente), elija GitHub. Si no está conectado a GitHub, siga las instrucciones para conectarse.
 - En Repository (Repositorio), elija Public repository (Repositorio público).
 - En URL de repositorio, especifique `https://github.com/aws/aws-sdk-ruby.git`
 - En Environment (Entorno):
 - En Environment image (Imagen de entorno), elija Managed image (Imagen administrada).
 - En Operating system (Sistema operativo), elija Amazon Linux 2.
 - En Runtime(s) (Tiempo de ejecución), elija Standard (Estándar).
 - Para Image (Imagen), elija `aws/codebuild/amazonlinux2-x86_64-standard:3.0`.

3. En Build specifications (Especificaciones de compilación), elija Insert build commands (Insertar comandos de compilación) y, a continuación, elija Switch to editor (Cambiar a editor).
4. En Build commands (Comandos de compilación), reemplace el texto del marcador de posición por lo siguiente:

```
version: 0.2

phases:
  install:
    runtime-versions:
      ruby: 2.6
  build:
    commands:
      - echo $CODEBUILD_RESOLVED_SOURCE_VERSION
```

La sección `runtime-versions` es obligatoria si utiliza la imagen estándar de Ubuntu 2.0. Aquí, se especifica el tiempo de ejecución de la versión de Ruby 2.6, pero puede usar cualquier tiempo de ejecución. El comando `echo` muestra la versión del código de origen almacenado en la variable de entorno de `CODEBUILD_RESOLVED_SOURCE_VERSION`.

5. En Build configuration (Configuración de la compilación), acepte los valores predeterminados y elija Start build (Comenzar compilación).
6. En Versión de origen, especifique **046e8b67481d53bdc86c3f6affdd5d1afae6d369**. Este es el SHA de una confirmación en el repositorio <https://github.com/aws/aws-sdk-ruby.git>.
7. Seleccione Start build.
8. Cuando se complete la compilación, debería ver lo siguiente:

- En la pestaña Registros de compilación, la versión del origen de proyecto que se ha utilizado. A continuación se muestra un ejemplo.

```
[Container] Date Time Running command echo $CODEBUILD_RESOLVED_SOURCE_VERSION
046e8b67481d53bdc86c3f6affdd5d1afae6d369

[Container] Date Time Phase complete: BUILD State: SUCCEEDED
```

- En la pestaña Variables de entorno, Versión de origen resuelta coincide con el ID de confirmación utilizado para crear la compilación.
- En la pestaña Detalles de fase, la duración de la fase `DOWNLOAD_SOURCE`.

Estos pasos muestran cómo crear una compilación con la misma versión del origen. Esta vez, la versión del origen se especifica mediante una referencia con el ID de confirmación.

Para especificar una versión del repositorio GitHub versión con un ID de confirmación y una referencia, realice el siguiente procedimiento:

1. En el panel de navegación izquierdo, elija Proyectos de compilación y, a continuación, elija el proyecto que creó anteriormente.
 2. Seleccione Start build.
 3. En Versión de origen, especifique **`refs/heads/main^{046e8b67481d53bdc86c3f6affdd5d1afae6d369}`**. Es el mismo ID de confirmación y una referencia a una ramificación en el formato **`refs/heads/branchname^{full-commit-SHA}`**.
 4. Seleccione Start build.
 5. Cuando se complete la compilación, debería ver lo siguiente:
- En la pestaña Registros de compilación, la versión del origen de proyecto que se ha utilizado. A continuación se muestra un ejemplo.

```
[Container] Date Time Running command echo $CODEBUILD_RESOLVED_SOURCE_VERSION
046e8b67481d53bdc86c3f6affdd5d1afae6d369

[Container] Date Time Phase complete: BUILD State: SUCCEEDED
```

- En la pestaña Variables de entorno, Versión de origen resuelta coincide con el ID de confirmación utilizado para crear la compilación.
- En la pestaña Detalles de fase, la duración de la fase `DOWNLOAD_SOURCE` debe ser inferior a la duración cuando utilizó solo el ID de confirmación para especificar la versión del origen.

Ejemplo de un registro privado conAWS Secrets Managerejemplo de CodeBuild

En este ejemplo, se explica cómo se utiliza una imagen de Docker almacenada en un registro privado como entorno de ejecución de AWS CodeBuild. Las credenciales del registro privado están almacenadas en AWS Secrets Manager. Cualquier registro privado funciona con CodeBuild. En esta muestra se utiliza Docker Hub.

Requisitos del ejemplo de un registro privado

Para poder utilizar un registro privado con AWS CodeBuild, debe tener lo siguiente:

- Un secreto de Secrets Manager que almacena sus credenciales de Docker Hub. Las credenciales se utilizan para obtener acceso al repositorio privado.
- Un repositorio privado o una cuenta.
- Una política de IAM del rol de servicio de CodeBuild que conceda acceso al secreto de Secrets Manager.

Siga estos pasos para crear los recursos y, a continuación, cree un proyecto de compilación de CodeBuild utilizando las imágenes de Docker almacenadas en su registro privado.

Cree un proyecto de CodeBuild con un registro privado

1. Para obtener información sobre cómo crear un repositorio privado gratuito, consulte [Repositorios en Docker Hub](#). También puede ejecutar los siguientes comandos en un terminal para extraer una imagen, obtener su ID y enviarla a un nuevo repositorio.

```
docker pull amazonlinux
docker images amazonlinux --format {{.ID}}
docker tag image-id your-username/repository-name:tag
docker login
docker push your-username/repository-name
```

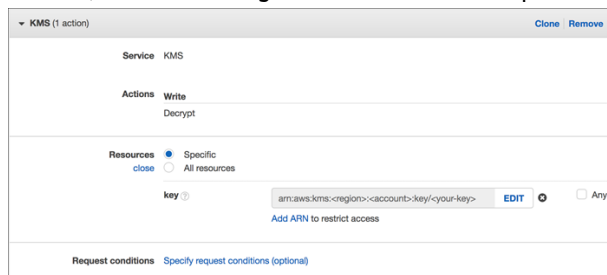
2. Siga los pasos de [Creación de un secreto básico](#) en la Guía del usuario de AWS Secrets Manager. En el paso 3, en Select secret type (Seleccionar tipo de secreto), haga lo siguiente:
 - a. Elija Other type of secrets (Otro tipo de secretos).
 - b. En Secret key/value (Clave/valor de secreto), crear un par de clave-valor para el nombre de usuario de Docker Hub y otro par de clave-valor para la contraseña de Docker Hub.
 - c. En Secret Name (Nombre del secreto), escriba un nombre, por ejemplo, **dockerhub**. Puede escribir una descripción opcional que le ayude a recordar que es un secreto para Docker Hub.
 - d. Deje seleccionada la opción Disable automatic rotation (Deshabilitar rotación automática) porque las claves corresponden a sus credenciales de Docker Hub.

- e. Elija Store secret (Almacenar secreto).
- f. Cuando revise la configuración, anote el ARN para utilizarlo más adelante en esta muestra.

Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#)

3. Al crear un AWS CodeBuild en la consola de, CodeBuild asocia el permiso necesario necesario para usted. Si utiliza una clave de AWS KMS distinta de `DefaultEncryptionKey`, debe añadirla al rol de servicio. Para obtener más información, consulte [Modificación de un rol \(consola\)](#) en la IAM User Guide.

Para que su rol de servicio funcione con Secrets Manager, debe tener, como mínimo, `secretsmanager:GetSecretValue` permiso.



4. Para usar la consola para crear un proyecto con un entorno almacenado en un registro privado, haga lo siguiente mientras crea un proyecto. Para obtener información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).

Note

Si el registro privado está en su VPC, debe tener acceso público a Internet. CodeBuild no puede extraer una imagen de una dirección IP privada de una VPC.

- a. En Environment (Entorno), elija Custom image (Imagen personalizada).
- b. En Environment type (Tipo de entorno), elija Linux o Windows.
- c. En Custom image type (Tipo de imagen personalizada), elija Other location (Otra ubicación).
- d. En Otra ubicación, escriba la ubicación de la imagen y el ARN o nombre de sus credenciales de Secrets Manager.

Note

Si las credenciales no existen en la región actual, debe usar el ARN. No puede utilizar el nombre de la credencial si las credenciales existen en otra región.

Ejemplo de varios orígenes de entrada y varios artefactos de salida

Puede crear un proyecto de compilación de AWS CodeBuild con varios orígenes de entrada y varios conjuntos de artefactos de salida. En este ejemplo se muestra cómo configurar un proyecto de compilación que:

- Utiliza varios orígenes y repositorios de distintos tipos.
- Publica los artefactos de compilación en varios buckets de S3 en una sola compilación.

En este ejemplo se crea un proyecto de compilación que se utiliza para ejecutar una compilación. El ejemplo usa el archivo `buildspec` del proyecto de compilación para mostrar cómo incorporar más de un origen y crear más de un conjunto de artefactos.

1. Cargue los orígenes en uno o varios buckets de S3 o en uno o varios repositorios de, CodeCommit, GitHub, GitHub Enterprise Server o Bitbucket.
2. Elija cuál es el origen principal. Este será el origen en el que CodeBuild busca el archivo buildspec y lo ejecuta.
3. Cree un proyecto de compilación. Para obtener más información, consulte [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#).
4. Siga las instrucciones en [Ejecutar AWS CodeBuild directamente \(p. 406\)](#) para crear el proyecto de compilación, ejecutar la compilación y obtener información sobre esta.
5. Si usa la AWS CLI para crear el proyecto de compilación, es posible que el resultado con formato JSON del comando `create-project` tenga un aspecto similar al siguiente:

```
{
  "name": "sample-project",
  "source": {
    "type": "S3",
    "location": "bucket/sample.zip"
  },
  "secondarySources": [
    {
      "type": "CODECOMMIT",
      "location": "https://git-codecommit.us-west-2.amazonaws.com/v1/repos/repo"
      "sourceIdentifier": "source1"
    },
    {
      "type": "GITHUB",
      "location": "https://github.com/aws-labs/aws-codebuild-jenkins-plugin"
      "sourceIdentifier": "source2"
    }
  ],
  "secondaryArtifacts": [
    {
      "type": "S3",
      "location": "output-bucket",
      "artifactIdentifier": "artifact1"
    },
    {
      "type": "S3",
      "location": "other-output-bucket",
      "artifactIdentifier": "artifact2"
    }
  ],
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:4.0",
    "computeType": "BUILD_GENERAL1_SMALL"
  },
  "serviceRole": "arn:aws:iam::account-ID:role/role-name",
  "encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

El origen principal se define en el atributo `source`. El resto de orígenes se denominan secundarios y aparecen en `secondarySources`. Todos los orígenes secundarios se instalan en su propio directorio. Este directorio se almacena en la variable de entorno integrada `CODEBUILD_SRC_DIR_`*sourceIdentifier*. Para obtener más información, consulte [Variables de entorno en los entornos de compilación \(p. 172\)](#).

El atributo `secondaryArtifacts` contiene una lista de definiciones de artefacto. Estos artefactos usan el bloque `secondary-artifacts` del archivo buildspec anidado en el bloque `artifacts`.

Los artefactos secundarios del archivo buildspec tienen la misma estructura que el resto de artefactos y se separan mediante el identificador correspondiente.

Note

En el navegador [API de CodeBuild](#), el `artifactIdentifier` en un artefacto secundario es un atributo obligatorio en `CreateProject` y `UpdateProject`. Debe utilizarse para hacer referencia a un artefacto secundario.

Si se utiliza la entrada con formato JSON anterior, el archivo buildspec del proyecto puede tener este aspecto:

```
version: 0.2

phases:
  install:
    runtime-versions:
      java: openjdk11
  build:
    commands:
      - cd $CODEBUILD_SRC_DIR_source1
      - touch file1
      - cd $CODEBUILD_SRC_DIR_source2
      - touch file2

artifacts:
  files:
    - '**.*'
  secondary-artifacts:
    artifact1:
      base-directory: $CODEBUILD_SRC_DIR_source1
      files:
        - file1
    artifact2:
      base-directory: $CODEBUILD_SRC_DIR_source2
      files:
        - file2
```

Puede reemplazar la versión del origen principal mediante la API con el atributo `sourceVersion` en `StartBuild`. Para reemplazar una o varias versiones de origen secundario, utilice el atributo `secondarySourceVersionOverride`.

Es posible que la entrada con formato JSON del comando `start-build` de la AWS CLI tenga este aspecto:

```
{
  "projectName": "sample-project",
  "secondarySourcesVersionOverride": [
    {
      "sourceIdentifier": "source1",
      "sourceVersion": "codecommit-branch"
    },
    {
      "sourceIdentifier": "source2",
      "sourceVersion": "github-branch"
    }
  ]
}
```

Ejemplo de proyecto sin un origen

Puede configurar un proyecto de CodeBuild eligiendo el tipo de origen `NO_SOURCE` al configurar el origen de. Cuando el tipo de origen es `NO_SOURCE`, no puede especificar un archivo de especificación de compilación porque su proyecto no tiene un origen. En su lugar, debe especificar una cadena de especificación de compilación con formato YAML en el atributo `buildspec` de la entrada con formato JSON al comando de la CLI `create-project`. Podría ser como el siguiente:

```
{
  "name": "project-name",
  "source": {
    "type": "NO_SOURCE",
    "buildspec": "version: 0.2\n\nphases:\n  build:\n    commands:\n      - command"
  },
  "environment": {
    "type": "LINUX_CONTAINER",
    "image": "aws/codebuild/standard:4.0",
    "computeType": "BUILD_GENERAL1_SMALL",
  },
  "serviceRole": "arn:aws:iam::account-ID:role/role-name",
  "encryptionKey": "arn:aws:kms:region-ID:account-ID:key/key-ID"
}
```

Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

Para obtener información sobre cómo crear una canalización que utilice varias entradas de origen en CodeBuild para crear varios artefactos de salida, consulte [AWS CodePipeline Ejemplo de integración de integración con CodeBuild y varios orígenes de entrada y varios artefactos](#) (p. 62).

Ejemplo para utilizar el control de versiones semánticas para asignar nombre a artefactos de compilación

El ejemplo contiene archivos `buildspec` que muestran cómo especificar un nombre de artefacto que se crea en el momento de la compilación. El nombre que se especifica en un archivo `buildspec` puede incorporar comandos de shell y variables de entorno para hacerlo único. Dicho nombre anulará otro nombre que escriba en la consola al crear el proyecto.

Si compila varias veces, el uso de un nombre de artefacto especificado en el archivo `buildspec` le ayuda a garantizar que los nombres de los archivos de artefactos de salida son únicos. Por ejemplo, puede utilizar una marca de fecha y hora para que se inserte en el nombre del artefacto en el momento de la compilación.

Si desea reemplazar el nombre de artefacto que ha escrito en la consola por un nombre del archivo `buildspec`, haga lo siguiente:

1. Establezca que el proyecto de compilación reemplace el nombre del artefacto por un nombre incluido en el archivo `buildspec`.
 - Si utiliza la consola para crear su proyecto de compilación, seleccione **Enable semantic versioning** (Habilitar control semántico de versiones). Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).
 - Si utiliza la AWS CLI, establezca `overrideArtifactName` en `true` en el archivo con formato JSON que se transfiere a `create-project`. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

- Si utiliza la API de AWS CodeBuild, establezca la marca `overrideArtifactName` en el objeto `ProjectArtifacts` cuando cree o actualice un proyecto o cuando inicie una compilación.
2. Especifique un nombre en el archivo `buildspec`. Utilice los siguientes archivos `buildspec` de ejemplo como guía.

Este ejemplo de Linux muestra cómo especificar un nombre de artefacto que incluye la fecha de creación de la compilación:

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
    - '**/*'
  name: myname-$(date +%Y-%m-%d)
```

Este ejemplo de Linux muestra cómo especificar un nombre de artefacto que utiliza una variable de entorno de CodeBuild. Para obtener más información, consulte [Variables de entorno en los entornos de compilación](#) (p. 172).

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
    - '**/*'
  name: myname-$AWS_REGION
```

Este ejemplo de Windows muestra cómo especificar un nombre de artefacto que incluye la fecha y la hora de creación de la compilación:

```
version: 0.2
env:
  variables:
    TEST_ENV_VARIABLE: myArtifactName
phases:
  build:
    commands:
      - cd samples/helloworld
      - dotnet restore
      - dotnet run
artifacts:
  files:
    - '**/*'
  name: $Env:TEST_ENV_VARIABLE-$(Get-Date -UFormat "%Y%m%d-%H%M%S")
```

Este ejemplo de Windows muestra cómo especificar un nombre de artefacto que utiliza una variable declarada en el archivo `buildspec` y una variable de entorno de CodeBuild. Para obtener más información, consulte [Variables de entorno en los entornos de compilación](#) (p. 172).

```
version: 0.2
env:
  variables:
    TEST_ENV_VARIABLE: myArtifactName
```

```
phases:
  build:
    commands:
      - cd samples/helloworld
      - dotnet restore
      - dotnet run
artifacts:
  files:
    - '**/*'
  name: $Env:TEST_ENV_VARIABLE-$Env:AWS_REGION
```

Para obtener más información, consulte [Referencia de la especificación de compilación para CodeBuild \(p. 136\)](#).

Planificar una compilación en AWS CodeBuild

Antes de usar AWS CodeBuild, debe responder estas preguntas:

1. ¿Dónde se encuentra el código fuente? CodeBuild admite actualmente compilaciones de los siguientes proveedores de repositorios de código fuente. El código fuente debe contener un archivo de especificación de compilación (buildspec). ASpeces una colección de comandos de compilación y opciones de configuración relacionadas, en formato YAML, que CodeBuild utiliza para ejecutar una compilación. Puede declarar un archivo buildspec en una definición del proyecto de compilación.

Proveedor de repositorio	Obligatorio	Documentación
CodeCommit	Nombre del repositorio. (Opcional) ID de confirmación asociado con el código fuente.	Consulte estos temas en la Guía del usuario de AWS CodeCommit: Crear un repositorio de CodeCommit Crear una confirmación en CodeCommit
Amazon S3	Nombre del bucket de entrada. Nombre de objeto correspondiente al archivo ZIP de entrada de la compilación que contiene el código fuente. (Opcional) ID de versión asociado al archivo ZIP de entrada de la compilación.	Consulte estos temas en Guía de introducción a Amazon S3: Crear un bucket Añadir un objeto a un bucket.
GitHub	Nombre del repositorio. (Opcional) ID de confirmación asociado con el código fuente.	Consulte este tema en el sitio web de ayuda de GitHub: Crear un repositorio
Bitbucket	Nombre del repositorio. (Opcional) ID de confirmación	Consulte este tema en el sitio web de documentación de Bitbucket Cloud: Crear un repositorio de

Proveedor de repositorio	Obligatorio	Documentación
	asociado con el código fuente.	

2. ¿Qué comandos de compilación necesita ejecutar y en qué orden? De forma predeterminada, CodeBuild descarga la entrada de la compilación del proveedor que especifique y carga la salida de la compilación en el bucket que especifique. Debe usar la especificación de compilación para indicar cómo convertir la entrada de compilación descargada en la salida de compilación prevista. Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

3. ¿Qué tiempos de ejecución y herramientas necesita para ejecutar la compilación? Por ejemplo, ¿va a compilar para Java, Ruby, Python o Node.js? ¿Necesita la compilación Maven o Ant, o un compilador para Java, Ruby o Python? ¿Necesita la compilación Git, la AWS CLI u otras herramientas?

CodeBuild ejecuta las compilaciones en entornos de compilación que usan imágenes de Docker. Estas imágenes de Docker deberán almacenarse en un tipo de repositorio compatible con CodeBuild. Entre estos se incluyen el repositorio de imágenes de Docker de CodeBuild, Docker Hub y Amazon Elastic Container Registry (Amazon ECR). Para obtener más información sobre el repositorio de imágenes de Docker de, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).

4. ¿Necesita AWS? ¿Necesita recursos que no proporciona automáticamente CodeBuild? En tal caso, ¿qué políticas de seguridad necesitan estos recursos? Por ejemplo, es posible que tenga que modificar la función de servicio de CodeBuild para permitir que CodeBuild funcione con esos recursos.

5. ¿Desea que CodeBuild funcione con su VPC? En tal caso, necesita el ID de la VPC, los identificadores de subred y los identificadores de grupo de seguridad de la configuración de su VPC. Para obtener más información, consulte [Usar AWS CodeBuild Amazon Virtual Private Cloud \(p. 180\)](#).

Después de responder a estas preguntas, debe disponer de la configuración y los recursos que necesita para ejecutar la compilación correctamente. Para ejecutar la compilación, puede:

- Usar la consola de AWS CodeBuild, AWS CLI o los SDK de AWS. Para obtener más información, consulte [Ejecutar CodeBuild directamente \(p. 406\)](#).
- Crear o identificar una canalización en AWS CodePipeline, a continuación, añada una acción de compilación o prueba que indique a CodeBuild que pruebe su código, ejecute la compilación o ambas cosas. Para obtener más información, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#).

Referencia de la especificación de compilación para CodeBuild

En este tema, se proporciona información de referencia importante sobre los archivos de especificación de compilación (buildspec). A BuildSpec esta es una colección de comandos de compilación y opciones relacionadas, en formato YAML, que CodeBuild utiliza para ejecutar una compilación. Puede incluir una especificación de compilación como parte del código fuente o puede incluir una especificación de compilación cuando cree un proyecto de compilación. Para obtener información sobre cómo funciona una especificación de compilación, consulte [Funcionamiento de CodeBuild \(p. 3\)](#).

Temas

- [Nombre de archivo y ubicación de almacenamiento de buildspec \(p. 137\)](#)
- [Sintaxis de buildspec \(p. 137\)](#)
- [Ejemplo de un archivo buildspec \(p. 152\)](#)
- [Versiones de buildspec \(p. 154\)](#)
- [Referencia de la especificación de compilación por Batch \(p. 154\)](#)

Nombre de archivo y ubicación de almacenamiento de buildspec

Si incluye una especificación de compilación como parte del código fuente, de forma predeterminada, el archivo de especificación de compilación debe llamarse `buildspec.yml` y debe encontrarse en la raíz del directorio de código fuente.

Puede invalidar el nombre y la ubicación predeterminados del archivo de especificación de compilación. Por ejemplo, puede hacer lo siguiente:

- Usar un archivo de especificación de compilación diferente para las distintas compilaciones del mismo repositorio, como `buildspec_debug.yml` y `buildspec_release.yml`.
- Almacenar un archivo de especificación de compilación en otro lugar que no sea la raíz de su directorio de origen, como en `config/buildspec.yml` o en un bucket de S3. El bucket de S3 debe estar en la misma AWS región que el proyecto de compilación. Especifique el archivo `buildspec` utilizando su ARN (por ejemplo, `arn:aws:s3:::my-codebuild-sample2/buildspec.yml`).

Solo puede especificar una especificación de compilación para un proyecto de compilación, independientemente del nombre del archivo de especificación de compilación.

Para invalidar el nombre y/o la ubicación del archivo de especificación de compilación, realice alguna de las siguientes operaciones:

- Ejecute el comando `create-project` o `update-project` de la AWS CLI, estableciendo el valor `buildspec` en la ruta del archivo de especificación de compilación alternativo relativo al valor de la variable de entorno integrada `CODEBUILD_SRC_DIR`. También puede hacer lo mismo con la operación `create-project` en los SDK de AWS. Para obtener más información, consulte [Creación de un proyecto de compilación \(p. 197\)](#) o [Cambiar la configuración de un proyecto de compilación \(p. 254\)](#).
- Ejecute el comando `start-build` de la AWS CLI, estableciendo el valor `buildspecOverride` en la ruta del archivo de especificación de compilación alternativo relativo al valor de la variable de entorno integrada `CODEBUILD_SRC_DIR`. También puede hacer lo mismo con la operación `start-build` en los SDK de AWS. Para obtener más información, consulte [Ejecutar una compilación \(p. 281\)](#).
- En una plantilla de AWS CloudFormation, establezca la propiedad `BuildSpec` de `Source` en un recurso de tipo `AWS::CodeBuild::Project` en la ruta del archivo de especificación de compilación alternativo relativo al valor de la variable de entorno integrada `CODEBUILD_SRC_DIR`. Para obtener más información, consulte la propiedad `BuildSpec` en [AWS CodeBuildOrigen de proyecto](#) en la [AWS CloudFormation Guía del usuario](#) de.

Sintaxis de buildspec

Los archivos de especificación de compilación deben expresarse en formato [YAML](#).

Si un comando contiene un carácter, o una cadena de caracteres, que no es compatible con YAML, debe encerrar el comando entre comillas (""). El siguiente comando se incluye entre comillas porque no se permiten dos puntos (:) seguidos de un espacio en YAML. La comilla en el comando utiliza la secuencia de escape (`\`).

```
"export PACKAGE_NAME=$(cat package.json | grep name | head -1 | awk -F: '{ print $2 }' | sed 's/[\",,]/\\&/'")"
```

La especificación de compilación tiene la siguiente sintaxis:

```
version (p. 139): 0.2
```



```
run-as (p. 139): Linux-user-name

env (p. 139):
  shell (p. 140): shell-tag
  variables (p. 140):
    key: "value"
    key: "value"
  parameter-store (p. 141):
    key: "value"
    key: "value"
  exported-variables (p. 142):
    - variable
    - variable
  secrets-manager (p. 141):
    key: secret-id:json-key:version-stage:version-id
  git-credential-helper (p. 143): no | yes

proxy (p. 143):
  upload-artifacts (p. 143): no | yes
  logs (p. 143): no | yes

batch (p. 154):
  fast-fail: false | true
  # build-list:
  # build-matrix:
  # build-graph:

phases (p. 143):
  install (p. 144):
    run-as (p. 143): Linux-user-name
    on-failure (p. 143): ABORT | CONTINUE
    runtime-versions (p. 144):
      runtime: version
      runtime: version
    commands (p. 145):
      - command
      - command
    finally (p. 144):
      - command
      - command
  pre_build (p. 145):
    run-as (p. 143): Linux-user-name
    on-failure (p. 143): ABORT | CONTINUE
    commands (p. 145):
      - command
      - command
    finally (p. 144):
      - command
      - command
  build (p. 145):
    run-as (p. 143): Linux-user-name
    on-failure (p. 143): ABORT | CONTINUE
    commands (p. 145):
      - command
      - command
    finally (p. 144):
      - command
      - command
  post_build (p. 145):
    run-as (p. 143): Linux-user-name
    on-failure (p. 143): ABORT | CONTINUE
    commands (p. 145):
      - command
      - command
    finally (p. 144):
      - command
```

```
- command
reports (p. 145):
  report-group-name-or-arn:
    files (p. 146):
      - location
      - location
    base-directory (p. 147): location
    discard-paths (p. 147): no | yes
    file-format (p. 146): report-format
artifacts (p. 147):
  files (p. 147):
    - location
    - location
  name (p. 147): artifact-name
  discard-paths (p. 149): no | yes
  base-directory (p. 149): location
  exclude-paths (p. 150): excluded paths
  enable-symlinks (p. 150): no | yes
  s3-prefix (p. 150): prefix
  secondary-artifacts (p. 150):
    artifactIdentifier:
      files (p. 147):
        - location
        - location
      name (p. 147): secondary-artifact-name
      discard-paths (p. 149): no | yes
      base-directory (p. 149): location
    artifactIdentifier:
      files (p. 147):
        - location
        - location
      discard-paths (p. 149): no | yes
      base-directory (p. 149): location
cache (p. 151):
  paths (p. 151):
    - path
    - path
```

La especificación de compilación contiene lo siguiente:

version

Mapeo obligatorio. Representa la versión de la especificación de compilación. Le recomendamos que utilice 0.2.

Note

Aunque la versión 0.1 sigue siendo compatible, le recomendamos que utilice la versión 0.2 siempre que sea posible. Para obtener más información, consulte [Versiones de buildspec \(p. 154\)](#).

run-as

Secuencia opcional. Disponible solo para usuarios de Linux. Especifica un usuario de Linux que ejecuta comandos en este archivo buildspec. `run-as` otorga al usuario especificado permisos de lectura y ejecución. Cuando se especifica `run-as` en la parte superior del archivo buildspec, se aplica globalmente a todos los comandos. Si no desea especificar un usuario para todos los comandos de archivo buildspec, puede especificar uno para comandos en una fase utilizando `run-as` en uno de los bloques `phases`. Si no se especifica `run-as`, todos los comandos se ejecutan como usuario raíz.

env

Secuencia opcional. Representa información para una o más variables de entorno personalizadas.

Note

Para proteger la información confidencial, lo siguiente está oculto en los registros de CodeBuild:

- ID de clave de acceso de AWS. Para obtener más información, consulte [Administración de las claves de acceso de los usuarios de IAM](#) en la [AWS Identity and Access Management Guía del usuario](#) de.
- Cadenas especificadas mediante el almacén de parámetros. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial de la consola de almacén de parámetros de Systems Manager](#) en la [Guía del usuario de Amazon EC2 Systems Manager](#).
- Cadenas especificadas mediante AWS Secrets Manager. Para obtener más información, consulte [Administración de claves \(p. 353\)](#).

env/shell

Secuencia opcional. Especifica el shell admitido para sistemas operativos Linux o Windows.

Para los sistemas operativos Linux, las etiquetas de shell compatibles son:

- `bash`
- `/bin/sh`

Para los sistemas operativos Windows, las etiquetas de shell compatibles son:

- `powershell.exe`
- `cmd.exe`

env/variables

Obligatorio si se especifica `env` y desea definir variables de entorno personalizadas en texto sin formato. Contiene una asignación de pares `clave/valor` escalares, donde cada asignación representa una única variable de entorno personalizada en texto sin formato. La `clave` es el nombre de la variable de entorno personalizada, mientras que el `valor` es el valor de la variable.

Important

Se desaconseja el almacenamiento de valores confidenciales, especialmente los identificadores de clave de acceso y las claves de acceso secretas de AWS en variables de entorno. Las variables de entorno se pueden mostrar en texto sin formato con herramientas como la consola de CodeBuild y la AWS CLI. Para valores confidenciales, le recomendamos utilizar el mapeo `parameter-store` o `secrets-manager` en su lugar, tal y como se describe más adelante en esta sección.

Las variables de entorno que defina reemplazan las variables de entorno existentes. Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y establece una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`. Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y establece una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No establezca variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad. Al crear una compilación, puede añadir o anular las variables de entorno. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#).

- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad. Al crear o editar un proyecto, puede añadir las variables de entorno en el nivel del proyecto. Para obtener más información, consulte [Creación de un proyecto de compilación en AWS CodeBuild](#) (p. 197) y [Cambiar la configuración de un proyecto de compilación en AWS CodeBuild](#) (p. 254).
- El valor en la declaración de especificación de compilación es el que menos prioridad tiene.

env/parameter-store

Obligatorio `env`Y desea recuperar variables de entorno personalizadas almacenadas en el almacén de parámetros Amazon EC2 Systems Manager. Contiene una asignación de `clave/valor`. Los escalares, en los que cada asignación representa una única variable de entorno personalizada almacenada en el almacén de parámetros Amazon EC2 Systems Manager. `clave` Este es el nombre que utiliza más adelante en los comandos de compilación para referirse a esta variable de entorno personalizada, y `valor` clave es el nombre de la variable de entorno personalizada almacenada en el almacén de parámetros Amazon EC2 Systems Manager. Para almacenar valores confidenciales, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial: Creación y prueba de parámetros de cadena \(consola\)](#) en la Guía del usuario de Amazon EC2 Systems Manager.

Important

Para permitir que CodeBuild recupere variables de entorno personalizadas almacenadas en el almacén de parámetros de Amazon EC2 Systems Manager, debe añadir el comando `ssm:GetParameters` a su función de servicio CodeBuild. Para obtener más información, consulte [Cree un rol de servicio de CodeBuild](#) (p. 396).

Todas las variables de entorno que recupere del almacén de parámetros Amazon EC2 Systems Manager reemplazan las variables de entorno existentes. Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y recupera una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`. Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y recupera una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No almacene variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad. Al crear una compilación, puede añadir o anular las variables de entorno. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild](#) (p. 281).
- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad. Al crear o editar un proyecto, puede añadir las variables de entorno en el nivel del proyecto. Para obtener más información, consulte [Creación de un proyecto de compilación en AWS CodeBuild](#) (p. 197) y [Cambiar la configuración de un proyecto de compilación en AWS CodeBuild](#) (p. 254).
- El valor en la declaración de especificación de compilación es el que menos prioridad tiene.

env/secrets-manager

: obligatorio si desea recuperar variables de entorno personalizadas almacenadas en AWS Secrets Manager. Especifique un Administrador de Secrets Manager `reference-key` Utilizando el siguiente patrón:

```
<key>: <secret-id>: <json-key>: <version-stage> | <version-id>  
<key>
```

(Requerido) El nombre de la variable de entorno local. Utilice este nombre para acceder a la variable durante la compilación.

<secret-id>

(Obligatorio) El nombre o nombre de recurso de Amazon (ARN) que sirve como identificador único del secreto. Para acceder a un secreto en su cuenta de AWS, simplemente especifique el nombre del secreto. Para acceder a un secreto en una cuenta de AWS diferente, especifique el ARN del secreto.

<json-key>

(Opcional) Especifica el nombre de la clave del par clave-valor del Secrets Manager cuyo valor desea recuperar. Si no se especifica un error de `json-key`, CodeBuild recupera todo el texto secreto.

<version-stage>

(Opcional) Especifica la versión del secreto que desea recuperar mediante la etiqueta de fase que se asocia a la versión. Las etiquetas de fase se utilizan para realizar un seguimiento de las diferentes versiones durante el proceso de rotación. Si usa `version-stage`, no especifique `version-id`. Si no especifica una fase o un ID de versión, el valor predeterminado será recuperar la versión con el valor de la fase de versión `AWSCURRENT`.

<version-id>

(Opcional) Especifica el identificador único de la versión del secreto que desea utilizar. Si especifica `version-id`, no especifique `version-stage`. Si no especifica una fase o un ID de versión, el valor predeterminado será recuperar la versión con el valor de la fase de versión `AWSCURRENT`.

En el siguiente ejemplo, `TestSecret` El nombre del par clave-valor almacenado en Secrets Manager. La clave de `TestSecretesMY_SECRET_VAR`. Puede acceder a la variable durante la compilación mediante el comando `LOCAL_SECRET_VAR` Nombre de de:

```
env:
  secrets-manager:
    LOCAL_SECRET_VAR: "TestSecret:MY_SECRET_VAR"
```

Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

env/exported-variables

Mapeo opcional. Se utiliza para enumerar las variables de entorno que desea exportar. Especifique el nombre de cada variable en la que desee exportar en una línea independiente `exported-variables`. La variable que desea exportar debe estar disponible en su contenedor durante la compilación. La variable exportada puede ser una variable de entorno.

Las variables de entorno exportadas se utilizan junto con AWS CodePipeline para exportar variables de entorno de la etapa de compilación actual a etapas posteriores de la canalización. Para obtener más información, consulte [Trabajar con variables](#) en la AWS CodePipeline Guía del usuario de.

Durante una compilación, el valor de una variable está disponible a partir de la fase `install`. Se puede actualizar entre el inicio de la fase `install` y el final de la fase `post_build`. Una vez finalizada la fase `post_build`, el valor de las variables exportadas no puede cambiar.

Note

No se pueden exportar los siguientes elementos:

- Secretos del almacén de parámetros de Amazon EC2 Systems Manager especificados en el proyecto de compilación.
- Secretos de Secrets Manager especificados en el proyecto de compilación
- Variables de entorno que empiezan por `AWS_`.

env/git-credential-helper

Mapeo opcional. Se utiliza para indicar si CodeBuild utiliza su auxiliar de credenciales de Git para proporcionar las credenciales de Git. `yes` si se utiliza. De lo contrario, seleccione `no` o sin especificar. Para obtener más información, consulte [gitcredentials](#) en el sitio web de Git.

Note

`git-credential-helper` no es compatible con compilaciones desencadenadas por un webhook para un repositorio de Git público.

proxy

Secuencia opcional. Se utiliza para representar configuraciones si ejecuta la compilación en un servidor de proxy explícito. Para obtener más información, consulte [Ejecución de CodeBuild en un servidor proxy explícito](#) (p. 191).

proxy/upload-artifacts

Mapeo opcional. Establezca en `yes` si desea que la compilación de un servidor de proxy explícito cargue artefactos. El valor predeterminado es `no`.

proxy/logs

Mapeo opcional. Definir en `yes` Para obtener una compilación en un servidor proxy explícito para crear registros de CloudWatch. El valor predeterminado es `no`.

phases

Secuencia obligatoria. Representa los comandos que ejecuta CodeBuild durante cada fase de la compilación.

Note

En la versión de especificación de compilación 0.1, CodeBuild ejecuta cada comando en una instancia distinta del shell predeterminado en el entorno de compilación. Esto significa que cada comando se ejecuta con independencia de los demás. Por lo tanto, de forma predeterminada, no puede ejecutar un comando que se base en el estado de comandos anteriores (por ejemplo, cambiar directorios o configurar variables de entorno). Para solventar esta limitación, le recomendamos utilizar la versión 0.2, que soluciona este problema. Si debe utilizar la versión de especificación de compilación 0.1, se recomiendan los enfoques que se describen en [Shells y comandos de los entornos de compilación](#) (p. 171).

phases/*/run-as

Secuencia opcional. Utilice una fase de compilación para especificar un usuario de Linux que ejecuta sus comandos. Si `run-as` también se especifica globalmente para todos los comandos en la parte superior del archivo buildspec, entonces el usuario de nivel de fase tiene prioridad. Por ejemplo, si `run-as` especifica globalmente `User-1` y para la fase `install` solo una instrucción `run-as` especifica `User-2`, todos los comandos del archivo buildspec se ejecutan como `User-1` excepto los comandos de la fase `install`, que se ejecutan como `User-2`.

phases/*/en caso de error

Secuencia opcional. Especifica la acción que se debe realizar si se produce un error durante la fase. Puede ser uno de los siguientes valores:

- `ABORT`- Aborta la construcción.
- `CONTINUE`Continúe en la siguiente fase.

Si no se especifica esta propiedad, el proceso de error sigue las fases de transición como se muestra en [Transiciones de fases de compilación \(p. 291\)](#).

`phases/*/finally`

Bloque opcional. Los comandos especificados en `unfinally` se ejecutan después de los comandos en `elcommandsbloque`. Los comandos de `unfinally` se ejecutan incluso si un comando en el bloque `commands` falla en bloque. Por ejemplo, si `elcommands` contiene tres comandos y el primero produce un error, CodeBuild omite los dos comandos restantes y ejecuta los comandos del directorio `finally` bloque. La fase se lleva a cabo correctamente cuando todos los comandos de los bloques `commands` y `finally` se ejecutan sin problemas. Si un comando de una fase produce un error, la fase también producirá un error.

Los nombres de las fases de compilación permitidos son:

`phases/install`

Secuencia opcional. Representa los comandos, si hay alguno, que ejecuta CodeBuild durante la instalación. Le recomendamos que utilice la fase `install` únicamente para instalar paquetes en el entorno de compilación. Por ejemplo, puede utilizar esta fase para instalar una plataforma de comprobación de código como Mocha o RSpec.

`phases/install/runtime-versions`

Secuencia opcional. Una versión en tiempo de ejecución es compatible con la imagen estándar de Ubuntu 2.0 o posterior y la imagen estándar de Amazon Linux 2 1.0 o posterior. Si se especifica, al menos debe haber un runtime incluido en esta sección. Especifique un tiempo de ejecución utilizando una versión específica, una versión principal seguida de `.x` para especificar que CodeBuild utiliza esa versión principal con su última versión secundaria, o `latest` para utilizar la versión principal y secundaria más reciente (por ejemplo, `java: openjdk11`, `ruby: 2.6`, `nodejs: 12.x`, o bien `java: latest`). Puede especificar el runtime mediante un número o una variable de entorno. Por ejemplo, si utiliza la versión 2.0 de la imagen estándar de Amazon Linux 2, entonces lo siguiente especifica que se instala la versión 8 de Java, la versión secundaria más reciente de la versión 3 de Python y una versión incluida en una variable de entorno de Ruby. Para obtener más información, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).

```
phases:
  install:
    runtime-versions:
      java: corretto8
      python: 3.x
      ruby: "$MY_RUBY_VAR"
```

Puede especificar uno o más tiempos de ejecución en la sección `runtime-versions` del archivo de especificación de compilación. Si el tiempo de ejecución depende de otro tiempo de ejecución, también puede especificar el tiempo de ejecución dependiente en el archivo de especificación de compilación. Si no especifica ningún motor de ejecución en el archivo `buildspec`, CodeBuild elige los tiempos de ejecución predeterminados que están disponibles en la imagen que utiliza. Si especifica uno o más tiempos de ejecución, CodeBuild utiliza solo esos tiempos de ejecución. Si no se especifica un motor de ejecución dependiente, CodeBuild intenta elegir el motor de ejecución dependiente.

Si dos runtimes especificados están en conflicto, la compilación produce un error. Por ejemplo, `android: 29` y `java: openjdk11` están en conflicto, por lo que si se especifican ambos, la compilación produce un error.

Para obtener más información acerca de los tiempos de ejecución disponibles, consulte [Runtimes de ejecución disponibles \(p. 162\)](#).

Note

Si especifica `runtime-versions` Para utilizar una imagen distinta de Ubuntu Standard Image 2.0 o posterior, o la imagen estándar de Amazon Linux 2 (AL2) 1.0 o posterior, la compilación emite la advertencia `Skipping install of runtimes. Runtime version selection is not supported by this build image.`

phases/install/commands

Secuencia opcional. Contiene una secuencia de valores escalares, en la que cada valor escalar representa un comando que CodeBuild ejecuta durante la instalación. CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

phases/pre_build

Secuencia opcional. Representa los comandos, si hay alguno, que CodeBuild ejecuta antes de la compilación. Por ejemplo, puede utilizar esta fase para iniciar sesión en Amazon ECR o puede instalar dependencias npm.

phases/pre_build/commands

Secuencia obligatoria si se especifica `pre_build`. Contiene una secuencia de valores escalares, en la que cada valor escalar representa un comando que CodeBuild ejecuta antes de la compilación. CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

phases/build

Secuencia opcional. Representa los comandos, si hay alguno, que CodeBuild ejecuta durante la compilación. Por ejemplo, puede utilizar esta fase para ejecutar Mocha, RSpec o sbt.

phases/build/commands

Obligatoria si `build` está especificado. Contiene una secuencia de valores escalares, en la que cada valor escalar representa un comando que CodeBuild ejecuta durante la compilación. CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

phases/post_build

Secuencia opcional. Representa los comandos, si hay alguno, que CodeBuild ejecuta después de la compilación. Por ejemplo, puede utilizar Maven para empaquetar los artefactos de la compilación en un archivo JAR o WAR, o puede insertar una imagen de Docker en Amazon ECR. A continuación, puede enviar una notificación de compilación a través de Amazon SNS.

phases/post_build/commands

Obligatoria si `post_build` está especificado. Contiene una secuencia de valores escalares, en la que cada valor escalar representa un comando que CodeBuild ejecuta después de la compilación. CodeBuild ejecuta cada comando, uno cada vez, en el orden indicado, de principio a fin.

reports

report-group-name-or-arn

Secuencia opcional. Especifica el grupo de informes al que se envían los informes. Un proyecto puede tener un máximo de cinco grupos de informes. Especifique el ARN de un grupo de informes existente o el nombre de un nuevo grupo de informes. Si especifica un nombre, CodeBuild crea un grupo de informes utilizando el nombre del proyecto y el nombre especificado en el formato `<project-name>-<report-group-name>`. Para obtener más información, consulte [Nomenclatura de grupos de informes \(p. 314\)](#).

reports/<grupo-informes>/files

Secuencia obligatoria. Representa las ubicaciones que contienen los datos sin procesar de los resultados de las pruebas generados por el informe. Contiene una secuencia de valores escalares, en la que cada valor escalar representa una ubicación independiente donde CodeBuild puede encontrar los archivos de prueba, en relación con la ubicación de la compilación original o, si se ha definido, el `comandobase-directory`. Las ubicaciones pueden ser las siguientes:

- Un archivo único (por ejemplo, `my-test-report-file.json`).
- Un único archivo de un subdirectorio (por ejemplo, `my-subdirectory/my-test-report-file.json` o `my-parent-subdirectory/my-subdirectory/my-test-report-file.json`).
- `'**/*'` representa todos los archivos recursivamente.
- `my-subdirectory/*` representa todos los archivos de un subdirectorio denominado `my-subdirectory`.
- `my-subdirectory/**/*` representa todos los archivos recursivamente a partir de un subdirectorio denominado `my-subdirectory`.

reports/<grupo-informes>/file-format

Mapeo opcional. Representa el formato del archivo de informe. Si no se especifica, se utiliza `JUNITXML`. No distingue entre mayúsculas y minúsculas. Los valores posibles son:

Informes de pruebas

CUCUMBERJSON

Cucumber JSON

JUNITXML

JUnit XML

NUNITXML

NUnit XML

NUNIT3XML

NUnit 3 XML

TESTNGXML

TestNG XML

VISUALSTUDIOTRX

Visual Studio TRX

Informes de cobertura de códigos

CLOVERXML

XML de trébol

COBERTURAXML

Cobertura XML

JACOCOXML

XML de JaCoco

SIMPLECOV

SimpleCov JSON

Note

CodeBuild acepta informes de cobertura de código JSON generados por [simplecov](#), [no-simplecov-json](#).

`reports/<grupo-informes>/base-directory`

Mapeo opcional. Representa uno o varios directorios de nivel superior, en relación con la ubicación de la compilación original, que CodeBuild utiliza para determinar dónde encontrar los archivos de prueba sin procesar.

`reports/<grupo-informes>/discard-paths`

Opcional. Especifica si los directorios del archivo de informes se aplanan en la salida. Si esto no se especifica o contiene `no`, los archivos de informes se generan con su estructura de directorios intacta. Si esto contiene `yes`, todos los archivos de prueba se colocan en el mismo directorio de salida. Por ejemplo, si una ruta a un resultado de prueba es `com/myapp/mytests/TestResult.xml`, especificar `yes` colocará este archivo en `/TestResult.xml`.

artifacts

Secuencia opcional. Representa la información sobre dónde CodeBuild puede encontrar el resultado de compilación y cómo CodeBuild lo prepara para cargarlo en el bucket de salida de S3. Esta secuencia no es necesaria si, por ejemplo, va a crear e insertar una imagen de Docker en Amazon ECR o si va a ejecutar pruebas unitarias en el código fuente pero no va a compilarlas.

`artifacts/files`

Secuencia obligatoria. Representa las ubicaciones que contienen los artefactos de salida de la compilación en el entorno de compilación. Contiene una secuencia de valores escalares, en la que cada valor escalar representa una ubicación independiente donde CodeBuild puede encontrar artefactos de salida de la compilación en relación con la ubicación de la compilación original o, si se ha definido, el directorio base. Las ubicaciones pueden ser las siguientes:

- Un archivo único (por ejemplo, `my-file.jar`).
- Un único archivo de un subdirectorio (por ejemplo, `my-subdirectory/my-file.jar` o `my-parent-subdirectory/my-subdirectory/my-file.jar`).
- `'**/*'` representa todos los archivos recursivamente.
- `my-subdirectory/*` representa todos los archivos de un subdirectorio denominado `my-subdirectory`.
- `my-subdirectory/**/*` representa todos los archivos recursivamente a partir de un subdirectorio denominado `my-subdirectory`.

Cuando especifique las ubicaciones de artefactos de salida de la compilación, CodeBuild podrá encontrar la ubicación de la compilación original en el entorno de compilación. No tiene que anexar las ubicaciones de los artefactos de salida de la compilación a la ruta de acceso de la ubicación de la compilación original ni especificar `./` o similar. Si desea conocer la ruta a esta ubicación, puede ejecutar un comando como `echo $CODEBUILD_SRC_DIR` durante una compilación. La ubicación de cada entorno de compilación puede ser ligeramente diferente.

`artifacts/name`

Nombre opcional. Especifica un nombre para su artefacto de compilación. Este nombre se utiliza cuando se cumple alguna de las condiciones siguientes.

- Utilice la API de CodeBuild para crear sus compilaciones y `overrideArtifactName` se establece en el `ProjectArtifactsObjeto` cuando se actualiza un proyecto, se crea un proyecto o se inicia una compilación.

- Puede utilizar la consola de CodeBuild para crear sus compilaciones, se especifica un nombre en el archivo buildspec y se selecciona [Habilitar el control de versiones semánticas](#) Al crear o actualizar un proyecto de. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).

Puede especificar un nombre en el archivo de especificación de compilación que se calcula en el momento de la compilación. El nombre especificado en un archivo de especificación utiliza el lenguaje de comandos Shell. Por ejemplo, puede adjuntar una fecha y una hora al nombre del artefacto para que siempre sea único. Los nombres de artefactos únicos impiden que los artefactos se sobrescriban. Para obtener más información, consulte [Lenguaje de comandos Shell](#).

- Este es un ejemplo de una nombre de artefacto asociado con la fecha de creación del artefacto.

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
    - '**/*'
  name: myname-$(date +%Y-%m-%d)
```

- Este es un ejemplo de un nombre de artefacto que utiliza una variable de entorno de CodeBuild. Para obtener más información, consulte [Variables de entorno en los entornos de compilación](#) (p. 172).

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
    - '**/*'
  name: myname-$AWS_REGION
```

- Este es un ejemplo de un nombre de artefacto que utiliza una variable de entorno de CodeBuild con la fecha de creación del artefacto adjunta.

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
    - '**/*'
  name: $AWS_REGION-$(date +%Y-%m-%d)
```

Puede agregar información de ruta al nombre para que los artefactos con nombre se coloquen en directorios basados en la ruta del nombre. En este ejemplo, los artefactos de compilación se colocan en la salida de `bajobuilds/<build number>/my-artifacts`.

```
version: 0.2
phases:
  build:
    commands:
      - rspec HelloWorld_spec.rb
artifacts:
  files:
```

```
- '**/*'
name: builds/$CODEBUILD_BUILD_NUMBER/my-artifacts
```

artifacts/discard-paths

Opcional. Especifica si los directorios de artefactos de compilación se aplanan en la salida. Si esto no se especifica o contiene `no`, los artefactos de compilación se generan con su estructura de directorios intacta. Si esto contiene `yes`, todos los artefactos de compilación se colocan en el mismo directorio de salida. Por ejemplo, si una ruta a un archivo en el artefacto de salida de compilación es `com/mycompany/app/HelloWorld.java`, especificar `yes` colocará este archivo en `/HelloWorld.java`.

artifacts/base-directory

Mapeo opcional. Representa uno o varios directorios de nivel superior, en relación con la ubicación de la compilación original, que CodeBuild utiliza para determinar qué archivos y subdirectorios debe incluir en el artefacto de salida de la compilación. Los valores válidos son:

- Un único directorio de nivel superior (por ejemplo, `my-directory`).
- `'my-directory*'` representa todos los directorios de nivel superior con nombres que empiezan por `my-directory`.

Los directorios de nivel superior coincidentes no se incluyen en el artefacto de salida de la compilación, solo sus archivos y subdirectorios.

Puede utilizar `files` y `discard-paths` para restringir aún más los archivos y subdirectorios que se incluyen. Por ejemplo, para la siguiente estructura de directorios:

```
.
### my-build-1
#   ### my-file-1.txt
### my-build-2
    ### my-file-2.txt
    ### my-subdirectory
        ### my-file-3.txt
```

Y para la siguiente secuencia `artifacts`:

```
artifacts:
  files:
    - '*my-file-3.txt'
  base-directory: my-build-2
```

Se incluiría el siguiente subdirectorio y archivo en el artefacto de salida de la compilación:

```
.
### my-subdirectory
    ### my-file-3.txt
```

Sin embargo, para la siguiente secuencia `artifacts`:

```
artifacts:
  files:
    - '**/*'
  base-directory: 'my-build*'
  discard-paths: yes
```

Se incluirían los siguientes archivos en el artefacto de salida de la compilación:

```
.  
### my-file-1.txt  
### my-file-2.txt  
### my-file-3.txt
```

Artefactos deexcluye rutas

Mapeo opcional. Representa una o más rutas, relativas a `base-directory`, que CodeBuild excluirá de los artefactos de compilación.

Artefactos deactivar-enlaces simbólicos

Opcional. Si el tipo de salida es `ZIP`, especifica si los vínculos simbólicos internos se conservan en el archivo ZIP. Si esto contiene `yes`, todos los enlaces simbólicos internos de la fuente se conservarán en el archivo ZIP de artefactos.

Artefactos des3-prefijo

Opcional. Especifica un prefijo utilizado cuando los artefactos se envían a un bucket de Amazon S3 y el tipo de espacio de nombres es `BUILD_ID`. Cuando se utiliza, la ruta de salida en el bucket es `s3-prefix/<build-id>/<name>.zip`.

artifacts/secondary-artifacts

Secuencia opcional. Representa una o varias definiciones de artefacto como mapeo entre un identificador de artefacto y una definición de este. Cada uno de los identificadores de artefacto de este bloque debe coincidir con un artefacto definido en el atributo `secondaryArtifacts` del proyecto. Todas las definiciones tienen la misma sintaxis que el bloque `artifacts` anterior.

Note

La [artifacts/files \(p. 147\)](#) siempre es necesaria, incluso cuando solo hay artefactos secundarios definidos.

Por ejemplo, si el proyecto tiene la estructura siguiente:

```
{  
  "name": "sample-project",  
  "secondaryArtifacts": [  
    {  
      "type": "S3",  
      "location": "output-bucket1",  
      "artifactIdentifier": "artifact1",  
      "name": "secondary-artifact-name-1"  
    },  
    {  
      "type": "S3",  
      "location": "output-bucket2",  
      "artifactIdentifier": "artifact2",  
      "name": "secondary-artifact-name-2"  
    }  
  ]  
}
```

El archivo `buildspec` tiene este aspecto:

```
version: 0.2  
  
phases:  
  build:  
    commands:  
      - echo Building...
```

```
artifacts:
  files:
    - '**/*'
  secondary-artifacts:
    artifact1:
      files:
        - directory/file1
      name: secondary-artifact-name-1
    artifact2:
      files:
        - directory/file2
      name: secondary-artifact-name-2
```

cache

Secuencia opcional. Representa la información sobre en lugar en el que CodeBuild puede preparar los archivos para cargar la caché en un bucket de caché de S3. Esta secuencia no es necesaria si el tipo de caché del proyecto es No Cache.

cache/paths

Secuencia obligatoria. Representa las ubicaciones de la caché. Contiene una secuencia de valores escalares, en la que cada valor escalar representa una ubicación independiente donde CodeBuild puede encontrar artefactos de salida de la compilación en relación con la ubicación de la compilación original o, si se ha definido, el directorio base. Las ubicaciones pueden ser las siguientes:

- Un archivo único (por ejemplo, `my-file.jar`).
- Un único archivo de un subdirectorio (por ejemplo, `my-subdirectory/my-file.jar` o `my-parent-subdirectory/my-subdirectory/my-file.jar`).
- `'**/*'` representa todos los archivos recursivamente.
- `my-subdirectory/*` representa todos los archivos de un subdirectorio denominado `my-subdirectory`.
- `my-subdirectory/**/*` representa todos los archivos recursivamente a partir de un subdirectorio denominado `my-subdirectory`.

Important

Como una declaración de especificación de compilación debe ser una declaración YAML válida, los espacios de la declaración son importantes. Si el número de espacios en la declaración de la especificación de compilación no es válido, es posible que las compilaciones produzcan un error inmediatamente. Puede utilizar un validador YAML para comprobar si sus declaraciones de especificación de compilación son declaraciones YAML válidas.

Si utiliza la AWS CLI o los SDK de AWS para declarar una especificación de compilación cuando crea o actualiza un proyecto de compilación, la especificación de compilación debe ser una cadena única expresada en formato YAML, junto con los espacios en blanco y los caracteres de escape de nueva línea necesarios. Encontrará un ejemplo en la siguiente sección.

Si utiliza el código CodeBuild o AWS CodePipeline en lugar de un archivo `buildspec.yml`, puede insertar comandos para el comando `build` Fase solamente. En lugar de utilizar la sintaxis anterior, incluirá en una sola línea todos los comandos que desea ejecutar durante la fase de compilación. En caso de que haya varios comandos, separe cada comando con `&&` (por ejemplo, `mvn test && mvn package`).

Puede utilizar las consolas de CodeBuild o CodePipeline en lugar de un archivo `buildspec.yml` para especificar las ubicaciones de los artefactos de salida de la compilación en el entorno de compilación. En lugar de utilizar la sintaxis anterior, incluirá en una sola línea todas las ubicaciones. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, `buildspec.yml, target/my-app.jar`).

Ejemplo de un archivo buildspec

A continuación se muestra un ejemplo de un archivo buildspec.yml.

```
version: 0.2

env:
  variables:
    JAVA_HOME: "/usr/lib/jvm/java-8-openjdk-amd64"
  parameter-store:
    LOGIN_PASSWORD: /CodeBuild/dockerLoginPassword

phases:
  install:
    commands:
      - echo Entered the install phase...
      - apt-get update -y
      - apt-get install -y maven
    finally:
      - echo This always runs even if the update or install command fails
  pre_build:
    commands:
      - echo Entered the pre_build phase...
      - docker login -u User -p $LOGIN_PASSWORD
    finally:
      - echo This always runs even if the login command fails
  build:
    commands:
      - echo Entered the build phase...
      - echo Build started on `date`
      - mvn install
    finally:
      - echo This always runs even if the install command fails
  post_build:
    commands:
      - echo Entered the post_build phase...
      - echo Build completed on `date`

reports:
  arn:aws:codebuild:your-region:your-aws-account-id:report-group/report-group-name-1:
    files:
      - "**/*"
    base-directory: 'target/tests/reports'
    discard-paths: no
  reportGroupCucumberJson:
    files:
      - 'cucumber/target/cucumber-tests.xml'
    discard-paths: yes
    file-format: CUCUMBERJSON # default is JUNITXML

artifacts:
  files:
    - target/messageUtil-1.0.jar
  discard-paths: yes
  secondary-artifacts:
    artifact1:
      files:
        - target/artifact-1.0.jar
      discard-paths: yes
    artifact2:
      files:
        - target/artifact-2.0.jar
      discard-paths: yes
  cache:
    paths:
```

```
- '/root/.m2/**/*'
```

A continuación se muestra un ejemplo de la especificación de compilación anterior, expresada como una sola cadena, para su uso con la AWS CLI o los SDK de AWS.

```
"version: 0.2\n\nenv:\n  variables:\n    JAVA_HOME: \"/usr/lib/jvm/java-8-openjdk-amd64\n\nparameter-store:\n  LOGIN_PASSWORD: /CodeBuild/dockerLoginPassword\n\nphases:\n  install:\n    commands:\n      - echo Entered the install phase...\n      - apt-get update\n      - apt-get install -y maven\n  finally:\n    - echo This always runs even if the update or install command fails\n  pre_build:\n    commands:\n      - echo Entered the pre_build phase...\n      - docker login -u User -p $LOGIN_PASSWORD\n  finally:\n    - echo This always runs even if the login command fails\n  build:\n    commands:\n      - echo Entered the build phase...\n      - echo Build started on `date`\n      - mvn install\n  finally:\n    - echo This always runs even if the install command fails\n  post_build:\n    commands:\n      - echo Entered the post_build phase...\n      - echo Build completed on `date`\n\nreports:\n  reportGroupJUnitXml:\n    files:\n      - \"**/*\"\n    base-directory: 'target/tests/reports'\n    discard-paths: false\n  reportGroupCucumberJson:\n    files:\n      - 'cucumber/target/cucumber-tests.xml'\n    file-format: CUCUMBERJSON\n\nartifacts:\n  files:\n    - target/messageUtil-1.0.jar\n  discard-paths: yes\n  secondary-artifacts:\n    artifact1:\n      files:\n        - target/messageUtil-1.0.jar\n      discard-paths: yes\n    artifact2:\n      files:\n        - target/messageUtil-1.0.jar\n      discard-paths: yes\n  cache:\n    paths:\n      - '/root/.m2/**/*'
```

A continuación, se muestra un ejemplo de los comandos de labuild, para su uso con las consolas CodeBuild o CodePipeline.

```
echo Build started on `date` && mvn install
```

En estos ejemplos:

- Se establece una variable de entorno personalizada, en texto sin formato, con la clave `JAVA_HOME` y el valor `/usr/lib/jvm/java-8-openjdk-amd64`.
- Una variable de entorno personalizada denominada `dockerLoginPassword` Para hacer referencia en los comandos de compilación en el almacén de parámetros Amazon EC2 Systems Manager se usa la clave `deLOGIN_PASSWORD`.
- No puede cambiar estos nombres de fases de compilación. Los comandos que se ejecutan en este ejemplo son `apt-get update` `apt-get install -y maven` (para instalar Apache Maven), `mvn install` (para compilar, probar y empaquetar el código fuente en un artefacto de salida de la compilación e instalar el artefacto de salida de la compilación en su repositorio interno), `docker login` (para iniciar sesión en Docker con la contraseña que corresponde al valor de la variable de entorno personalizada `dockerLoginPassword` Se establece en el almacén de parámetros Amazon EC2 Systems Manager) y varios `echo` `commands`. Los comandos se incluyen aquí para mostrar cómo CodeBuild ejecuta los comandos y en el orden en que los ejecuta.
- `files` representa los archivos que se cargan en la ubicación de salida de la compilación. En este ejemplo, CodeBuild carga un solo archivo: `messageUtil-1.0.jar`. El archivo `messageUtil-1.0.jar` se encuentra en el directorio relativo denominado `target` en el entorno de compilación. Como se ha especificado `discard-paths: yes`, `messageUtil-1.0.jar` se carga directamente (y no en un directorio `target` intermedio). El nombre de archivo `messageUtil-1.0.jar` y el nombre del directorio relativo `target` se basan en la forma en la que Apache Maven crea y almacena los artefactos de salida de la compilación solo para este ejemplo. En sus propios escenarios, estos nombres de archivos y directorios serán diferentes.
- `reports` representa dos grupos de informes que generan informes durante la compilación:
 - `arn:aws:codebuild:your-region:your-aws-account-id:report-group/report-group-name-1` especifica el ARN de un grupo de informes. Los resultados de la prueba generados por el marco de prueba están en el directorio `target/tests/reports`. El formato del archivo es `JUnitXml` y la ruta no se elimina de los archivos que contienen los resultados de prueba.

- `reportGroupCucumberJson` especifica un nuevo grupo de informes. Si el nombre del proyecto es `my-project`, se crea un grupo de informes con el nombre `my-project-reportGroupCucumberJson` cuando se ejecuta una compilación. Los resultados de prueba generados por el marco de pruebas están en `cucumber/target/cucumber-tests.xml`. El formato del archivo de prueba es `CucumberJson` y la ruta se elimina de los archivos que contienen los resultados de prueba.

Versiones de buildspec

En la siguiente tabla se muestran las versiones de especificaciones de compilación y los cambios entre versiones.

Versión	Cambios
0.2	<ul style="list-style-type: none">• <code>environment_variables</code> ahora se llama <code>env</code>.• <code>plaintext</code> ahora se llama <code>variables</code>.• La propiedad <code>type</code> de <code>artifacts</code> ya no se utiliza.• En la versión 0.1, AWS CodeBuild ejecuta cada comando de compilación en una instancia distinta del shell predeterminado del entorno de compilación. En la versión 0.2, CodeBuild ejecuta todos los comandos de compilación en la misma instancia del shell predeterminado en el entorno de compilación.
0.1	Esta es la primera definición del formato de especificación de compilación.

Referencia de la especificación de compilación por Batch

Este tema contiene la referencia buildspec para las propiedades de compilación por lotes.

batch

Mapeo opcional. La configuración de compilación por lotes para el proyecto.

`lote/rápida falla`

Opcional. Especifica el comportamiento de la compilación por lotes cuando se produce un error en una o más tareas de compilación.

`false`

El valor predeterminado. Se completarán todas las compilaciones en ejecución.

`true`

Todas las compilaciones en ejecución se detendrán cuando se produzca un error en una de las tareas de compilación.

De forma predeterminada, todas las tareas de compilación por lotes se ejecutan con la configuración de compilación, como `envyphases`, especificado en el archivo buildspec. Para anular la configuración

de compilación predeterminada, especifique diferentes `env` en un archivo `buildspec` diferente en `batch/<batch-type>/buildspec` parámetro.

El contenido de `batch` La propiedad varía en función del tipo de compilación por lotes que se especifica. Los posibles tipos de compilación por lotes son:

- [batch/build-graph](#) (p. 155)
- [batch/build-list](#) (p. 156)
- [batch/build-matrix](#) (p. 158)

batch/build-graph

Define un Gráfico de compilación. Un gráfico de compilación define un conjunto de tareas que tienen dependencias de otras tareas del lote. Para obtener más información, consulte [Gráfico de compilación](#) (p. 277).

Este elemento contiene una matriz de tareas de compilación. Cada tarea de compilación contiene las siguientes propiedades.

identifier

Obligatorio. El identificador de la tarea.

BuildSpec

Opcional. Ruta y nombre de archivo del archivo `buildspec` que se usará para esta tarea. Si no se especifica este parámetro, se usará el archivo `buildspec` actual.

depuración de

Opcional. Un valor booleano que indica si la depuración de sesión está habilitada para esta compilación por lotes. Para obtener más información acerca de la depuración de sesiones, consulte [Ver una compilación en ejecución en el Administrador de sesiones](#) (p. 300).

`false`

La depuración de sesión está deshabilitada.

`true`

La depuración de sesión está habilitada.

depend-on

Opcional. Una matriz de identificadores de tarea de los que depende esta tarea. Esta tarea no se ejecutará hasta que se completen estas tareas.

env

Opcional. El entorno de compilación reemplaza para la tarea. Esto puede contener las siguientes propiedades:

Tipo de computación

Identificador del tipo de cálculo que se va a utilizar para la tarea. Consulte `computeType` in the section called “Tipos de computación del entorno de compilación” (p. 169) para los posibles valores.

image

El identificador de la imagen que se usará para la tarea. Consulte `imageIdentifier` in the section called “Imágenes de Docker proporcionadas por CodeBuild” (p. 161) para los posibles valores.

modo privilegiado

Un valor booleano que indica si se ejecuta el demonio de Docker dentro de un contenedor Docker. Definir en `true` Solo si el proyecto de compilación se utiliza para compilar imágenes de Docker. De lo contrario, se produce un error las compilaciones que intentan interactuar con el demonio de Docker. El valor predeterminado es `false`.

type

Identificador del tipo de entorno que se va a utilizar para la tarea. Consulte [Tipo de entorno](#) in [the section called "Tipos de computación del entorno de compilación" \(p. 169\)](#) para los posibles valores.

variables

Las variables de entorno que estarán presentes en el entorno de compilación. Consulte [env/variables \(p. 140\)](#) Para obtener más información,.

error de ignorar

Opcional. Valor booleano que indica si se puede ignorar un error de esta tarea de compilación.

`false`

El valor predeterminado. Si esta tarea de compilación falla, la compilación por lotes fallará.

`true`

Si se produce un error en esta tarea de compilación, la compilación por lotes aún puede tener éxito.

A continuación se muestra un ejemplo de una entrada `buildspec` de compilación:

```
batch:
  fast-fail: false
  build-graph:
    - identifier: build1
      env:
        variables:
          BUILD_ID: build1
      ignore-failure: false
    - identifier: build2
      buildspec: build2.yml
      env:
        variables:
          BUILD_ID: build2
      depend-on:
        - build1
    - identifier: build3
      env:
        variables:
          BUILD_ID: build3
      depend-on:
        - build2
```

batch/build-list

Define una lista de compilación. Una lista de compilación se utiliza para definir una serie de tareas que se ejecutan en paralelo. Para obtener más información, consulte [Lista de compilación \(p. 278\)](#).

Este elemento contiene una matriz de tareas de compilación. Cada tarea de compilación contiene las siguientes propiedades.

identifier

Obligatorio. El identificador de la tarea.

BuildSpec

Opcional. Ruta y nombre de archivo del archivo buildspec que se usará para esta tarea. Si no se especifica este parámetro, se utiliza el archivo buildspec actual.

depuración de

Opcional. Un valor booleano que indica si la depuración de sesión está habilitada para esta compilación por lotes. Para obtener más información acerca de la depuración de sesiones, consulte [Ver una compilación en ejecución en el Administrador de sesiones \(p. 300\)](#).

false

La depuración de sesión está deshabilitada.

true

La depuración de sesión está habilitada.

env

Opcional. El entorno de compilación reemplaza para la tarea. Esto puede contener las siguientes propiedades:

Tipo de computación

Identificador del tipo de cálculo que se va a utilizar para la tarea. Consulte `computeType` en [the section called "Tipos de computación del entorno de compilación" \(p. 169\)](#) para los posibles valores.

image

El identificador de la imagen que se usará para la tarea. Consulte `imageIdentifier` en [the section called "Imágenes de Docker proporcionadas por CodeBuild" \(p. 161\)](#) para los posibles valores.

modo privilegiado

Un valor booleano que indica si se ejecuta el demonio de Docker dentro de un contenedor Docker. Definir en `true` Solo si el proyecto de compilación se utiliza para compilar imágenes de Docker. De lo contrario, se produce un error las compilaciones que intentan interactuar con el demonio de Docker. El valor predeterminado es `false`.

type

Identificador del tipo de entorno que se va a utilizar para la tarea. Consulte `Tipo de entorno` en [the section called "Tipos de computación del entorno de compilación" \(p. 169\)](#) para los posibles valores.

variables

Las variables de entorno que estarán presentes en el entorno de compilación. Consulte [env/variables \(p. 140\)](#) Para obtener más información,.

error de ignorar

Opcional. Valor booleano que indica si se puede ignorar un error de esta tarea de compilación.

false

El valor predeterminado. Si esta tarea de compilación falla, la compilación por lotes fallará.

true

Si se produce un error en esta tarea de compilación, la compilación por lotes aún puede tener éxito.

A continuación se muestra un ejemplo de una entrada buildspec de lista de compilación:

```
batch:
  fast-fail: false
  build-list:
    - identifier: build1
      env:
        variables:
          BUILD_ID: build1
      ignore-failure: false
    - identifier: build2
      buildspec: build2.yml
      env:
        variables:
          BUILD_ID: build2
      ignore-failure: true
```

batch/build-matrix

Define unConstruir matriz. Una matriz de compilación define tareas con diferentes configuraciones que se ejecutan en paralelo. CodeBuild crea una compilación independiente para cada combinación de configuración posible. Para obtener más información, consulte [Compilar matriz \(p. 278\)](#).

estático

Las propiedades estáticas se aplican a todas las tareas de compilación.

error de ignorar

Opcional. Valor booleano que indica si se puede ignorar un error de esta tarea de compilación.

false

El valor predeterminado. Si esta tarea de compilación falla, la compilación por lotes fallará.

true

Si se produce un error en esta tarea de compilación, la compilación por lotes aún puede tener éxito.

env

Opcional. El entorno de compilación reemplaza para todas las tareas.

Tipo de computación

Identificador del tipo de cálculo que se va a utilizar para la tarea. Consulte `computeType` in [the section called “Tipos de computación del entorno de compilación” \(p. 169\)](#) para los posibles valores.

image

El identificador de la imagen que se usará para la tarea. Consulte `imageIdentifier` in [the section called “Imágenes de Docker proporcionadas por CodeBuild” \(p. 161\)](#) para los posibles valores.

modo privilegiado

Un valor booleano que indica si se ejecuta el demonio de Docker dentro de un contenedor Docker. Definir en `true` Solo si el proyecto de compilación se utiliza para compilar imágenes

de Docker. De lo contrario, se produce un error las compilaciones que intentan interactuar con el demonio de Docker. El valor predeterminado es `false`.

type

Identificador del tipo de entorno que se va a utilizar para la tarea. Consulte [Tipo de entorno](#) en [the section called “Tipos de computación del entorno de compilación” \(p. 169\)](#) para los posibles valores.

variables

Las variables de entorno que estarán presentes en el entorno de compilación. Consulte [env/variables \(p. 140\)](#) Para obtener más información,.

modo privilegiado

Un valor booleano que indica si se ejecuta el demonio de Docker dentro de un contenedor Docker. Definir en `true` Solo si el proyecto de compilación se utiliza para compilar imágenes de Docker. De lo contrario, se produce un error las compilaciones que intentan interactuar con el demonio de Docker. El valor predeterminado es `false`.

type

Identificador del tipo de entorno que se va a utilizar para estas tareas. Consulte [Tipo de entorno](#) en [the section called “Tipos de computación del entorno de compilación” \(p. 169\)](#) para los posibles valores.

dinámica

Las propiedades dinámicas definen la matriz de compilación.

BuildSpec

Opcional. Matriz que contiene la ruta de acceso y los nombres de archivo de los archivos `buildspec` que se utilizarán para estas tareas. Si no se especifica este parámetro, se utiliza el archivo `buildspec` actual.

env

Opcional. El entorno de compilación reemplaza para estas tareas.

Tipo de computación

Matriz que contiene los identificadores de los tipos de proceso que se van a utilizar para estas tareas. Consulte [computeType](#) en [the section called “Tipos de computación del entorno de compilación” \(p. 169\)](#) para los posibles valores.

image

Matriz que contiene los identificadores de las imágenes que se van a utilizar para estas tareas. Consulte [identificador de imagen](#) en [the section called “Imágenes de Docker proporcionadas por CodeBuild” \(p. 161\)](#) para los posibles valores.

variables

Una matriz que contiene las variables de entorno que estarán presentes en los entornos de compilación para estas tareas. Para obtener más información, consulte [env/variables \(p. 140\)](#).

A continuación se muestra un ejemplo de una entrada `buildspec` de matriz de compilación:

```
batch:
  build-matrix:
    static:
      ignore-failure: false
```

```
env:
  type: LINUX_CONTAINER
  image: aws/codebuild/amazonlinux2-x86_64-standard:3.0
  privileged-mode: true
dynamic:
  buildspec:
    - matrix1.yml
    - matrix2.yml
  env:
    variables:
      MY_VAR:
        - VALUE1
        - VALUE2
        - VALUE3
```

Para obtener más información, consulte [Compilar matriz \(p. 278\)](#).

Referencia de entornos de compilación de AWS CodeBuild

Si llama a AWS CodeBuild para ejecutar una compilación, deberá proporcionar información sobre el entorno que va a usar. Un entorno de compilación representa una combinación de sistema operativo, el tiempo de ejecución de un lenguaje de programación y herramientas que CodeBuild utiliza para ejecutar una compilación. Para obtener información sobre cómo funciona un entorno de compilación, consulte [Funcionamiento de CodeBuild \(p. 3\)](#).

Un entorno de compilación contiene una imagen de Docker. Para obtener información, consulte el [glosario de Docker](#) en el sitio web de Docker Docs.

Cuando proporciona información a CodeBuild sobre el entorno de compilación, debe especificar el identificador de una imagen de Docker en un tipo de repositorio admitido. Entre estos se incluyen el repositorio de imágenes de Docker de CodeBuild, imágenes disponibles públicamente en Docker Hub y repositorios de Amazon Elastic Container Registry (Amazon ECR) que su AWS tiene permisos para obtener acceso.

- Le recomendamos que utilice imágenes de Docker almacenadas en el repositorio de imágenes de Docker de CodeBuild, ya que están optimizadas para su uso con el servicio. Para obtener más información, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).
- Para obtener el identificador de una imagen de Docker disponible públicamente almacenada en Docker Hub, consulte [Búsqueda de repositorios](#) en el sitio web de Docker Docs.
- Para obtener información sobre cómo trabajar con imágenes de Docker almacenadas en repositorios de Amazon ECR en su AWS cuenta, consulte [Ejemplo de Amazon ECR \(p. 46\)](#).

Además de un identificador de imágenes de Docker, deberá especificar también el conjunto de recursos informáticos que utiliza el entorno. Para obtener más información, consulte [Tipos de computación del entorno de compilación \(p. 169\)](#).

Temas

- [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#)
- [Tipos de computación del entorno de compilación \(p. 169\)](#)
- [Shells y comandos de los entornos de compilación \(p. 171\)](#)
- [Variables de entorno en los entornos de compilación \(p. 172\)](#)
- [Tareas en segundo plano en entornos de compilación \(p. 176\)](#)

Imágenes de Docker proporcionadas por CodeBuild

AWS CodeBuild administra las siguientes imágenes de Docker que se encuentran disponibles en CodeBuild y AWS CodePipeline consolas.

Plataforma	Identificador de imagen	Definición
Amazon Linux 2	<code>aws/codebuild/amazonlinux2-x86_64-standard:2.0</code>	al2/standard/2.0
Amazon Linux 2	<code>aws/codebuild/amazonlinux2-x86_64-standard:3.0</code>	al2/standard/3.0
Amazon Linux 2	<code>aws/codebuild/amazonlinux2-aarch64-standard:1.0</code>	al2/aarch64/standard/1.0
Amazon Linux 2	<code>aws/codebuild/amazonlinux2-aarch64-standard:2.0</code>	al2/aarch64/standard/2.0
Ubuntu 18.04	<code>aws/codebuild/standard:3.0¹</code>	ubuntu/standard/3.0
Ubuntu 18.04	<code>aws/codebuild/standard:4.0</code>	ubuntu/standard/4.0
Ubuntu 20.04	<code>aws/codebuild/standard:5.0</code>	ubuntu/standard/5.0
Windows Server Core 2019	<code>aws/codebuild/windows-base:2019-1.0</code>	N/A

¹ No se mantiene después de mayo de 2021.

La imagen base de la plataforma Windows Server Core 2019 solo está disponible en las siguientes regiones:

- EE. UU. Este (Norte de Virginia)
- EE. UU. Este (Ohio)
- EE. UU. Oeste (Oregon)
- Europa (Irlanda)

La última versión de cada imagen se almacena en la caché. Si especifica una versión más concreta, CodeBuild aprovisiona esa versión en lugar de la versión en la caché. Esto puede dar lugar a tiempos de desarrollo más largos. Por ejemplo, para beneficiarse del almacenamiento en caché, especifique `aws/codebuild/amazonlinux2-x86_64-standard:3.0` en lugar de una versión más detallada, como `aws/codebuild/amazonlinux2-x86_64-standard:3.0-1.0.0`.

CodeBuild actualiza frecuentemente la lista de imágenes de Docker. Para obtener la lista más actualizada, realice alguna de las siguientes operaciones:

- En la consola de CodeBuild, en el **Crear un proyecto de compilación** asistente o **Editar proyecto de compilación**, para **imagen de entorno**, elija **imagen administrada**. Elija valores en las listas desplegables

Operating system (Sistema operativo), Runtime (Tiempo de ejecución) y Runtime version (Versión de tiempo de ejecución). Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) o [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).

- Para la AWS CLI, ejecute el comando `list-curated-environment-images`:

```
aws codebuild list-curated-environment-images
```

- Para los SDK de AWS, llame a la operación `ListCuratedEnvironmentImages` correspondiente a su lenguaje de programación de destino. Para obtener más información, consulte la [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Temas

- [Runtimes de ejecución disponibles](#) (p. 162)
- [Versiones de tiempo de ejecución](#) (p. 168)

Runtimes de ejecución disponibles

Puede especificar uno o más tiempos de ejecución en la sección `runtime-versions` del archivo de especificación de compilación. Si el tiempo de ejecución depende de otro tiempo de ejecución, también puede especificar el tiempo de ejecución dependiente en el archivo de especificación de compilación. Si no especifica ningún tiempo de ejecución en el archivo de especificación de compilación, CodeBuild elige los runtimes predeterminados que están disponibles en la imagen que utiliza. Si especifica uno o más tiempos de ejecución, CodeBuild utiliza solo esos runtimes. Si no se especifica un motor de ejecución dependiente, CodeBuild intenta elegir el motor de ejecución dependiente. Para obtener más información, consulte [Specify runtime versions in the buildspec file](#).

Temas

- [tiempos de ejecución de imágenes Linux](#) (p. 162)
- [Runtimes de imágenes de Windows](#) (p. 167)

tiempos de ejecución de imágenes Linux

En la siguiente tabla se incluyen los tiempos de ejecución disponibles y las imágenes Linux estándar que los admiten.

Tiempos de ejecución de la plataforma de Ubuntu/Amazon Linux 2

Nombre del runtime	Versión	Imágenes
android	28	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Ubuntu:3.0 Estándar de Ubuntu:4.0
	29	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64

Nombre del runtime	Versión	Imágenes
dotnet		Estándar de Ubuntu:3.0
		Estándar de Ubuntu:4.0
	3.0	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Ubuntu:3.0
	3.1	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:4.0 Estándar de Ubuntu:5.0
	5.0	Estándar de Ubuntu:5.0
golang	1.12	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:1.0 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:3.0 Estándar de Ubuntu:4.0
	1.13	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:1.0 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:3.0 Estándar de Ubuntu:4.0

Nombre del runtime	Versión	Imágenes
	1.14	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:4.0
	1.15	Estándar de Ubuntu:5.0
	1.16	Estándar de Ubuntu:5.0
nodejs	8	Estándar de Amazon Linux 2 AArch64:1.0
	10	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:1.0 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:3.0 Estándar de Ubuntu:4.0
	12	Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 x86_64 x86_64 Estándar de Amazon Linux 2 AArch64:1.0 Estándar de Amazon Linux 2 AArch64:2.0 Estándar de Ubuntu:3.0 Estándar de Ubuntu:4.0 Estándar de Ubuntu:5.0
	14	Estándar de Ubuntu:5.0
java	openjdk8	Estándar de Ubuntu:3.0
	openjdk11	Estándar de Ubuntu:3.0

Nombre del runtime	Versión	Imágenes
	corretto8	<p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
	corretto11	<p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
php	7.3	<p>Estándar de Amazon Linux 2 x86_64:2.0</p> <p>Estándar de Amazon Linux 2 x86_64:3.0</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:3.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>

Nombre del runtime	Versión	Imágenes
	7.4	<p>Estándar de Amazon Linux 2 x86_64:3.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
	8.0	Estándar de Ubuntu:5.0
python	3.7	<p>Estándar de Amazon Linux 2 x86_64:3.0</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
	3.8	<p>Estándar de Amazon Linux 2 x86_64:2.0</p> <p>Estándar de Amazon Linux 2 x86_64:3.0</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:3.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
	3.9	Estándar de Ubuntu:5.0

Nombre del runtime	Versión	Imágenes
ruby	2.6	<p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 x86_64 x86_64</p> <p>Estándar de Amazon Linux 2 AArch64:1.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:3.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>
	2.7	<p>Estándar de Amazon Linux 2 x86_64:3.0</p> <p>Estándar de Amazon Linux 2 AArch64:2.0</p> <p>Estándar de Ubuntu:4.0</p> <p>Estándar de Ubuntu:5.0</p>

Runtimes de imágenes de Windows

La imagen base de Windows Server Core 2019 contiene los siguientes runtimes.

Runtimes de la plataforma Windows

Nombre del runtime	Versiones disponibles en enwindows-base:2019-1.0	
dotnet	3.1.404 5.0	
golang	1.14	
nodejs	12.18	
java	corretto11	
php	7.4.7	
powershell	7.0.2	
python	3.8.3	
ruby	2.7	

Versiones de tiempo de ejecución

Cuando especifica un tiempo de ejecución en la sección `runtime-versions` (p. 144) del archivo `buildspec`, puede especificar una versión específica, una versión principal específica y la última versión secundaria o la última versión. En la tabla siguiente se muestran los tiempos de ejecución disponibles y cómo especificarlos.

Versiones en tiempo de ejecución de la plataforma Ubuntu y Amazon Linux 2

Nombre del runtime	Versión	Versión específica
android	28	android: 28
	29	android: 29
dotnet	3.0	dotnet: 3.0
	3.1	dotnet: 3.1
	5.0	dotnet: 5.0
golang	1.12	golang: 1.12
	1.13	golang: 1.13
	1.14	golang: 1.14
	1.15	golang: 1.15
	1.16	golang: 1.16
nodejs	8	nodejs: 8
	10	nodejs: 10
	12	nodejs: 12
	14	nodejs: 14
java	openjdk8	java: openjdk8
	openjdk11	java: openjdk11
	corretto8	java: corretto8
	corretto11	java: corretto11
php	7.3	php: 7.3
	7.4	php: 7.4
	8.0	php: 8.0
python	3.7	python: 3.7
	3.8	python: 3.8

Nombre del runtime	Versión	Versión específica
	3.9	python: 3.9
ruby	2.6	ruby: 2.6
	2.7	ruby: 2.7

Note

La imagen `aws/codebuild/amazonlinux2-aarch64-standard:1.0` no admite Android Runtime (ART).

Puede utilizar una especificación de compilación para instalar otros componentes (por ejemplo, la AWS CLI, Apache Maven, Apache Ant, Mocha, RSpec o similares) durante la fase de compilación `install`. Para obtener más información, consulte [Ejemplo de un archivo buildspec \(p. 152\)](#).

Tipos de computación del entorno de compilación

AWS CodeBuild proporciona entornos de compilación con la siguiente memoria, CPU virtuales y espacio en disco disponibles:

Tipo de computación	Valor de computeType del entorno	Valor del tipo de entorno	Memoria	vCPU	Espacio en disco
ARM pequeña	BUILD_GENERAL1_SMALL_CONTAINER	ARM_CONTAINER	4 GB	2	50 GB
ARM Grande	BUILD_GENERAL1_2X_LARGE_CONTAINER	ARM_2X_LARGE_CONTAINER	16 GB	8	50 GB
Linux pequeño	BUILD_GENERAL1_SMALL_CONTAINER	LINUX_CONTAINER	2 GB	2	64 GB
Media para Linux	BUILD_GENERAL1_MEDIUM_CONTAINER	LINUX_CONTAINER	7 GB	4	128 GB
Linux grande	BUILD_GENERAL1_LARGE_CONTAINER	LINUX_CONTAINER	15 GB	8	128 GB
Linux 2xlarge	BUILD_GENERAL1_2X_LARGE_CONTAINER	LINUX_2X_LARGE_CONTAINER	45 GB	72	824 GB (SSD)
GPU Linux grande	BUILD_GENERAL1_LARGE_GPU_CONTAINER	LINUX_GPU_CONTAINER	25 GB	32	50 GB
Windows Media	BUILD_GENERAL1_MEDIUM_SERVER_19_CONTAINER	WINDOWS_SERVER_19_CONTAINER	7 GB	4	128 GB
Ventanas grandes	BUILD_GENERAL1_2X_LARGE_SERVER_19_CONTAINER	WINDOWS_SERVER_19_CONTAINER	45 GB	72	128 GB

El espacio de disco indicado para cada entorno de compilación solo está disponible en el directorio especificado por la variable de entorno `CODEBUILD_SRC_DIR`.

Para elegir un tipo de computación:

- En la consola de CodeBuild, en el **Crear un proyecto de compilación** asistente o **Editar proyecto de compilación**, en **Environment (Entorno)**: expanda **Configuración adicional**, a continuación, elija una de las opciones de **Tipo de computación**. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198) o [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).
- Para la AWS CLI, ejecute el comando `create-project` o `update-project`, especificando el valor de `computeType` del objeto `environment`. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209) o [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267).
- Para el SDK de AWS, llame al equivalente de la operación `CreateProject` o `UpdateProject` para su lenguaje de programación de destino, especificando el equivalente del valor de `computeType` del objeto `environment`. Para obtener más información, consulte la [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Algunos tipos de entorno y computación tienen limitaciones de disponibilidad de la región:

- El tipo de entorno `LINUX_GPU_CONTAINERS` solo está disponible en estas regiones:
 - EE. UU. Este (Norte de Virginia)
 - EE. UU. Oeste (Oregon)
 - Asia Pacífico (Seúl)
 - Asia Pacífico (Singapur)
 - Asia Pacífico (Sídney)
 - Asia Pacífico (Tokio)
 - Canadá (Central)
 - China (Pekín)
 - China (Ningxia)
 - Europa (Fráncfort)
 - Europa (Irland)
 - Europa (Londres)
- El tipo de entorno `ARM_CONTAINERS` solo está disponible en estas regiones:
 - EE. UU. Este (Ohio)
 - EE.UU. Este (Norte de Virginia)
 - EE.UU. Oeste (Norte de California)
 - US West (Oregon)
 - Asia Pacífico (Mumbai)
 - Asia Pacífico (Singapur)
 - Asia Pacífico (Sídney)
 - Asia Pacífico (Tokio)
 - Europa (Fráncfort)
 - Europa (Irlanda)
- El tipo de computación `BUILD_GENERAL1_2XLARGE` solo está disponible en estas regiones:
 - EE. UU. Este (Ohio)
 - EE.UU. Este (Norte de Virginia)
 - EE.UU. Oeste (Norte de California)
 - EE. UU. Oeste (Oregon)
 - Asia Pacífico (Hong Kong)
 - Asia Pacífico (Mumbai)

- Asia Pacific (Seoul)
- Asia Pacífico (Singapur)
- Asia Pacífico (Sídney)
- Asia Pacífico (Tokio)
- Canadá (Central)
- China (Pekín)
- China (Ningxia)
- Europa (Fráncfort)
- Europe (Ireland)
- Europe (London)
- Europe (Paris)
- Europe (Stockholm)
- Middle East (Bahrain)
- América del Sur (São Paulo)

Para el tipo de proceso `BUILD_GENERAL1_2XLARGE`, se admiten las imágenes de Docker de hasta 100 GB sin comprimir.

Note

Para crear imágenes del entorno de compilación personalizadas, CodeBuild permite el uso de imágenes de Docker de hasta 50 GB sin comprimir en Linux y Windows, independientemente del tipo de computación. Para comprobar el tamaño de la imagen de compilación, use Docker para ejecutar el comando `docker images` **REPOSITORY:TAG**.

Puede utilizar Amazon EFS para tener acceso a más espacio en el contenedor de compilación. Para obtener más información, consulte [Ejemplo de Amazon Elastic File System para AWS CodeBuild \(p. 50\)](#). Si desea manipular el espacio en disco del contenedor durante una compilación, la compilación debe ejecutarse en modo privilegiado.

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

Shells y comandos de los entornos de compilación

Proporcione un conjunto de comandos de AWS CodeBuild que se ejecuten en un entorno de compilación durante el ciclo de vida de una compilación (por ejemplo, la instalación de dependencias de compilación, y la comprobación y compilación del código fuente). Hay varias formas de especificar estos comandos:

- Cree un archivo de especificación de compilación e inclúyalo con su código fuente. En este archivo, especifique los comandos que desea ejecutar en cada fase del ciclo de vida de compilación. Para obtener más información, consulte la [Referencia de la especificación de compilación para CodeBuild \(p. 136\)](#).
- Utilice la consola de CodeBuild para crear un proyecto de compilación. En Insert build commands (Insertar comandos de compilación), para Build commands (Comandos de compilación), especifique los comandos que desea ejecutar en la fase `build`. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\) \(p. 198\)](#).
- Utilice la consola de CodeBuild para cambiar la configuración de un proyecto de compilación. En Insert build commands (Insertar comandos de compilación), para Build commands (Comandos de

compilación), especifique los comandos que desea ejecutar en la fase `build`. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).

- Utilice la AWS CLI o los SDK de AWS para crear un proyecto de compilación o para cambiar la configuración de un proyecto de compilación. Haga referencia al código fuente que contiene un archivo `buildspec` con los comandos o especifique una única cadena que incluya el contenido de un archivo `buildspec` equivalente. Para obtener más información, consulte [Creación de un proyecto de compilación](#) (p. 197) o [Cambiar la configuración de un proyecto de compilación](#) (p. 254).
- Utilice la AWS CLI o los SDK de AWS para comenzar una compilación, especificando un archivo `buildspec` o una única cadena que incluya el contenido de un archivo `buildspec` equivalente. Para obtener más información, consulte la descripción del valor `buildspecOverride` en [Ejecutar una compilación](#) (p. 281).

Puede especificar cualquier comando de Lenguaje de comandos Shell (`sh`). En la versión de especificación de compilación 0.1, CodeBuild ejecuta cada comando del Shell en una instancia independiente del entorno de compilación. Esto significa que cada comando se ejecuta con independencia de los demás. Por lo tanto, de forma predeterminada, no puede ejecutar un comando que se base en el estado de comandos anteriores (por ejemplo, cambiar directorios o configurar variables de entorno). Para solventar esta limitación, le recomendamos utilizar la versión 0.2, que soluciona este problema. Si debe utilizar la versión 0.1, se recomiendan los siguientes enfoques:

- Incluya un script del shell en el código fuente que contenga los comandos que desea ejecutar en una única instancia del shell predeterminado. Por ejemplo, podría incluir un archivo denominado `my-script.sh` en su código fuente que contenga comandos como `cd MyDir; mkdir -p mySubDir; cd mySubDir; pwd`; . A continuación, en el archivo `buildspec`, especifique el comando `./my-script.sh`.
- En el archivo `buildspec`, o en la opción `Build commands` (Comandos de compilación) para la fase `build` únicamente, especifique un solo comando que incluya todos los comandos que desee ejecutar en una sola instancia en el shell predeterminado (por ejemplo, `cd MyDir && mkdir -p mySubDir && cd mySubDir && pwd`).

Si CodeBuild encuentra un error, el error podría ser más difícil de solucionar que cuando se ejecuta un solo comando en su propia instancia del shell predeterminado.

Los comandos que se ejecutan en una imagen de Windows Server Core usan el shell PowerShell.

Variables de entorno en los entornos de compilación

AWS CodeBuild proporciona varias variables de entorno que puede utilizar en los comandos de compilación:

AWS_DEFAULT_REGION

LaAWSRegión de donde se ejecuta la compilación (por ejemplo, `us-east-1`). Esta variable de entorno se utiliza principalmente en la AWS CLI.

AWS_REGION

LaAWSRegión de donde se ejecuta la compilación (por ejemplo, `us-east-1`). Esta variable de entorno se utiliza principalmente en los SDK de AWS.

CODEBUILD_BATCH_BUILD_IDENTIFIER

Identificador de la compilación en una compilación por lotes. Esto se especifica en la especificación de compilación por lotes. Para obtener más información, consulte [the section called “Referencia de la especificación de compilación de Batch”](#) (p. 154).

CODEBUILD_BUILD_ARN

Nombre de recurso de Amazon (ARN) de la compilación (por ejemplo, `arn:aws:codebuild:region-ID:account-ID:build/codebuild-demo-project:b1e6661e-e4f2-4156-9ab9-82a19EXAMPLE`).

CODEBUILD_BUILD_ID

El ID de CodeBuild de la compilación (por ejemplo, `codebuild-demo-project:b1e6661e-e4f2-4156-9ab9-82a19EXAMPLE`).

CODEBUILD_BUILD_IMAGE

El identificador de la imagen de compilación de CodeBuild (por ejemplo, `aws/codebuild/standard:2.0`).

CODEBUILD_BUILD_NUMBER

El número de compilación actual del proyecto.

CODEBUILD_BUILD_SUCCEEDING

Si la compilación actual se ha realizado correctamente. Se establece en 0 si la compilación ha producido un error o en 1 si la compilación se ha realizado correctamente.

CODEBUILD_INITIATOR

La entidad que inició la compilación. Si CodePipeline inició la compilación, este es el nombre de la canalización (por ejemplo, `codepipeline/my-demo-pipeline`). Si un usuario de IAM inició la compilación, este es el nombre del usuario (por ejemplo, `MyUserName`). Si el complemento Jenkins para CodeBuild inició la compilación, esta es la cadena `CodeBuild-Jenkins-Plugin`.

CODEBUILD_KMS_KEY_ID

Identificador de la AWS KMS clave que CodeBuild utiliza para cifrar el artefacto de salida de la compilación (por ejemplo, `arn:aws:kms:region-ID:account-ID:key/key-IDoralias/key-alias`).

CODEBUILD_LOG_PATH

El nombre del flujo de logs de CloudWatch Logs para la compilación.

CODEBUILD_PUBLIC_BUILD_URL

La URL de los resultados de la compilación de esta compilación en el sitio web de compilaciones públicas. Esta variable solo se establece si el proyecto de compilación tiene habilitadas las compilaciones públicas. Para obtener más información, consulte [Proyectos de compilación pública en AWS CodeBuild](#) (p. 279).

CODEBUILD_RESOLVED_SOURCE_VERSION

El identificador de versión del código fuente de una compilación. El contenido depende del repositorio del código fuente:

CodeCommit, GitHub, GitHub Enterprise Server y Bitbucket

Esta variable contiene el ID de confirmación.

CodePipeline

Esta variable contiene la revisión de origen proporcionada por CodePipeline.

Si CodePipeline no puede resolver la revisión de origen, por ejemplo, cuando el origen es un bucket de Amazon S3 que no tiene habilitado el control de versiones, esta variable de entorno no se establece.

Amazon S3

Esta variable no está establecida.

Cuando proceda, el `CODEBUILD_RESOLVED_SOURCE_VERSION` la variable solo está disponible después de que `DOWNLOAD_SOURCE` fase.

`CODEBUILD_SOURCE_REPO_URL`

La dirección URL del artefacto de entrada o el repositorio del código fuente. Para Amazon S3, esto es `s3://` seguido del nombre del bucket y la ruta del artefacto de entrada. Para CodeCommit y GitHub, esta es la dirección URL de la clonación del repositorio. Si una compilación se origina de CodePipeline, esta variable de entorno puede estar vacía.

Para los orígenes secundarios, la variable de entorno de la URL del repositorio de origen secundario es `CODEBUILD_SOURCE_REPO_URL_<sourceIdentifier>`, donde `<sourceIdentifier>` es el identificador de origen que creas.

`CODEBUILD_SOURCE_VERSION`

El formato del valor depende del repositorio del código fuente.

- Para Amazon S3, es el ID de versión asociado con el artefacto de entrada.
- Para CodeCommit, es el ID de confirmación o el nombre de ramificación asociado con la versión del código fuente que se va a compilar.
- Para GitHub, GitHub Enterprise Server y Bitbucket es el ID de confirmación, el nombre de ramificación o el nombre de etiqueta asociado con la versión del código fuente que se va a compilar.

Note

Para una compilación de GitHub o GitHub Enterprise Server activada por un evento de solicitud de extracción de webhook, es `pr/pull-request-number`.

Para los orígenes secundarios, la variable de entorno de la versión de origen secundaria es `CODEBUILD_SOURCE_VERSION_<sourceIdentifier>`, donde `<sourceIdentifier>` es el identificador de origen que creas. Para obtener más información, consulte [Ejemplo de varios orígenes de entrada y varios artefactos de salida](#) (p. 129).

`CODEBUILD_SRC_DIR`

La ruta del directorio que CodeBuild utiliza para la compilación (por ejemplo, `/tmp/src123456789/src`).

Para los orígenes secundarios, la variable de entorno de la ruta del directorio de origen secundario es `CODEBUILD_SRC_DIR_<sourceIdentifier>`, donde `<sourceIdentifier>` es el identificador de origen que creas. Para obtener más información, consulte [Ejemplo de varios orígenes de entrada y varios artefactos de salida](#) (p. 129).

`CODEBUILD_START_TIME`

Hora de inicio de la compilación especificada como marca temporal Unix en milisegundos.

`CODEBUILD_WEBHOOK_ACTOR_ACCOUNT_ID`

: ID de la cuenta del usuario que ha desencadenado el evento de webhook.

`CODEBUILD_WEBHOOK_BASE_REF`

Nombre de referencia de base del evento de webhook que desencadena la compilación actual. En las solicitudes de extracción, se trata de la referencia de ramificación.

`CODEBUILD_WEBHOOK_EVENT`

El evento de webhook que desencadena la compilación actual.

CODEBUILD_WEBHOOK_MERGE_COMMIT

Identificador de la confirmación de fusión utilizada para la compilación. Esta variable se establece cuando una solicitud de extracción de Bitbucket se fusiona con la estrategia squash y se cierra la rama de solicitud de extracción. En este caso, la confirmación de solicitud de extracción original ya no existe, por lo que esta variable de entorno contiene el identificador de la confirmación de combinación aplastada.

CODEBUILD_WEBHOOK_PREV_COMMIT

: ID de la confirmación más reciente antes del evento push del webhook que desencadena la compilación actual.

CODEBUILD_WEBHOOK_HEAD_REF

Nombre de referencia de encabezado del evento de webhook que desencadena la compilación actual. Puede ser una referencia de ramificación o de etiqueta.

CODEBUILD_WEBHOOK_TRIGGER

Muestra el evento webhook que activó la compilación. Esta variable solo está disponible para compilaciones activadas por un webhook. El valor se analiza desde la carga que GitHub, GitHub Enterprise Server o Bitbucket envían a CodeBuild. El formato del valor depende del tipo de evento que desencadenó la compilación.

- Para las compilaciones desencadenadas por una solicitud de extracción, es `pr/pull-request-number`.
- Para las compilaciones desencadenadas por la creación de una nueva rama o la inserción de una confirmación en una rama, es `branch/branch-name`.
- Para compilaciones activadas por el envío de una etiqueta a un repositorio, es `tag/tag-name`.

INICIO

Esta variable de entorno siempre está establecida en `/root`.

También puede proporcionar entornos de compilación con sus propias variables de entorno. Para obtener más información, consulte los siguientes temas:

- [Usar CodePipeline con CodeBuild \(p. 407\)](#)
- [Creación de un proyecto de compilación \(p. 197\)](#)
- [Cambiar la configuración de un proyecto de compilación \(p. 254\)](#)
- [Ejecutar una compilación \(p. 281\)](#)
- [Referencia de la especificación de compilación \(p. 136\)](#)

Para enumerar todas las variables de entorno disponibles en un entorno de compilación, puede ejecutar el comando `printenv` (para un entorno de compilación basado en Linux) o `"Get-ChildItem Env:"` (para entornos de compilación basados en Windows) durante una compilación. A excepción de las indicadas anteriormente, las variables de entorno que comienzan por `CODEBUILD_` son para uso interno de CodeBuild. No deben utilizarse en los comandos de compilación.

Important

Se desaconseja usar variables de entorno para almacenar valores confidenciales, especialmente los identificadores de clave de acceso y las claves de acceso secretas de AWS. Las variables de entorno se pueden mostrar en texto sin formato con herramientas como la consola de CodeBuild y la AWS CLI.

Le recomendamos que almacene los valores confidenciales en el almacén de parámetros de Amazon EC2 Systems Manager y que después los recupere de la especificación de compilación.

Para almacenar valores sensibles, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Guía del tutorial: Creación y prueba de parámetros de cadena \(consola\)](#) en la Guía del usuario de Amazon EC2 Systems Manager. Para recuperarlos, consulte la asignación de `parameter-store` en [Sintaxis de buildspec \(p. 137\)](#).

Tareas en segundo plano en entornos de compilación

Puede ejecutar tareas en segundo plano en entornos de compilación. Para ello, en la especificación de compilación, use el comando `nohup` para ejecutar un comando como una tarea en segundo plano, aunque el proceso de compilación cierre el shell. Utilice el comando `disown` para detener una tarea que se esté ejecutando en segundo plano.

Ejemplos:

- Iniciar un proceso en segundo plano y esperar a que se complete:

```
nohup sleep 30 & echo $! > pidfile
...
wait $(cat pidfile)
```

- Iniciar un proceso en segundo plano y no esperar a que se complete:

```
nohup sleep 30 & disown $!
```

- Iniciar un proceso en segundo plano y cancelarlo más adelante:

```
nohup sleep 30 & echo $! > pidfile
...
kill $(cat pidfile)
```

Ejecute las compilaciones localmente con elAWS CodeBuildagente de

Puede utilizar elAWS CodeBuildagente de para ejecutar las compilaciones de CodeBuild en un equipo local. Hay agentes disponibles para plataformas x86_64 y ARM.

También puede suscribirse para recibir notificaciones cuando se publiquen nuevas versiones del agente de.

Prerequisites

Antes de comenzar, necesita hacer lo siguiente:

- Instale Git en su equipo local.
- Instalar y configurar[Docker](#)En el equipo local.

Configurar la imagen de compilación

Solo tiene que configurar la imagen de compilación la primera vez que ejecuta el agente o cuando la imagen ha cambiado.

Para configurar la imagen de compilación

1. Si desea utilizar una imagen seleccionada de Amazon Linux 2, puede extraerlo del repositorio público Amazon ECR de CodeBuild en https://gallery.ecr.aws/codebuild/amazonlinux2-x86_64-standard Con el comando siguiente:

```
$ docker pull public.ecr.aws/codebuild/amazonlinux2-x86_64-standard:3.0
```

Como alternativa, si desea utilizar otra imagen de Linux, siga estos pasos:

- a. Clonar el repositorio de imágenes de CodeBuild:

```
$ git clone https://github.com/aws/aws-codebuild-docker-images.git
```

- b. Cambie al directorio de imágenes. Para este ejemplo, use la `aws/codebuild/standard:5.0` image:

```
$ cd aws-codebuild-docker-images/ubuntu/standard/5.0
```

- c. Cree la imagen. Esto puede tardar varios minutos.

```
$ docker build -t aws/codebuild/standard:5.0 .
```

2. Descargue el agente de .

Para descargar la versión x86_64 del agente, ejecute el comando siguiente:

```
$ docker pull public.ecr.aws/codebuild/local-builds:latest
```

Para descargar la versión ARM del agente, ejecute el comando siguiente:

```
$ docker pull public.ecr.aws/codebuild/local-builds:aarch64
```

3. El agente de CodeBuild está disponible en <https://gallery.ecr.aws/codebuild/local-builds>.

La firma Secure Hash Algorithm (SHA) para la versión x86_64 del agente es:

```
sha256:fdfff9470520c53dcd522606a3cc2b5df195ae8a5546697b08249b48175f45ed
```

La firma SHA para la versión ARM del agente es:

```
sha256:5480b70cf48435e276c21789c61280cfada24e17701ede6386e5d82088bc41ca
```

Puede utilizar el SHA para identificar la versión del agente. Ejecute el siguiente comando para ver la firma SHA del agente:

```
$ docker inspect public.ecr.aws/codebuild/local-builds:latest
```

Ejecutar el agente de CodeBuild

Para ejecutar el agente de CodeBuild

1. Cambie al directorio que contiene el origen del proyecto de compilación.

2. Descarga de `codebuild_build.sh` que `esscript`:

```
$ wget https://raw.githubusercontent.com/aws/aws-codebuild-docker-images/master/local_builds/codebuild_build.sh
$ chmod +x codebuild_build.sh
```

3. Ejecute `codebuild_build.sh` script y especifique la imagen del contenedor y el directorio de salida.

Para ejecutar una compilación x86_64, ejecute el comando siguiente:

```
$ ./codebuild_build.sh -i <container-image> -a <output directory>
```

Para ejecutar una compilación de ARM, ejecute el comando siguiente:

```
$ ./codebuild_build.sh -i <container-image> -a <output directory> -l public.ecr.aws/codebuild/local-builds:aarch64
```

Reemplazar `<container-image>` con el nombre de la imagen del contenedor, como `aws/codebuild/standard:5.0` o `public.ecr.aws/codebuild/amazonlinux2-x86_64-standard:3.0`.

El script inicia la imagen de compilación y la ejecuta en el proyecto en el directorio actual. Para especificar la ubicación del proyecto de compilación, añada la `-s <build project directory>` opción para el comando de script.

Recepción de notificaciones sobre nuevas versiones del agente de CodeBuild

Puede suscribirse a las notificaciones de Amazon SNS para recibir una notificación cuando se produzca una nueva versión del AWS CodeBuild agente se liberan.

Para suscribirse al agente de CodeBuild

1. Abra la consola de Amazon SNS en <https://console.aws.amazon.com/sns/v3/home>.
2. En la barra de navegación, si aún no está seleccionada, cambie la AWS Región a EE.UU. Este (Norte de Virginia). Debe seleccionar esto AWS Región porque las notificaciones de Amazon SNS a las que se va a suscribir se crean en esa región.
3. En el panel de navegación, seleccione Subscriptions.
4. Seleccione Create subscription.
5. En Crear suscripción, realice una de las siguientes opciones:

- a. En Topic ARN, use el siguiente nombre de recurso de Amazon (ARN):

```
arn:aws:sns:us-east-1:850632864840:AWS-CodeBuild-Local-Agent-Updates
```

- b. En Protocol (Protocolo), elija Email (Correo electrónico) o SMS.
- c. En Endpoint (Punto de enlace), elija dónde desea recibir las notificaciones (por correo electrónico o SMS). Escriba un correo electrónico, una dirección o un número de teléfono, incluido el código de área.
- d. Elija Create subscription (Crear suscripción).
- e. Seleccione Email para recibir un correo electrónico donde se solicita que confirme la suscripción. Siga las instrucciones del mensaje para completar la suscripción.

Si ya no desea recibir estas notificaciones, utilice el siguiente procedimiento para cancelar la suscripción.

Para cancelar la suscripción a las notificaciones del agente de CodeBuild

1. Abra la consola de Amazon SNS en <https://console.aws.amazon.com/sns/v3/home>.
2. En el panel de navegación, seleccione Subscriptions.
3. Elija la suscripción y, en Actions (Acciones), elija Delete subscriptions (Eliminar suscripciones). Cuando se le pida confirmación, elija Delete.

Usar AWS CodeBuild en Amazon Virtual Private Cloud

Normalmente, AWS CodeBuild no puede obtener acceso a los recursos de una VPC. Para habilitar el acceso, debe proporcionar información de configuración específica de la VPC adicional en la configuración del proyecto de CodeBuild. Esta información incluye el ID de la VPC, los identificadores de subred de la VPC y los identificadores de grupo de seguridad de la VPC. Las compilaciones habilitadas para VPC pueden acceder a los recursos que se encuentran dentro de la VPC. Para obtener más información acerca de cómo configurar una VPC en Amazon VPC, consulte la [Guía del usuario de Amazon VPC](#).

Note

La conectividad con la VPC desde CodeBuild no es compatible con Windows.

Temas

- [Casos de uso \(p. 180\)](#)
- [Permitir el acceso a Amazon VPC en sus proyectos de CodeBuild \(p. 180\)](#)
- [Prácticas recomendadas para las VPC \(p. 181\)](#)
- [Solución de problemas con la configuración de la VPC \(p. 182\)](#)
- [Uso de puntos de enlace de la VPC \(p. 182\)](#)
- [Plantilla de VPC de AWS CloudFormation \(p. 184\)](#)
- [Uso de AWS CodeBuild con un servidor proxy \(p. 188\)](#)

Casos de uso

La conectividad con la VPC desde las compilaciones de AWS CodeBuild permite:

- Ejecutar pruebas de integración desde la compilación con los datos de una base de datos de Amazon RDS que esté aislada en una subred privada.
- Consultar datos en un clúster de Amazon ElastiCache directamente desde las pruebas.
- Interactuar con los servicios web internos alojados en Amazon EC2, Amazon ECS o servicios que utilizan un Elastic Load Balancing interno.
- Recuperar las dependencias de repositorios de artefactos internos y autoalojados, como PyPI para Python, Maven para Java y npm para Node.js.
- Obtener acceso a los objetos de un bucket de S3 configurado para permitir el acceso únicamente a través de un punto de enlace de Amazon VPC.
- Consultar servicios web externos que requieren direcciones IP fijas a través de la dirección IP elástica de la gateway NAT o la instancia NAT asociada a la subred.

Sus compilaciones pueden tener acceso a cualquier recurso alojado en su VPC.

Permitir el acceso a Amazon VPC en sus proyectos de CodeBuild

Incluya estos ajustes en la configuración de su VPC:

- Para `Id` de la VPC, elija el ID de la VPC que usa CodeBuild.
- Para `Subredes`, elija una subred privada con traducción de NAT que incluya o tenga rutas a los recursos utilizados por CodeBuild.
- Para `Grupos de seguridad`, elija los grupos de seguridad que usa CodeBuild para permitir el acceso a los recursos de las VPC.

Si desea utilizar la consola para crear un proyecto de compilación, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198). Cuando crea o cambia su proyecto CodeBuild, en `VPC`, elija el ID de la VPC, las subredes y los grupos de seguridad.

Si desea utilizar AWS CLI para crear un proyecto de compilación, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209). Si utiliza la AWS CLI con CodeBuild, la función de servicio utilizada por CodeBuild para interactuar con los servicios en nombre del usuario de IAM debe tener asociada una política. Para obtener información, consulte [Permitir acceso a CodeBuild a AWS servicios necesarios para crear una interfaz de red de VPC](#) (p. 375).

El objeto `vpcConfig` debe incluir su `vpcId`, `securityGroupIds` y `subnets`.

- `vpcId`: Obligatorio. El ID de la VPC que usa CodeBuild. Ejecute este comando para obtener una lista de todos los ID de Amazon VPC de su región:

```
aws ec2 describe-vpcs
```

- `Subredes`: Obligatorio. Los ID de subred que incluyen recursos utilizados por CodeBuild. Ejecute este comando para obtener estos identificadores:

```
aws ec2 describe-subnets --filters "Name=vpc-id,Values=<vpc-id>" --region us-east-1
```

Note

Reemplace `us-east-1` por la región.

- `securityGroupIds`: Obligatorio. Los ID de grupo de seguridad utilizados por CodeBuild para permitir el acceso a los recursos de las VPC. Ejecute este comando para obtener estos identificadores:

```
aws ec2 describe-security-groups --filters "Name=vpc-id,Values=<vpc-id>" --region us-east-1
```

Note

Reemplace `us-east-1` por la región.

Prácticas recomendadas para las VPC

Utilice esta lista de comprobación cuando configure una VPC para que funcione con CodeBuild.

- Configure la VPC con subredes públicas y privadas y una gateway NAT. La gateway NAT debe residir en una subred pública. Para obtener más información, consulte [VPC son subredes públicas y privadas \(NAT\)](#) en la Guía del usuario de Amazon VPC.

Important

Necesita una gateway NAT o una instancia NAT para utilizar CodeBuild con la VPC de modo que CodeBuild pueda tener acceso a los puntos de enlace públicos (por ejemplo, para ejecutar

comandos de la CLI durante las compilaciones). No puede utilizar la gateway de Internet en lugar de una gateway NAT o instancia NAT porque CodeBuild no permite asignar direcciones IP elásticas a las interfaces de red que crea, y Amazon EC2 no permite la asignación automática de una dirección IP pública para las interfaces de red creadas fuera de Amazon EC2. Lance una instancia.

- Incluya varias zonas de disponibilidad con su VPC.
- Asegúrese de que los grupos de seguridad no permitan el tráfico de entrada de sus compilaciones. CodeBuild no tiene requisitos específicos para el tráfico saliente, pero debe permitir el acceso a los recursos de Internet necesarios para su compilación, como GitHub o Amazon S3.

Para obtener más información, consulte [Reglas del grupo de seguridad](#) en la Guía del usuario de Amazon VPC.

- Configure subredes independientes para sus compilaciones.
- Cuando configure sus proyectos de CodeBuild para tener acceso a la VPC, elija únicamente subredes privadas.

Para obtener más información acerca de cómo configurar una VPC en Amazon VPC, consulte la [Guía del usuario de Amazon VPC](#).

Para obtener más información acerca del uso de AWS CloudFormation para configurar una VPC de modo que utilice la característica de CodeBuild VPC, consulte la [Plantilla de VPC de AWS CloudFormation](#) (p. 184).

Solución de problemas con la configuración de la VPC

Utilice la información que aparece en el mensaje de error para ayudarle a identificar, diagnosticar y resolver los problemas.

A continuación, se incluyen varias recomendaciones para ayudarle a solucionar un error de CodeBuild de `VPC de:Build does not have internet connectivity. Please check subnet network configuration.`

1. [Asegúrese de que la gateway de Internet está conectada a la VPC.](#)
2. [Asegúrese de que la tabla de enrutamiento de su subred pública apunta a la gateway de Internet.](#)
3. [Asegúrese de que las ACL de red permiten el tráfico.](#)
4. [Asegúrese de que los grupos de seguridad permiten el tráfico.](#)
5. [Solucione los problemas de la gateway NAT.](#)
6. [Asegúrese de que la tabla de enrutamiento de las subredes privadas apunta a la gateway NAT.](#)
7. [Asegúrese de que el rol de servicio utilizado por CodeBuild para interactuar con los servicios en nombre del usuario de IAM tiene los permisos de \[Esta política\]\(#\). Para obtener más información, consulte \[Cree un rol de servicio de CodeBuild\]\(#\) \(p. 396\).](#)

Si faltan permisos en CodeBuild, es posible que aparezca el error siguiente: `Unexpected EC2 error: UnauthorizedOperation`. Este error puede producirse si CodeBuild no tiene los permisos de Amazon EC2 necesarios para trabajar con una VPC.

Uso de puntos de enlace de la VPC

Puede mejorar la seguridad de sus compilaciones si configura AWS CodeBuild para usar una interfaz de punto de enlace de VPC de interfaz. Los puntos de enlace de la interfaz funcionan con PrivateLink, una

tecnología que puede utilizar para obtener acceso de forma privada a Amazon EC2 y CodeBuild utilizando direcciones IP privadas. PrivateLink restringe todo el tráfico de red entre las instancias administradas, CodeBuild y Amazon EC2 a la red de Amazon. Las instancias administradas no tienen acceso a Internet. Asimismo, no necesita una gateway de Internet, ni un dispositivo NAT, ni una gateway privada virtual. No es necesario configurar PrivateLink, pero es recomendable. Para obtener más información sobre PrivateLink y los puntos de enlace de la VPC, consulte [Acceso a servicios a través de AWS PrivateLink](#) en la Amazon VPC User Guide.

Antes de crear los puntos de enlace de la VPC

Antes de configurar los puntos de enlace de la VPC para AWS CodeBuild, debe conocer las siguientes restricciones y limitaciones.

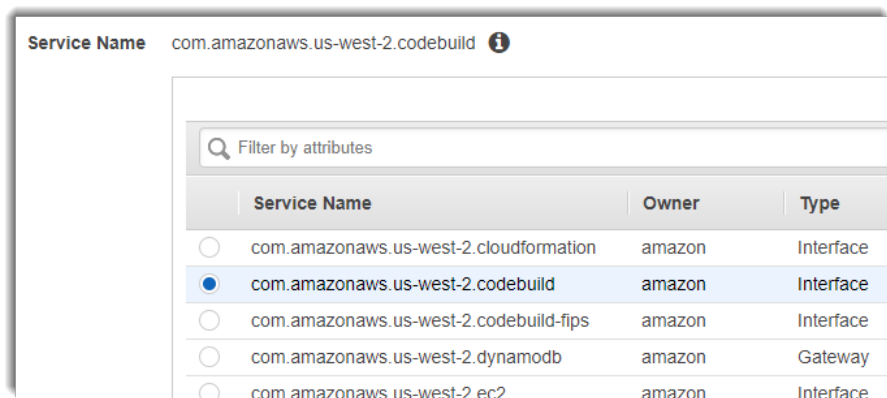
Note

Usar un [gateway NAT](#) si quieres usar CodeBuild con AWS servicios que no admiten conexiones de Amazon VPC PrivateLink.

- Los puntos de enlace de la VPC solo admiten DNS proporcionadas por Amazon a través de Amazon Route 53. Si desea utilizar su propio DNS, puede utilizar el enrutamiento de DNS condicional. Para obtener más información, consulte [Conjuntos de opciones de DHCP](#) en la Amazon VPC User Guide.
- En la actualidad, los puntos de enlace de la VPC no admiten las solicitudes entre regiones. Asegúrese de crear su punto de enlace en el mismo AWS Región como cualquier bucket de S3 que almacenan la entrada y la salida de la compilación. Puede utilizar la consola de Amazon S3 o la [get-bucket-location](#) para buscar la ubicación de su bucket. Utilice un punto de enlace de Amazon S3 específico de la región para obtener acceso al bucket (por ejemplo, `mybucket.s3-us-west-2.amazonaws.com`). Para obtener más información acerca de los puntos de enlace específicos de regiones para Amazon S3, consulte [Amazon Simple Storage Service](#) en la Referencia general de Amazon Web Services. Si utiliza el AWS CLI para realizar solicitudes a Amazon S3, establezca la región predeterminada en la misma región en la que se creó el bucket o utilice la `--region` parámetro en tus solicitudes.

Creación de puntos de enlace de la VPC para CodeBuild

Siga las instrucciones de [Creación de un punto de enlace de interfaz](#) para crear el punto de enlace `com.amazonaws.región.codebuild`. Este es un punto de enlace de la VPC para AWS CodeBuild.



región representa el identificador de región de un AWS Región compatible con CodeBuild, como `us-east-2` para la región EE. UU. Este (Ohio). Para obtener una lista de los admitidos AWS Regiones, consulte [CodeBuild](#) en la AWS Referencia general de. El punto de enlace ya contiene la región que

especificó al iniciar sesión en AWS. Si cambia la región, el punto de enlace de la VPC se actualizará en consecuencia.

Creación de una política de puntos de enlace de la VPC para CodeBuild

Puede crear una política para los puntos de enlace de la VPC de Amazon para AWS CodeBuild en la que puede especificar:

- La entidad principal que puede realizar acciones.
- Las acciones que se pueden realizar.
- Los recursos en los que se pueden realizar acciones.

La siguiente política de ejemplo especifica que todas las entidades principales solo pueden iniciar y consultar compilaciones para el proyecto `project-name`.

```
{
  "Statement": [
    {
      "Action": [
        "codebuild:ListBuildsForProject",
        "codebuild:StartBuild",
        "codebuild:BatchGetBuilds"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:codebuild:region-ID:account-ID:project/project-name",
      "Principal": "*"
    }
  ]
}
```

Para obtener más información, consulte [Control del acceso a los servicios con puntos de enlace de la VPC](#) en la guía del usuario de Amazon VPC.

Plantilla de VPC de AWS CloudFormation

AWS CloudFormation le permite crear y aprovisionar AWS Implementaciones de infraestructura de manera predecible y repetida, mediante el uso de archivos de plantilla para crear y eliminar una colección de recursos de forma conjunta como una única unidad (pila). Para obtener más información, consulte la [Guía del usuario de AWS CloudFormation](#).

A continuación, se muestra una plantilla de YAML de AWS CloudFormation para configurar una VPC que va a utilizar AWS CodeBuild. Este archivo también está disponible en [samples.zip](#).

```
Description: This template deploys a VPC, with a pair of public and private subnets spread
across two Availability Zones. It deploys an internet gateway, with a default
route on the public subnets. It deploys a pair of NAT gateways (one in each AZ),
and default routes for them in the private subnets.

Parameters:
  EnvironmentName:
    Description: An environment name that is prefixed to resource names
    Type: String

  VpcCIDR:
```

```
Description: Please enter the IP range (CIDR notation) for this VPC
Type: String
Default: 10.192.0.0/16

PublicSubnet1CIDR:
  Description: Please enter the IP range (CIDR notation) for the public subnet in the
  first Availability Zone
  Type: String
  Default: 10.192.10.0/24

PublicSubnet2CIDR:
  Description: Please enter the IP range (CIDR notation) for the public subnet in the
  second Availability Zone
  Type: String
  Default: 10.192.11.0/24

PrivateSubnet1CIDR:
  Description: Please enter the IP range (CIDR notation) for the private subnet in the
  first Availability Zone
  Type: String
  Default: 10.192.20.0/24

PrivateSubnet2CIDR:
  Description: Please enter the IP range (CIDR notation) for the private subnet in the
  second Availability Zone
  Type: String
  Default: 10.192.21.0/24

Resources:
  VPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: !Ref VpcCIDR
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGateway:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: !Ref EnvironmentName

  InternetGatewayAttachment:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      InternetGatewayId: !Ref InternetGateway
      VpcId: !Ref VPC

  PublicSubnet1:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref VPC
      AvailabilityZone: !Select [ 0, !GetAZs '' ]
      CidrBlock: !Ref PublicSubnet1CIDR
      MapPublicIpOnLaunch: true
      Tags:
        - Key: Name
          Value: !Sub ${EnvironmentName} Public Subnet (AZ1)

  PublicSubnet2:
    Type: AWS::EC2::Subnet
    Properties:
```



```
VpcId: !Ref VPC
AvailabilityZone: !Select [ 1, !GetAZs '' ]
CidrBlock: !Ref PublicSubnet2CIDR
MapPublicIpOnLaunch: true
Tags:
  - Key: Name
    Value: !Sub ${EnvironmentName} Public Subnet (AZ2)

PrivateSubnet1:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    AvailabilityZone: !Select [ 0, !GetAZs '' ]
    CidrBlock: !Ref PrivateSubnet1CIDR
    MapPublicIpOnLaunch: false
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Subnet (AZ1)

PrivateSubnet2:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref VPC
    AvailabilityZone: !Select [ 1, !GetAZs '' ]
    CidrBlock: !Ref PrivateSubnet2CIDR
    MapPublicIpOnLaunch: false
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Subnet (AZ2)

NatGateway1EIP:
  Type: AWS::EC2::EIP
  DependsOn: InternetGatewayAttachment
  Properties:
    Domain: vpc

NatGateway2EIP:
  Type: AWS::EC2::EIP
  DependsOn: InternetGatewayAttachment
  Properties:
    Domain: vpc

NatGateway1:
  Type: AWS::EC2::NatGateway
  Properties:
    AllocationId: !GetAtt NatGateway1EIP.AllocationId
    SubnetId: !Ref PublicSubnet1

NatGateway2:
  Type: AWS::EC2::NatGateway
  Properties:
    AllocationId: !GetAtt NatGateway2EIP.AllocationId
    SubnetId: !Ref PublicSubnet2

PublicRouteTable:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Public Routes

DefaultPublicRoute:
  Type: AWS::EC2::Route
  DependsOn: InternetGatewayAttachment
  Properties:
```

```
RouteTableId: !Ref PublicRouteTable
DestinationCidrBlock: 0.0.0.0/0
GatewayId: !Ref InternetGateway

PublicSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PublicRouteTable
    SubnetId: !Ref PublicSubnet1

PublicSubnet2RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PublicRouteTable
    SubnetId: !Ref PublicSubnet2

PrivateRouteTable1:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Routes (AZ1)

DefaultPrivateRoute1:
  Type: AWS::EC2::Route
  Properties:
    RouteTableId: !Ref PrivateRouteTable1
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NatGateway1

PrivateSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PrivateRouteTable1
    SubnetId: !Ref PrivateSubnet1

PrivateRouteTable2:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref VPC
    Tags:
      - Key: Name
        Value: !Sub ${EnvironmentName} Private Routes (AZ2)

DefaultPrivateRoute2:
  Type: AWS::EC2::Route
  Properties:
    RouteTableId: !Ref PrivateRouteTable2
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NatGateway2

PrivateSubnet2RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref PrivateRouteTable2
    SubnetId: !Ref PrivateSubnet2

NoIngressSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupName: "no-ingress-sg"
    GroupDescription: "Security group with no ingress rule"
    VpcId: !Ref VPC
```

```
Outputs:
  VPC:
    Description: A reference to the created VPC
    Value: !Ref VPC

  PublicSubnets:
    Description: A list of the public subnets
    Value: !Join [ ",", [ !Ref PublicSubnet1, !Ref PublicSubnet2 ] ]

  PrivateSubnets:
    Description: A list of the private subnets
    Value: !Join [ ",", [ !Ref PrivateSubnet1, !Ref PrivateSubnet2 ] ]

  PublicSubnet1:
    Description: A reference to the public subnet in the 1st Availability Zone
    Value: !Ref PublicSubnet1

  PublicSubnet2:
    Description: A reference to the public subnet in the 2nd Availability Zone
    Value: !Ref PublicSubnet2

  PrivateSubnet1:
    Description: A reference to the private subnet in the 1st Availability Zone
    Value: !Ref PrivateSubnet1

  PrivateSubnet2:
    Description: A reference to the private subnet in the 2nd Availability Zone
    Value: !Ref PrivateSubnet2

  NoIngressSecurityGroup:
    Description: Security group with no ingress rule
    Value: !Ref NoIngressSecurityGroup
```

Uso de AWS CodeBuild con un servidor proxy

Puede utilizar AWS CodeBuild con un servidor proxy para regular el tráfico HTTP y HTTPS de entrada y salida de Internet. Para ejecutar CodeBuild con un servidor proxy, debe instalar un servidor proxy en una subred pública y CodeBuild en una subred privada de una VPC.

Existen dos casos de uso principales en los que se ejecuta CodeBuild en un servidor proxy:

- Elimina la necesidad de utilizar un gateway NAT o una instancia NAT en la VPC.
- Permite especificar las URL a las que las instancias del servidor proxy pueden obtener acceso y a las que el servidor proxy deniega el acceso.

Puede utilizar CodeBuild con dos tipos de servidores proxy. En ambos casos, el servidor proxy se ejecuta en una subred pública, mientras que CodeBuild se ejecuta en una subred privada.

- Proxy explícito: Si utiliza un servidor proxy explícito, debe configurar `NO_PROXY`, `HTTP_PROXY`, y `HTTPS_PROXY` en CodeBuild en el nivel del proyecto. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación en AWS CodeBuild \(p. 254\)](#) y [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#).
- Proxy transparente: Si utiliza un servidor proxy transparente, no se necesita ninguna configuración especial.

Temas

- [Componentes necesarios para ejecutar CodeBuild en un servidor proxy \(p. 189\)](#)

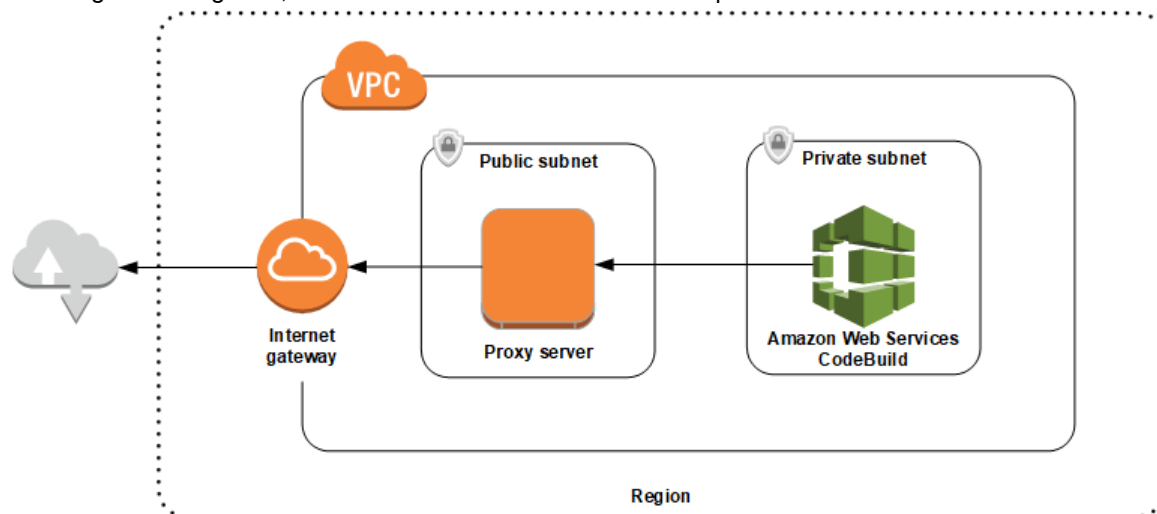
- [Ejecución de CodeBuild en un servidor proxy explícito \(p. 191\)](#)
- [Ejecución de CodeBuild en un servidor proxy transparente \(p. 194\)](#)
- [Ejecución de un administrador de paquetes y otras herramientas en un servidor proxy \(p. 195\)](#)

Componentes necesarios para ejecutar CodeBuild en un servidor proxy

Para ejecutar AWS CodeBuild en un servidor proxy transparente o explícito, necesita estos componentes:

- Una VPC.
- Una subred pública en la VPC para el servidor proxy.
- Una subred privada en la VPC para CodeBuild.
- Una gateway de Internet que permita la comunicación entre la VPC e Internet.

En el siguiente diagrama, se muestran cómo interactúan los componentes.



Configuración de una VPC, varias subredes y una gateway de red

A continuación, se presentan los pasos necesarios para ejecutar AWS CodeBuild en un servidor proxy transparente o explícito.

1. Cree una VPC. Para obtener más información, consulte [Creación de una VPC](#) en la Guía del usuario de Amazon VPC.
2. Cree dos subredes en la VPC. Una será una subred pública llamada `Public Subnet` en la que se ejecutará el servidor proxy. La otra será una subred privada llamada `Private Subnet` en la que se ejecuta CodeBuild.

Para obtener más información, consulte [Creación de una subred en la VPC](#).

3. Cree un puerto de enlace a Internet y vincúlelo a su VPC. Para obtener más información, consulte [Creación y asociación de una gateway de Internet](#).
4. Añada una regla a la tabla de ruteo predeterminada que dirija el tráfico saliente desde la VPC (0.0.0.0/0) a la gateway de Internet. Para obtener más información, consulte [Adición y eliminación de rutas de una tabla de ruteo](#).

5. Añada una regla al grupo de seguridad predeterminado de la VPC que permita la entrada de tráfico SSH (TCP 22) procedente de la VPC (0.0.0.0/0).
 6. Siga las instrucciones en [Lanzamiento de una instancia con el asistente de lanzamiento de instancias](#) en la Guía del usuario de Amazon EC2 para lanzar una instancia de Amazon Linux. Cuando el asistente esté en ejecución, seleccione las siguientes opciones:
 - En **Página Choose an Instance Type**, seleccione una imagen de máquina de Amazon Linux (AMI) de.
 - En **Subnet (Subred)**, seleccione la subred pública que creó anteriormente en este tema. Si utilizó el nombre recomendado, se llamará **Public Subnet**.
 - En **Auto-assign Public IP**, elija **Enable**.
 - En la página **Configure Security Group (Configurar grupo de seguridad)**, en **Assign a security group (Asignar un grupo de seguridad)**, seleccione **Select an existing security group (Seleccionar un grupo de seguridad existente)**. A continuación, seleccione el grupo de seguridad predeterminado.
 - Cuando haga clic en **Launch (Lanzar)**, seleccione un par de claves existente o cree uno.
- Seleccione la configuración predeterminada de todas las demás opciones.
7. Una vez que la instancia EC2 esté en ejecución, deshabilite las comprobaciones de origen o destino. Para obtener más información, consulte [Deshabilitación de comprobaciones de origen/destino](#) en la Guía del usuario de Amazon VPC.
 8. Cree una tabla de ruteo en la VPC. Añada una regla a la tabla de ruteo que dirija el tráfico destinado a Internet al servidor proxy. Asocie esta tabla de ruteo con la subred privada. Esta operación es necesaria para que las solicitudes de salida de las instancias de la subred privada en la que se ejecuta CodeBuild se dirijan siempre a través del servidor proxy.

Instalación y configuración de un servidor proxy

Existen muchos servidores proxy entre los que elegir. Aquí, vamos a utilizar un servidor proxy de código abierto, Squid, para mostrar cómo se ejecuta AWS CodeBuild en un servidor proxy. Puede emplear estos mismos conceptos con otros servidores proxy.

Para instalar Squid, utilice un repositorio yum ejecutando los siguientes comandos:

```
sudo yum update -y
sudo yum install -y squid
```

Después de instalar Squid, edite el archivo `squid.conf` siguiendo las instrucciones que se indican más adelante en este tema.

Configuración de Squid para el tráfico HTTPS

En HTTPS, el tráfico HTTP se encapsula en una conexión Transport Layer Security (TLS). Squid utiliza una función llamada [SslPeekAndSplice](#) para recuperar la indicación de nombre de servidor (SNI) del inicio de TLS que contiene el host de Internet solicitado. Esta operación es necesaria, ya que Squid no necesita descifrar el tráfico HTTPS. Para habilitar `SslPeekAndSplice`, Squid necesita un certificado. Cree este certificado con OpenSSL:

```
sudo mkdir /etc/squid/ssl
cd /etc/squid/ssl
sudo openssl genrsa -out squid.key 2048
sudo openssl req -new -key squid.key -out squid.csr -subj "/C=XX/ST=XX/L=squid/O=squid/CN=squid"
sudo openssl x509 -req -days 3650 -in squid.csr -signkey squid.key -out squid.crt
sudo cat squid.key squid.crt | sudo tee squid.pem
```

Note

En el caso de HTTP, Squid no requiere configuración. En todos los mensajes de solicitud HTTP/1.1, puede recuperar el campo del encabezado de host, donde se especifica el host de Internet que se está solicitando.

Ejecución de CodeBuild en un servidor proxy explícito

Temas

- [Configuración de Squid como un servidor proxy explícito \(p. 191\)](#)
- [Crear un proyecto de CodeBuild \(p. 192\)](#)
- [Archivo squid.conf de ejemplo de un servidor proxy explícito \(p. 192\)](#)

Para ejecutar AWS CodeBuild en un servidor proxy explícito, debe configurar el servidor proxy de forma que permita o deniegue el tráfico dirigido a sitios externos o el tráfico procedente de estos sitios y debe configurar las variables de entorno `HTTPS_PROXY` y `HTTP_PROXY`.

Configuración de Squid como un servidor proxy explícito

Si desea configurar el servidor proxy de Squid para que sea explícito, debe realizar las siguientes modificaciones en el archivo `/etc/squid/squid.conf`:

- Elimine las siguientes reglas de lista de control de acceso (ACL) predeterminadas.

```
acl localnet src 10.0.0.0/8
acl localnet src 172.16.0.0/12
acl localnet src 192.168.0.0/16
acl localnet src fc00::/7
acl localnet src fe80::/10
```

Añada lo siguiente en lugar de las reglas de ACL predeterminadas que ha eliminado. La primera línea admite las solicitudes procedentes de la VPC. Las siguientes dos líneas conceden al servidor proxy acceso a una URL de destino que AWS CodeBuild podría estar utilizando. Modifique la expresión regular de la última línea para especificar buckets de S3 o un repositorio de CodeCommit en una AWS Región. Por ejemplo:

- Si el origen es Amazon S3, use el comando `acl download_src dstdomain_regex .*s3\.us-west-1\.amazonaws\.com` Para conceder acceso a los buckets de S3 en la `us-west-1` Región.
- Si la fuente es AWS CodeCommit, use `git-codecommit.<your-region>.amazonaws.com` Para agregar una AWS Región a una lista de permitidos.

```
acl localnet src 10.1.0.0/16 #Only allow requests from within the VPC
acl allowed_sites dstdomain .github.com #Allows to download source from GitHub
acl allowed_sites dstdomain .bitbucket.com #Allows to download source from Bitbucket
acl download_src dstdomain_regex .*\.amazonaws\.com #Allows to download source from Amazon S3 or CodeCommit
```

- Reemplace `http_access allow localnet` por lo siguiente:

```
http_access allow localnet allowed_sites
http_access allow localnet download_src
```

- Si desea que la compilación cargue registros y artefactos, realice una de las siguientes acciones:
 1. Antes de la instrucción `http_access deny all`, inserte las siguientes instrucciones. Permiten a CodeBuild acceder a CloudWatch y Amazon S3. Se requiere acceso a CloudWatch para que

CodeBuild pueda crear registros de CloudWatch. Para poder cargar artefactos y el almacenamiento en caché de Amazon S3 es necesario acceder a Amazon S3.

```
https_port 3130 cert=/etc/squid/ssl/squid.pem ssl-bump intercept
acl SSL_port port 443
http_access allow SSL_port
acl allowed_https_sites ssl::server_name .amazonaws.com
acl step1 at_step SslBump1
acl step2 at_step SslBump2
acl step3 at_step SslBump3
ssl_bump peek step1 all
ssl_bump peek step2 allowed_https_sites
ssl_bump splice step3 allowed_https_sites
ssl_bump terminate step2 all
```

- Después de guardarsquid.confEn, ejecute el siguiente comando:

```
sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 3130
sudo service squid restart
```

2. Añada proxy al archivo buildspec. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

```
version: 0.2
proxy:
  upload-artifacts: yes
  logs: yes
phases:
  build:
    commands:
      - command
```

Note

Si recibe un error de tiempo de espera agotado RequestError, consulte [Error de tiempo de espera agotado de RequestError al ejecutar CodeBuild en un servidor proxy \(p. 440\)](#).

Para obtener más información, consulte [Archivo squid.conf de ejemplo de un servidor proxy explícito \(p. 192\)](#) más adelante en este tema.

Crear un proyecto de CodeBuild

Para ejecutar AWS CodeBuild con el servidor proxy explícito, defina las variables de entorno HTTPS_PROXY y HTTP_PROXY con la dirección IP privada de la instancia EC2 que creó para el servidor proxy y el puerto 3128 en el nivel del proyecto. La dirección IP privada será parecida a la siguiente: `http://your-ec2-private-ip-address:3128`. Para obtener más información, consulte [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#) y [Cambiar la configuración de un proyecto de compilación en AWS CodeBuild \(p. 254\)](#).

Utilice el comando siguiente para ver el registro de acceso del proxy de Squid:

```
sudo tail -f /var/log/squid/access.log
```

Archivo squid.conf de ejemplo de un servidor proxy explícito

A continuación, se incluye un ejemplo de un archivo squid.conf configurado para un servidor proxy explícito.

```
acl localnet src 10.0.0.0/16 #Only allow requests from within the VPC
```

```
# add all URLs to be whitelisted for download source and commands to be run in build
environment
acl allowed_sites dstdomain .github.com      #Allows to download source from github
acl allowed_sites dstdomain .bitbucket.com   #Allows to download source from bitbucket
acl allowed_sites dstdomain ppa.launchpad.net #Allows to run apt-get in build environment
acl download_src dstdom_regex .*\.amazonaws\.com #Allows to download source from S3 or
CodeCommit
acl SSL_ports port 443
acl Safe_ports port 80  # http
acl Safe_ports port 21  # ftp
acl Safe_ports port 443 # https
acl Safe_ports port 70  # gopher
acl Safe_ports port 210  # wais
acl Safe_ports port 1025-65535 # unregistered ports
acl Safe_ports port 280  # http-mgmt
acl Safe_ports port 488  # gss-http
acl Safe_ports port 591  # filemaker
acl Safe_ports port 777  # multiling http
acl CONNECT method CONNECT
#
# Recommended minimum Access Permission configuration:
#
# Deny requests to certain unsafe ports
http_access deny !Safe_ports
# Deny CONNECT to other than secure SSL ports
http_access deny CONNECT !SSL_ports
# Only allow cachemgr access from localhost
http_access allow localhost manager
http_access deny manager
# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
http_access allow localnet allowed_sites
http_access allow localnet download_src
http_access allow localhost
# Add this for CodeBuild to access CWL end point, caching and upload artifacts S3 bucket
end point
https_port 3130 cert=/etc/squid/ssl/squid.pem ssl-bump intercept
acl SSL_port port 443
http_access allow SSL_port
acl allowed_https_sites ssl::server_name .amazonaws.com
acl step1 at_step SslBump1
acl step2 at_step SslBump2
acl step3 at_step SslBump3
ssl_bump peek step1 all
ssl_bump peek step2 allowed_https_sites
ssl_bump splice step3 allowed_https_sites
ssl_bump terminate step2 all
# And finally deny all other access to this proxy
http_access deny all
# Squid normally listens to port 3128
http_port 3128
# Uncomment and adjust the following to add a disk cache directory.
#cache_dir ufs /var/spool/squid 100 16 256
# Leave coredumps in the first cache dir
coredump_dir /var/spool/squid
#
# Add any of your own refresh_pattern entries above these.
#
```



```
refresh_pattern ^ftp: 1440 20% 10080
refresh_pattern ^gopher: 1440 0% 1440
refresh_pattern -i (/cgi-bin/|\?) 0 0% 0
refresh_pattern . 0 20% 4320
```

Ejecución de CodeBuild en un servidor proxy transparente

Para ejecutar AWS CodeBuild en un servidor proxy transparente, debe configurar el servidor proxy para que tenga acceso a los sitios web y los dominios con los que interactúa.

Configuración de Squid como un servidor proxy transparente

Si desea configurar un servidor proxy para que sea transparente, debe concederle acceso a los dominios y sitios web que desee. Para ejecutar AWS CodeBuild con un servidor proxy transparente, debe concederle acceso a `amazonaws.com`. También debe concederle acceso a otros sitios web que utiliza CodeBuild. Los valores variarán en función de cómo cree los proyectos de CodeBuild. Existen sitios web de ejemplo con repositorios, como GitHub, Bitbucket, Yum y Maven. Para conceder a Squid acceso a determinados dominios y sitios web, utilice un comando similar al siguiente para actualizar el archivo `squid.conf`. Este comando de ejemplo concede acceso a `amazonaws.com`, `github.com` y `bitbucket.com`. Puede editar este ejemplo para conceder acceso a otros sitios web.

```
cat | sudo tee /etc/squid/squid.conf #EOF
visible_hostname squid
#Handling HTTP requests
http_port 3129 intercept
acl allowed_http_sites dstdomain .amazonaws.com
#acl allowed_http_sites dstdomain domain_name [uncomment this line to add another domain]
http_access allow allowed_http_sites
#Handling HTTPS requests
https_port 3130 cert=/etc/squid/ssl/squid.pem ssl-bump intercept
acl SSL_port port 443
http_access allow SSL_port
acl allowed_https_sites ssl::server_name .amazonaws.com
acl allowed_https_sites ssl::server_name .github.com
acl allowed_https_sites ssl::server_name .bitbucket.com
#acl allowed_https_sites ssl::server_name [uncomment this line to add another website]
acl step1 at_step SslBump1
acl step2 at_step SslBump2
acl step3 at_step SslBump3
ssl_bump peek step1 all
ssl_bump peek step2 allowed_https_sites
ssl_bump splice step3 allowed_https_sites
ssl_bump terminate step2 all
http_access deny all
EOF
```

Las solicitudes entrantes procedentes de las instancias de la subred privada deben redirigirse a los puertos de Squid. Squid escucha el tráfico HTTP en el puerto 3129 (en lugar de en el 80) y el tráfico HTTPS en el puerto 3130 (en lugar de en el 443). Utilice el comando `iptables` para direccionar el tráfico:

```
sudo iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 3129
sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 3130
sudo service iptables save
sudo service squid start
```

Crear un proyecto de CodeBuild

Una vez configurado el servidor proxy, podrá utilizarlo con AWS CodeBuild en una subred privada sin necesidad de configurar nada más. Todas las solicitudes HTTP y HTTPS pasan por el servidor proxy público. Utilice el comando siguiente para ver el registro de acceso del proxy de Squid:

```
sudo tail -f /var/log/squid/access.log
```

Ejecución de un administrador de paquetes y otras herramientas en un servidor proxy

Para ejecutar una herramienta, como un administrador de paquetes, en un servidor proxy

1. Incorpore la herramienta a la lista de permitidos en su servidor proxy agregando instrucciones en el archivo `squid.conf`.
2. Añada una línea al archivo `buildspec` que apunte al punto de enlace privado del servidor proxy.

En los siguientes ejemplos, se muestra cómo hacerlo con `apt-get`, `curl` y `maven`. Si utiliza una herramienta diferente, se aplicarán los mismos principios. Añádelo a una lista de permitidos en `elsquid.conf` Para que CodeBuild conozca el punto de enlace del servidor proxy, agregue un comando al archivo `buildspec` para que codeBuild conozca el punto de enlace del servidor proxy.

Para ejecutar **apt-get** en un servidor proxy

1. Agregue las siguientes instrucciones en el archivo `squid.conf` para incluir `apt-get` en la lista de permitidos del servidor proxy. Las tres primeras líneas permiten `apt-get` para ejecutarse en el entorno de compilación.

```
acl allowed_sites dstdomain ppa.launchpad.net # Required for apt-get to run in the
build environment
acl apt_get dstdom_regex .*\.launchpad.net # Required for CodeBuild to run apt-get in
the build environment
acl apt_get dstdom_regex .*\.ubuntu.com # Required for CodeBuild to run apt-get in
the build environment
http_access allow localnet allowed_sites
http_access allow localnet apt_get
```

2. Añada la siguiente instrucción en el archivo `buildspec` para que los comandos `apt-get` busquen la configuración del proxy en `/etc/apt/apt.conf.d/00proxy`.

```
echo 'Acquire::http::Proxy "http://<private-ip-of-proxy-server>:3128";
Acquire::https::Proxy "http://<private-ip-of-proxy-server>:3128"; Acquire::ftp::Proxy
"http://<private-ip-of-proxy-server>:3128";' > /etc/apt/apt.conf.d/00proxy
```

Para ejecutar **curl** en un servidor proxy

1. Agregue lo siguiente al archivo `squid.conf` para incluir `curl` en la lista de permitidos del entorno de compilación.

```
acl allowed_sites dstdomain ppa.launchpad.net # Required to run apt-get in the build
environment
acl allowed_sites dstdomain google.com # Required for access to a webiste. This example
uses www.google.com.
```

```
http_access allow localnet allowed_sites  
http_access allow localnet apt_get
```

2. Añada la siguiente instrucción al archivo `buildspec`, de modo que `curl` utilice el servidor proxy privado para obtener acceso al sitio web que añadió a `squid.conf`. En este ejemplo, el sitio web es `google.com`.

```
curl -x <private-ip-of-proxy-server>:3128 https://www.google.com
```

Para ejecutar **maven** en un servidor proxy

1. Agregue lo siguiente al archivo `squid.conf` para incluir maven en la lista de permitidos del entorno de compilación.

```
acl allowed_sites dstdomain ppa.launchpad.net # Required to run apt-get in the build  
environment  
acl maven dstdom_regex .*\.maven.org # Allows access to the maven repository in the  
build environment  
http_access allow localnet allowed_sites  
http_access allow localnet maven
```

2. Añada la siguiente instrucción al archivo `buildspec`.

```
maven clean install -DproxySet=true -DproxyHost=<private-ip-of-proxy-server> -  
DproxyPort=3128
```

Trabajar con proyectos de compilación y compilaciones en AWS CodeBuild

Para comenzar, siga los pasos de [Creación de un proyecto de compilación \(p. 197\)](#). Y, a continuación, siga los pasos de [Ejecutar una compilación \(p. 281\)](#). Para obtener más información sobre los proyectos de compilación y las compilaciones, consulte los siguientes temas.

Temas

- [Trabajar con proyectos de compilación \(p. 197\)](#)
- [Trabajar con compilaciones en AWS CodeBuild \(p. 280\)](#)

Trabajar con proyectos de compilación

Un proyecto de compilación incluye información sobre cómo ejecutar una compilación, incluido dónde se obtiene el código fuente, qué entorno de compilación se va a usar, qué comandos de compilación se van a ejecutar y dónde se va a almacenar la salida de la compilación.

Puede realizar estas tareas cuando trabaje con proyectos de compilación:

Temas

- [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#)
- [Creación de una regla de notificación \(p. 223\)](#)
- [Ver una lista de nombres de proyectos de compilación en AWS CodeBuild \(p. 225\)](#)
- [Ver los detalles de un proyecto de compilación en AWS CodeBuild \(p. 227\)](#)
- [Almacenamiento en caché de compilaciones de AWS CodeBuild \(p. 229\)](#)
- [Crear desencadenadores de AWS CodeBuild \(p. 233\)](#)
- [Editar desencadenadores de AWS CodeBuild \(p. 236\)](#)
- [Uso de webhooks con AWS CodeBuild \(p. 237\)](#)
- [Cambiar la configuración de un proyecto de compilación en AWS CodeBuild \(p. 254\)](#)
- [Eliminar un proyecto de compilación en AWS CodeBuild \(p. 268\)](#)
- [Uso de proyectos compartidos \(p. 269\)](#)
- [Etiquetado de proyectos en AWS CodeBuild \(p. 273\)](#)
- [Compilaciones Batch en AWS CodeBuild \(p. 276\)](#)
- [Proyectos de compilación pública en AWS CodeBuild \(p. 279\)](#)

Creación de un proyecto de compilación en AWS CodeBuild

Puede usar la consola de AWS CodeBuild, AWS CLI o los SDK de AWS para crear un proyecto de compilación.

Prerequisites

Antes de crear un proyecto de compilación, responda a las preguntas de [Planificación de una compilación](#) (p. 135).

Temas

- [Creación de un proyecto de compilación \(consola\)](#) (p. 198)
- [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209)
- [Crear un proyecto de compilación \(SDK de AWS\)](#) (p. 223)
- [Crear un proyecto de compilación \(AWS CloudFormation\)](#) (p. 223)

Creación de un proyecto de compilación (consola)

Abra el icono **AWS CodeBuild Consola** de <https://console.aws.amazon.com/codesuite/codebuild/home>.

Si aparece una página de información de CodeBuild, elija **Crear un proyecto de compilación**. De lo contrario, en el panel de navegación, expanda **Build**, elija **Proyectos de compilación** y luego seleccione **Crear un proyecto de compilación**.

Elija **Create build project** (**Crear proyecto de compilación**).

Rellene las siguientes secciones. Una vez completado, seleccione **Crear un proyecto de compilación** en la parte inferior de la página.

Secciones:

- [Configuración de proyectos](#) (p. 198)
- [Source](#) (p. 199)
- [Environment](#) (p. 203)
- [Buildspec](#) (p. 206)
- [Configuración Batch](#) (p. 206)
- [Artifacts](#) (p. 207)
- [Logs](#) (p. 209)

Configuración de proyectos

Project name (Nombre de proyecto)

Escriba un nombre para este proyecto de compilación. Los nombres de los proyectos de compilación debe ser únicos en cada cuenta de AWS.

Descripción

Introduzca una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.

Crear insignias de

(Opcional) Seleccione **Habilitar insignias** para convertir el estado de compilación del proyecto en visible e incrustable. Para obtener más información, consulte [Ejemplo de insignias de compilación](#) (p. 78).

Note

Crear insignia no se aplica si el proveedor de código fuente es Amazon S3.

Habilitar límite de compilación simultánea

(Opcional) Si desea limitar el número de compilaciones simultáneas de este proyecto, siga estos pasos:

1. `SelectRestringir` el número de compilaciones simultáneas que este proyecto puede iniciar.
2. En `Límite de compilación simultánea`, introduzca el número máximo de versiones simultáneas que están permitidas para este proyecto. Este límite no puede ser superior al límite de compilación simultáneo establecido para la cuenta. Si intenta introducir un número mayor que el límite de la cuenta, se muestra un mensaje de error.

Las nuevas versiones solo se inician si el número actual de versiones es menor o igual a este límite. Si el recuento de versiones actual cumple con este límite, las nuevas versiones se limitan y no se ejecutan.

Información adicional

(Opcional) En `Tags` (Etiquetas), escriba el nombre y el valor de las etiquetas que desea que usen los servicios compatibles de AWS. Use `Add row` para añadir una etiqueta. Puede añadir hasta 50 etiquetas.

Source

Proveedor de origen

Elegir el tipo de proveedor de código fuente. Utilice las siguientes listas para realizar las selecciones apropiadas para el proveedor de código fuente:

Note

CodeBuild no admite Bitbucket Server.

Amazon S3

Bucket

Elija el nombre del bucket de entrada que contiene el código fuente.

Clave de objeto de S3 o carpeta de S3

Introduzca el nombre del archivo ZIP o la ruta a la carpeta que contiene el código fuente. Incluya una barra diagonal (/) para descargarlo todo en el bucket de S3.

Versión de origen

Introduzca el ID de versión del objeto que representa la compilación del archivo de entrada. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

CodeCommit

Repositorio

Elegir el repositorio que desea utilizar.

Tipo de referencia

Seleccionar `Crear ramificaciones`, `Etiqueta Git`, o bien `ID de confirmación` para especificar la versión del código fuente. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Elija crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione Use Git submodules (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Bitbucket

Repositorio

SeleccionarConnect mediante OAuthConnect con una contraseña de aplicación de Bitbuckety siga las instrucciones para conectarse (o volver a conectarse) a Bitbucket.

Seleccione un repositorio público o un repositorio de su cuenta.

Versión de origen

Introduzca una ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte[Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#)

Profundidad del clon de Git

Seleccione Git clone depth (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione Use Git submodules (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

SelectInformar de los estados de compilación al proveedor de origen cuando se inician y finalizan las compilacionessi desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

ParaContexto de estado, escriba el valor que se va a utilizar para elnameen el estado de confirmación de Bitbucket. Para obtener más información, consulte la [compilación](#) en la documentación de la API de Bitbucket.

ParaURL de destino, escriba el valor que se va a utilizar para elurlen el estado de confirmación de Bitbucket. Para obtener más información, consulte la [compilación](#) en la documentación de la API de Bitbucket.

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

En **Eventos de webhooks de origen principal**, seleccione **Volver a compilar cada vez que se envíe un cambio de código a este repositorio** si desea que CodeBuild compile el código fuente cada vez que se envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de Bitbucket \(p. 238\)](#).

GitHub

Repositorio

Seleccione **Connect** mediante **OAuth** con un token de acceso personal de GitHub y siga las instrucciones para conectarse (o volver a conectarse) a GitHub y autorizar el acceso a AWS CodeBuild.

Elija un repositorio público o un repositorio de su cuenta.

Versión de origen

Introduzca una ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Seleccione **Git clone depth** (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija **Full** (Completo).

Submódulos de Git

Seleccione **Use Git submodules** (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

Seleccione **Informar de los estados de compilación al proveedor de origen cuando se inician y finalizan las compilaciones** si desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

Para **Contexto de estado**, escriba el valor que se va a utilizar para el **context** en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación en la guía para desarrolladores de GitHub](#).

Para **URL de destino**, escriba el valor que se va a utilizar para el **target_url** en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación en la guía para desarrolladores de GitHub](#).

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

En **Eventos de webhooks de origen principal**, seleccione **Volver a compilar cada vez que se envíe un cambio de código a este repositorio** si desea que CodeBuild compile el código fuente cada vez que se envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de GitHub \(p. 246\)](#).

GitHub Enterprise Server

Token de acceso personal de GitHub Enterprise

Consulte [Ejemplo de GitHub Enterprise Server \(p. 107\)](#) Para obtener información sobre cómo copiar un token de acceso personal en el portapapeles. Pegue el token en el campo de texto y seleccione Save Token (Guardar token).

Note

Solo tiene que introducir y guardar el token de acceso personal una vez. CodeBuild utiliza este token en todos los proyectos futuros.

Versión de origen

Introduzca una solicitud de extracción, ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Seleccione Git clone depth (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione Use Git submodules (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

SelectInformar de los estados de compilación al proveedor de origen cuando se inician y finalizan las compilaciones si desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

Para Contexto de estado, escriba el valor que se va a utilizar para el context en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

Para URL de destino, escriba el valor que se va a utilizar para el target_url en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

SSL inseguro

Seleccione Enable insecure SSL (Habilitar SSL no seguro) para omitir las advertencias de SSL al conectarse al repositorio del proyecto de GitHub Enterprise.

En Eventos de webhooks de origen principal, seleccione Volver a compilar cada vez que se envíe un cambio de código a este repositorio si desea que CodeBuild compile el código fuente cada vez que se

envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de GitHub](#) (p. 246).

Environment

Imagen de entorno

Aplique alguna de las siguientes acciones:

- Para usar una imagen de Docker administrada por AWS CodeBuild, elija Managed image (Imagen administrada) y después elija una opción en Operating system (Sistema operativo), Runtime (Tiempo de ejecución), Image (Imagen) e Image version (Versión de imagen). Realice una selección en Environment type (Tipo de entorno) si está disponible.
- Para usar otra imagen de Docker, elija Custom image (Imagen personalizada). Para Tipo de entorno, elige ARM, Linux, GPU de Linux, o bien Windows. Si elige Other registry (Otro registro), para External registry URL (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige Amazon ECR, use Repositorio de Amazon ECR para elegir la imagen de Docker en su AWS account.
- Para utilizar una imagen privada de Docker, elija imagen personalizada. Para Tipo de entorno, elige ARM, Linux, GPU de Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Secrets Manager debe crear las credenciales. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Note

CodeBuild anula el `ENTRYPOINT` para imágenes de Docker personalizadas.

Privilegiado

(Opcional) Seleccione Privilegiado únicamente si tiene previsto usar este proyecto de compilación para compilar imágenes de Docker y la imagen del entorno de compilación que eligió no es una de las proporcionadas por CodeBuild con compatibilidad con Docker. De lo contrario, todas las compilaciones asociadas que intenten interactuar con el daemon de Docker producirán un error. También debe iniciar el daemon de Docker para que las compilaciones puedan interactuar con él. Una forma de hacer esto es inicializar el daemon de Docker en la fase `install` de la especificación de compilación ejecutando los siguientes comandos de compilación. No ejecute estos comandos si eligió una imagen del entorno de compilación proporcionada por CodeBuild compatible con Docker.

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

```
- nohup /usr/local/bin/dockerd --host=unix:///var/run/docker.sock --  
host=tcp://127.0.0.1:2375 --storage-driver=overlay2 &  
- timeout 15 sh -c "until docker info; do echo .; sleep 1; done"
```

Rol de servicio

Aplique alguna de las siguientes acciones:

- Si no tiene un rol de servicio CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.

- Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola de para crear un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

Configuración adicional

Timeout (Tiempo de espera)

Especifique un valor entre 5 minutos y 8 horas después de los cuales CodeBuild detendrá la compilación si no se ha completado. Si hours y minutes se dejan en blanco, se usará el valor predeterminado de 60 minutos.

VPC

Si desea que CodeBuild funcione con su VPC:

- Para VPC:, elija el ID de VPC que utiliza CodeBuild.
- Para Subredes de la VPC, elija las subredes que incluyen recursos usados por CodeBuild.
- Para Grupos de seguridad de la VPC, elija los grupos de seguridad que usa CodeBuild para permitir el acceso a los recursos de las VPC.

Para obtener más información, consulte [Usar AWS CodeBuild con Amazon Virtual Private Cloud](#) (p. 180).

Computación

Elegir una de las opciones disponibles.

Variables de entorno

Escriba el nombre y el valor de las variables de entorno que se van a usar y seleccione su tipo.

Note

CodeBuild establece la variable de entorno para su AWS Región automáticamente. Debe definir las siguientes variables de entorno, si no las ha agregado a su archivo buildspec.yml:

- AWS_ACCOUNT_ID
- IMAGE_REPO_NAME
- IMAGE_TAG

Los usuarios de la consola y de AWS CLI pueden ver las variables de entorno. Si no le preocupa que la variable de entorno esté visible, establezca los campos Name y Value, y después establezca Type en Plaintext.

Le recomendamos que almacene una variable de entorno con un valor confidencial, como un AWS ID de clave de acceso, un AWS clave de acceso secreta o contraseña como parámetro en el almacén de parámetros Amazon EC2 Systems Manager o AWS Secrets Manager.

Si utiliza Amazon EC2 Systems Manager, en Tipo, elige Parámetro. Para Nombre, escriba un identificador para CodeBuild al que haga referencia. Para Valor, escriba el nombre del parámetro tal como está almacenado en el almacén de parámetros Amazon EC2 Systems Manager.

Utilizando el parámetro denominado `/CodeBuild/dockerLoginPassword` como ejemplo, en Type (Tipo), seleccione Parameter (Parámetro). En Name (Nombre) escriba `LOGIN_PASSWORD`. En Value (Valor), escriba `/CodeBuild/dockerLoginPassword`.

Important

Si utiliza el almacén de parámetros de Amazon EC2 Systems Manager, le recomendamos que almacene los parámetros con nombres de parámetros que comiencen por `/CodeBuild/` (por ejemplo, `/CodeBuild/dockerLoginPassword`). Puede utilizar la consola de CodeBuild para crear un parámetro en Amazon EC2 Systems Manager. Seleccione Create a parameter (Crear parámetro) y siga las instrucciones del cuadro de diálogo. (En ese cuadro de diálogo, para Clave de KMS, puede especificar el ARN de un AWS KMS clave en tu cuenta. Amazon EC2 Systems Manager utiliza esta clave para cifrar el valor del parámetro durante el almacenamiento y descifrarlo durante la recuperación). Si usa la consola de CodeBuild para crear un parámetro, la consola asigna al principio del nombre del parámetro con `/CodeBuild/` ya que se está almacenando. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial de la consola del almacén de parámetros de Systems Manager](#) en la Guía del usuario de Amazon EC2 Systems Manager.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager, el rol de servicio del proyecto de compilación debe permitir `assm:GetParametersAction`. Si eligió Nuevo rol de servicio anteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó Existing service role (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager con nombres de parámetros que no empiezan por `/CodeBuild/` y eligió Nuevo rol de servicio, debe actualizar ese rol de servicio para permitir el acceso a nombres de parámetros que no empiezan por `/CodeBuild/`. Esto es así porque el rol de servicio permite el acceso únicamente a los nombres de parámetro que empiezan por `/CodeBuild/`.

Si eligió Nuevo rol de servicio, el rol de servicio incluye permisos para descifrar todos los parámetros en el `/CodeBuild/namespace` en el almacén de parámetros Amazon EC2 Systems Manager.

Las variables de entorno que defina reemplazan las variables de entorno existentes.

Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y establece una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`.

Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y establece una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No establezca variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad.
- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad.
- El valor en la declaración de especificación de compilación es el que menos prioridad tiene.

Si utilizas Secrets Manager, para Tipo, elige Secrets Manager. Para Nombre, escriba un identificador para CodeBuild al que haga referencia. En Value (Valor), introduzca un `reference-`

key utilizando el patrón `secret-id:json-key:version-stage:version-id`. Para obtener información, consulte [Secrets Manager reference-key in the buildspec file](#).

Important

Si utiliza Secrets Manager, le recomendamos que almacene secretos con nombres que comiencen por `/CodeBuild/` (por ejemplo, `/CodeBuild/dockerLoginPassword`). Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Si el proyecto de compilación hace referencia a secretos almacenados en Secrets Manager, el rol de servicio del proyecto de compilación debe permitir `lasecretsmanager:GetSecretValue` acción. Si eligió `Nuevo` rol de servicio anteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó `Existing service role` (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado.

Si tu proyecto de compilación hace referencia a secretos almacenados en Secrets Manager con nombres secretos que no empiezan por `/CodeBuild/` y eligió `Nuevo` rol de servicio, debe actualizar el rol de servicio para permitir el acceso a nombres secretos que no empiezan por `/CodeBuild/`. Esto es así porque el rol de servicio permite el acceso únicamente a nombres secretos que empiezan por `/CodeBuild/`.

Si eligió `Nuevo` rol de servicio, el rol de servicio incluye permisos para descifrar todos los secretos en el `/CodeBuild/` espacio de nombres en Secrets Manager.

Buildspec

Especificaciones de compilación

Aplique alguna de las siguientes acciones:

- Si el código fuente incluye un archivo de especificación de compilación, elija `Use a buildspec file` (Usar un archivo buildspec). De forma predeterminada, CodeBuild busca un archivo llamado `buildspec.yml` en el directorio raíz del código fuente. Si su archivo buildspec usa un nombre o ubicación diferente, escriba su ruta desde la raíz de origen en `Nombre de la especificación de compilación` (por ejemplo, `buildspec-two.yml` o `configuration/buildspec.yml`). Si el archivo buildspec está en un bucket de S3, debe estar en el mismo `AWS` región como proyecto de compilación. Especifique el archivo buildspec utilizando su ARN (por ejemplo, `arn:aws:s3:::my-codebuild-sample2/buildspec.yml`).
- Si el código fuente no incluye un archivo de especificación de compilación o si desea ejecutar comandos de compilación diferentes de los especificados para la fase `build` en el archivo `buildspec.yml` del directorio raíz de código fuente, elija `Insert build commands` (Insertar comandos de compilación). En `Build commands` (Comandos de compilación), escriba los comandos que desea ejecutar en la fase `build`. En caso de que haya varios comandos, separe cada comando con `&&` (por ejemplo, `mvn test && mvn package`). Para ejecutar comandos en otras fases, o si tiene una larga lista de comandos para el `build` fase, añadir un `buildspec.yml` al directorio raíz del código fuente, agregue los comandos al archivo y, a continuación, elija `Use the buildspec.yml in the directory root of the source code`.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

Configuración Batch

Puede ejecutar un grupo de compilaciones como una sola operación. Para obtener más información, consulte [Compilaciones Batch en AWS CodeBuild \(p. 276\)](#).

Definición de configuración de lotes

Seleccione esta opción para permitir compilaciones por lotes en este proyecto.

Rol de servicio de Batch

Proporciona el rol de servicio para las compilaciones por lotes.

Elija una de las siguientes opciones:

- Si no tiene un rol de servicio por lotes, elija **Nuevo rol de servicio**. En **Rol de servicio**, especifique un nombre para el nuevo rol.
- Si dispone de un rol de servicio por lotes, elija **Rol de servicio existente**. En **Rol de servicio**, seleccione el rol de servicio.

Las compilaciones Batch introducen un nuevo rol de seguridad en la configuración de lotes. Se requiere este nuevo rol, ya que CodeBuild debe poder llamar a `StartBuild`, `StopBuild`, y `RetryBuild` acciones en su nombre para ejecutar compilaciones como parte de un lote. Los clientes deben utilizar un nuevo rol, y no el mismo rol que usan en su compilación, por dos razones:

- Dar el rol de compilación `StartBuild`, `StopBuild`, y `RetryBuild` los permisos permitirían que una sola compilación inicie más compilaciones a través de la especificación de compilación.
- Las compilaciones por lotes de CodeBuild proporcionan restricciones que restringen el número de compilaciones y tipos de cómputo que se pueden utilizar para las compilaciones del lote. Si el rol de compilación tiene estos permisos, es posible que las compilaciones mismas puedan eludir estas restricciones.

Tipos de cómputo permitidos para lotes

Seleccione los tipos de cómputo permitidos para el lote. Seleccione todo lo que corresponda.

Número máximo de compilaciones permitidas por lotes

Escriba el número máximo de compilaciones permitidas en el lote. Si un lote supera este límite, el lote fallará.

Tiempo de espera del lote

Introduzca la cantidad máxima de tiempo que dura la compilación por lotes.

Combine Artifacts

Select Combina todos los artefactos del lote en una única ubicación para que todos los artefactos del lote se combinen en una única ubicación.

Modo de informe por lotes

Seleccione el modo de informe de estado de compilación deseado para las compilaciones por lotes.

Note

Este campo solo está disponible cuando el origen del proyecto es Bitbucket, GitHub o GitHub Enterprise, y **Informar de los estados de compilación al proveedor de origen** cuando se inician y finalizan las compilaciones se selecciona en **Fuente**.

Builds agregadas

Seleccione esta opción para combinar los estados de todas las compilaciones del lote en un único informe de estado.

Builds individuales

Seleccione esta opción para que los estados de compilación de todas las compilaciones del lote se informen por separado.

Artifacts

Type

Aplique alguna de las siguientes acciones:

- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts. por ejemplo, si solo va a ejecutar pruebas de compilación o si desea insertar una imagen de Docker en un repositorio de Amazon ECR.
- Para almacenar la salida de la compilación en un bucket de S3, elija Amazon S3y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. (Si desea producir un archivo ZIP y desea que el archivo ZIP tenga una extensión de archivo, asegúrese de incluirlo detrás del nombre del archivo ZIP).
 - Seleccione Enable semantic versioning (Habilitar control semántico de versiones) si desea que el nombre especificado en el archivo buildspec invalide cualquier nombre especificado en la consola. El nombre en un archivo buildspec se calcula en tiempo de compilación y utiliza el lenguaje de comandos Shell. Por ejemplo, puede adjuntar una fecha y una hora al nombre del artefacto para que siempre sea único. Los nombres de artefactos únicos impiden que los artefactos se sobrescriban. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).
 - En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.
 - Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, appspec.yml, target/my-app.jar). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec \(p. 137\)](#).
 - Si no desea que los artefactos de compilación se cifren, seleccione Remove artifacts encryption (Quitar cifrado de artefactos),

Para cada conjunto de artefactos secundarios que desee:

1. Para Source identifier (Identificador de artefacto), escriba un valor que tenga como máximo 128 caracteres, únicamente de tipo alfanumérico y guiones bajos.
2. Elija Add artifact (Añadir artefacto).
3. Siga los pasos anteriores para configurar los artefactos secundarios.
4. Elija Save artifact (Guardar artefacto).

Configuración adicional

Clave de cifrado

(opcional) Realice una de las operaciones siguientes:

- Para utilizar el Clave administrada por AWS para Amazon S3 de su cuenta para cifrar los artefactos de salida de la compilación, deje Clave de cifrado en blanco. Esta es la opción predeterminada.
- Para utilizar una clave administrada por el cliente para cifrar los artefactos de salida de la compilación, en Clave de cifrado, especifique el ARN de la clave KMS. Utilice el formato `arn:aws:kms:region-ID:account-ID:key/key-ID`.

Tipo de caché

En Cache type (Tipo de caché), seleccione una de las opciones siguientes:

- Si no desea usar una memoria caché, elija No cache.
- Si desea utilizar una caché de Amazon S3, elija Amazon S3y, a continuación, haga lo siguiente:
 - En Bucket, elija el nombre del bucket de S3 donde se almacena la caché.
 - (Opcional) Para Prefijo de ruta de caché, especifique un prefijo de ruta de Amazon S3. El valor Cache path prefix (Prefijo de ruta de caché) es similar a un nombre de directorio. Le permite almacenar la caché en el mismo directorio en un bucket.

Important

No añada una barra (/) al final del prefijo de ruta.

- Si desea utilizar una caché local, seleccione Local y elija uno o varios modos de caché local.

Note

El modo Docker layer cache (Caché de capas de Docker) solo está disponible para Linux. Si lo selecciona, el proyecto deberá ejecutarse en modo con privilegios. La `ARM_CONTAINER` y `LINUX_GPU_CONTAINER` tipos de entorno `yBUILD_GENERAL1_2XLARGE` el tipo de cómputo no admite el uso de una caché local.

El uso de memoria caché ahorra mucho tiempo de compilación, ya que algunas partes reutilizables del entorno de compilación se almacenan en ella y se usan en las distintas compilaciones. Para obtener información acerca de cómo especificar una caché en el archivo de especificación de compilación, consulte [Sintaxis de buildspec \(p. 137\)](#). Para obtener más información acerca del almacenamiento en caché, consulte [Almacenamiento en caché de compilaciones de AWS CodeBuild \(p. 229\)](#).

Logs

Elija los registros que desea crear. Puede crear Amazon CloudWatch Logs, registros Amazon S3 o ambos.

CloudWatch

Si desea registros de Amazon CloudWatch Logs:

CloudWatch Logs

SelectCloudWatch Logs.

Group name

Escriba el nombre del grupo de registros de Amazon CloudWatch Logs.

Nombre de transmisión

Introduzca el nombre del flujo de registro de Amazon CloudWatch Logs.

S3

Si quieres registros de Amazon S3:

Registros de S3

Seleccione registros de S3.

Bucket

Seleccione el nombre del bucket de S3 para los registros.

Prefijo de la ruta

Escriba el prefijo de los registros.

Deshabilitar el cifrado de registros de S3

Seleccione si no quiere que los registros de S3 se cifren.

Crear un proyecto de compilación (AWS CLI)

Para obtener más información acerca del uso de la AWS CLI con CodeBuild, consulte la [Referencia de la línea de comandos \(p. 402\)](#).

Para crear un proyecto de compilación de CodeBuild mediante elAWS CLI, crea un formato JSONProyecto de estructura, rellene la estructura y llame alcreate-projectpara crear el proyecto.

Creación del archivo JSON

Cree un archivo JSON esqueleto con elcreate-project, utilizando el comando--generate-cli-skeletonopción:

```
aws codebuild create-project --generate-cli-skeleton > <json-file>
```

Esto crea un archivo JSON con la ruta y el nombre de archivo especificados por<json-file>.

Rellene el archivo JSON

Modifique los datos JSON de la siguiente manera y guarde los resultados.

```
{
  "name (p. 212)": "<project-name>",
  "description (p. 212)": "<description>",
  "source (p. 212)": {
    "type (p. 212)": "CODECOMMIT" | "CODEPIPELINE" | "GITHUB" | "GITHUB_ENTERPRISE" |
    "BITBUCKET" | "S3" | "NO_SOURCE",
    "location (p. 213)": "<source-location>",
    "gitCloneDepth (p. 214)": "<git-clone-depth>",
    "buildspec (p. 214)": "<buildspec>",
    "InsecureSsl (p. 215)": "<insecure-ssl>",
    "reportBuildStatus (p. 214)": "<report-build-status>",
    "buildStatusConfig": {
      "context (p. 214)": "<context>",
      "targetUrl (p. 214)": "<target-url>"
    },
    "gitSubmodulesConfig": {
      "fetchSubmodules (p. 215)": "<fetch-submodules>"
    },
    "auth": {
      "type (p. 214)": "<auth-type>",
      "resource (p. 214)": "<auth-resource>"
    },
    "sourceIdentifier (p. 215)": "<source-identifier>"
  },
  "secondarySources (p. 215)": [
    {
      "type": "CODECOMMIT" | "CODEPIPELINE" | "GITHUB" | "GITHUB_ENTERPRISE" |
      "BITBUCKET" | "S3" | "NO_SOURCE",
      "location": "<source-location>",
      "gitCloneDepth": "<git-clone-depth>",
      "buildspec": "<buildspec>",
      "InsecureSsl": "<insecure-ssl>",
      "reportBuildStatus": "<report-build-status>",
      "auth": {
        "type": "<auth-type>",
        "resource": "<auth-resource>"
      },
      "sourceIdentifier": "<source-identifier>"
    }
  ],
  "secondarySourceVersions (p. 215)": [
    {
      "sourceIdentifier": "<secondary-source-identifier>",
      "sourceVersion": "<secondary-source-version>"
    }
  ]
}
```

```
"sourceVersion (p. 215)": "<source-version>",
"artifacts (p. 216)": {
  "type (p. 216)": "CODEPIPELINE" | "S3" | "NO_ARTIFACTS",
  "location (p. 216)": "<artifacts-location>",
  "path (p. 216)": "<artifacts-path>",
  "namespaceType (p. 216)": "<artifacts-namespacetype>",
  "name (p. 216)": "<artifacts-name>",
  "overrideArtifactName (p. 217)": "<override-artifact-name>",
  "packaging (p. 217)": "<artifacts-packaging>"
},
"secondaryArtifacts (p. 217)": [
  {
    "type": "CODEPIPELINE" | "S3" | "NO_ARTIFACTS",
    "location": "<secondary-artifact-location>",
    "path": "<secondary-artifact-path>",
    "namespaceType": "<secondary-artifact-namespacetype>",
    "name": "<secondary-artifact-name>",
    "packaging": "<secondary-artifact-packaging>",
    "artifactIdentifier": "<secondary-artifact-identifier>"
  }
],
"cache (p. 217)": {
  "type": "<cache-type>",
  "location": "<cache-location>",
  "mode": [
    "<cache-mode>"
  ]
},
"environment (p. 217)": {
  "type (p. 217)": "LINUX_CONTAINER" | "LINUX_GPU_CONTAINER" | "ARM_CONTAINER" |
  "WINDOWS_SERVER_2019_CONTAINER",
  "image (p. 217)": "<image>",
  "computeType (p. 217)": "BUILD_GENERAL1_SMALL" | "BUILD_GENERAL1_MEDIUM" |
  "BUILD_GENERAL1_LARGE" | "BUILD_GENERAL1_2XLARGE",
  "certificate (p. 218)": "<certificate>",
  "environmentVariables (p. 218)": [
    {
      "name": "<environmentVariable-name>",
      "value": "<environmentVariable-value>",
      "type": "<environmentVariable-type>"
    }
  ],
  "registryCredential (p. 219)": [
    {
      "credential": "<credential-arn-or-name>",
      "credentialProvider": "<credential-provider>"
    }
  ],
  "imagePullCredentialsType (p. 220)": "CODEBUILD" | "SERVICE_ROLE",
  "privilegedMode (p. 220)": "<privileged-mode>"
},
"serviceRole (p. 220)": "<service-role>",
"timeoutInMinutes (p. 220)": <timeout>,
"queuedTimeoutInMinutes (p. 220)": <queued-timeout>,
"encryptionKey (p. 221)": "<encryption-key>",
"tags (p. 221)": [
  {
    "key": "<tag-key>",
    "value": "<tag-value>"
  }
],
"vpcConfig (p. 221)": {
  "securityGroupIds": [
    "<security-group-id>"
  ],
  "subnets": [
```

```
        "<subnet-id>"
    ],
    "vpcId": "<vpc-id>"
  },
  "badgeEnabled (p. 221)": "<badge-enabled>",
  "logsConfig (p. 221)": {
    "cloudWatchLogs (p. 221)": {
      "status": "<cloudwatch-logs-status>",
      "groupName": "<group-name>",
      "streamName": "<stream-name>"
    },
    "s3Logs (p. 221)": {
      "status": "<s3-logs-status>",
      "location": "<s3-logs-location>",
      "encryptionDisabled": "<s3-logs-encryption-disabled>"
    }
  },
  "fileSystemLocations (p. 222)": [
    {
      "type": "EFS",
      "location": "<EFS-DNS-name-1>:/<directory-path>",
      "mountPoint": "<mount-point>",
      "identifier": "<efs-identifier>",
      "mountOptions": "<efs-mount-options>"
    }
  ],
  "buildBatchConfig (p. 222)": {
    "serviceRole": "<batch-service-role>",
    "combineArtifacts": "<combine-artifacts>",
    "restrictions": {
      "maximumBuildsAllowed": "<max-builds>",
      "computeTypesAllowed": [
        "<compute-type>"
      ]
    },
    "timeoutInMins": "<batch-timeout>",
    "batchReportMode": "REPORT_AGGREGATED_BATCH" | "REPORT_INDIVIDUAL_BUILDS"
  },
  "concurrentBuildLimit (p. 222)": "<concurrent-build-limit>"
}
```

Sustituya lo siguiente:

name

Obligatorio. El nombre de este proyecto de compilación. Este nombre debe ser único en todos los proyectos de compilación de su cuenta de AWS.

description

Opcional. La descripción de este proyecto de compilación.

source

Obligatorio. [A Origen del proyecto](#) que contiene información sobre la configuración del código fuente de este proyecto de compilación. Después de añadir un objeto `source`, puede añadir hasta 12 orígenes más mediante el [the section called "secondarySources"](#). Esta configuración incluye lo siguiente:

origen/type

Obligatorio. El tipo de repositorio que contiene el código fuente que se va a compilar. Los valores válidos son:

- CODECOMMIT
- CODEPIPELINE
- GITHUB
- GITHUB_ENTERPRISE
- BITBUCKET
- S3
- NO_SOURCE

Si usa `NO_SOURCE`, la especificación de compilación no puede ser un archivo porque el proyecto no tiene un origen. En su lugar, debe usar el atributo `buildspec` para especificar una cadena con formato YAML para su especificación de compilación. Para obtener más información, consulte [Ejemplo de proyecto sin un origen](#) (p. 132).

origen/location

Obligatorio, a menos que `<source-type>` sea `CODEPIPELINE`. La ubicación del código fuente para el tipo de repositorio especificado.

- Para CodeCommit, la URL clon HTTPS al repositorio que contiene el código fuente y el archivo `buildspec` (por ejemplo, `https://git-codecommit.<region-id>.amazonaws.com/v1/repos/<repo-name>`).
- Para Amazon S3, el nombre del bucket de entrada de la compilación, seguido de la ruta y el nombre del archivo ZIP que contiene el código fuente y la especificación de compilación. Por ejemplo:
 - En el caso de un archivo ZIP ubicado en la raíz del depósito de entrada: `<bucket-name>/<object-name>.zip`.
 - Para un archivo ZIP ubicado en una subcarpeta del bucket de entrada: `<bucket-name>/<subfolder-path>/<object-name>.zip`.
- Para GitHub, la URL clon HTTPS al repositorio que contiene el código fuente y el archivo `buildspec`. La dirección URL debe contener `github.com`. Debe conectar su cuenta de AWS a su cuenta de GitHub. Para ello, utilice la consola de CodeBuild para crear un proyecto de compilación.
 1. En GitHub Autorizar solicitud, en la Acceso a la organización sección, elija. Solicitud de acceso junto a cada repositorio al que desee que CodeBuild pueda obtener acceso en él.
 2. Elija Authorize application. (Una vez conectado a su cuenta de GitHub, no es necesario que termine de crear el proyecto de compilación. Puede cerrar la consola de CodeBuild.)
- En GitHub Enterprise Server, la URL clon HTTP o HTTPS al repositorio que contiene el código fuente y el archivo `buildspec`. Además, debe conectar su cuenta de AWS con su cuenta de GitHub Enterprise Server. Para ello, utilice la consola de CodeBuild para crear un proyecto de compilación.
 1. Cree un token de acceso personal en GitHub Enterprise Server.
 2. Copie dicho token en el portapapeles de forma que pueda utilizarlo cuando cree su proyecto de CodeBuild. Para obtener más información, consulte [Creating a personal access token for the command line](#) en el sitio web GitHub Help.
 3. Cuando utilice la consola para crear su proyecto de CodeBuild, en Fuente, para Proveedor de origen, elija GitHub Enterprise.
 4. En Personal Access Token, pegue el token que ha copiado en el portapapeles. Elija Save Token. Su cuenta de CodeBuild ya está conectada a su cuenta de GitHub Enterprise Server.
- Para Bitbucket, la URL clon HTTPS al repositorio que contiene el código fuente y el archivo `buildspec`. La dirección URL debe contener `bitbucket.org`. Además, debe conectar su cuenta de AWS a su cuenta de Bitbucket. Para ello, utilice la consola de CodeBuild para crear un proyecto de compilación.
 1. Cuando use la consola para conectarse (o volver a conectarse) a Bitbucket, en la página Confirm access to your account de Bitbucket, elija Grant access. (Una vez conectado a su cuenta de Bitbucket, no es necesario que termine de crear el proyecto de compilación. Puede cerrar la consola de CodeBuild.)

- En AWS CodePipeline, no especifique el valor `location` para `source`. CodePipeline no tiene en cuenta este valor porque, cuando se crea una canalización en CodePipeline, se especifica la ubicación del código fuente en la etapa Source de la canalización.

`origen/gitCloneDepth`

Opcional. La profundidad del historial que se va a descargar. El valor mínimo es 0. Si esta valor es 0, superior a 25 o no se facilita, se descargará el historial completo con cada proyecto de compilación. Si el tipo de origen es Amazon S3, este valor no se admite.

`origen/buildspec`

Opcional. La definición o archivo de especificación de compilación que va a usar. Si este valor no se proporciona o se establece en una cadena vacía, el código fuente debe contener un archivo `buildspec.yml` en su directorio raíz. Si se establece este valor, puede ser una definición de especificación de compilación insertada, la ruta a un archivo de especificación de compilación alternativo en relación con el directorio raíz de su fuente principal o la ruta a un bucket de S3. El bucket debe encontrarse en la misma región de AWS que el proyecto de compilación. Especifique el archivo `buildspec` utilizando su ARN (por ejemplo, `arn:aws:s3:::my-codebuild-sample2/buildspec.yml`). Para obtener más información, consulte [Nombre de archivo y ubicación de almacenamiento de buildspec \(p. 137\)](#).

`origen/auth`

No utilizar. Este objeto se utiliza únicamente en la consola de CodeBuild.

`origen/reportBuildStatus`

Especifica si se debe enviar al proveedor de código fuente el estado de inicio y finalización de una compilación. Si define este valor con un proveedor de código fuente distinto de GitHub, GitHub Enterprise Server o Bitbucket, se genera una excepción `invalidInputException`.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

`origen/BuildStatusConfig`

Contiene información que define cómo el proyecto de compilación de CodeBuild informa del estado de compilación al proveedor de origen. Esta opción se utiliza únicamente cuando el tipo de origen es `ESGITHUB`, `GITHUB_ENTERPRISE`, o bien `BITBUCKET`.

Fuente/BuildStatusConfig/context

En el caso de los orígenes de Bitbucket, este parámetro se utiliza para `lanameen` el estado de confirmación de Bitbucket. En los orígenes de GitHub, este parámetro se utiliza para `lacontexten` el estado de confirmación de GitHub.

Por ejemplo, puede tener `elcontext` contiene el número de compilación y el activador webhook utilizando las variables de entorno CodeBuild:

```
AWS CodeBuild sample-project Build #${CODEBUILD_BUILD_NUMBER} -  
${CODEBUILD_WEBHOOK_TRIGGER}
```

Esto da como resultado que el contexto de la compilación #24 desencadena por un evento de solicitud de extracción de webhook:

```
AWS CodeBuild sample-project Build #24 - pr/8
```

Fuente/BuildStatusConfig/TargetURL

En el caso de los orígenes de Bitbucket, este parámetro se utiliza para `laurl` en el estado de confirmación de Bitbucket. En los orígenes de GitHub, este parámetro se utiliza para `laurl` en el estado de confirmación de GitHub.

Por ejemplo, puede establecer la opción `targetUrl` de `https://aws.amazon.com/codebuild/<path to build>` y el estado de confirmación se vinculará a esta URL.

También puede incluir variables de entorno de CodeBuild en el `targetUrl` para añadir información adicional a la URL. Por ejemplo, para agregar la región de compilación a la URL, establezca `laurl`:

```
"targetUrl": "https://aws.amazon.com/codebuild/<path to build>?region=$AWS_REGION"
```

Si la región de compilación es `us-east-2`, se ampliará a:

```
https://aws.amazon.com/codebuild/<path to build>?region=us-east-2
```

origen/gitSubmodulesConfig

Opcional. Información acerca de la configuración de submódulos de Git. Solo se utiliza con CodeCommit, GitHub Enterprise Server y Bitbucket.

Submódulos de fuente/GIT Config/FetchSubmodules

Establezca `fetchSubmodules` en `true` si desea incluir submódulos de Git en el repositorio. Los submódulos de Git que se incluyan deben estar configurados como HTTPS.

origen/InsecureSsl

Opcional. Solo se utiliza con GitHub Enterprise Server. Establezca este valor en `true` para omitir las advertencias de TLS al conectarse al repositorio del proyecto de GitHub Enterprise Server. El valor predeterminado es `false`. `InsecureSsl` debe utilizarse con fines de prueba únicamente. No debe utilizarse en un entorno de producción.

origen/SourceIdentifier

Un identificador definido por el usuario para el origen del proyecto. Opcional para el origen principal. Necesario para fuentes secundarias.

secondarySources

Opcional. Una matriz de [Origen del proyecto](#) objetos que contienen información sobre los orígenes secundarios de un proyecto de compilación. Puede añadir hasta 12 fuentes secundarias. Las `secondarySources` los objetos utilizan las mismas propiedades que utiliza el [the section called "source"](#) objeto. En un objeto de origen secundario, el `sourceIdentifier` es obligatorio.

secondarySourceVersions

Opcional. Una matriz de [ProjectSourceVersion](#) objetos. Si se especifica `secondarySourceVersions` en el nivel de compilación, prevalece sobre esto.

sourceVersion

Opcional. La versión de la entrada de la compilación que se creará para este proyecto. Si no se especifica, se utiliza la versión más reciente. Si se especifica, debe ser una de las siguientes opciones:

- En CodeCommit, el ID de confirmación, la ramificación o la etiqueta Git que se va a utilizar.

- Para GitHub, el ID de confirmación, el ID de solicitud de inserción, el nombre de ramificación o el nombre de etiqueta correspondiente a la versión de código fuente que desea compilar. Si se especifica un ID de solicitud de inserción, este debe tener el formato `pr/pull-request-ID` (por ejemplo, `pr/25`). Si se especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si no se especifica, se usa el ID de confirmación HEAD de la ramificación personalizada.
- Para Bitbucket, el ID de confirmación, el nombre de ramificación o el nombre de etiqueta correspondiente a la versión de código fuente que desea compilar. Si se especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si no se especifica, se usa el ID de confirmación HEAD de la ramificación personalizada.
- En Amazon S3, el ID de versión del objeto que representa el archivo ZIP de entrada de compilación que se va a utilizar.

Si se especifica `sourceVersion` en el nivel de compilación y, a continuación, esa versión prevalece sobre `sourceVersion` (en el nivel del proyecto). Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild](#) (p. 125).

artifacts

Obligatorio. [Artefactos del proyecto](#) que contiene información sobre la configuración del artefacto de salida de este proyecto de compilación. Después de añadir un objeto `artifacts`, puede añadir hasta 12 artefactos más mediante el [the section called "secondaryArtifacts"](#). Esta configuración incluye lo siguiente:

artifacts/type

Obligatorio. El tipo de artefacto de salida de la compilación. Los valores válidos son:

- `CODEPIPELINE`
- `NO_ARTIFACTS`
- `S3`

artifacts de/location

Solo se utiliza con el `s3` tipo artefacto. No se utiliza para otros tipos de artefactos.

El nombre del bucket de salida que creó o identificó en los requisitos previos.

artifacts de/path

Solo se utiliza con el `s3` tipo artefacto. No se utiliza para otros tipos de artefactos.

Ruta del depósito de salida para colocar el archivo ZIP o la carpeta. Si no especifica ningún valor `parapath`, CodeBuild utiliza `namespaceType` (si se especifica) y `name` para determinar la ruta y el nombre del archivo ZIP o carpeta de salida de la compilación. Por ejemplo, si especifica `MyPath`: `parapathMyArtifact.zip`: `paraname`, el camino y el nombre serían `MyPath/MyArtifact.zip`.

artifacts de/namespaceType

Solo se utiliza con el `s3` tipo artefacto. No se utiliza para otros tipos de artefactos.

El espacio de nombres del archivo ZIP o carpeta de salida de la compilación. Los valores válidos son `BUILD_ID` y `NONE`. Utilice `BUILD_ID` para insertar el ID de compilación en la ruta del archivo ZIP o carpeta de salida de la compilación. De lo contrario, utilice `NONE`. Si no especifica ningún valor `paranamespaceType`, CodeBuild utiliza `path` (si se especifica) y `name` para determinar la ruta y el nombre del archivo ZIP o carpeta de salida de la compilación. Por ejemplo, si especifica `MyPath`: `parapath, BUILD_ID`: `paranamespaceType`, y `MyArtifact.zip`: `paraname`, el camino y el nombre serían `MyPath/build-ID/MyArtifact.zip`.

artifacts/name

Solo se utiliza con el `s3` tipo artefacto. No se utiliza para otros tipos de artefactos.

El nombre del archivo ZIP o carpeta de salida de la compilación dentro de `location`. Por ejemplo, si especifica `MyPath: parapathMyArtifact.zip`: `paraname`, el camino y el nombre serían `MyPath/MyArtifact.zip`.

artefactos de/OverrideArtifactName

Solo se utiliza con el tipo de artefacto S3. No se utiliza para otros tipos de artefactos.

Opcional. Si se establece en `true`, el nombre especificado en el `artifacts` bloque de las modificaciones de archivo `buildspecname`. Para obtener más información, consulte [Referencia de la especificación de compilación para CodeBuild \(p. 136\)](#).

artefactos de/empaquetado

Solo se utiliza con el `s3` tipo artefacto. No se utiliza para otros tipos de artefactos.

Opcional. Especifica cómo empaquetar los artefactos. Los valores permitidos son:

NONE

Cree una carpeta que contenga los artefactos de compilación. Este es el valor predeterminado.

ZIP

Cree un archivo ZIP que contenga los artefactos de compilación.

secondaryArtifacts

Opcional. Una matriz de [Artefactos del proyecto](#) objetos que contienen información sobre la configuración de artefactos secundarios de un proyecto de compilación. Puede añadir hasta 12 artefactos secundarios. El objeto `secondaryArtifacts` usa muchos de los mismos valores que utiliza el objeto [the section called "artifacts"](#).

cache

Obligatorio. A [ProjectCache](#) que contiene información sobre la configuración de la memoria caché de este proyecto de compilación. Para obtener más información, consulte [Almacenamiento en caché de compilaciones \(p. 229\)](#).

environment

Obligatorio. A [Entorno del proyecto](#) que contiene información sobre la configuración del entorno de compilación de este proyecto. Esta configuración incluye:

ambiente/type

Obligatorio. El tipo del entorno de compilación. Para obtener más información, consulte [type](#) en la Referencia de la API de código de.

ambiente/image

Obligatorio. El identificador de imagen de Docker usado por este entorno de compilación. Normalmente, este identificador se expresa como `imagen-name:etiqueta`. Por ejemplo, en el repositorio de Docker que CodeBuild utiliza para administrar sus imágenes de Docker, este podría ser `aws/codebuild/standard:4.0`. En Docker Hub, `maven:3.3.9-jdk-8`. En Amazon ECR, `account-id.dkr.ecr.region-id.amazonaws.com/your-Amazon-ECR-repo-name:tag`. Para obtener más información, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).

ambiente/computeType

Obligatorio. Especifica los recursos informáticos utilizados por este entorno de compilación. Para obtener más información, consulte [computeType](#) en la Referencia de la API de código de.

ambiente/certificado

Opcional. El ARN del bucket de Amazon S3, el prefijo de ruta y la clave de objeto que contiene el certificado codificado en PEM. La clave de objeto puede ser únicamente el archivo .pem o un archivo .zip que contenga el certificado codificado en PEM. Por ejemplo, si el nombre del bucket de Amazon S3 es my-bucket, tu prefijo de ruta es cert, y el nombre de la clave de objeto es certificate.pem, a continuación, formatos aceptables para certificate son my-bucket/cert/certificate.pem o arn:aws:s3:::my-bucket/cert/certificate.pem.

ambiente/EnvronVariables

Opcional. Una matriz de [EnvironmentVariable](#) objetos que contienen las variables de entorno que desea especificar para este entorno de compilación. Cada variable de entorno se expresa como un objeto que contiene un name, value, y type de name, value, y type.

Consola y AWS CLI los usuarios pueden ver todas las variables de entorno. Si no le preocupa que la variable de entorno esté visible, establezca name y value, y set type de aPLAINTEXT.

Le recomendamos que almacene variables de entorno con valores sensibles, como un AWSID de clave de acceso, un AWS clave de acceso secreta, o contraseña, como un parámetro en el almacén de parámetros Amazon EC2 Systems Manager o AWS Secrets Manager. Para name, para ese parámetro almacenado, defina un identificador de CodeBuild a que haga referencia.

Si utiliza Amazon EC2 Systems Manager Parameter Store de, para value, defina el nombre del parámetro tal como está almacenado en el almacén de parámetros. Establezca type en PARAMETER_STORE. Uso de un parámetro denominado /CodeBuild/dockerLoginPassword como ejemplo, establezca name de a LOGIN_PASSWORD. Establezca value en /CodeBuild/dockerLoginPassword. Establezca type en PARAMETER_STORE.

Important

Si utiliza el almacén de parámetros de Amazon EC2 Systems Manager, le recomendamos que almacene los parámetros con nombres de parámetros que comiencen por /CodeBuild/ (por ejemplo, /CodeBuild/dockerLoginPassword). Puede utilizar la consola de CodeBuild para crear un parámetro en Amazon EC2 Systems Manager. Seleccione Create a parameter (Crear parámetro) y siga las instrucciones del cuadro de diálogo. (En ese cuadro de diálogo, para Clave de KMS, puede especificar el ARN de un AWS KMS clave en tu cuenta. Amazon EC2 Systems Manager utiliza esta clave para cifrar el valor del parámetro durante el almacenamiento y descifrarlo durante la recuperación). Si utiliza la consola de CodeBuild para crear un parámetro, la consola asigna al principio del nombre del parámetro con /CodeBuild/ ya que se está almacenando. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial de la consola del almacén de parámetros de Systems Manager](#) en la Guía del usuario de Amazon EC2 Systems Manager.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager, el rol de servicio del proyecto de compilación debe permitir `iam:GetParameters` acción. Si eligió Nuevo rol de servicio anteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó Existing service role (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager con nombres de parámetros que no empiezan por /CodeBuild/ y eligió Nuevo rol de servicio, debe actualizar ese rol de servicio para permitir el acceso a nombres de parámetros que no empiezan por /CodeBuild/. Esto es así porque el rol de servicio permite el acceso únicamente a los nombres de parámetro que empiezan por /CodeBuild/.

Si elige Nuevo rol de servicio, el rol de servicio incluye permisos para descifrar todos los parámetros en el /CodeBuild/ namespace en el almacén de parámetros Amazon EC2 Systems Manager.

Las variables de entorno que defina reemplazan las variables de entorno existentes. Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y establece una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`. Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y establece una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No establezca variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad.
- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad.
- El valor en la declaración de especificación de compilación es el que menos prioridad tiene.

Si utilizas Secrets Manager, `paravalue`, defina el nombre del parámetro tal como está almacenado en Secrets Manager. Establezca `type` en `SECRETS_MANAGER`. Uso de un secreto llamado `/CodeBuild/dockerLoginPassword` como ejemplo, establezca `name` de `aLOGIN_PASSWORD`. Establezca `value` en `/CodeBuild/dockerLoginPassword`. Establezca `type` en `SECRETS_MANAGER`.

Important

Si utiliza Secrets Manager, le recomendamos que almacene secretos con nombres que comiencen por `/CodeBuild/` (por ejemplo, `/CodeBuild/dockerLoginPassword`). Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Si el proyecto de compilación hace referencia a secretos almacenados en Secrets Manager, el rol de servicio del proyecto de compilación debe permitir la `secretsmanager:GetSecretValue` acción. Si eligió `Nuevo` rol de servicio anteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó `Existing service role` (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado.

Si tu proyecto de compilación hace referencia a secretos almacenados en Secrets Manager con nombres secretos que no empiezan por `/CodeBuild/` y eligió `Nuevo` rol de servicio, debe actualizar el rol de servicio para permitir el acceso a nombres secretos que no empiezan por `/CodeBuild/`. Esto es así porque el rol de servicio permite el acceso únicamente a nombres secretos que empiezan por `/CodeBuild/`.

Si elige `Nuevo` rol de servicio, el rol de servicio incluye permisos para descifrar todos los secretos en el `/CodeBuild/` espacio de nombres en Secrets Manager.

ambiente/Credencial de registro

Opcional. [ARegistryCredential](#) que especifica las credenciales que proporcionan acceso a un registro de Docker privado.

Entorno/Credencial de registro/Credenciales de

Especifica el ARN o nombre de las credenciales creadas mediante `AWS Managed Services`. Puede utilizar el nombre de las credenciales solo si existen en su región actual.

Entorno/Credencial de registro/CredentialProvider

El único valor válido es `SECRETS_MANAGER`.

Cuando se ha establecido:

- `imagePullCredentials` se debe establecer en `SERVICE_ROLE`.
- La imagen no puede ser una imagen preparada ni una imagen de Amazon ECR.

ambiente/imagePullCredentialsType

Opcional. Tipo de credenciales que CodeBuild utiliza para extraer imágenes de la compilación. Hay dos valores válidos:

CODEBUILD

CODEBUILD especifica que CodeBuild utiliza sus propias credenciales. Debe editar la política de repositorio de Amazon ECR para que confíe en el principal de servicio de CodeBuild.

SERVICE_ROLE

Especifica que CodeBuild utiliza el rol de servicio del proyecto de compilación.

Cuando utilice una imagen de registro entre cuentas o privada, debe usar credenciales de `SERVICE_ROLE`. Cuando utilice una imagen preparada de CodeBuild, debe utilizar `CODEBUILD` Credenciales de .

ambiente/privilegedMode

Establecer `true` únicamente si tiene previsto usar este proyecto de compilación para compilar imágenes de Docker y la imagen del entorno de compilación que especificó no es una de las proporcionadas por CodeBuild con compatibilidad con Docker. De lo contrario, todas las compilaciones asociadas que intenten interactuar con el daemon de Docker producirán un error. También debe iniciar el daemon de Docker para que las compilaciones puedan interactuar con él. Una forma de hacerlo es iniciar el daemon Docker en la fase de `install` de su archivo `buildspec` ejecutando los siguientes comandos de compilación. No ejecute estos comandos si especificó una imagen del entorno de compilación proporcionada por CodeBuild compatible con Docker.

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

```
- nohup /usr/local/bin/dockerd --host=unix:///var/run/docker.sock --  
host=tcp://127.0.0.1:2375 --storage-driver=overlay2 &  
- timeout 15 sh -c "until docker info; do echo .; sleep 1; done"
```

serviceRole

Obligatorio. El ARN del rol de servicio que usa CodeBuild para interactuar con los servicios en nombre del usuario de IAM (por ejemplo, `arn:aws:iam::account-id:role/role-name`).

timeoutInMinutes

Opcional. El número de minutos comprendidos entre 5 y 480 (8 horas) tras los cuales CodeBuild detendrá la compilación si no se ha completado. Si no se especifica, se usa el valor predeterminado de 60. Para determinar si y cuándo CodeBuild ha detenido una compilación debido a que se ha agotado el tiempo de espera, ejecute `elbatch-get-builds` comando. Para determinar si la compilación se ha detenido, busque en la salida un valor de `buildStatus` de `FAILED`. Para determinar cuándo se ha agotado el tiempo de espera de la compilación, busque en la salida el valor de `endTime` asociado a un valor de `phaseStatus` de `TIMED_OUT`.

queuedTimeoutInMinutes

Opcional. El número de minutos comprendidos entre 5 y 480 (8 horas) tras los cuales CodeBuild detendrá la compilación si sigue en cola. Si no se especifica, se usa el valor predeterminado de 60.

encryptionKey

Opcional. El alias o ARN del AWS KMS key utilizado por CodeBuild para cifrar la salida de compilación. Si especifica un alias, utilice el formato `arn:aws:kms:region-ID:account-ID:key/key-ID` o si existe un alias, utilice el formato `alias/key-alias`. Si no se especifica, el AWS- se utiliza la clave KMS administrada para Amazon S3.

tags

Opcional. Una matriz de [Etiquetas](#) objetos que proporcionan las etiquetas que desea asociar a este proyecto de compilación. Puede especificar hasta 50 etiquetas. Estas etiquetas se pueden utilizar en cualquier AWS que admite etiquetas de proyecto de compilación de CodeBuild. Cada etiqueta se expresa como un objeto con un `key` y un `value`.

vpcConfig

Opcional. A [VpcConfig](#) que contiene información sobre la configuración de VPC del proyecto de. Estas propiedades incluyen:

vpclId

Obligatorio. El ID de la VPC que usa CodeBuild. Ejecute este comando para obtener una lista de todos los ID de la VPC de su región:

```
aws ec2 describe-vpcs --region <region-ID>
```

subredes

Obligatorio. Una matriz de los IDs de subred que incluyen recursos utilizados por CodeBuild. Para obtener estos ID, ejecute este comando:

```
aws ec2 describe-subnets --filters "Name=vpc-id,Values=<vpc-id>" --region <region-ID>
```

securityGroupIds

Obligatorio. Un conjunto de ID de grupo de seguridad utilizado por CodeBuild para permitir el acceso a los recursos de la VPC. Para obtener estos ID, ejecute este comando:

```
aws ec2 describe-security-groups --filters "Name=vpc-id,Values=<vpc-id>" --<region-ID>
```

badgeEnabled

Opcional. Especifica si se deben incluir insignias de compilación en el proyecto CodeBuild. Establecer `true` para habilitar insignias de compilación, o `false` de lo contrario, . Para obtener más información, consulte [Ejemplo de insignias de compilación con Code \(p. 78\)](#).

logsConfig

A [LogsConfig](#) que contiene información sobre el lugar en que se encuentran los registros de esta compilación.

LogsConfig/CloudWatchLogs

A [CloudWatchLogsConfig](#) que contiene información sobre cómo enviar registros a CloudWatch Logs.

LogsConfig/S3Logs

Un registro [S3LogsConfig](#) que contiene información acerca de la inserción de registros a Amazon S3.

`fileSystemLocations`

Opcional. Una matriz de [Ubicación de los sistemas de archivos del proyecto](#) objetos que contienen información sobre la configuración de Amazon EFS.

`buildBatchConfig`

Opcional. La `buildBatchConfig` object es un [ProjectBuildBatchConfig](#) estructura que contiene la información de configuración de compilación por lotes del proyecto.

`BuildBatchConfig/serviceRole`

El ARN del rol de servicio para el proyecto de compilación por lotes.

`BuildBatchConfig/CombineArtifacts`

Un valor booleano que especifica si se van a combinar los artefactos de compilación de la compilación por lotes en una única ubicación de artefacto.

`Configuración/Restricciones de compilación por lotes/MaximumBuilds permitidos`

El número máximo de compilaciones permitidas.

`Configuración/Restricciones de compilación por lotes/ComputeTypesAllowed`

Una matriz de cadenas que especifican los tipos de computación permitidos para la compilación por lotes. Consulte [Tipos de computación del entorno de compilación](#) para estos valores.

`BuildBatchConfig/timeoutInMinutes`

La cantidad máxima de tiempo, en minutos, en que se debe completar la compilación por lotes.

`BuildBatchConfig/Modo de informe por lotes`

Especifica cómo se envían los informes de estado de compilación al proveedor de origen para la compilación por lotes. Los valores válidos son:

`REPORT_AGGREGATED_BATCH`

(Predeterminado) Agrega todos los estados de compilación en un único informe de estado.

`REPORT_INDIVIDUAL_BUILDS`

Envía un informe de estado independiente para cada compilación individual.

`concurrentBuildLimit`

Establece el número máximo de versiones simultáneas que están permitidas para este proyecto.

Las nuevas versiones solo se inician si el número actual de versiones es menor o igual a este límite. Si el recuento de versiones actual cumple con este límite, las nuevas versiones se limitan y no se ejecutan.

Creación del proyecto

Para crear el proyecto, ejecute el `create-project` de nuevo, pasando su archivo JSON:

```
aws codebuild create-project --cli-input-json file://<json-file>
```

Si tiene éxito, la representación JSON de un [Proyecto de](#) aparece en la salida de la consola. Consulte [la Sintaxis de la respuesta CreateProject](#) para ver un ejemplo de estos datos.

Excepto el nombre del proyecto de compilación, puede cambiar cualquiera de los ajustes del proyecto de compilación más adelante. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267).

Para empezar a ejecutar una compilación, consulte [Ejecutar una compilación \(AWS CLI\)](#) (p. 282).

Si su código fuente está almacenado en un repositorio de GitHub y desea que CodeBuild lo vuelva a compilar siempre que se envíe un cambio de código al repositorio, consulte [Iniciar la ejecución de compilaciones automáticamente \(AWS CLI\)](#) (p. 288).

Crear un proyecto de compilación (SDK de AWS)

Para obtener información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Crear un proyecto de compilación (AWS CloudFormation)

Para obtener más información acerca del uso de AWS CodeBuild por AWS CloudFormation, consulte [el AWS CloudFormation plantilla para CodeBuild](#) en la AWS CloudFormation Guía del usuario de.

Creación de una regla de notificación

Puede utilizar reglas de notificación para notificar a los usuarios cuando se producen cambios importantes, como éxitos y errores de compilación. Las reglas de notificación especifican tanto los eventos como el tema de Amazon SNS que se utiliza para enviar notificaciones. Para obtener más información, consulte [¿Qué son las notificaciones?](#)

Puede utilizar la consola o la AWS CLI para crear reglas de notificación para AWS CodeBuild.

Para crear una regla de notificación (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. Elija Build (Compilar), elija Build projects (Proyectos de compilación), y, a continuación, elija un proyecto de compilación en el que desee agregar notificaciones.
3. En la página del proyecto de compilación, elija Notify (Notificar) y, a continuación, elija Create notification rule (Crear regla de notificación). También puede ir a la página Settings (Configuración) del proyecto de compilación y elegir Create notification rule (Crear regla de notificación).
4. En Notification name (Nombre de la notificación), introduzca un nombre para la regla.
5. En Detail type (Tipo de detalle), elija Basic (Básico) si desea que solo la información proporcionada en Amazon EventBridge se incluya en la notificación. Seleccionar Complete si desea incluir la información proporcionada a Amazon EventBridge y otra información que CodeBuild o el administrador de notificaciones podría suministrar.

Para obtener más información, consulte [Descripción del contenido y la seguridad de las notificaciones](#).

6. En Events that trigger notifications (Eventos que activan notificaciones), seleccione los eventos para los que desea enviar notificaciones. Para obtener más información, consulte [Eventos de reglas de notificación en proyectos de compilación](#).
7. En Targets (Destinos), realice una de las siguientes operaciones:
 - Si ya ha configurado un recurso para utilizarlo con notificaciones, en Choose target type (Elegir tipo de destino) elija AWS Chatbot (Slack) o SNS topic (Tema de SNS). En Seleccione objetivo, elija el nombre del cliente (para un cliente Slack configurado en AWS Chatbot) o el nombre de recurso de Amazon (ARN) del tema Amazon SNS (para los temas de Amazon SNS que ya están configurados con la política necesaria para las notificaciones).

- Si no ha configurado un recurso para utilizarlo con notificaciones, elija Create target (Crear destino) y, a continuación, elija SNS topic (Tema de SNS). Indique el nombre del tema después de codestar-notifications- y, a continuación, elija Create (Crear).

Note

- Si crea el tema de Amazon SNS durante la creación de la regla de notificación, se aplica la política que permite a la característica de notificaciones publicar eventos en el tema. El uso de un tema creado para las reglas de notificación lo ayuda a garantizar que solo suscriba a los usuarios que desea recibir notificaciones sobre este recurso.
 - No se puede crear un cliente de AWS Chatbot durante la creación de una regla de notificación. Si elige AWS Chatbot (Slack), aparecerá un botón que le llevará a configurar un cliente en AWS Chatbot. Al elegir esa opción, se abrirá la consola de AWS Chatbot. Para obtener más información, consulte [Configurar la integración entre notificaciones y AWS Chatbot](#).
 - Si desea utilizar un tema de Amazon SNS ya existente como destino, debe agregar la política necesaria para AWS CodeStar Notifications además de otras políticas que puedan existir para ese tema. Para obtener más información, consulte [Configuración de temas de Amazon SNS para notificaciones](#) y [Descripción del contenido y la seguridad de las notificaciones](#).
8. Para terminar de crear la regla, elija Submit (Enviar).
 9. Debe suscribir a los usuarios al tema de Amazon SNS para la regla antes de que puedan recibir notificaciones. Para obtener más información, consulte [Suscribir usuarios a temas de Amazon SNS que son destinos](#). También puede configurar la integración entre notificaciones y AWS Chatbot para enviar notificaciones a las salas de chat de Amazon Chime. Para obtener más información, vea [Configurar la integración entre notificaciones y AWS Chatbot](#).

Para crear una regla de notificación (AWS CLI)

1. En un terminal o símbolo del sistema, ejecute el comando create-notification rule para generar el esqueleto JSON:

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

Puede asignar al archivo el nombre que desee. En este ejemplo, el archivo se denomina *rule.json*.

2. Abra el archivo JSON en un editor de texto sin formato y editelo para incluir el recurso, los tipos de eventos y el destino que desea para la regla. En el siguiente ejemplo se muestra una regla de notificación denominada *MyNotificationRule* para un proyecto de compilación denominado *MyBuildProject* en un AWS cuenta con el ID *123456789012*. Las notificaciones se envían con el tipo de detalle completo a un tema Amazon SNS denominado *Codestar - Notificaciones - Mi tema de notificación* cuando las compilaciones se realizan correctamente:

```
{
  "Name": "MyNotificationRule",
  "EventTypeIds": [
    "codebuild-project-build-state-succeeded"
  ],
  "Resource": "arn:aws:codebuild:us-east-2:123456789012:MyBuildProject",
  "Targets": [
    {
      "TargetType": "SNS",
      "TargetAddress": "arn:aws:sns:us-east-2:123456789012:codestar-
notificaciones-MyNotificationTopic"
    }
  ]
}
```



```
],  
  "Status": "ENABLED",  
  "DetailType": "FULL"  
}
```

Guarde el archivo.

3. Mediante el archivo que acaba de modificar, en el terminal o línea de comandos, vuelva a ejecutar el comando `create-notification-rule` para crear la regla de notificación:

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

4. Si se ejecuta correctamente, el comando devuelve el ARN de la regla de notificación, similar a lo siguiente:

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

Ver una lista de nombres de proyectos de compilación en AWS CodeBuild

Puede utilizar el [AWS CodeBuild Consola](#) de, [AWS CLI](#), o bien [AWS SDK](#) para ver una lista de proyectos de compilación en CodeBuild.

Temas

- [Ver una lista de nombres de proyectos de compilación \(consola\)](#) (p. 225)
- [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) (p. 225)
- [Ver una lista de nombres de proyectos de compilación \(SDK de AWS\)](#) (p. 226)

Ver una lista de nombres de proyectos de compilación (consola)

Puede ver una lista de proyectos de compilación en una región de AWS en la consola. La información incluye el nombre, el proveedor de origen, el repositorio, el estado de compilación más reciente y la descripción, en su caso.

1. Abra el icono [AWS CodeBuild Consola](https://console.aws.amazon.com/codesuite/codebuild/home) de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija **Build projects**.

Note

De forma predeterminada, solo se muestran los 10 últimos proyectos de compilación. Para ver más proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para **Projects per page** (Proyectos por página) o use las flechas atrás y adelante.

Ver una lista de nombres de proyectos de compilación (AWS CLI)

Ejecute el comando `list-projects`:

```
aws codebuild list-projects --sort-by sort-by --sort-order sort-order --next-token next-token
```


En el comando anterior, sustituya los siguientes marcadores de posición:

- **ordenación por**: Cadena opcional que se utiliza para indicar el criterio que se utiliza para mostrar los nombres de proyectos de compilación. Los valores válidos son:
 - **CREATED_TIME**: muestra los nombres de proyectos de compilación en función de la fecha de creación de cada proyecto.
 - **LAST_MODIFIED_TIME**: muestra los nombres de proyectos de compilación en función de la fecha de última modificación de cada proyecto.
 - **NAME**: muestra los nombres de proyectos de compilación en función del nombre de cada proyecto.
- **orden de clasificación**: Cadena opcional que se utiliza para indicar el orden en que se muestran los proyectos de compilación, en función del **ordenación por**. Los valores válidos son **ASCENDING** y **DESCENDING**.
- **siguiente token**: Cadena opcional. Durante una ejecución anterior, si hubiera más de 100 elementos en la lista, solo se devolverían los 100 primeros, junto con una única cadena denominada next token. Para obtener el siguiente lote de elementos de la lista, ejecute de nuevo este comando añadiendo el siguiente token a la llamada. Para obtener todos los elementos de la lista, siga ejecutando el comando con cada uno de los siguientes tokens hasta que no se devuelvan más tokens.

Por ejemplo, si ejecuta este comando:

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
  "nextToken": "Ci33ACF6...The full token has been omitted for brevity...U+AkMx8=",
  "projects": [
    "codebuild-demo-project",
    "codebuild-demo-project2",
    ... The full list of build project names has been omitted for brevity ...
    "codebuild-demo-project99"
  ]
}
```

Si ejecuta este comando de nuevo:

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING --next-token
Ci33ACF6...The full token has been omitted for brevity...U+AkMx8=
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
  "projects": [
    "codebuild-demo-project100",
    "codebuild-demo-project101",
    ... The full list of build project names has been omitted for brevity ...
    "codebuild-demo-project122"
  ]
}
```

Ver una lista de nombres de proyectos de compilación (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS \(p. 403\)](#).

Ver los detalles de un proyecto de compilación en AWS CodeBuild

Puede utilizar el [AWS CodeBuild Consola](#), el [AWS CLI](#), o bien el [AWS SDK](#) para ver los detalles de un proyecto de compilación en CodeBuild.

Temas

- [Ver los detalles de un proyecto de compilación \(consola\)](#) (p. 227)
- [Ver los detalles de un proyecto de compilación \(AWS CLI\)](#) (p. 227)
- [Ver los detalles de un proyecto de compilación \(SDK de AWS\)](#) (p. 229)

Ver los detalles de un proyecto de compilación (consola)

1. Abra el icono [AWS CodeBuild Consola](https://console.aws.amazon.com/codesuite/codebuild/home) de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.

Note

De forma predeterminada, solo se muestran los 10 últimos proyectos de compilación. Para ver más proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Projects per page (Proyectos por página) o use las flechas atrás y adelante.

3. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el enlace del proyecto de compilación.
4. En la página Build project: (Proyecto de compilación:) **nombre-proyecto**, elija Build details (Detalles de compilación).

Ver los detalles de un proyecto de compilación (AWS CLI)

Ejecute el comando batch-get-projects:

```
aws codebuild batch-get-projects --names names
```

En el comando anterior, sustituya el siguiente marcador de posición:

- **names**: Cadena obligatoria utilizada para indicar uno o varios nombres de proyectos de compilación para los que desea ver los detalles. Para especificar varios proyectos de compilación, separe el nombre de cada proyecto de compilación con un espacio. Puede especificar hasta 100 nombres de proyecto de compilación. Para obtener una lista de proyectos de compilación, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) (p. 225).

Por ejemplo, si ejecuta este comando:

```
aws codebuild batch-get-projects --names codebuild-demo-project codebuild-demo-project2 my-other-demo-project
```

Un resultado similar al siguiente podría aparecer en la salida. Los puntos suspensivos (...) se utilizan para representar datos omitidos por razones de brevedad.

```
{  
  "projectsNotFound": [  

```

```
    "my-other-demo-project"
  ],
  "projects": [
    {
      ...
      "name": "codebuild-demo-project",
      ...
    },
    {
      ...
      "name": "codebuild-demo-project2",
      ...
    }
  ]
}
```

En el resultado anterior, la matriz `projectsNotFound` muestra todos los nombres de proyectos de compilación especificados, pero no encontrados. La matriz `projects` muestra los detalles de cada proyecto de compilación donde se encontró información. Los detalles del proyecto de compilación se han omitido del resultado anterior por razones de brevedad. Para obtener más información, consulte el resultado de [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

`Labatch-get-projects` no admite el filtrado de determinados valores de propiedades, pero puede escribir un script que enumere las propiedades de un proyecto. Por ejemplo, el siguiente script de shell de Linux enumera los proyectos de la región actual de la cuenta corriente e imprime la imagen utilizada por cada proyecto.

```
#!/usr/bin/sh

# This script enumerates all of the projects for the current account
# in the current region and prints out the image that each project is using.

imageName=""

function getImageName(){
    local environmentValues=(${1//'\t'/ })
    imageName=${environmentValues[1]}
}

function processProjectInfo() {
    local projectInfo=$1

    while IFS=$'\t' read -r section value; do
        if [[ "$section" == *"ENVIRONMENT"* ]]; then
            getImageName "$value"
        fi
    done <<< "$projectInfo"
}

# Get the list of projects.
projectList=$(aws codebuild list-projects --output=text)

for projectName in $projectList
do
    if [[ "$projectName" != *"PROJECTS"* ]]; then
        echo "====="

        # Get the detailed information for the project.
        projectInfo=$(aws codebuild batch-get-projects --output=text --names "$projectName")

        processProjectInfo "$projectInfo"

        printf 'Project "%s" has image "%s"\n' "$projectName" "$imageName"
    fi
done
```

done

Para obtener más información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos](#) (p. 402).

Ver los detalles de un proyecto de compilación (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Almacenamiento en caché de compilaciones de AWS CodeBuild

Puede ahorrar tiempo si compila su proyecto utilizando una caché. Una caché puede almacenar fragmentos reutilizables del entorno de compilación y utilizarlos en diferentes compilaciones. Su proyecto de compilación puede utilizar uno de los dos tipos de almacenamiento en caché: Amazon S3 o local. Si utiliza una caché local, debe elegir uno o varios de los tres modos disponibles: caché de origen, caché de capas de Docker y caché personalizada.

Note

El modo de caché de capas de Docker solamente está disponible en los entornos Linux. Si elige este modo, debe ejecutar la compilación en modo con privilegios. Los proyectos de CodeBuild concedidos en modo privilegiado otorgan acceso a su contenedor a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

Temas

- [Almacenamiento en caché de Amazon S3](#) (p. 229)
- [Almacenamiento en la caché local](#) (p. 229)

Almacenamiento en caché de Amazon S3

El almacenamiento en caché de Amazon S3 almacena la caché en un bucket de Amazon S3 disponible en varios hosts de compilación. Esta opción resulta adecuada para artefactos de compilación de tamaño pequeño o intermedio que son más costosos de compilar que de descargar. Sin embargo, no es la mejor opción para los artefactos de compilación grandes, ya que tardan bastante tiempo en transferirse a través de la red, lo que puede afectar al rendimiento de la compilación. Tampoco es la mejor opción si usa capas de Docker.

Almacenamiento en la caché local

El almacenamiento en caché local aloja en una caché localmente en un host de compilación, que es el único host de compilación para el que está disponible la caché. Esta opción resulta adecuada para artefactos de compilación intermedios o grandes porque la caché está disponible de forma inmediata en el host de compilación. Esta no es la mejor opción si no hace muchas compilaciones. De este modo, el rendimiento de la compilación no se verá afectado por el tiempo de transferencia de la red. Si elige la opción de almacenamiento en caché local, debe elegir uno o varios de los siguientes modos de caché:

- El modo de caché de origen almacena en la caché los metadatos de Git del origen principal y los orígenes secundarios. Una vez que se ha creado la caché, las compilaciones que se realicen posteriormente solo extraerán los cambios realizados entre las confirmaciones. Este modo es conveniente para los proyectos que tienen un directorio de trabajo limpio y un origen que es un gran repositorio de Git. Si elige esta opción y el proyecto no utiliza un repositorio de Git (GitHub, GitHub Enterprise Server o Bitbucket), esta opción no se tiene en cuenta.

- El modo de caché de capas de Docker almacena en caché las capas de Docker existentes. Este modo es conveniente para los proyectos que compilan o extraen imágenes grandes de Docker. Puede evitar los problemas de rendimiento derivados de extraer imágenes grandes de Docker de la red.

Note

- Las cachés de capas de Docker solo pueden utilizarse en entornos Linux.
- Es necesario establecer la marca `privileged` para que el proyecto tenga los permisos de Docker necesarios.

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

- Debe tener en cuenta la implicación de seguridad antes de utilizar una caché de capas de Docker.
- El modo de caché personalizada almacena en caché los directorios que se especifican en el archivo `buildspec`. Este modo es conveniente si el escenario de compilación no se ajusta a ninguno de los otros dos modos de caché local. Si utiliza una caché personalizada:
 - Solo se pueden especificar directorios para el almacenamiento en caché. No se pueden especificar archivos individuales.
 - Para hacer referencia a los directorios almacenados en la caché, se utiliza Symlinks.
 - Los directorios de la caché se asocian a la compilación antes de que se descarguen los orígenes del proyecto. Los elementos almacenados en la memoria caché invalidan los elementos de origen si tienen el mismo nombre. Los directorios se especifican utilizando rutas de caché en la archivo `buildspec`. Para obtener más información, consulte [Sintaxis de buildspec](#) (p. 137).
 - Evite los nombres de directorio que sean los mismos en el origen y en la memoria caché. Los directorios almacenados localmente en la memoria caché pueden invalidar o eliminar el contenido de los directorios del repositorio de origen que tienen el mismo nombre.

Note

`LaARM_CONTAINER``Y``LINUX_GPU_CONTAINER`tipos de entorno y `BUILD_GENERAL1_2XLARGE`el tipo de cómputo no admite el uso de una caché local. Para obtener más información, consulte [Tipos de computación del entorno de compilación](#) (p. 169).

Temas

- [Especificar el almacenamiento en la caché local \(CLI\)](#) (p. 230)
- [Especificar el almacenamiento en la caché local \(consola\)](#) (p. 231)
- [Especificar el almacenamiento en la caché local \(AWS CloudFormation\)](#) (p. 233)

Puede utilizar el `AWS CLI`, consola, SDK o `AWS CloudFormation` para especificar una caché local.

Especificar el almacenamiento en la caché local (CLI)

Puede utilizar el parámetro `--cache` de la `AWS CLI` para especificar cada uno de los tres tipos de caché local.

- Para especificar una caché de origen:

```
--cache type=LOCAL,mode=[LOCAL_SOURCE_CACHE]
```

- Para especificar una caché de capas de Docker:

```
--cache type=LOCAL,mode=[LOCAL_DOCKER_LAYER_CACHE]
```

- Para especificar una caché personalizada:

```
--cache type=LOCAL,mode=[LOCAL_CUSTOM_CACHE]
```

Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\) \(p. 209\)](#).

Especificar el almacenamiento en la caché local (consola)

Puede especificar una caché en la sección Artifacts (Artefactos) de la consola. Para Tipo de caché, elige Amazon S3 o Local. Si elige Local, seleccione una o varias de las tres opciones de caché local.

Cache type

Local

Select one or more local

☐

Docker layer cache

Caches existing Docker

☐

Source cache

Caches .git metadata so

☐

Custom cache

Caches directories speci

Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).

Especificar el almacenamiento en la caché local (AWS CloudFormation)

Si utiliza AWS CloudFormation para especificar una caché local, en la propiedad `Cache`, en `Type`, especifique `LOCAL`. En el siguiente ejemplo de código de AWS CloudFormation con formato YAML, se especifican los tres tipos de caché local. Puede especificar cualquier combinación de tipos. Si utiliza una caché de capas de Docker, en `Environment`, debe establecer `PrivilegedMode` en `true` y `Type` en `LINUX_CONTAINER`.

```
CodeBuildProject:
  Type: AWS::CodeBuild::Project
  Properties:
    Name: MyProject
    ServiceRole: <service-role>
    Artifacts:
      Type: S3
      Location: myBucket
      Name: myArtifact
      EncryptionDisabled: true
      OverrideArtifactName: true
    Environment:
      Type: LINUX_CONTAINER
      ComputeType: BUILD_GENERAL1_SMALL
      Image: aws/codebuild/standard:5.0
      Certificate: bucket/cert.zip
      # PrivilegedMode must be true if you specify LOCAL_DOCKER_LAYER_CACHE
      PrivilegedMode: true
    Source:
      Type: GITHUB
      Location: <github-location>
      InsecureSsl: true
      GitCloneDepth: 1
      ReportBuildStatus: false
    TimeoutInMinutes: 10
    Cache:
      Type: LOCAL
      Modes: # You can specify one or more cache mode,
        - LOCAL_CUSTOM_CACHE
        - LOCAL_DOCKER_LAYER_CACHE
        - LOCAL_SOURCE_CACHE
```

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CloudFormation\)](#) (p. 223).

Crear desencadenadores de AWS CodeBuild

CrearAWS CodeBuilddesencadenadores (consola)

Puede crear un disparador en un proyecto para programar una compilación una vez cada hora, día o semana. También puede crear un disparador mediante una regla personalizada con una expresión cron de Amazon CloudWatch. Por ejemplo, mediante una expresión cron puede programar una compilación en un momento específico todos los días de la semana.

Note

No es posible iniciar una compilación por lotes desde un desencadenador de compilación, un evento de Amazon EventBridge o un AWS Step Function tarea.

Para crear un disparador

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Elija el enlace del proyecto de compilación al que desea agregar un disparador y, a continuación, elija la pestaña Build triggers (Disparadores de compilación).

Note

De forma predeterminada, se muestran los 100 últimos proyectos de compilación. Para ver más proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Projects per page (Proyectos por página) o use las flechas atrás y adelante.

4. Elija Create trigger.
5. Escriba un nombre en Trigger name (Nombre del disparador).
6. En la lista desplegable Frequency (Frecuencia), elija la frecuencia del disparador. Si desea crear la frecuencia con una expresión Cron, seleccione Custom (Personalizado).
7. Especifique los parámetros de frecuencia del disparador. Puede introducir los primeros caracteres de su selección en el cuadro de texto para filtrar los elementos del menú desplegable.

Note

Las horas de inicio y los minutos se basan en cero. El minuto de inicio es un número entre cero y 59. La hora de inicio es un número comprendido entre cero y 23. Por ejemplo, un disparador diario que comienza todos los días a las 12:15 h tiene una hora de inicio de 12 y un minuto de inicio de 15. Un disparador diario que comienza todos los días a medianoche tiene una hora de inicio cero y un minuto de inicio de cero. Un disparador diario que comienza todos los días a las 23:59 h tiene una hora de inicio de 23 y un minuto de inicio de 59.

Frecuencia	Parámetros requeridos	Detalles
Por hora	Minuto de inicio	Utilice el menú desplegable Start minute (Minuto de inicio).
Por día	Minuto de inicio Hora de inicio	Utilice el menú desplegable Start minute (Minuto de inicio). Utilice el menú desplegable Start hour (Hora de inicio).
Semanal	Minuto de inicio Hora de inicio Día de inicio	Utilice el menú desplegable Start minute (Minuto de inicio). Utilice el menú desplegable Start hour (Hora de inicio). Utilice el menú desplegable Start day (Día de inicio).
Personalizado	Expresión Cron	Introduzca una expresión Cron en Cron expression (Expresión Cron). Las expresiones Cron

Frecuencia	Parámetros requeridos	Detalles
		<p>tienen seis campos obligatorios, que están separados por un espacio en blanco. Los campos especifican un valor de inicio de minuto, hora, día del mes, día de la semana y año. Puede utilizar caracteres comodín para especificar un intervalo, valores adicionales, etc. Por ejemplo, la expresión Crono</p> <p>9 ? * MON-FRI * programa una compilación todos los días de la semana a las 9:00 a. m. Para obtener más información, consulte Expresiones Cronen laAmazon CloudWatch Events Guía del usuario.</p>

8. Seleccione Enable this trigger (Activar este disparador).
9. (Opcional) Expanda la sección Advanced (Avanzado). En Source version (Versión de código fuente), escriba una versión del código fuente.
 - En Amazon S3, introduzca el identificador de versión correspondiente a la versión del artefacto de entrada que desea compilar. Si la Source version (Versión del código fuente) se deja en blanco, se usará la versión más reciente.
 - En AWS CodeCommit, especifique un ID de confirmación. Si Source version (Versión del código fuente) se deja en blanco, se utilizará el ID de confirmación HEAD de la ramificación predeterminada.
 - En GitHub o GitHub Enterprise, especifique un ID de confirmación, un ID de solicitud de extracción, un nombre de ramificación o un nombre de etiqueta que se corresponda con la versión del código fuente que desea compilar. Si especifica un ID de solicitud de extracción, este debe tener el formato `pr/pull-request-ID` (por ejemplo, `pr/25`). Si especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si Source version (Versión del código fuente) se deja en blanco, se usa el ID de confirmación HEAD de la ramificación predeterminada.
 - En Bitbucket, especifique un ID de confirmación, un nombre de ramificación o un nombre de etiqueta que se corresponda con la versión de código fuente que desea compilar. Si especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si Source version (Versión del código fuente) se deja en blanco, se usa el ID de confirmación HEAD de la ramificación predeterminada.
10. (Opcional) Especifique un tiempo de espera entre 5 minutos y 480 minutos (8 horas). Este valor especifica durante cuánto tiempo AWS CodeBuild intenta ejecutar una compilación antes de detenerse. Si los campos Hours (Horas) y Minutes (Minutos) se dejan en blanco, se usa el valor de tiempo de espera predeterminado especificado en el proyecto.
11. Elija Create trigger.

CrearAWS CodeBuildse activa mediante programación

CodeBuild utiliza reglas de Amazon EventBridge para los disparadores de compilación. Puede utilizar la API de EventBridge para crear programáticamente activadores de compilación para sus proyectos de CodeBuild. Consulte [Referencia de API de Amazon EventBridge](#) para obtener más información.

Editar desencadenadores de AWS CodeBuild

EditarAWS CodeBuilddesencadenadores (consola)

Puede editar un disparador en un proyecto para programar una compilación una vez cada hora, día o semana. También puede editar un disparador para usar una regla personalizada con una expresión cron de Amazon CloudWatch. Por ejemplo, mediante una expresión cron puede programar una compilación en un momento específico todos los días de la semana. Para obtener información acerca de la creación de un disparador, consulte [Crear desencadenadores de AWS CodeBuild](#) (p. 233).

Para editar un desencadenador

1. Abra el iconoAWS CodeBuildConsola de:<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Elija el enlace del proyecto de compilación que desea cambiar y, a continuación, seleccione la pestaña Build triggers (Desencadenadores de compilación).

Note

De forma predeterminada, se muestran los 100 últimos proyectos de compilación. Para ver más proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Projects per page (Proyectos por página) o use las flechas atrás y adelante.

4. Elija el botón de opción junto al desencadenador que desea cambiar y, a continuación, elija Edit (Editar).
5. En la lista desplegable Frequency (Frecuencia), elija la frecuencia del disparador. Si desea crear la frecuencia con una expresión Cron, seleccione Custom (Personalizado).
6. Especifique los parámetros de frecuencia del disparador. Puede introducir los primeros caracteres de su selección en el cuadro de texto para filtrar los elementos del menú desplegable.

Note

Las horas de inicio y los minutos se basan en cero. El minuto de inicio es un número entre cero y 59. La hora de inicio es un número comprendido entre cero y 23. Por ejemplo, un disparador diario que comienza todos los días a las 12:15 h tiene una hora de inicio de 12 y un minuto de inicio de 15. Un disparador diario que comienza todos los días a medianoche tiene una hora de inicio cero y un minuto de inicio de cero. Un disparador diario que comienza todos los días a las 23:59 h tiene una hora de inicio de 23 y un minuto de inicio de 59.

Frecuencia	Parámetros requeridos	Detalles
Por hora	Minuto de inicio	Utilice el menú desplegable Start minute (Minuto de inicio).
Por día	Minuto de inicio	Utilice el menú desplegable Start minute (Minuto de inicio).
	Hora de inicio	Utilice el menú desplegable Start hour (Hora de inicio).
Semanal	Minuto de inicio	Utilice el menú desplegable Start minute (Minuto de inicio).
	Hora de inicio	
	Día de inicio	Utilice el menú desplegable Start hour (Hora de inicio).

Frecuencia	Parámetros requeridos	Detalles
		Utilice el menú desplegable Start day (Día de inicio).
Personalizado	Expresión Cron	Introduzca una expresión Cron en Cron expression (Expresión Cron). Las expresiones Cron tienen seis campos obligatorios, que están separados por un espacio en blanco. Los campos especifican un valor de inicio de minuto, hora, día del mes, día de la semana y año. Puede utilizar caracteres comodín para especificar un intervalo, valores adicionales, etc. Por ejemplo, la expresión Cron 0 9 ? * MON-FRI * programa una compilación todos los días de la semana a las 9:00 a. m. Para obtener más información, consulte Expresiones Cron en la Amazon CloudWatch Events Guía del usuario.

7. Seleccione Enable this trigger (Activar este disparador).

Note

Puede utilizar la consola Amazon CloudWatch en <https://console.aws.amazon.com/cloudwatch/> para editar la versión de origen, el tiempo de espera y otras opciones que no están disponibles en AWS CodeBuild.

Editar AWS CodeBuild se activa mediante programación

CodeBuild utiliza reglas de Amazon EventBridge para los disparadores de compilación. Puede utilizar la API de EventBridge para editar mediante programación los activadores de compilación de sus proyectos de CodeBuild. Consulte [Referencia de API de Amazon EventBridge](#) para obtener más información.

Uso de webhooks con AWS CodeBuild

AWS CodeBuild admite la integración webhook con GitHub, GitHub Enterprise Server y Bitbucket.

Temas

- [Prácticas recomendadas para utilizar webhooks con AWS CodeBuild \(p. 237\)](#)
- [Eventos de webhooks de Bitbucket \(p. 238\)](#)
- [Eventos de webhooks de GitHub \(p. 246\)](#)

Prácticas recomendadas para utilizar webhooks con AWS CodeBuild

Para los proyectos que utilizan repositorios públicos para configurar webhooks, le recomendamos las siguientes opciones:

Configuración `ACTOR_ACCOUNT_ID` filters

Añada `ACTOR_ACCOUNT_ID` filtros en los grupos de filtros de webhooks del proyecto para especificar qué usuarios pueden desencadenar una compilación. Cada evento webhook entregado a CodeBuild incluye información del remitente que especifica el identificador del actor. CodeBuild filtrará los webhooks según el patrón de expresión regular proporcionado en los filtros. Puede especificar los usuarios específicos a los que se les permite activar compilaciones con este filtro. Para obtener más información, consulte [Eventos de webhooks de GitHub \(p. 246\)](#) y [Eventos de webhooks de Bitbucket \(p. 238\)](#).

Configuración `FILE_PATH` filters

Añada `FILE_PATH` filtros a los grupos de filtros webhook de su proyecto para incluir o excluir los archivos que pueden desencadenar una compilación cuando se cambia. Por ejemplo, puede denegar las solicitudes de compilación de cambios en el `buildspec.yml` mediante un patrón de expresión regular, como `^buildspec.yml$`, junto con `excludeMatchedPattern` propiedad. Para obtener más información, consulte [Eventos de webhooks de GitHub \(p. 246\)](#) y [Eventos de webhooks de Bitbucket \(p. 238\)](#).

Reduzca el alcance de los permisos de su rol de compilación de IAM

Las compilaciones activadas por un webhook utilizan el rol de servicio de IAM especificado en el proyecto. Recomendamos configurar los permisos del rol de servicio en el conjunto mínimo de permisos necesarios para ejecutar la compilación. Por ejemplo, en un escenario de prueba e implementación, cree un proyecto para pruebas y otro proyecto para la implementación. El proyecto de prueba acepta compilaciones webhook del repositorio, pero no proporciona permisos de escritura a los recursos. El proyecto de implementación proporciona permisos de escritura a los recursos y el filtro webhook está configurado para permitir que los usuarios de confianza activen compilaciones únicamente.

Utilizar una especificación de compilación almacenada en línea o Amazon S3

Si define la especificación de compilación en línea dentro del propio proyecto o almacena el archivo `buildspec` en un bucket de Amazon S3, el archivo `buildspec` solo estará visible para el propietario del proyecto. Esto evita que las solicitudes de extracción realicen cambios de código en el archivo `buildspec` y desencadenen compilaciones no deseadas. Para obtener más información, consulte [ProjectSource.Buildspec](#) en la Referencia de la API de código de.

Eventos de webhooks de Bitbucket

Puede utilizar grupos de filtros de webhooks para especificar qué eventos de webhooks de Bitbucket van a desencadenar una compilación. Por ejemplo, puede especificar que una compilación se desencadene únicamente con cambios en ramificaciones específicas.

Puede especificar varios grupos de filtros de webhooks. Si los filtros de uno o varios grupos se evalúan como `true`, se desencadenará una compilación. Cuando cree un grupo de filtros, deberá especificar:

Un evento

En Bitbucket, puede seleccionar uno o varios de los siguientes eventos:

- `PUSH`
- `PULL_REQUEST_CREATED`
- `PULL_REQUEST_UPDATED`
- `PULL_REQUEST_MERGED`

El tipo de evento de webhook está en su encabezado en el campo `X-Event-Key`. En la siguiente tabla se muestra cómo los valores del encabezado `X-Event-Key` se asignan a los tipos de eventos.

Note

Debe habilitar el evento `merged` en su configuración de webhook de Bitbucket si crea un grupo de filtros de webhook que utilice el tipo de evento `PULL_REQUEST_MERGED`.

Valor del encabezado X-Event-Key	Tipo de evento
<code>repo:push</code>	<code>PUSH</code>
<code>pullrequest:created</code>	<code>PULL_REQUEST_CREATED</code>
<code>pullrequest:updated</code>	<code>PULL_REQUEST_UPDATED</code>
<code>pullrequest:fulfilled</code>	<code>PULL_REQUEST_MERGED</code>

Para `PULL_REQUEST_MERGED`, si una solicitud de extracción se fusiona con la estrategia squash y la rama de solicitud de extracción está cerrada, la confirmación de solicitud de extracción original ya no existe. En este caso, el campo `CODEBUILD_WEBHOOK_MERGE_COMMIT` variable de entorno contiene el identificador de la confirmación de combinación aplastada.

Uno o varios filtros opcionales

Utilice una expresión regular para especificar los filtros. En el caso de los eventos que desencadenan una compilación, todos los filtros asociados con ellos deben evaluarse como `true`.

ACTOR_ACCOUNT_ID(`ACTOR_ID` en la consola)

Un evento de webhook desencadena una compilación cuando el ID de una cuenta de Bitbucket coincide con el patrón de la expresión regular. Este valor se encuentra en la propiedad `account_id` del objeto `actor` de la carga del filtro de webhook.

HEAD_REF

Un evento de webhook desencadena una compilación cuando la referencia del encabezado coincide con el patrón de la expresión regular (por ejemplo, `refs/heads/branch-name` y `refs/tags/tag-name`). Un filtro `HEAD_REF` evalúa el nombre de referencia de Git de la ramificación o etiqueta. El nombre de la ramificación o la etiqueta se encuentra en el campo `name` del objeto `new` incluido en el objeto `push` de la carga del webhook. En el caso de los eventos de las solicitudes de extracción, el nombre de la ramificación se encuentra en el campo `name` del objeto `branch` incluido en el objeto `source` de la carga del webhook.

BASE_REF

Un evento de webhook desencadena una compilación cuando la referencia de base coincide con el patrón de la expresión regular. Un filtro `BASE_REF` solamente funciona con eventos de solicitudes de extracción (por ejemplo, `refs/heads/branch-name`). Un filtro `BASE_REF` evalúa el nombre de referencia de Git de la ramificación. El nombre de la ramificación se encuentra en el campo `name` del objeto `branch` incluido en el objeto `destination` de la carga del webhook.

FILE_PATH

Un webhook desencadena una compilación cuando la ruta de un archivo modificado coincide con el patrón de la expresión regular.

COMMIT_MESSAGE

Un webhook desencadena una compilación cuando el mensaje de confirmación del encabezado coincide con el patrón de la expresión regular.

Note

Puede encontrar la carga de webhook en la configuración de webhook del repositorio de Bitbucket.

Temas

- [Filtrar eventos de webhooks de Bitbucket \(consola\) \(p. 240\)](#)
- [Filtrar eventos de webhooks de Bitbucket \(SDK\) \(p. 243\)](#)
- [Filtrar eventos de webhooks de Bitbucket \(AWS CloudFormation\) \(p. 245\)](#)

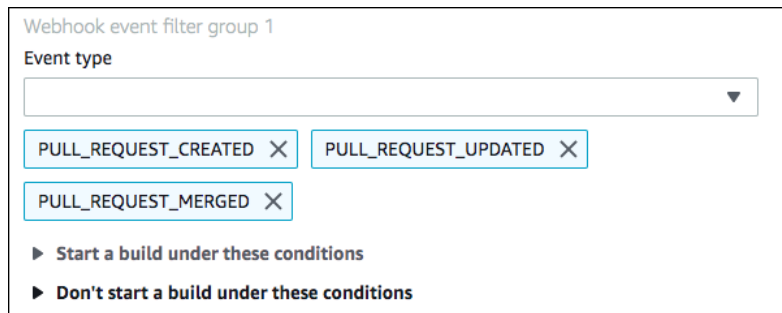
Filtrar eventos de webhooks de Bitbucket (consola)

Si desea utilizar la AWS Management Console para filtrar eventos de webhooks:

1. Cuando cree el proyecto, seleccione **Rebuild every time a code change is pushed to this repository** (Volver a compilar cada vez que se inserte un cambio de código en este repositorio).
2. En **Event type** (Tipo de evento), seleccione uno o varios eventos.
3. Para filtrar en función de cuándo un evento va a desencadenar una compilación, en **Start a build under these conditions** (Iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
4. Para filtrar en función de cuándo no se va a desencadenar un evento, en **Don't start a build under these conditions** (No iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
5. Seleccione **Add filter group** (Añadir grupo de filtros) para añadir otro grupo de filtros.

Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\) \(p. 198\)](#) y [WebhookFilter](#) en la Referencia de la API de AWS CodeBuild.

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación únicamente con solicitudes de extracción:



The screenshot shows the 'Webhook event filter group 1' configuration in the AWS Management Console. It features a dropdown menu for 'Event type' which is currently empty. Below the dropdown, there are three filter tags: 'PULL_REQUEST_CREATED', 'PULL_REQUEST_UPDATED', and 'PULL_REQUEST_MERGED', each with a close button (X). At the bottom, there are two expandable sections: 'Start a build under these conditions' and 'Don't start a build under these conditions', both currently collapsed.

Si utilizamos un ejemplo con dos grupos de filtros, la compilación se desencadenaría cuando uno de los grupos o los dos se evalúen como true:

- El primer grupo de filtros especifica las solicitudes de extracción que se crean o actualizan en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/main$` y las referencias de encabezado que coinciden con `^refs/heads/branch1!`.
- El segundo grupo de filtros especifica solicitudes de inserción en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/branch1$`.

Webhook event filter group 1

Event type
Add one or more a webhook event filter groups to specify which events trigger a new build. If you do not add a webhook event filter group, then a new build is triggered every time a code change is pushed to your repository.

PULL_REQUEST_CREATED X PULL_REQUEST_UPDATED X

▼ Start a build under these conditions

ACTOR_ID - optional HEAD_REF - optional BASE_REF - optional FILE_PATH - optional

^refs/heads/branch1\$ ^refs/heads/main\$

COMMIT_MESSAGE - optional

► Don't start a build under these conditions

Webhook event filter group 2

Remove filter group

Event type
Add one or more a webhook event filter groups to specify which events trigger a new build. If you do not add a webhook event filter group, then a new build is triggered every time a code change is pushed to your repository.

PUSH X

▼ Start a build under these conditions

ACTOR_ID - optional HEAD_REF - optional BASE_REF - optional FILE_PATH - optional

^refs/heads/branch1\$

COMMIT_MESSAGE - optional

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación de todas las solicitudes excepto los eventos de etiquetas.

Webhook event filter group 1

Event type

PUSH X PULL_REQUEST_CREATED X PULL_REQUEST_UPDATED X

PULL_REQUEST_MERGED X

► Start a build under these conditions

▼ Don't start a build under these conditions

ACTOR_ID - optional HEAD_REF - optional BASE_REF - optional FILE_PATH - optional

^refs/tags/.*

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación solo cuando se modifican los archivos cuyos nombres coinciden con la expresión regular `^buildspec.*`.

Webhook event filter group 1

Event type

PUSH X

▼ Start a build under these conditions

ACTOR_ID - *optional*

HEAD_REF - *optional*

BASE_REF - *optional*

FILE_PATH - *optional*

`^buildspec.*`

COMMIT_MESSAGE -
optional

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación únicamente cuando hay un cambio realizado por un usuario de Bitbucket que no tiene un ID de cuenta que coincida con la expresión regular `actor-account-id`.

Note

Para obtener información acerca de cómo encontrar el ID de cuenta de Bitbucket, consulte <https://api.bitbucket.org/2.0/users/nombre-usuario>, donde *nombre-usuario* es el nombre de usuario de Bitbucket.

Webhook event filter group 1

Event type

PUSH X PULL_REQUEST_CREATED X PULL_REQUEST_UPDATED X

PULL_REQUEST_MERGED X

▼ Start a build under these conditions

ACTOR_ID - <i>optional</i>	HEAD_REF - <i>optional</i>	BASE_REF - <i>optional</i>	FILE_PATH - <i>optional</i>
<input type="text" value="actor-account-id"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhook desencadena una compilación para un evento de inserción cuando el mensaje de confirmación de la cabeza coincide con la expresión regular `\[CodeBuild\]`.

Webhook event filter group 1

Event type

PUSH X

▼ Start a build under these conditions

ACTOR_ID - *optional*

HEAD_REF - *optional*

BASE_REF - *optional*

FILE_PATH - *optional*

COMMIT_MESSAGE -
optional

\[CodeBuild\]

► Don't start a build under these conditions

Filtrar eventos de webhooks de Bitbucket (SDK)

Si desea utilizar el SDK de AWS CodeBuild para filtrar eventos de webhooks, utilice el campo `filterGroups` de la sintaxis de la solicitud de los métodos de API `CreateWebhook` o `UpdateWebhook`. Para obtener más información, consulte [WebhookFilter](#) en la Referencia de la API de código de.

Si desea crear un filtro de webhook que desencadene una compilación únicamente con las solicitudes de extracción, inserte lo siguiente en la sintaxis de la solicitud:

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",  
      "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_MERGED"  
    }  
  ]  
]
```

Si desea crear un filtro de webhooks que desencadene una compilación únicamente con las ramificaciones especificadas, utilice el parámetro `pattern` para especificar una expresión regular que filtre los nombres de las ramificaciones. Si utilizamos un ejemplo con dos grupos de filtros, la compilación se desencadenaría cuando uno de los grupos o los dos se evalúen como `true`:

- El primer grupo de filtros especifica las solicitudes de extracción que se crean o actualizan en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/main$` y las referencias de encabezado que coinciden con `^refs/heads/myBranch$`.
- El segundo grupo de filtros especifica solicitudes de inserción en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/myBranch$`.

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",
```

```

    "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED"
  },
  {
    "type": "HEAD_REF",
    "pattern": "^refs/heads/myBranch$"
  },
  {
    "type": "BASE_REF",
    "pattern": "^refs/heads/main$"
  }
],
[
  {
    "type": "EVENT",
    "pattern": "PUSH"
  },
  {
    "type": "HEAD_REF",
    "pattern": "^refs/heads/myBranch$"
  }
]
]

```

Puede utilizar el parámetro `excludeMatchedPattern` para especificar qué eventos no desencadenan una compilación. En este ejemplo, se desencadena una compilación con todas las solicitudes, excepto los eventos de etiquetas.

```

"filterGroups": [
  [
    {
      "type": "EVENT",
      "pattern": "PUSH, PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_MERGED"
    },
    {
      "type": "HEAD_REF",
      "pattern": "^refs/tags/.*",
      "excludeMatchedPattern": true
    }
  ]
]

```

Puede crear un filtro que desencadene una compilación solo cuando un usuario de Bitbucket con el ID de cuenta `actor-account-id` realice algún cambio.

Note

Para obtener información acerca de cómo encontrar el ID de cuenta de Bitbucket, consulte <https://api.bitbucket.org/2.0/users/nombre-usuario>, donde *nombre-usuario* es el nombre de usuario de Bitbucket.

```

"filterGroups": [
  [
    {
      "type": "EVENT",
      "pattern": "PUSH, PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_MERGED"
    },
    {
      "type": "ACTOR_ACCOUNT_ID",
      "pattern": "actor-account-id"
    }
  ]
]

```

Puede crear un filtro que desencadene una compilación solo cuando se modifiquen los archivos cuyos nombres coincidan con la expresión regular del argumento `pattern`. En este ejemplo, el grupo de filtros especifica que la compilación solo debe desencadenarse cuando se modifiquen los archivos cuyos nombres coincidan con la expresión regular `^buildspec.*`.

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",  
      "pattern": "PUSH"  
    },  
    {  
      "type": "FILE_PATH",  
      "pattern": "^buildspec.*"  
    }  
  ]  
]
```

Puede crear un filtro que desencadene una compilación solo cuando el mensaje de confirmación del encabezado coincida con la expresión regular en el argumento `pattern`. En este ejemplo, el grupo de filtros especifica que una compilación se desencadena solo cuando el mensaje de confirmación del encabezado del evento de inserción coincide con la expresión regular `\[CodeBuild\]`.

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",  
      "pattern": "PUSH"  
    },  
    {  
      "type": "COMMIT_MESSAGE",  
      "pattern": "\[CodeBuild\]"  
    }  
  ]  
]
```

Filtrar eventos de webhooks de Bitbucket (AWS CloudFormation)

Si desea usar una plantilla de AWS CloudFormation para filtrar eventos de webhooks, utilice la propiedad `FilterGroups` del proyecto de AWS CodeBuild. El siguiente fragmento con formato YAML de una plantilla AWS CloudFormation crea dos grupos de filtros. Juntos, desencadenan una compilación cuando uno de los grupos o los dos se evalúan como `true`:

- El primer grupo de filtros especifica las solicitudes de extracción que un usuario de Bitbucket que no tiene el ID de cuenta 12345 crea o actualiza en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/main$`.
- El segundo grupo de filtros especifica solicitudes de inserción que se crean en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/.*`.
- El tercer grupo de filtros especifica una solicitud de inserción con un mensaje de confirmación del encabezado que coincida con la expresión regular `\[CodeBuild\]`.

```
CodeBuildProject:  
  Type: AWS::CodeBuild::Project  
  Properties:  
    Name: MyProject  
    ServiceRole: service-role  
    Artifacts:  
      Type: NO_ARTIFACTS
```

```
Environment:
  Type: LINUX_CONTAINER
  ComputeType: BUILD_GENERAL1_SMALL
  Image: aws/codebuild/standard:4.0
Source:
  Type: BITBUCKET
  Location: source-location
Triggers:
  Webhook: true
  FilterGroups:
    - Type: EVENT
      Pattern: PULL_REQUEST_CREATED,PULL_REQUEST_UPDATED
    - Type: BASE_REF
      Pattern: ^refs/heads/main$
      ExcludeMatchedPattern: false
    - Type: ACTOR_ACCOUNT_ID
      Pattern: 12345
      ExcludeMatchedPattern: true
    - Type: EVENT
      Pattern: PUSH
    - Type: HEAD_REF
      Pattern: ^refs/heads/.*
    - Type: EVENT
      Pattern: PUSH
    - Type: COMMIT_MESSAGE
    - Pattern: \[CodeBuild\]
```

Eventos de webhooks de GitHub

Puede utilizar grupos de filtros de webhooks para especificar qué eventos de webhooks de GitHub van a desencadenar una compilación. Por ejemplo, puede especificar que una compilación se desencadene únicamente con cambios en ramificaciones específicas.

Puede crear uno o varios grupos de filtros de webhooks para especificar qué eventos de webhooks van a desencadenar una compilación. Si todos los filtros de uno o varios grupos se evalúan como true, se desencadenará una compilación. Cuando cree un grupo de filtros, deberá especificar:

Un evento

En GitHub, puede seleccionar uno o varios de los siguientes eventos: PUSH, PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED y PULL_REQUEST_MERGED. El tipo de evento de webhook está en el encabezado X-GitHub-Event de la carga de webhook. En el encabezado X-GitHub-Event, es posible que vea pull_request o push. Para un evento de solicitud de extracción, el tipo está en el campo action de la carga del evento de webhook. En la siguiente tabla se muestra cómo los valores del encabezado X-GitHub-Event y los valores del campo action de la carga de solicitud de extracción de webhook se asignan a los tipos de eventos disponibles.

Valor del encabezado X-GitHub-Event	Valor action de la carga del evento de webhook	Tipo de evento
pull_request	opened	PULL_REQUEST_CREATED
pull_request	reopened	PULL_REQUEST_REOPENED
pull_request	synchronize	PULL_REQUEST_UPDATED
pull_request	closed y el campo merged es true	PULL_REQUEST_MERGED
push	n/a	PUSH

Note

El tipo de evento `PULL_REQUEST_REOPENED` solo se puede utilizar con GitHub y GitHub Enterprise Server.

Uno o varios filtros opcionales

Utilice una expresión regular para especificar los filtros. En el caso de los eventos que desencadenan una compilación, todos los filtros asociados con ellos deben evaluarse como `true`.

`ACTOR_ACCOUNT_ID`(`ACTOR_ID` en la consola)

Un evento de webhook desencadena una compilación cuando el ID de una cuenta del servidor de GitHub o GitHub Enterprise Server coincide con el patrón de la expresión regular. Este valor se encuentra en la propiedad `id` del objeto `sender` en la carga de webhook.

`HEAD_REF`

Un evento de webhook desencadena una compilación cuando la referencia del encabezado coincide con el patrón de la expresión regular (por ejemplo, `refs/heads/branch-name` o `refs/tags/tag-name`). Para un evento, el nombre de referencia se encuentra en la propiedad `ref` de la carga de webhook. Para los eventos de solicitud de extracción, el nombre de la ramificación se encuentra en la propiedad `ref` del objeto `head` de la carga de webhook.

`BASE_REF`

Un evento de webhook desencadena una compilación cuando la referencia de base coincide con el patrón de la expresión regular (por ejemplo, `refs/heads/branch-name`). Un filtro `BASE_REF` solo se puede utilizar con eventos de solicitud de extracción. El nombre de la ramificación se encuentra en la propiedad `ref` del objeto `base` de la carga de webhook.

`FILE_PATH`

Un webhook desencadena una compilación cuando la ruta de un archivo modificado coincide con el patrón de la expresión regular. Un filtro `FILE_PATH` se puede usar con eventos de solicitud de inserción y extracción de GitHub y eventos de inserción de GitHub Enterprise Server. No se puede usar con eventos de solicitud de extracción de GitHub Enterprise Server.

`COMMIT_MESSAGE`

Un webhook desencadena una compilación cuando el mensaje de confirmación del encabezado coincide con el patrón de la expresión regular. Un filtro `COMMIT_MESSAGE` se puede usar con eventos de solicitud de inserción y extracción de GitHub y eventos de inserción de GitHub Enterprise Server. No se puede usar con eventos de solicitud de extracción de GitHub Enterprise Server.

Note

Puede encontrar la carga de webhook en la configuración de webhook del repositorio de GitHub.

Temas

- [Filtrar eventos de webhooks en GitHub \(consola\)](#) (p. 247)
- [Filtrar eventos de webhooks en GitHub \(SDK\)](#) (p. 251)
- [Filtrar eventos de webhooks en GitHub \(AWS CloudFormation\)](#) (p. 253)

Filtrar eventos de webhooks en GitHub (consola)

En **Eventos de webhooks de origen principal**, seleccione lo siguiente. Esta sección solo está disponible cuando eliges **Repositorio** en mi cuenta de GitHub para el repositorio de código fuente.

1. Cuando cree el proyecto, seleccione Rebuild every time a code change is pushed to this repository (Volver a compilar cada vez que se inserte un cambio de código en este repositorio).
2. En Event type (Tipo de evento), seleccione uno o varios eventos.
3. Para filtrar en función de cuándo un evento va a desencadenar una compilación, en Start a build under these conditions (Iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
4. Para filtrar en función de cuándo no se va a desencadenar un evento, en Don't start a build under these conditions (No iniciar una compilación en estas condiciones), añada uno o varios filtros opcionales.
5. Seleccionar Agregar grupo de filtros para añadir otro grupo de filtros, si es necesario.

Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\) \(p. 198\)](#) y [WebhookFilter](#) en la Referencia de la API de AWS CodeBuild.

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación únicamente con solicitudes de extracción:

The screenshot shows the 'Webhook event filter group 1' configuration in the AWS CodeBuild console. It features a dropdown menu for 'Event type' which is currently empty. Below the dropdown, there are four buttons representing different pull request events: 'PULL_REQUEST_CREATED', 'PULL_REQUEST_UPDATED', 'PULL_REQUEST_REOPENED', and 'PULL_REQUEST_MERGED'. Each button has a small 'X' icon in the top right corner, indicating it can be removed. At the bottom of the configuration area, there are two expandable sections: 'Start a build under these conditions' and 'Don't start a build under these conditions', both with a right-pointing triangle icon.

Si utilizamos un ejemplo con dos grupos de filtros de webhooks, la compilación se desencadenará cuando uno de los grupos o los dos se evalúen como true:

- El primer grupo de filtros especifica las solicitudes de extracción que se crean, actualizan o vuelven a abrirse en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/main$` y las referencias de encabezado que coinciden con `^refs/heads/branch1$`.
- El segundo grupo de filtros especifica solicitudes de inserción en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/branch1$`.

Webhook event filter group 1

Event type
Add one or more a webhook event filter groups to specify which events trigger a new build. If you do not add a webhook event filter group, then a new build is triggered every time a code change is pushed to your repository.

PULL_REQUEST_CREATED X

PULL_REQUEST_UPDATED X

PULL_REQUEST_REOPENED X

▼ Start a build under these conditions

ACTOR_ID - optional

HEAD_REF - optional

BASE_REF - optional

FILE_PATH - optional

^refs/heads/branch1\$

^refs/heads/main\$

COMMIT_MESSAGE - optional

► Don't start a build under these conditions

Webhook event filter group 2

Remove filter group

Event type
Add one or more a webhook event filter groups to specify which events trigger a new build. If you do not add a webhook event filter group, then a new build is triggered every time a code change is pushed to your repository.

PUSH X

▼ Start a build under these conditions

ACTOR_ID - optional

HEAD_REF - optional

BASE_REF - optional

FILE_PATH - optional

^refs/heads/branch1\$

COMMIT_MESSAGE - optional

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación de todas las solicitudes excepto los eventos de etiquetas.

Webhook event filter group 1

Event type

PUSH X

PULL_REQUEST_CREATED X

PULL_REQUEST_UPDATED X

PULL_REQUEST_REOPENED X

PULL_REQUEST_MERGED X

► Start a build under these conditions

▼ Don't start a build under these conditions

ACTOR_ID - optional

HEAD_REF - optional

BASE_REF - optional

FILE_PATH - optional

^refs/tags/.*

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación solo cuando se modifican los archivos cuyos nombres coinciden con la expresión regular `^buildspec.*`.

Webhook event filter group 1

Event type

PUSH X

▼ Start a build under these conditions

ACTOR_ID - *optional*

HEAD_REF - *optional*

BASE_REF - *optional*

FILE_PATH - *optional*

`^buildspec.*`

COMMIT_MESSAGE -
optional

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhooks desencadena una compilación solamente cuando hay un cambio realizado por un usuario de GitHub o GitHub Enterprise Server especificado que tiene un ID de cuenta que coincide con la expresión regular `actor-account-id`.

Note

Para obtener información acerca de cómo encontrar el ID de cuenta de GitHub, consulte <https://api.github.com/users/nombre-usuario>, donde *nombre-usuario* es el nombre de usuario de GitHub.

Webhook event filter group 1

Event type

PUSH X PULL_REQUEST_CREATED X PULL_REQUEST_UPDATED X

PULL_REQUEST_REOPENED X PULL_REQUEST_MERGED X

▼ Start a build under these conditions

ACTOR_ID - *optional* HEAD_REF - *optional* BASE_REF - *optional* FILE_PATH - *optional*

actor-account-id

► Don't start a build under these conditions

En este ejemplo, un grupo de filtros de webhook desencadena una compilación para un evento de inserción cuando el mensaje de confirmación de la cabeza coincide con la expresión regular `\[CodeBuild\]`.

Webhook event filter group 1

Event type

PUSH X

▼ Start a build under these conditions

ACTOR_ID - *optional*

HEAD_REF - *optional*

BASE_REF - *optional*

FILE_PATH - *optional*

COMMIT_MESSAGE -
optional

\[CodeBuild\]

► Don't start a build under these conditions

Filtrar eventos de webhooks en GitHub (SDK)

Si desea utilizar el SDK de AWS CodeBuild para filtrar eventos de webhooks, utilice el campo `filterGroups` de la sintaxis de la solicitud de los métodos de API `CreateWebhook` o `UpdateWebhook`. Para obtener más información, consulte [WebhookFilter](#) en la Referencia de la API de código de.

Si desea crear un filtro de webhook que desencadene una compilación únicamente con las solicitudes de extracción, inserte lo siguiente en la sintaxis de la solicitud:

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",  
      "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED,  
PULL_REQUEST_MERGED"  
    }  
  ]  
]
```

Si desea crear un filtro de webhooks que desencadene una compilación únicamente con las ramificaciones especificadas, utilice el parámetro `pattern` para especificar una expresión regular que filtre los nombres de las ramificaciones. Si utilizamos un ejemplo con dos grupos de filtros, la compilación se desencadenaría cuando uno de los grupos o los dos se evalúen como `true`:

- El primer grupo de filtros especifica las solicitudes de extracción que se crean, actualizan o vuelven a abrirse en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/main$` y las referencias de encabezado que coinciden con `^refs/heads/myBranch$`.
- El segundo grupo de filtros especifica solicitudes de inserción en ramificaciones con nombres de referencia de Git que coinciden con la expresión regular `^refs/heads/myBranch$`.

```
"filterGroups": [  
  [  
    {  
      "type": "EVENT",
```

```

        "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED"
    },
    {
        "type": "HEAD_REF",
        "pattern": "^refs/heads/myBranch$"
    },
    {
        "type": "BASE_REF",
        "pattern": "^refs/heads/main$"
    }
],
[
    {
        "type": "EVENT",
        "pattern": "PUSH"
    },
    {
        "type": "HEAD_REF",
        "pattern": "^refs/heads/myBranch$"
    }
]
]

```

Puede utilizar el parámetro `excludeMatchedPattern` para especificar qué eventos no desencadenan una compilación. Por ejemplo, en este caso, se desencadena una compilación con todas las solicitudes, excepto los eventos de etiquetas.

```

"filterGroups": [
    [
        {
            "type": "EVENT",
            "pattern": "PUSH, PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
PULL_REQUEST_REOPENED, PULL_REQUEST_MERGED"
        },
        {
            "type": "HEAD_REF",
            "pattern": "^refs/tags/.+",
            "excludeMatchedPattern": true
        }
    ]
]

```

Puede crear un filtro que desencadene una compilación solo cuando se modifiquen los archivos cuyos nombres coincidan con la expresión regular del argumento `pattern`. En este ejemplo, el grupo de filtros especifica que la compilación solo debe desencadenarse cuando se modifiquen los archivos cuyos nombres coincidan con la expresión regular `^buildspec.*`.

```

"filterGroups": [
    [
        {
            "type": "EVENT",
            "pattern": "PUSH"
        },
        {
            "type": "FILE_PATH",
            "pattern": "^buildspec.*"
        }
    ]
]

```

Puede crear un filtro que desencadene una compilación solo cuando un usuario de GitHub o GitHub Enterprise Server especificado con el ID de cuenta `actor-account-id` realice algún cambio.

Note

Para obtener información acerca de cómo encontrar el ID de cuenta de GitHub, consulte <https://api.github.com/users/nombre-usuario>, donde *nombre-usuario* es el nombre de usuario de GitHub.

```
"filterGroups": [
  [
    {
      "type": "EVENT",
      "pattern": "PUSH, PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
PULL_REQUEST_REOPENED, PULL_REQUEST_MERGED"
    },
    {
      "type": "ACTOR_ACCOUNT_ID",
      "pattern": "actor-account-id"
    }
  ]
]
```

Puede crear un filtro que desencadene una compilación solo cuando el mensaje de confirmación del encabezado coincida con la expresión regular en el argumento pattern. En este ejemplo, el grupo de filtros especifica que una compilación se desencadena solo cuando el mensaje de confirmación del encabezado del evento de inserción coincide con la expresión regular `\[CodeBuild\]`.

```
"filterGroups": [
  [
    {
      "type": "EVENT",
      "pattern": "PUSH"
    },
    {
      "type": "COMMIT_MESSAGE",
      "pattern": "\[CodeBuild\]"
    }
  ]
]
```

Filtrar eventos de webhooks en GitHub (AWS CloudFormation)

Si desea usar una plantilla de AWS CloudFormation para filtrar eventos de webhooks, utilice la propiedad `FilterGroups` del proyecto de AWS CodeBuild. El siguiente fragmento con formato YAML de una plantilla AWS CloudFormation crea dos grupos de filtros. Juntos, desencadenan una compilación cuando uno de los grupos o los dos se evalúan como true:

- El primer grupo de filtros especifica las solicitudes de extracción creadas o actualizadas en ramificaciones con nombres de referencia que coinciden con la expresión regular `^refs/heads/main$` por un usuario de Git que no tiene el ID de cuenta 12345.
- El segundo grupo de filtros especifica las solicitudes de inserción que se crean en archivos cuyos nombres coinciden con la expresión regular `READ_ME` en las ramificaciones que tienen un nombre que coincide con la expresión regular `^refs/heads/.*`.
- El tercer grupo de filtros especifica una solicitud de inserción con un mensaje de confirmación del encabezado que coincida con la expresión regular `\[CodeBuild\]`.

```
CodeBuildProject:
  Type: AWS::CodeBuild::Project
  Properties:
```

```
Name: MyProject
ServiceRole: service-role
Artifacts:
  Type: NO_ARTIFACTS
Environment:
  Type: LINUX_CONTAINER
  ComputeType: BUILD_GENERAL1_SMALL
  Image: aws/codebuild/standard:4.0
Source:
  Type: GITHUB
  Location: source-location
Triggers:
  Webhook: true
  FilterGroups:
    - Type: EVENT
      Pattern: PULL_REQUEST_CREATED,PULL_REQUEST_UPDATED
    - Type: BASE_REF
      Pattern: ^refs/heads/main$
      ExcludeMatchedPattern: false
    - Type: ACTOR_ACCOUNT_ID
      Pattern: 12345
      ExcludeMatchedPattern: true
    - Type: EVENT
      Pattern: PUSH
    - Type: HEAD_REF
      Pattern: ^refs/heads/.*$
    - Type: FILE_PATH
      Pattern: READ_ME
      ExcludeMatchedPattern: true
    - Type: EVENT
      Pattern: PUSH
    - Type: COMMIT_MESSAGE
      Pattern: \[CodeBuild\]
```

Cambiar la configuración de un proyecto de compilación en AWS CodeBuild

Puede usar la consola de AWS CodeBuild, la AWS CLI o los SDK de AWS para cambiar la configuración de un proyecto de compilación.

Si añade un informe de pruebas a un proyecto de compilación, asegúrese de que su rol de IAM tenga los permisos descritos en [Uso de permisos de informes de pruebas](#) (p. 323).

Temas

- [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254)
- [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267)
- [Cambiar la configuración de un proyecto de compilación \(SDK de AWS\)](#) (p. 268)

Cambiar la configuración de un proyecto de compilación (consola)

Para cambiar la configuración de un proyecto de compilación, siga el procedimiento que se indica a continuación.

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Aplique alguna de las siguientes acciones:

- Elija el enlace del proyecto de compilación que desea cambiar y, a continuación, seleccione Build details (Detalles de compilación).
- Seleccione el botón de opción situado junto al proyecto de compilación que desea cambiar, elija View details (Ver detalles) y, a continuación, elija Build details (Detalles de compilación).

Puede modificar las siguientes secciones:

Secciones

- [Configuración de proyectos \(p. 255\)](#)
- [Source \(p. 256\)](#)
- [Environment \(p. 260\)](#)
- [Buildspec \(p. 263\)](#)
- [Configuración Batch \(p. 264\)](#)
- [Artifacts \(p. 265\)](#)
- [Logs \(p. 267\)](#)

Configuración de proyectos

En el navegador Configuración de proyectos sección, elija Editar. Cuando se hayan completado los cambios, elija Actualización de configuración de para guardar la nueva configuración.

Puede modificar las siguientes propiedades:

Descripción

Introduzca una descripción opcional del proyecto de compilación para ayudar a otros usuarios a entender para qué se utiliza este proyecto.

Crear insignias de

Seleccione Enable build badge (Habilitar insignia de compilación) para convertir el estado de compilación del proyecto en visible e incrustable. Para obtener más información, consulte [Ejemplo de insignias de compilación \(p. 78\)](#).

Note

Crear insignia no se aplica si el proveedor de código fuente es Amazon S3.

Habilitar límite de compilación simultánea

Si desea limitar el número de compilaciones simultáneas de este proyecto, siga estos pasos:

1. **Select** Restringir el número de compilaciones simultáneas que este proyecto puede iniciar.
2. En **Límite de compilación simultánea**, introduzca el número máximo de versiones simultáneas que están permitidas para este proyecto. Este límite no puede ser superior al límite de compilación simultáneo establecido para la cuenta. Si intenta introducir un número mayor que el límite de la cuenta, se muestra un mensaje de error.

Las nuevas versiones solo se inician si el número actual de versiones es menor o igual a este límite. Si el recuento de versiones actual cumple con este límite, las nuevas versiones se limitan y no se ejecutan.

Activar el acceso a compilación

Para que los resultados de compilación de su proyecto estén disponibles para el público, incluidos los usuarios sin acceso a una AWS cuenta, seleccione Activar el acceso a compilación y confirme que desea

que los resultados de la compilación se hagan públicos. Las siguientes propiedades se utilizan para proyectos de compilación pública:

Función de servicio de compilación pública

SelectNuevo rol de serviciosi desea que CodeBuild cree un nuevo rol de servicio para usted, oRol de servicio existentesi desea utilizar un rol de servicio existente.

El rol de servicio de compilación pública permite a CodeBuild leer los CloudWatch Logs y descargar los artefactos de Amazon S3 para las compilaciones del proyecto. Esto es necesario para que los registros de compilación y artefactos del proyecto estén disponibles para el público.

Rol de servicio

Introduzca el nombre del nuevo rol de servicio o un rol de servicio existente.

Para que los resultados de compilación de su proyecto sean privados, clarosActivar el acceso a compilación.

Para obtener más información, consulte [Proyectos de compilación pública enAWS CodeBuild \(p. 279\)](#).

Warning

Tenga en cuenta lo siguiente a la hora de hacer públicos los resultados de compilación de su proyecto:

- Todos los resultados de compilación, registros y artefactos de un proyecto, incluidas las compilaciones que se ejecutaron cuando el proyecto era privado, están disponibles para el público.
- Todos los registros y artefactos de compilación están disponibles para el público. Es posible que se hayan emitido variables de entorno, código fuente y otra información confidencial en los registros y artefactos de compilación. Debe tener cuidado con la información que se genera en los registros de compilación. Algunas prácticas recomendadas son:
 - No almacene valores confidenciales, especialmenteAWSlos identificadores de clave de acceso y las claves de acceso secretas de variables de entorno. Le recomendamos que utilice un almacén de parámetros Amazon EC2 Systems Manager oAWS Secrets Managerpara almacenar valores sensibles.
 - Seguimiento[Prácticas recomendadas para el uso de webhooks \(p. 237\)](#)para limitar qué entidades pueden desencadenar una compilación y no almacenar la especificación de compilación en el propio proyecto, para garantizar que sus webhooks sean lo más seguros posible.
- Un usuario malintencionado puede utilizar compilaciones públicas para distribuir artefactos maliciosos. Recomendamos que los administradores del proyecto revisen todas las solicitudes de extracción para verificar que la solicitud de extracción sea un cambio legítimo. También te recomendamos que valides cualquier artefacto con sus sumas de comprobación para asegurarte de que se están descargando los artefactos correctos.

Información adicional

ParaEtiquetas, introduzca el nombre y el valor de las etiquetas que desea admitirAWSservicios para usar. Use Add row para añadir una etiqueta. Puede añadir hasta 50 etiquetas.

Source

En el navegadorFuenteseccción, elija.Editar. Cuando se hayan completado los cambios, elijaActualización de configuración depara guardar la nueva configuración.

Puede modificar las siguientes propiedades:

Proveedor de origen

Elegir el tipo de proveedor de código fuente. Utilice las siguientes listas para realizar las selecciones apropiadas para el proveedor de código fuente:

Note

CodeBuild no admite Bitbucket Server.

Amazon S3

Bucket

Elija el nombre del bucket de entrada que contiene el código fuente.

Clave de objeto de S3 o carpeta de S3

Introduzca el nombre del archivo ZIP o la ruta a la carpeta que contiene el código fuente. Incluya una barra diagonal (/) para descargarlo todo en el bucket de S3.

Versión de origen

Introduzca el ID de versión del objeto que representa la compilación del archivo de entrada. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

CodeCommit

Repositorio

Elegir el repositorio que desea utilizar.

Tipo de referencia

SeleccionarCrear ramificaciones,Etiqueta Git, o bienID de confirmaciónpara especificar la versión del código fuente. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Elija crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione Use Git submodules (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Bitbucket

Repositorio

SeleccionarConnect mediante OAuthoConnect con una contraseña de aplicación de Bitbuckety siga las instrucciones para conectarse (o volver a conectarse) a Bitbucket.

Elija un repositorio público o un repositorio de su cuenta.

Versión de origen

Introduzca una ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte[Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#)

Profundidad del clon de Git

Seleccione Git clone depth (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione Use Git submodules (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

SelectInformar de los estados de compilación al proveedor de origen cuando se inician y finalizan las compilaciones si desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

Para Contexto de estado, escriba el valor que se va a utilizar para el name en el estado de confirmación de Bitbucket. Para obtener más información, consulte la [compilación](#) en la documentación de la API de Bitbucket.

Para URL de destino, escriba el valor que se va a utilizar para el url en el estado de confirmación de Bitbucket. Para obtener más información, consulte la [compilación](#) en la documentación de la API de Bitbucket.

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

En Eventos de webhooks de origen principal, seleccione Volver a compilar cada vez que se envíe un cambio de código a este repositorio si desea que CodeBuild compile el código fuente cada vez que se envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de Bitbucket \(p. 238\)](#).

GitHub

Repositorio

Seleccionar Connect mediante OAuth con un token de acceso personal de GitHub y siga las instrucciones para conectarse (o volver a conectarse) a GitHub y autorizar el acceso a AWS CodeBuild.

Elija un repositorio público o un repositorio de su cuenta.

Versión de origen

Introduzca una ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Seleccione Git clone depth (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija Full (Completo).

Submódulos de Git

Seleccione **Use Git submodules** (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

Seleccione **Informar de los estados de compilación al proveedor de origen** cuando se inician y finalizan las compilaciones si desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

Para **Contexto de estado**, escriba el valor que se va a utilizar para el **context** en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

Para **URL de destino**, escriba el valor que se va a utilizar para el **target_url** en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

En **Eventos de webhooks de origen principal**, seleccione **Volver a compilar cada vez que se envíe un cambio de código a este repositorio** si desea que CodeBuild compile el código fuente cada vez que se envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de GitHub \(p. 246\)](#).

GitHub Enterprise Server

Token de acceso personal de GitHub Enterprise

Consulte [Ejemplo de GitHub Enterprise Server \(p. 107\)](#) para obtener información sobre cómo copiar un token de acceso personal en el portapapeles. Pegue el token en el campo de texto y seleccione **Save Token** (Guardar token).

Note

Solo tiene que introducir y guardar el token de acceso personal una vez. CodeBuild utiliza este token en todos los proyectos futuros.

Versión de origen

Introduzca una solicitud de extracción, ramificación, ID de confirmación, etiqueta o referencia y un ID de confirmación. Para obtener más información, consulte [Ejemplo de versión de origen con AWS CodeBuild \(p. 125\)](#).

Profundidad del clon de Git

Seleccione **Git clone depth** (Profundidad del clon de Git) para crear un clon superficial con un historial truncado en el número de confirmaciones especificado. Si desea un clon completo, elija **Full** (Completo).

Submódulos de Git

Seleccione **Use Git submodules** (Usar submódulos de Git) si desea incluir submódulos de Git en el repositorio.

Estado de la compilación

Seleccione `Informar de los estados de compilación al proveedor de origen` cuando se inician y finalizan las compilaciones si desea que el estado del inicio y finalización de la compilación se registre en el proveedor de origen.

Para poder informar del estado de compilación al proveedor de origen, el usuario asociado al proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen](#) (p. 385).

Para `Contexto de estado`, escriba el valor que se va a utilizar para el contexto en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

Para `URL de destino`, escriba el valor que se va a utilizar para el `target_url` en el estado de confirmación de GitHub. Para obtener más información, consulte [Crear un estado de confirmación](#) en la guía para desarrolladores de GitHub.

El estado de una compilación activada por un webhook se registra siempre en el proveedor de origen. Para que el estado de una compilación iniciada desde la consola o una llamada a la API se informe al proveedor de origen, debe seleccionar esta configuración.

Si las compilaciones de tu proyecto se activan mediante un webhook, debes enviar una nueva confirmación al repositorio para que surta efecto un cambio en esta configuración.

SSL inseguro

Seleccione `Enable insecure SSL` (Habilitar SSL no seguro) para omitir las advertencias de SSL al conectarse al repositorio del proyecto de GitHub Enterprise.

En `Eventos de webhooks de origen principal`, seleccione `Volver a compilar cada vez que se envíe un cambio de código a este repositorio` si desea que CodeBuild compile el código fuente cada vez que se envíe un cambio de código al repositorio. Para obtener más información acerca de los webhooks y los grupos de filtros de, consulte [Eventos de webhooks de GitHub](#) (p. 246).

Environment

En el navegador `Environment` (Entorno):sección, elija `Editar`. Cuando se hayan completado los cambios, elija `Configuración de actualización` para guardar la nueva configuración.

Puede modificar las siguientes propiedades:

Imagen de entorno

Para cambiar la imagen de compilación, elija `Imagen de anulación` y realice una de las siguientes operaciones:

- Para usar una imagen de Docker administrada por AWS CodeBuild, elija `Managed image` (Imagen administrada) y después elija una opción en `Operating system` (Sistema operativo), `Runtime` (Tiempo de ejecución), `Image` (Imagen) e `Image version` (Versión de imagen). Realice una selección en `Environment type` (Tipo de entorno) si está disponible.
- Para usar otra imagen de Docker, elija `Custom image` (Imagen personalizada). Para `Tipo de entorno`, elija `ARM`, `Linux`, `GPU de Linux`, o bien `Windows`. Si elige `Other registry` (Otro registro), para `External registry URL` (URL de registro externo), introduzca el nombre y etiqueta de la imagen de Docker en Docker Hub, usando el formato `docker repository/docker image name`. Si elige `Amazon ECR`, use `Repositorio de Amazon ECR` para elegir la imagen de Docker en su `AWS account`.

- Para utilizar una imagen privada de Docker, elija imagen personalizada. Para Tipo de entorno, elija ARM, Linux, GPU de Linux, o bien Windows. En Image registry (Registro de imagen), elija Other registry (Otro registro) y, a continuación, introduzca el ARN de las credenciales para su imagen de Docker privada. Secrets Manager debe crear las credenciales. Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Note

CodeBuild anula el `ENTRYPOINT` para imágenes de Docker personalizadas.

Privilegiado

SelectPrivilegiado únicamente si tiene previsto usar este proyecto de compilación para compilar imágenes de Docker y la imagen del entorno de compilación que eligió no es una de las proporcionadas por CodeBuild con compatibilidad con Docker. De lo contrario, todas las compilaciones asociadas que intenten interactuar con el daemon de Docker producirán un error. También debe iniciar el daemon de Docker para que las compilaciones puedan interactuar con él. Una forma de hacer esto es inicializar el daemon de Docker en la fase `install` de la especificación de compilación ejecutando los siguientes comandos de compilación. No ejecute estos comandos si eligió una imagen del entorno de compilación proporcionada por CodeBuild compatible con Docker.

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

```
- nohup /usr/local/bin/dockerd --host=unix:///var/run/docker.sock --  
host=tcp://127.0.0.1:2375 --storage-driver=overlay2 &  
- timeout 15 sh -c "until docker info; do echo .; sleep 1; done"
```

Rol de servicio

Aplique alguna de las siguientes acciones:

- Si no tiene un rol de servicio CodeBuild, elija Nuevo rol de servicio. Escriba un nombre para el nuevo rol en el campo Role Name.
- Si dispone de un rol de servicio CodeBuild, elija Rol de servicio existente. En ARN de rol, seleccione el rol de servicio.

Note

Cuando utiliza la consola de para crear un proyecto de compilación, puede crear un rol de servicio de CodeBuild al mismo tiempo. De forma predeterminada, el rol funciona solamente con ese proyecto de compilación. Si utiliza la consola para asociar este rol de servicio a otro proyecto de compilación, el rol se actualizará para que funcione con el otro proyecto de compilación. Un rol de servicio puede funcionar con hasta diez proyectos de compilación.

Configuración adicional

Timeout (Tiempo de espera)

Especifique un valor entre 5 minutos y 8 horas después de los cuales CodeBuild detendrá la compilación si no se ha completado. Si hours y minutes se dejan en blanco, se usará el valor predeterminado de 60 minutos.

VPC

Si desea que CodeBuild funcione con su VPC:

- Para VPC:, elija el ID de VPC que utiliza CodeBuild.
- Para Subredes de la VPC, elija las subredes que incluyen recursos usados por CodeBuild.
- Para Grupos de seguridad de la VPC, elija los grupos de seguridad que usa CodeBuild para permitir el acceso a los recursos de las VPC.

Para obtener más información, consulte [UsarAWS CodeBuildAmazon Virtual Private Cloud \(p. 180\)](#).

Computación

Elegir una de las opciones disponibles.

Variables de entorno

Escriba el nombre y el valor de las variables de entorno que se van a usar y seleccione su tipo.

Note

CodeBuild establece la variable de entorno para suAWSRegión automáticamente. Debe definir las siguientes variables de entorno, si no las ha agregado a su archivo buildspec.yml:

- AWS_ACCOUNT_ID
- IMAGE_REPO_NAME
- IMAGE_TAG

Los usuarios de la consola y de AWS CLI pueden ver las variables de entorno. Si no le preocupa que la variable de entorno esté visible, establezca los campos Name y Value, y después establezca Type en Plaintext.

Le recomendamos que almacene una variable de entorno con un valor confidencial, como unAWSID de clave de acceso, unAWSclave de acceso secreta o contraseña como parámetro en el almacén de parámetros Amazon EC2 Systems Manager oAWS Secrets Manager.

Si utiliza Amazon EC2 Systems Manager, enTipo, eligeParámetro. ParaNombre, escriba un identificador para CodeBuild al que haga referencia. ParaValor, escriba el nombre del parámetro tal como está almacenado en el almacén de parámetros Amazon EC2 Systems Manager. Utilizando el parámetro denominado /CodeBuild/dockerLoginPassword como ejemplo, en Type (Tipo), seleccione Parameter (Parámetro). En Name (Nombre) escriba LOGIN_PASSWORD. En Value (Valor), escriba /CodeBuild/dockerLoginPassword.

Important

Si utiliza el almacén de parámetros de Amazon EC2 Systems Manager, le recomendamos que almacene los parámetros con nombres de parámetros que comiencen por/CodeBuild/(por ejemplo,/CodeBuild/dockerLoginPassword). Puede utilizar la consola de CodeBuild para crear un parámetro en Amazon EC2 Systems Manager. Seleccione Create a parameter (Crear parámetro) y siga las instrucciones del cuadro de diálogo. (En ese cuadro de diálogo, paraClave de KMS, puede especificar el ARN de unAWS KMSclave en tu cuenta. Amazon EC2 Systems Manager utiliza esta clave para cifrar el valor del parámetro durante el almacenamiento y descifrarlo durante la recuperación). Si usa la consola de CodeBuild para crear un parámetro, la consola asigna al principio del nombre del parámetro con/CodeBuild/ya que se está almacenando. Para obtener más información, consulte[Almacén de parámetros de Administrador de sistemas](#)y[Tutorial de la consola del almacén de parámetros de Systems Manager](#)en laGuía del usuario de Amazon EC2 Systems Manager.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager, el rol de servicio del proyecto de compilación debe permitir `ssm:GetParameters`. Si eligióNuevo rol de servicioanteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó Existing service role (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado. Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager con nombres de parámetros que

no empiezan por `/CodeBuild/` y eligió Nuevo rol de servicio, debe actualizar ese rol de servicio para permitir el acceso a nombres de parámetros que no empiezan por `/CodeBuild/`. Esto es así porque el rol de servicio permite el acceso únicamente a los nombres de parámetro que empiezan por `/CodeBuild/`.

Si elige Nuevo rol de servicio, el rol de servicio incluye permisos para descifrar todos los parámetros en el `/CodeBuild/namespace` en el almacén de parámetros Amazon EC2 Systems Manager.

Las variables de entorno que defina reemplazan las variables de entorno existentes. Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y establece una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`. Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y establece una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No establezca variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad.
- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad.
- El valor en la declaración de especificación de compilación es el que menos prioridad tiene.

Si utilizas Secrets Manager, paraTipo, elige Secrets Manager. ParaNombre, escriba un identificador para CodeBuild al que haga referencia. En Value (Valor), introduzca un `reference-key` utilizando el patrón `secret-id:json-key:version-stage:version-id`. Para obtener información, consulte [Secrets Manager reference-key in the buildspec file](#).

Important

Si utiliza Secrets Manager, le recomendamos que almacene secretos con nombres que comiencen por `/CodeBuild/` (por ejemplo, `/CodeBuild/dockerLoginPassword`). Para obtener más información, consulte [¿Qué es AWS Secrets Manager?](#) en la Guía del usuario de AWS Secrets Manager.

Si el proyecto de compilación hace referencia a secretos almacenados en Secrets Manager, el rol de servicio del proyecto de compilación debe permitir `lasecretsmanager:GetSecretValue` acción. Si eligió Nuevo rol de servicio anteriormente, CodeBuild incluye esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si seleccionó Existing service role (Rol de servicio existente), deberá incluir esta acción en el rol de servicio por separado.

Si tu proyecto de compilación hace referencia a secretos almacenados en Secrets Manager con nombres secretos que no empiezan por `/CodeBuild/` y eligió Nuevo rol de servicio, debe actualizar el rol de servicio para permitir el acceso a nombres secretos que no empiezan por `/CodeBuild/`. Esto es así porque el rol de servicio permite el acceso únicamente a nombres secretos que empiezan por `/CodeBuild/`.

Si elige Nuevo rol de servicio, el rol de servicio incluye permisos para descifrar todos los secretos en el `/CodeBuild/espacio de nombres` en Secrets Manager.

Buildspec

En el navegador BuildSpec sección, elija Editar. Cuando se hayan completado los cambios, elija Configuración de actualización para guardar la nueva configuración.

Puede modificar las propiedades siguientes:

Especificaciones de compilación

Aplique alguna de las siguientes acciones:

- Si el código fuente incluye un archivo de especificación de compilación, elija Use a buildspec file (Usar un archivo buildspec). De forma predeterminada, CodeBuild busca un archivo llamado `buildspec.yml` en el directorio raíz del código fuente. Si su archivo buildspec usa un nombre o ubicación diferente, escriba su ruta desde la raíz de origen en Nombre de la especificación de compilación (por ejemplo, `buildspec-two.yml` o `configuration/buildspec.yml`). Si el archivo buildspec está en un bucket de S3, debe estar en el mismo AWSLa región como proyecto de compilación. Especifique el archivo buildspec utilizando su ARN (por ejemplo, `arn:aws:s3:::my-codebuild-sample2/buildspec.yml`).
- Si el código fuente no incluye un archivo de especificación de compilación o si desea ejecutar comandos de compilación diferentes de los especificados para la fase build en el archivo `buildspec.yml` del directorio raíz de código fuente, elija Insert build commands (Insertar comandos de compilación). En Build commands (Comandos de compilación), escriba los comandos que desea ejecutar en la fase build. En caso de que haya varios comandos, separe cada comando con `&&` (por ejemplo, `mvn test && mvn package`). Para ejecutar comandos en otras fases, o si tiene una larga lista de comandos para el build fase, añadir un `buildspec.yml` al directorio raíz del código fuente, agregue los comandos al archivo y, a continuación, elija Use el buildspec.yml en el directorio raíz del código fuente.

Para obtener más información, consulte la [Referencia de la especificación de compilación \(p. 136\)](#).

Configuración Batch

En el navegador Configuración Batch sección, elija Editar. Cuando se hayan completado los cambios, elija Configuración de actualización para guardar la nueva configuración. Para obtener más información, consulte [Compilaciones Batch en AWS CodeBuild \(p. 276\)](#).

Puede modificar las propiedades siguientes:

Rol de servicio de Batch

Proporciona el rol de servicio para las compilaciones por lotes.

Elija una de las siguientes opciones:

- Si no tiene un rol de servicio por lotes, elija Nuevo rol de servicio. En Rol de servicio, especifique un nombre para el nuevo rol.
- Si dispone de un rol de servicio por lotes, elija Rol de servicio existente. En Rol de servicio, seleccione el rol de servicio.

Las compilaciones Batch introducen un nuevo rol de seguridad en la configuración de lotes.

Se requiere este nuevo rol, ya que CodeBuild debe poder llamar al `startBuild`, `stopBuild`, y `retryBuild` acciones en su nombre para ejecutar compilaciones como parte de un lote. Los clientes deben utilizar un nuevo rol, y no el mismo rol que usan en su compilación, por dos razones:

- Dar el rol de compilación `startBuild`, `stopBuild`, y `retryBuild` los permisos permitirían que una sola compilación inicie más compilaciones a través de la especificación de compilación.
- Las compilaciones por lotes de CodeBuild proporcionan restricciones que restringen el número de compilaciones y tipos de cómputo que se pueden utilizar para las compilaciones del lote. Si el rol de compilación tiene estos permisos, es posible que las compilaciones mismas puedan eludir estas restricciones.

Tipos de cómputo permitidos para lotes

Seleccione los tipos de cómputo permitidos para el lote. Selecciona todo lo que corresponda.

Número máximo de compilaciones permitidas por lotes

Escriba el número máximo de compilaciones permitidas en el lote. Si un lote supera este límite, el lote fallará.

Tiempo de espera del lote

Introduzca la cantidad máxima de tiempo que dura la compilación por lotes.

CombineArtifacts

SelectCombina todos los artefactos del lote en una única ubicación para que todos los artefactos del lote se combinen en una única ubicación.

Modo de informe por lotes

Seleccione el modo de informe de estado de compilación deseado para las compilaciones por lotes.

Note

Este campo solo está disponible cuando el origen del proyecto es Bitbucket, GitHub o GitHub Enterprise, y Informar de los estados de compilación al proveedor de origen cuando se inician y finalizan las compilaciones se selecciona en Fuente.

Builds agregadas

Seleccione esta opción para combinar los estados de todas las compilaciones del lote en un único informe de estado.

Builds individuales

Seleccione esta opción para que los estados de compilación de todas las compilaciones del lote se informen por separado.

Artifacts

En el navegador Artifacts sección, elija Editar. Cuando se hayan completado los cambios, elija Configuración de actualización para guardar la nueva configuración.

Puede modificar las propiedades siguientes:

Type

Aplique alguna de las siguientes acciones:

- Si no desea crear ningún artefacto de salida de la compilación, elija No artifacts. por ejemplo, si solo va a ejecutar pruebas de compilación o si desea insertar una imagen de Docker en un repositorio de Amazon ECR.
- Para almacenar la salida de la compilación en un bucket de S3, elija Amazon S3 y, a continuación, haga lo siguiente:
 - Si desea utilizar su nombre de proyecto para el archivo ZIP o carpeta de salida de la compilación, deje en blanco Name (Nombre). De lo contrario, escriba el nombre. (Si desea producir un archivo ZIP y desea que el archivo ZIP tenga una extensión de archivo, asegúrese de incluirlo detrás del nombre del archivo ZIP).
 - Seleccione Enable semantic versioning (Habilitar control semántico de versiones) si desea que el nombre especificado en el archivo buildspec invalide cualquier nombre especificado en la consola. El nombre en un archivo buildspec se calcula en tiempo de compilación y utiliza el lenguaje de comandos Shell. Por ejemplo, puede adjuntar una fecha y una hora al nombre del artefacto para que siempre sea único. Los nombres de artefactos únicos impiden que los artefactos se sobrescriban. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).
 - En Bucket name (Nombre del bucket), seleccione el nombre del bucket de salida.

- Si eligió Insert build commands (Insertar comandos de compilación) anteriormente en este procedimiento, en Output files (Archivos de salida), escriba las ubicaciones de los archivos de la compilación que desea incluir en el archivo ZIP o carpeta de salida de la compilación. Si hay varias ubicaciones, separe cada una de las ubicaciones con una coma (por ejemplo, `appspec.yml, target/my-app.jar`). Para obtener más información, consulte la descripción de files en [Sintaxis de buildspec \(p. 137\)](#).
- Si no desea que los artefactos de compilación se cifren, seleccione Remove artifacts encryption (Quitar cifrado de artefactos),

Para cada conjunto de artefactos secundarios que desee:

1. Para Source identifier (Identificador de artefacto), escriba un valor que tenga como máximo 128 caracteres, únicamente de tipo alfanumérico y guiones bajos.
2. Elija Add artifact (Añadir artefacto).
3. Siga los pasos anteriores para configurar los artefactos secundarios.
4. Elija Save artifact (Guardar artefacto).

Configuración adicional

Clave de cifrado

Aplique alguna de las siguientes acciones:

- Para utilizar el Clave administrada por AWS Amazon S3 en su cuenta para cifrar los artefactos de salida de la compilación, deje Clave de cifrado en blanco. Esta es la opción predeterminada.
- Para utilizar una clave administrada por el cliente para cifrar los artefactos de salida de la compilación, en Clave de cifrado, escriba el ARN de la clave administrada por el cliente. Utilice el formato `arn:aws:kms:region-ID:account-ID:key/key-ID`.

Tipo de caché

En Cache type (Tipo de caché), seleccione una de las opciones siguientes:

- Si no desea usar una memoria caché, elija No cache.
- Si desea utilizar una caché de Amazon S3, elija Amazon S3y, a continuación, haga lo siguiente:
 - En Bucket, elija el nombre del bucket de S3 donde se almacena la caché.
 - (Opcional) Para Prefijo de ruta de caché, especifique un prefijo de ruta de Amazon S3. El valor Cache path prefix (Prefijo de ruta de caché) es similar a un nombre de directorio. Le permite almacenar la caché en el mismo directorio en un bucket.

Important

No añada una barra (/) al final del prefijo de ruta.

- Si desea utilizar una caché local, seleccione Local y elija uno o varios modos de caché local.

Note

El modo Docker layer cache (Caché de capas de Docker) solo está disponible para Linux. Si lo selecciona, el proyecto deberá ejecutarse en modo con privilegios. `LAARM_CONTAINER_Linux_GPU_CONTAINER` tipos de entorno `YBUILD_GENERAL1_2XLARGE` el tipo de cómputo no admite el uso de una caché local.

El uso de memoria caché ahorra mucho tiempo de compilación, ya que algunas partes reutilizables del entorno de compilación se almacenan en ella y se usan en las distintas compilaciones. Para obtener información acerca de cómo especificar una caché en el archivo de especificación de compilación, consulte [Sintaxis de buildspec \(p. 137\)](#). Para obtener más información acerca del almacenamiento en caché, consulte [Almacenamiento en caché de compilaciones de AWS CodeBuild \(p. 229\)](#).

Logs

En el navegadorRegistrossección, elija.Editar. Cuando se hayan completado los cambios, elijaConfiguración de actualizaciónpara guardar la nueva configuración.

Puede modificar las propiedades siguientes:

Elija los registros que desea crear. Puede crear Amazon CloudWatch Logs, registros Amazon S3 o ambos.

CloudWatch

Si desea registros de Amazon CloudWatch Logs:

CloudWatch Logs

SelectCloudWatch Logs.

Group name

Escriba el nombre del grupo de registros de Amazon CloudWatch Logs.

Nombre de transmisión

Introduzca el nombre del flujo de registro de Amazon CloudWatch Logs.

S3

Si quieres registros de Amazon S3:

Registros de S3

Seleccione registros de S3.

Bucket

Seleccione el nombre del bucket de S3 para los registros.

Prefijo de la ruta

Escriba el prefijo de los registros.

Deshabilitar el cifrado de registros de S3

Seleccione si no quiere que los registros de S3 se cifren.

Cambiar la configuración de un proyecto de compilación (AWS CLI)

Para obtener información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos](#) (p. 402).

Para actualizar un proyecto de CodeBuild con elAWS CLI, crea un archivo JSON con las propiedades actualizadas y pasa ese archivo a la`update-project`comando. Las propiedades no incluidas en el archivo de actualización permanecen sin cambios.

En el archivo JSON de actualización, solo el`name`y las propiedades modificadas son obligatorias. Lanameidentifica el proyecto que se va a modificar. Para las estructuras modificadas, también deben incluirse los parámetros requeridos para dichas estructuras. Por ejemplo, para modificar el entorno del proyecto, el`environment/type``environment/computeType`las propiedades son obligatorias. A continuación se muestra un ejemplo que actualiza la imagen del entorno:

```
{
  "name": "<project-name>",
  "environment": {
    "type": "LINUX_CONTAINER",
```

```
"computeType": "BUILD_GENERAL1_SMALL",  
"image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0"  
}  
}
```

Si necesita obtener los valores de propiedad actuales de un proyecto, utilice [batch-get-projects](#) para obtener las propiedades actuales del proyecto que está modificando y escribir la salida en un archivo.

```
aws codebuild batch-get-projects --names "<project-name>" > project-info.json
```

La [project-info.json](#) contiene una matriz de proyectos, por lo que no se puede utilizar directamente para actualizar un proyecto. Sin embargo, puede copiar las propiedades que desea modificar desde el [project-info.json](#) y péguelos en el archivo de actualización como línea base de las propiedades que desea modificar. Para obtener más información, consulte [Ver los detalles de un proyecto de compilación \(AWS CLI\)](#) (p. 227).

Modifique el archivo JSON de actualización tal y como se describe en [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209) y guarde los resultados. Cuando haya terminado de modificar el archivo JSON de actualización, ejecute [update-project](#), pasando el archivo JSON de actualización.

```
aws codebuild update-project --cli-input-json file://<update-project-file>
```

Si el comando se ejecuta correctamente, el JSON del proyecto actualizado aparece en el resultado. Si falta algún parámetro necesario, aparece un mensaje de error en la salida que identifica los parámetros que faltan. Por ejemplo, este es el mensaje de error que se muestra si el `environment/type` falta el parámetro:

```
aws codebuild update-project --cli-input-json file://update-project.json  
  
Parameter validation failed:  
Missing required parameter in environment: "type"
```

Cambiar la configuración de un proyecto de compilación (SDK de AWS)

Para obtener información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Eliminar un proyecto de compilación en AWS CodeBuild

Puede utilizar la consola CodeBuild, AWS CLI, o bien AWS SDK para eliminar un proyecto de compilación en CodeBuild. Si elimina un proyecto, sus compilaciones no se eliminan.

Warning

No se puede eliminar un proyecto que tenga compilaciones y una política de recursos. Para eliminar un proyecto con una política de recursos y compilaciones, primero debe quitar la política de recursos y eliminar sus compilaciones.

Temas

- [Eliminar un proyecto de compilación \(consola\)](#) (p. 269)
- [Eliminar un proyecto de compilación \(AWS CLI\)](#) (p. 269)
- [Eliminar un proyecto de compilación \(SDK de AWS\)](#) (p. 269)

Eliminar un proyecto de compilación (consola)

1. Abra el iconoAWS CodeBuildConsola de:<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Aplique alguna de las siguientes acciones:
 - Elija el botón de opción junto al proyecto de compilación que desea eliminar, elija Delete (Eliminar).
 - Elija el enlace del proyecto de compilación que desea eliminar y, a continuación, elija Delete.

Note

De forma predeterminada, se muestran los 10 últimos proyectos de compilación. Para ver más proyectos de compilación, elija un valor diferente para Projects per page (Proyectos por página) o use las flechas atrás y adelante para ver los proyectos.

Eliminar un proyecto de compilación (AWS CLI)

1. Ejecute el comando `delete-project`:

```
aws codebuild delete-project --name name
```

Reemplace el siguiente marcador de posición:

- ***Nombre de***: Cadena obligatoria. El nombre del proyecto de compilación que desea eliminar. Para obtener una lista de los proyectos de compilación disponibles, ejecute el comando `list-projects`. Para obtener más información, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\) \(p. 225\)](#).
2. Si el comando se ejecuta correctamente, no aparecerán datos ni errores en el resultado.

Para obtener más información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos \(p. 402\)](#).

Eliminar un proyecto de compilación (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS \(p. 403\)](#).

Uso de proyectos compartidos

El uso compartido de proyectos permite a los propietarios compartir susAWS CodeBuildproyectos con otrosAWS cuentas o usuarios. En este modelo, la cuenta propietaria del proyecto (propietario) comparte un proyecto con otras cuentas (consumidores). Un consumidor no puede editar ni ejecutar un proyecto.

Contenido

- [Requisitos previos para compartir proyectos \(p. 270\)](#)
- [Requisitos previos para acceder a proyectos compartidos con usted \(p. 270\)](#)
- [Servicios de relacionados \(p. 270\)](#)
- [Compartir un proyecto \(p. 270\)](#)
- [Dejar de compartir un proyecto compartido \(p. 272\)](#)
- [Identificación de un proyecto compartido \(p. 272\)](#)
- [Permisos de los proyectos compartidos \(p. 273\)](#)

Requisitos previos para compartir proyectos

Para compartir un proyecto, su cuenta de AWS debe ser la propietaria. No puede compartir un proyecto que se ha compartido con usted.

Requisitos previos para acceder a proyectos compartidos con usted

Para tener acceso a un proyecto compartido, el rol de IAM de un consumidor requiere el `BatchGetProjects` permiso. Puede adjuntar la siguiente política a su rol de IAM:

```
{
  "Effect": "Allow",
  "Resource": [
    "*"
  ],
  "Action": [
    "codebuild:BatchGetProjects"
  ]
}
```

Para obtener más información, consulte [Uso de políticas basadas en identidades para AWS CodeBuild](#) (p. 358).

Servicios de relacionados

El uso compartido de proyectos se integra con AWS Resource Access Manager (AWS RAM), un servicio que le permite compartir sus recursos de AWS con cualquier cuenta de AWS o a través de AWS Organizations. Con AWS RAM, comparte los recursos mediante la creación de un recurso compartido que especifica los recursos y los consumidores con los que los comparte. Los consumidores pueden ser cuentas de AWS individuales, unidades organizativas en AWS Organizations o toda una organización de AWS Organizations.

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

Compartir un proyecto

El consumidor puede utilizar tanto el `AWS CLI` como `AWS CodeBuild` para ver el proyecto y las compilaciones que ha compartido. El consumidor no puede editar ni ejecutar el proyecto.

Puede añadir un proyecto a un recurso compartido existente o puede crear uno en la consola de [AWS RAM](#).

Note

No se puede eliminar un proyecto con compilaciones que se haya añadido a un recurso compartido.

Para compartir un proyecto con unidades organizativas o con toda la organización, debe habilitar el uso compartido con AWS Organizations. Para obtener más información, consulte [Habilitar el uso compartido con AWS Organizations](#) en la Guía del usuario de AWS RAM.

Puede usar la consola de AWS CodeBuild, la consola de AWS RAM o AWS CLI para compartir un proyecto de su propiedad.

Para compartir un proyecto que posee (consola de CodeBuild)

1. Abra el icono `AWS CodeBuild` Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.

2. En el panel de navegación, elija Build projects.

Note

De forma predeterminada, solo se muestran los 10 últimos proyectos de compilación. Para ver más proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Projects per page (Proyectos por página) o use las flechas atrás y adelante.

3. Elija el proyecto que desea compartir y, a continuación, elija Share (Compartir). Para obtener más información, consulte [Crear un recurso compartido](#) en la Guía del usuario de AWS RAM.

Para compartir un proyecto que posee (consola de AWS RAM)

Consulte [Creación de un recurso compartido](#) en la AWS RAM Guía del usuario de.

Para compartir un proyecto que posee (comando AWS RAM)

Utilice el comando [create-resource-share](#).

Para compartir un proyecto que posee (comando CodeBuild)

Utilice el comando [put-resource-policy](#):

1. Cree un archivo llamado `policy.json` y copie lo siguiente en él.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": "<consumer-aws-account-id-or-user>"
    },
    "Action": [
      "codebuild:BatchGetProjects",
      "codebuild:BatchGetBuilds",
      "codebuild:ListBuildsForProject"
    ],
    "Resource": "<arn-of-project-to-share>"
  }]
}
```

2. Actualice `policy.json` con el ARN del proyecto y los identificadores para compartirlo. En el siguiente ejemplo se concede acceso de solo lectura al usuario raíz para el AWS cuenta identificada por 123456789012.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": [
        "123456789012"
      ]
    },
    "Action": [
      "codebuild:BatchGetProjects",
      "codebuild:BatchGetBuilds",
      "codebuild:ListBuildsForProject"
    ],
    "Resource": "arn:aws:codebuild:us-west-2:123456789012:project/my-project"
  }]
}
```

3. Ejecute el [put-resource-policy](#) comando.

```
aws codebuild put-resource-policy --resource-arn <project-arn> --policy file://  
policy.json
```

4. Consigue el AWS RAM ARN compartido de recursos.

```
aws ram list-resources --resource-owner SELF --resource-arns <project-arn>
```

Esto devolverá una respuesta similar a esta:

```
{  
  "resources": [  
    {  
      "arn": "<project-arn>",  
      "type": "<type>",  
      "resourceShareArn": "<resource-share-arn>",  
      "creationTime": "<creation-time>",  
      "lastUpdatedTime": "<last-update-time>"  
    }  
  ]  
}
```

En la respuesta, copie el `<resource-share-arn>` valor para utilizar en el paso siguiente.

5. Ejecute la AWS RAM `promote-recursos-compartido-creado-de-política` comando.

```
aws ram promote-resource-share-created-from-policy --resource-share-arn <resource-  
share-arn>
```

Dejar de compartir un proyecto compartido

Solo el propietario puede obtener acceso a un proyecto que se ha dejado de compartir, incluidas sus compilaciones. Si deja de compartir un proyecto, cualquier cuenta o usuario de AWS con el que lo haya compartido anteriormente no podrá obtener acceso al proyecto ni a sus compilaciones.

Para dejar de compartir un proyecto compartido que posee, debe quitarlo del recurso compartido. Puede usar la consola de AWS CodeBuild, la consola de AWS RAM o AWS CLI para realizar esta operación.

Para dejar de compartir un proyecto compartido que posee (consola de AWS RAM)

Consulte [Actualización de un recurso compartido](#) en la Guía del usuario de AWS RAM.

Para dejar de compartir un proyecto compartido que posee (AWS CLI)

Utilice el comando `disassociate-resource-share`.

Para dejar de compartir un proyecto que posee (comando CodeBuild)

Ejecute el comando `delete-resource-policy` y especifique el ARN del proyecto que desea dejar de compartir:

```
aws codebuild delete-resource-policy --resource-arn <project-arn>
```

Identificación de un proyecto compartido

Los propietarios y los consumidores pueden utilizar la AWS CLI para identificar proyectos compartidos.

Para identificar proyectos compartidos con su cuenta o usuario de AWS (AWS CLI)

Utilice el comando [lista de proyectos compartidos](#) para devolver los proyectos que se comparten con usted.

Permisos de los proyectos compartidos

Permisos de los propietarios

El propietario del proyecto puede editar el proyecto y utilizarlo para ejecutar compilaciones.

Permisos de los consumidores

Un consumidor de proyecto puede ver un proyecto y sus compilaciones, pero no puede editarlo ni usarlo para ejecutar compilaciones.

Etiquetado de proyectos en AWS CodeBuild

Una etiqueta es un atributo personalizado que usted o AWS asignan a un recurso de AWS. Cada etiqueta de AWS tiene dos partes:

- Una clave de etiqueta (por ejemplo, `CostCenter`, `Environment`, `Project` o `Secret`). Las claves de etiqueta distinguen entre mayúsculas y minúsculas.
- Un campo opcional que se denomina valor de etiqueta (por ejemplo, `111122223333` o `Production` o el nombre de un equipo). Omitir el valor de etiqueta es lo mismo que usar una cadena vacía. Al igual que las claves de etiqueta, los valores de etiqueta distinguen entre mayúsculas y minúsculas.

En conjunto, se conocen como pares clave-valor. Para obtener información sobre el número de etiquetas que puede tener en un proyecto y las restricciones en las claves y valores de las etiquetas, consulte [Tags](#) (p. 446).

Las etiquetas le ayudan a identificar y organizar los recursos de AWS. Muchos servicios de AWS admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, puede asignar la misma etiqueta a un proyecto de CodeBuild que asigna a un bucket de S3. Para obtener más información sobre el uso de etiquetas, consulte el documento técnico [Prácticas recomendadas de etiquetado](#).

En CodeBuild, los recursos principales son el proyecto y el grupo de informes. Puede utilizar la consola CodeBuild, la AWS CLI, API de CodeBuild o AWS SDK para agregar, administrar y quitar etiquetas de un proyecto. Además de utilizar etiquetas para identificar, organizar y realizar el seguimiento de su proyecto, puede utilizarlas en las políticas de IAM para ayudar a controlar quién puede ver e interactuar con el proyecto. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild](#) (p. 381).

Temas

- [Agregar una etiqueta a un proyecto](#) (p. 273)
- [Consultar etiquetas de un proyecto](#) (p. 274)
- [Editar etiquetas de un proyecto](#) (p. 275)
- [Quitar una etiqueta de un proyecto](#) (p. 276)

Agregar una etiqueta a un proyecto

Agregar etiquetas a un proyecto puede ayudarle a identificar y organizar sus recursos de AWS y a administrar el acceso a ellos. En primer lugar, agregue una o varias etiquetas (pares de clave-valor) a un proyecto. Tenga en cuenta que hay límites en el número de etiquetas que puede tener en un proyecto.

Existen restricciones sobre los caracteres que se pueden utilizar en los campos de clave y valor. Para obtener más información, consulte [Tags \(p. 446\)](#). Cuando tenga las etiquetas, puede crear políticas de IAM para administrar el acceso al proyecto en función de estas etiquetas. Puede utilizar la consola de CodeBuild o la AWS CLI para agregar etiquetas a un proyecto.

Important

Antes de añadir una etiqueta a un proyecto, asegúrese de revisar cualquier política de IAM que es posible que utilice etiquetas para controlar el acceso a recursos como los proyectos de compilación. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild \(p. 381\)](#).

Para obtener más información sobre cómo agregar etiquetas a un proyecto al crearlo, consulte [Agregar una etiqueta a un proyecto \(consola\) \(p. 274\)](#).

Temas

- [Agregar una etiqueta a un proyecto \(consola\) \(p. 274\)](#)
- [Agregar una etiqueta a un proyecto \(AWS CLI\) \(p. 274\)](#)

Agregar una etiqueta a un proyecto (consola)

Puede utilizar la consola de CodeBuild para agregar una o varias etiquetas a un proyecto de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Proyectos de compilación, elija el nombre del proyecto al que desea agregar etiquetas.
3. En el panel de navegación, seleccione Settings. Elija Etiquetas de proyecto de compilación.
4. Si no se ha agregado ninguna etiqueta al proyecto, elija Agregar etiqueta. De lo contrario, elija Edit (Editar) y, a continuación, seleccione Add tag (Añadir etiqueta).
5. En Key (Clave), escriba un nombre para la etiqueta. Puede añadir un valor opcional para la etiqueta en Value (Valor).
6. (Opcional) Para añadir otra etiqueta, vuelva a elegir Add tag (Añadir etiqueta).
7. Cuando haya terminado de añadir etiquetas, seleccione Submit (Enviar).

Agregar una etiqueta a un proyecto (AWS CLI)

Para agregar una etiqueta a un proyecto al crearlo, consulte [Crear un proyecto de compilación \(AWS CLI\) \(p. 209\)](#). En `create-project.json`, agregue sus etiquetas.

En estos pasos, se presupone que ya ha instalado una versión reciente de la AWS CLI o que la ha actualizado a la versión actual. Para obtener más información, consulte [Instalación de la AWS Command Line Interface](#).

Si se ejecuta correctamente, este comando no devuelve nada.

Consultar etiquetas de un proyecto

Las etiquetas pueden ayudarle a identificar y organizar sus recursos de AWS y administrar el acceso a ellos. Para obtener más información sobre el uso de etiquetas, consulte el documento técnico [Prácticas recomendadas de etiquetado](#). Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild \(p. 381\)](#).

Consultar etiquetas de un proyecto (consola)

Puede utilizar la consola de CodeBuild para consultar las etiquetas asociadas con un proyecto de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Proyectos de compilación, elija el nombre del proyecto en el que desea consultar las etiquetas.
3. En el panel de navegación, seleccione Settings. Elija Etiquetas de proyecto de compilación.

Consultar etiquetas de un proyecto (AWS CLI)

Para consultar las etiquetas de un proyecto de compilación, ejecute el siguiente comando. Utilice el nombre del proyecto para el parámetro `--names`.

```
aws codebuild batch-get-projects --names your-project-name
```

Si tiene éxito, este comando devuelve información con formato JSON sobre su proyecto de compilación que incluye algo como lo siguiente:

```
{
  "tags": {
    "Status": "Secret",
    "Team": "JanesProject"
  }
}
```

Si el proyecto no tiene etiquetas, la sección `tags` está vacía:

```
"tags": [ ]
```

Editar etiquetas de un proyecto

Puede cambiar el valor de una etiqueta asociada con un proyecto. También puede cambiar el nombre de la clave, lo que equivale a eliminar la etiqueta actual y añadir otra distinta con el nuevo nombre y el mismo valor que la otra clave. Tenga en cuenta que hay límites en los caracteres que puede utilizar en los campos de clave y valor. Para obtener más información, consulte [Tags \(p. 446\)](#).

Important

La edición de etiquetas de un proyecto puede afectar el acceso a ese proyecto. Antes de editar el nombre (clave) o valor de una etiqueta de un proyecto, asegúrese de revisar cualquier política de IAM que pueda utilizar la clave o el valor de una etiqueta para controlar el acceso a recursos como los proyectos de compilación. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild \(p. 381\)](#).

Editar una etiqueta de un proyecto (consola)

Puede utilizar la consola de CodeBuild para editar las etiquetas asociadas con un proyecto de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Proyectos de compilación, elija el nombre del proyecto en el que desea editar las etiquetas.
3. En el panel de navegación, seleccione Settings. Elija Etiquetas de proyecto de compilación.
4. Elija Edit.
5. Aplique alguna de las siguientes acciones:
 - Para cambiar la etiqueta, escriba un nuevo nombre en Key (Clave). Cambiar el nombre de la etiqueta equivale a quitar una etiqueta y añadir otra nueva con el nuevo nombre de clave.

- Para cambiar el valor de una etiqueta, escriba un nuevo valor. Si desea cambiar el valor por nada, elimine el valor actual y deje el campo en blanco.
6. Cuando haya terminado de editar las etiquetas, seleccione Submit (Enviar).

Editar etiquetas de un proyecto (AWS CLI)

Para agregar, cambiar o eliminar etiquetas de un proyecto de compilación, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267). Actualice la sección `tags` de los datos con formato JSON que utiliza para actualizar el proyecto.

Quitar una etiqueta de un proyecto

Puede quitar una o varias etiquetas asociadas con un proyecto. La eliminación de una etiqueta no la elimina de otros recursos de AWS que están asociados con esa etiqueta.

Important

Quitar las etiquetas de un proyecto puede afectar al acceso a ese proyecto. Antes de quitar una etiqueta de un proyecto, asegúrese de revisar cualquier política de IAM que es posible que utilice la clave o el valor de una etiqueta para controlar el acceso a recursos como los proyectos de compilación. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild](#) (p. 381).

Quitar una etiqueta de un proyecto (consola)

Puede usar la consola de CodeBuild para eliminar la asociación entre una etiqueta y un proyecto de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Proyectos de compilación, elija el nombre del proyecto en el que desea quitar las etiquetas.
3. En el panel de navegación, seleccione Settings. Elija Etiquetas de proyecto de compilación.
4. Elija Edit.
5. Busque la etiqueta que desea eliminar y, a continuación, seleccione Remove tag (Eliminar etiqueta).
6. Cuando haya terminado de eliminar las etiquetas, seleccione Submit (Enviar).

Quitar una etiqueta de un proyecto (AWS CLI)

Para eliminar una o varias etiquetas de un proyecto de compilación, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267). Actualice la sección `tags` de los datos con formato JSON con una lista actualizada de etiquetas que no contenga las que desea eliminar. Si desea eliminar todas las etiquetas, actualice la sección `tags` para:

```
"tags: []"
```

Note

Si elimina un proyecto de compilación de CodeBuild, todas las asociaciones de etiquetas se quitarán del proyecto de compilación eliminado. No es necesario quitar las etiquetas antes de eliminar un proyecto de compilación.

Compilaciones Batch enAWS CodeBuild

Puede usarAWS CodeBuildpara ejecutar compilaciones simultáneas y coordinadas de un proyecto con compilaciones por lotes.

Temas

- [Función de seguridad \(p. 277\)](#)
- [Tipos de creación por lotes \(p. 277\)](#)
- [Modo de informe por lotes \(p. 279\)](#)
- [Más información \(p. 279\)](#)

Función de seguridad

Las compilaciones Batch introducen un nuevo rol de seguridad en la configuración de lotes. Se requiere este nuevo rol, ya que CodeBuild debe poder llamar a `startBuild`, `stopBuild`, y `retryBuild` en su nombre para ejecutar compilaciones como parte de un lote. Los clientes deben utilizar un nuevo rol, y no el mismo rol que usan en su compilación, por dos razones:

- Dar el rol de compilación `startBuild`, `stopBuild`, y `retryBuild` los permisos permitirían que una sola compilación inicie más compilaciones a través de la especificación de compilación.
- Las compilaciones por lotes de CodeBuild proporcionan restricciones que restringen el número de compilaciones y tipos de cómputo que se pueden utilizar para las compilaciones del lote. Si el rol de compilación tiene estos permisos, es posible que las compilaciones mismas puedan eludir estas restricciones.

Tipos de creación por lotes

CodeBuild admite los siguientes tipos de compilación por lotes:

Tipos de creación por lotes

- [Gráfico de compilación \(p. 277\)](#)
- [Lista de compilación \(p. 278\)](#)
- [Compilar matriz \(p. 278\)](#)

Gráfico de compilación

Un gráfico de compilación define un conjunto de tareas que dependen de otras tareas del lote.

En el siguiente ejemplo se define un gráfico de compilación que crea una cadena de dependencias.

```
batch:
  fast-fail: false
  build-graph:
    - identifier: build1
      env:
        variables:
          BUILD_ID: build1
      ignore-failure: false
    - identifier: build2
      buildspec: build2.yml
      env:
        variables:
          BUILD_ID: build2
      depend-on:
        - build1
    - identifier: build3
      env:
        variables:
          BUILD_ID: build3
      depend-on:
```

```
- build2
```

En este ejemplo:

- build1 se ejecuta primero porque no tiene dependencias.
- build2 tiene una dependencia de build1, so build2 corre después build1 completa.
- build3 tiene una dependencia de build2, so build3 corre después build2 completa.

Para obtener más información acerca de la sintaxis de buildspec del gráfico de compilación, consulte [batch/build-graph](#) (p. 155).

Lista de compilación

Una lista de compilación define varias tareas que se ejecutan en paralelo.

En el siguiente ejemplo, se define una lista de compilación. La build1 y build2 Las compilaciones se ejecutarán en paralelo.

```
batch:
  fast-fail: false
  build-list:
    - identifier: build1
      env:
        variables:
          BUILD_ID: build1
      ignore-failure: false
    - identifier: build2
      buildspec: build2.yml
      env:
        variables:
          BUILD_ID: build2
      ignore-failure: true
```

Para obtener más información acerca de la sintaxis de buildspec de la lista de compilación, consulte [batch/build-list](#) (p. 156).

Compilar matriz

Una matriz de compilación define tareas con configuraciones diferentes que se ejecutan en paralelo. CodeBuild crea una compilación independiente para cada combinación de configuración posible.

En el ejemplo siguiente se muestra una matriz de compilación con dos archivos buildspec y tres valores para una variable de entorno.

```
batch:
  build-matrix:
    static:
      ignore-failure: false
      env:
        type: LINUX_CONTAINER
        image: aws/codebuild/amazonlinux2-x86_64-standard:3.0
        privileged-mode: true
    dynamic:
      buildspec:
        - matrix1.yml
        - matrix2.yml
      env:
        variables:
          MY_VAR:
```

```
- VALUE1
- VALUE2
- VALUE3
```

En este ejemplo, CodeBuild crea seis compilaciones:

- `matrix1.yml` por `$MY_VAR=VALUE1`
- `matrix1.yml` por `$MY_VAR=VALUE2`
- `matrix1.yml` por `$MY_VAR=VALUE3`
- `matrix2.yml` por `$MY_VAR=VALUE1`
- `matrix2.yml` por `$MY_VAR=VALUE2`
- `matrix2.yml` por `$MY_VAR=VALUE3`

Cada compilación tendrá la siguiente configuración:

- `ignore-failure` establecido en `false`
- `env/type` establecido en `LINUX_CONTAINER`
- `env/image` establecido en `aws/codebuild/amazonlinux2-x86_64-standard:3.0`
- `env/privileged-mode` establecido en `true`

Estas compilaciones se ejecutan en paralelo.

Para obtener más información acerca de la sintaxis de `buildspec` de la matriz de compilación, consulte [batch/build-matrix](#) (p. 158).

Modo de informe por lotes

Si el proveedor de origen de tu proyecto es Bitbucket, GitHub o GitHub Enterprise y tu proyecto está configurado para informar de los estados de compilación al proveedor de origen, puedes seleccionar cómo quieres que se envíen los estados de compilación por lotes al proveedor de origen. Puede seleccionar que los estados se envíen como un único informe de estado agregado para el lote o que se informe individualmente el estado de cada compilación del lote.

Para obtener más información, consulte los siguientes temas:

- [Configuración Batch \(crear\)](#) (p. 206)
- [Configuración Batch \(actualización\)](#) (p. 264)

Más información

Para obtener más información, consulte los siguientes temas:

- [Referencia de la especificación de compilación por Batch](#) (p. 154)
- [Configuración Batch](#) (p. 206)
- [Ejecutar una compilación por lotes \(AWS CLI\)](#) (p. 287)
- [Detener una compilación por lotes AWS CodeBuild](#) (p. 298)

Proyectos de compilación pública en AWS CodeBuild

AWS CodeBuild permite que los resultados de compilación, los registros y los artefactos de sus proyectos de compilación estén disponibles para el público en general. Esto permite a los colaboradores de los

repositorios de origen ver los resultados y descargar los artefactos de una compilación, sin necesidad de que tengan acceso a un AWS account.

Cuando pone las compilaciones de su proyecto a disposición del público, todos los resultados de compilación, registros y artefactos de un proyecto, incluidas las compilaciones que se ejecutaron cuando el proyecto era privado, se ponen a disposición del público. Del mismo modo, cuando hace que un proyecto de compilación público sea privado, los resultados de compilación de ese proyecto ya no están disponibles para el público.

Para obtener información sobre cómo cambiar la visibilidad pública de los resultados de compilación del proyecto de, consulte [Activar el acceso a compilación \(p. 255\)](#).

CodeBuild proporciona una URL para las compilaciones públicas del proyecto exclusivo de su proyecto de. Para obtener la URL pública de su proyecto de compilación, siga el procedimiento que se indica a continuación.

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. Elija el enlace del proyecto de compilación para el que desea obtener la URL pública.
4. La URL pública se muestra en el URL de proyecto público en el Configuración sección. Puede elegir el enlace para abrir la URL o copiar la URL con el botón copiar.

Warning

Tenga en cuenta lo siguiente a la hora de hacer públicos los resultados de compilación de su proyecto:

- Todos los resultados de compilación, registros y artefactos de un proyecto, incluidas las compilaciones que se ejecutaron cuando el proyecto era privado, están disponibles para el público.
- Todos los registros y artefactos de compilación están disponibles para el público. Es posible que se hayan emitido variables de entorno, código fuente y otra información confidencial en los registros y artefactos de compilación. Debe tener cuidado con la información que se genera en los registros de compilación. Algunas prácticas recomendadas son:
 - No almacene valores confidenciales, especialmente AWS los identificadores de clave de acceso y las claves de acceso secretas de en variables de entorno. Le recomendamos que utilice un almacén de parámetros Amazon EC2 Systems Manager o AWS Secrets Manager para almacenar valores sensibles.
 - Seguimiento [Prácticas recomendadas para el uso de webhooks \(p. 237\)](#) para limitar qué entidades pueden desencadenar una compilación y no almacenar la especificación de compilación en el propio proyecto, para garantizar que sus webhooks sean lo más seguros posible.
- Un usuario malintencionado puede utilizar compilaciones públicas para distribuir artefactos maliciosos. Recomendamos que los administradores del proyecto revisen todas las solicitudes de extracción para verificar que la solicitud de extracción sea un cambio legítimo. También te recomendamos que valides cualquier artefacto con sus sumas de comprobación para asegurarte de que se están descargando los artefactos correctos.

Trabajar con compilaciones en AWS CodeBuild

Una compilación representa un conjunto de acciones realizadas por AWS CodeBuild para crear artefactos de salida (por ejemplo, un archivo JAR) en función de una serie de artefactos de entrada (por ejemplo, una colección de archivos de clases de Java).

Las siguientes reglas se aplican cuando ejecuta varias compilaciones:

- Cuando es posible, las compilaciones se ejecutan simultáneamente. El número máximo de compilaciones en ejecución simultánea puede variar. Para obtener más información, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).
- Si el proyecto de compilación tiene un límite de compilación simultánea establecido, las compilaciones devuelven un error si el número de compilaciones en ejecución alcanza el límite de compilación simultánea para el proyecto. Para obtener más información, consulte [Habilitar límite de compilación simultánea \(p. 199\)](#).
- Si el proyecto de compilación no tiene establecido un límite de compilación simultánea, las compilaciones se ponen en cola si el número de compilaciones en ejecución alcanza el límite de compilación simultánea para la plataforma y el tipo de proceso. El número máximo de compilaciones de una cola es de cinco veces el límite de compilaciones simultáneas. Para obtener más información, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

Una compilación en una cola que no se inicia después del número de minutos especificado en su valor de tiempo de espera se elimina de la cola. El valor predeterminado del tiempo de espera es de ocho horas. Puede anular el tiempo de espera de la cola de compilación con un valor comprendido entre cinco minutos y ocho horas cuando ejecuta la compilación. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#).

No es posible predecir el orden en que se inician las compilaciones en cola.

Note

Puede acceder al historial de una compilación durante un año.

Puede realizar estas tareas cuando trabaje con compilaciones:

Temas

- [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#)
- [Ver detalles de las compilaciones de AWS CodeBuild \(p. 289\)](#)
- [Ver una lista de identificadores de compilación en AWS CodeBuild \(p. 292\)](#)
- [Ver una lista de identificadores de compilación de un proyecto de compilación en AWS CodeBuild \(p. 294\)](#)
- [Detener una compilación en AWS CodeBuild \(p. 297\)](#)
- [Detener una compilación por lotes AWS CodeBuild \(p. 298\)](#)
- [Reintentar una compilación en AWS CodeBuild \(p. 299\)](#)
- [Ver una compilación en ejecución en el Administrador de sesiones \(p. 300\)](#)
- [Eliminación de compilaciones en AWS CodeBuild \(p. 304\)](#)

Ejecutar una compilación en AWS CodeBuild

Puede utilizar el [AWS CodeBuild Consola de: AWS CLI](#), o bien [AWS Los SDK de para ejecutar una compilación en CodeBuild](#).

Temas

- [Ejecutar una compilación \(consola\) \(p. 282\)](#)
- [Ejecutar una compilación \(AWS CLI\) \(p. 282\)](#)
- [Ejecutar una compilación por lotes \(AWS CLI\) \(p. 287\)](#)
- [Iniciar la ejecución de compilaciones automáticamente \(AWS CLI\) \(p. 288\)](#)

- [Detener la ejecución de compilaciones automáticamente \(AWS CLI\) \(p. 289\)](#)
- [Ejecutar una compilación \(SDK de AWS\) \(p. 289\)](#)

Ejecutar una compilación (consola)

Para utilizar AWS CodePipeline para ejecutar una compilación con CodeBuild, omita estos pasos y siga las instrucciones de [Usar CodePipeline con CodeBuild \(p. 407\)](#).

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. En la lista de proyectos de compilación, elija el proyecto de compilación.
4. Puede ejecutar la compilación con la configuración predeterminada del proyecto de compilación o anular la configuración de compilación sólo para esta compilación.
 - a. Si desea ejecutar la compilación con la configuración predeterminada del proyecto de compilación, elija Iniciar compilación. La compilación comienza inmediatamente.
 - b. Si desea anular la configuración predeterminada del proyecto de compilación, elija Iniciar la compilación con invalidaciones. En el navegador Iniciar compilación, puede invalidar los elementos siguientes:
 - Build configuration (Configuración de la compilación)
 - Fuente
 - Reemplazos de variables de entorno

Si necesita seleccionar anulaciones más avanzadas, elija Sustitución avanzada de compilaciones. En esta página, puede invalidar los elementos siguientes:

- Build configuration (Configuración de la compilación)
- Fuente
- Entorno
- Especificación de desarrollo
- Artifacts
- Registros

Cuando haya realizado las selecciones de anulación, elija Iniciar compilación.

Para obtener información detallada acerca de esta compilación, consulte [Ver detalles de las compilaciones \(consola\) \(p. 290\)](#).

Ejecutar una compilación (AWS CLI)

Note

Para usar CodePipeline para ejecutar una compilación con AWS CodeBuild, omita estos pasos y siga las instrucciones de [Cree una canalización que use CodeBuild \(AWS CLI\) \(p. 412\)](#). Para obtener más información acerca del uso de AWS CLI con CodeBuild, consulte la [Referencia de la línea de comandos \(p. 402\)](#).

1. Ejecute el comando `start-build` de una de las siguientes formas:

```
aws codebuild start-build --project-name <project-name>
```

Utilice esta opción si desea ejecutar una compilación que utilice la última versión del artefacto de entrada de la compilación y la configuración existente del proyecto de compilación.

```
aws codebuild start-build --generate-cli-skeleton
```

Utilice esta opción si desea ejecutar una versión de compilación con una versión anterior del artefacto de entrada de compilación o si desea anular la configuración de los artefactos de salida de compilación, variables de entorno, especificación de compilación o período de tiempo de espera de compilación predeterminado.

2. Si ejecutará la consola `start-build` con el comando `--project-name` opción, reemplazar `<project-name>` Con el nombre del proyecto de compilación y vaya al paso 6 de este procedimiento. Para obtener una lista de proyectos de compilación, consulte [Ver una lista de nombres de proyectos de compilación \(p. 225\)](#).
3. Si ejecutará la consola `start-build` con el comando `--idempotency-token` En, se incluye un identificador único que distingue entre mayúsculas y minúsculas, con la opción `start-build-request`. El token es válido durante 5 horas después de la solicitud. Si repite el `start-build` con el mismo token, pero cambia un parámetro, CodeBuild devuelve un error de coincidencia de parámetro.
4. Si ejecuta el comando `start-build` con la opción `--generate-cli-skeleton`, se mostrarán en el resultado datos con formato JSON. Copie los datos en un archivo (por ejemplo, `start-build.json`) en la ubicación del equipo o instancia local en la que haya instalado la AWS CLI. Modifique los datos copiados para que coincidan con el siguiente formato y guarde los resultados:

```
{
  "projectName": "projectName",
  "sourceVersion": "sourceVersion",
  "artifactsOverride": {
    "type": "type",
    "location": "location",
    "path": "path",
    "namespaceType": "namespaceType",
    "name": "artifactsOverride-name",
    "packaging": "packaging"
  },
  "buildspecOverride": "buildspecOverride",
  "cacheOverride": {
    "location": "cacheOverride-location",
    "type": "cacheOverride-type"
  },
  "certificateOverride": "certificateOverride",
  "computeTypeOverride": "computeTypeOverride",
  "environmentTypeOverride": "environmentTypeOverride",
  "environmentVariablesOverride": {
    "name": "environmentVariablesOverride-name",
    "value": "environmentVariablesValue",
    "type": "environmentVariablesOverride-type"
  },
  "gitCloneDepthOverride": "gitCloneDepthOverride",
  "imageOverride": "imageOverride",
  "idempotencyToken": "idempotencyToken",
  "insecureSslOverride": "insecureSslOverride",
  "privilegedModeOverride": "privilegedModeOverride",
  "queuedTimeoutInMinutesOverride": "queuedTimeoutInMinutesOverride",
  "reportBuildStatusOverride": "reportBuildStatusOverride",
  "timeoutInMinutesOverride": "timeoutInMinutesOverride",
  "sourceAuthOverride": "sourceAuthOverride",
  "sourceLocationOverride": "sourceLocationOverride",
  "serviceRoleOverride": "serviceRoleOverride",
  "sourceTypeOverride": "sourceTypeOverride"
}
```

Reemplace los siguientes marcadores de posición:

- **projectName**: Cadena obligatoria. El nombre del proyecto de compilación que desea usar para esta compilación.
- **sourceVersion**: cadena opcional. Una versión del código fuente que se va a compilar, como la siguiente:
 - En Amazon S3, el ID de versión correspondiente a la versión del archivo ZIP de entrada que desea compilar. Si no se especifica **sourceVersion**, se usa la versión más reciente.
 - Para CodeCommit, el ID de confirmación correspondiente a la versión del código fuente que desea compilar. Si no se especifica **sourceVersion**, se usa el ID de confirmación HEAD de la ramificación personalizada. (No puede especificar un nombre de etiqueta para **sourceVersion**, pero puede especificar el ID de confirmación de la etiqueta).
 - Para GitHub, el ID de confirmación, el ID de solicitud de inserción, el nombre de ramificación o el nombre de etiqueta correspondiente a la versión de código fuente que desea compilar. Si se especifica un ID de solicitud de inserción, este debe tener el formato `pr/pull-request-ID` (por ejemplo, `pr/25`). Si se especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si no se especifica **sourceVersion**, se usa el ID de confirmación HEAD de la ramificación personalizada.
 - Para Bitbucket, el ID de confirmación, el nombre de ramificación o el nombre de etiqueta correspondiente a la versión de código fuente que desea compilar. Si se especifica un nombre de ramificación, se usa el ID de confirmación HEAD de la ramificación. Si no se especifica **sourceVersion**, se usa el ID de confirmación HEAD de la ramificación personalizada.
- Los siguientes marcadores de posición son para **artifactsOverride**.
 - **type**: Opcional. El tipo de artefacto de salida de la compilación que invalida el definido en el proyecto de compilación para esta compilación.
 - **location**: Opcional. La ubicación del artefacto de salida de la compilación que invalida la definida en el proyecto de compilación para esta compilación.
 - **path**: Opcional. La ruta del artefacto de salida de la compilación que invalida la definida en el proyecto de compilación para esta compilación.
 - **namespaceType**: Opcional. El tipo de ruta del artefacto de salida de la compilación que invalida el definido en el proyecto de compilación para esta compilación.
 - **Nombre de**: Opcional. El nombre del artefacto de salida de la compilación que invalida el definido en el proyecto de compilación para esta compilación.
 - **empaquetado**: Opcional. El tipo de empaquetado del artefacto de salida de la compilación que invalida el definido en el proyecto de compilación para esta compilación.
- **buildspecOverride**: Opcional. Declaración de buildspec que invalida la declaración definida en el proyecto de esta compilación. Si se establece este valor, puede ser una definición de buildspec en línea, la ruta a un archivo de buildspec alternativo en relación con el valor de la variable de entorno `CODEBUILD_SRC_DIR` incorporada, o la ruta a un bucket de S3. El bucket de S3 debe estar en la misma región de AWS que el proyecto de compilación. Especifique el archivo buildspec utilizando su ARN (por ejemplo, `arn:aws:s3:::my-codebuild-sample2/buildspec.yml`). Si este valor no se proporciona o se establece en una cadena vacía, el código fuente debe contener un archivo `buildspec.yml` en su directorio raíz. Para obtener más información, consulte [Nombre de archivo y ubicación de almacenamiento de buildspec](#) (p. 137).
- Los siguientes marcadores de posición son para **cacheOverride**.
 - **cacheOverride-location**: Opcional. La ubicación de un objeto `ProjectCache` para esta compilación que anula el objeto `ProjectCache` especificado en el proyecto de compilación. `cacheOverride` es opcional y toma un objeto `ProjectCache`. `location` es obligatorio en un objeto `ProjectCache`.
 - **Tipo de cacheOverride-**: Opcional. El tipo de un objeto `ProjectCache` para esta compilación que anula el objeto `ProjectCache` especificado en el proyecto de compilación.

`cacheOverride` es opcional y toma un objeto `ProjectCache`. `type` es obligatorio en un objeto `ProjectCache`.

- **`CertificateOverride`**: Opcional. El nombre de un certificado para esta compilación que sustituye el especificado en el proyecto de compilación.
- **`environmentTypeOverride`**: Opcional. Un tipo de contenedor para esta compilación que sustituye el especificado en el proyecto de compilación. La cadena válida actual es `LINUX_CONTAINER`.
- Los siguientes marcadores de posición son para `environmentVariablesOverride`.
 - **`EnvironmentVariablesOverride-name`**: Opcional. El nombre de una variable de entorno del proyecto de compilación cuyo valor desea invalidar para esta compilación.
 - **`EnvironmentVariablesOverride-type`**: Opcional. El tipo de variable de entorno del proyecto de compilación cuyo valor desea invalidar para esta compilación.
 - **`environmentVariablesValue`**: Opcional. El valor de la variable de entorno definida en el proyecto de compilación que desea invalidar para esta compilación.
- **`gitCloneDepthOverride`**: Opcional. El valor de Git clone depth en el proyecto de compilación cuyo valor desea invalidar para esta compilación. Si el tipo de origen es Amazon S3, este valor no se admite.
- **`ImageOverride`**: Opcional. El nombre de una imagen para esta compilación que sustituye el especificado en el proyecto de compilación.
- **`idempotencyToken`**: Opcional. Una cadena que sirve de token para especificar que la solicitud de compilación es idempotente. Puede elegir cualquier cadena de 64 caracteres o menos. El token es válido durante 5 horas después de la solicitud de compilación inicial. Si repite la solicitud de inicio y compilación con el mismo token, pero cambia un parámetro, CodeBuild devuelve un error de coincidencia de parámetro.
- **`insecureSslOverride`**: Valor booleano opcional que especifica si la configuración TLS no segura especificada se va a invalidar en el proyecto de compilación. La configuración de TLS no segura determina si deben omitirse las advertencias de TLS al conectarse al código fuente del proyecto. Esta sustitución se aplica únicamente si el origen de la compilación es GitHub Enterprise Server.
- **`privilegedModeOverride`**: Booleano opcional. Si se establece en `true`, la compilación sustituye el modo privilegiado en el proyecto de compilación.
- **`queuedTimeoutInMinutesOverride`**: Número entero opcional que especifica el número de minutos que se permite en cola una compilación antes de que transcurra el tiempo de espera. Su valor mínimo es de cinco minutos y su valor máximo es 480 minutos (ocho horas).
- **`reportBuildStatusOverride`**: Valor booleano opcional que especifica si se debe enviar al proveedor de código fuente el estado de inicio y finalización de una compilación. Si define este valor con un proveedor de código fuente distinto de GitHub, GitHub Enterprise Server o Bitbucket, se genera una excepción `invalidInputException`.
- **`sourceAuthOverride`**: cadena opcional. Un tipo de autorización para esta compilación que sustituye la definida en el proyecto de compilación. Esta sustitución se aplica únicamente si el origen del proyecto de compilación es Bitbucket o GitHub.
- **`sourceLocationOverride`**: cadena opcional. Una ubicación que sustituye la ubicación de código fuente de este proyecto por la definida en el proyecto de compilación.
- **`serviceRoleOverride`**: cadena opcional. El nombre de un rol de servicio para esta compilación que sustituye el especificado en el proyecto de compilación.
- **`sourceTypeOverride`**: cadena opcional. Un tipo de entrada de código fuente para esta compilación que sustituye la entrada de código fuente definida en el proyecto de compilación. Las cadenas válidas son `NO_SOURCE`, `CODECOMMIT`, `CODEPIPELINE`, `GITHUB`, `S3`, `BITBUCKET` y `GITHUB_ENTERPRISE`.
- **`timeoutInMinutesOverride`**: Número opcional. El número de minutos de tiempo de espera de la compilación que invalida el definido en el proyecto de compilación para esta compilación.

Le recomendamos que almacene una variable de entorno con un valor confidencial, como un `AWSID` de clave de acceso, una `AWS`La clave de acceso secreta de o una contraseña como un parámetro en el almacén de parámetros Amazon EC2 Systems Manager. CodeBuild puede utilizar un parámetro almacenado en el almacén de parámetros de Amazon EC2 Systems Manager solo si el nombre del parámetro comienza por `/CodeBuild/` (por ejemplo, `/CodeBuild/dockerLoginPassword`). Puede utilizar la consola de CodeBuild para crear un parámetro en Amazon EC2 Systems Manager. Elija `Create a parameter` (Crear un parámetro) y siga las instrucciones. (En ese cuadro de diálogo, para `Clave de KMS`, opcionalmente puede especificar el ARN de un `AWS KMS` en su cuenta. Amazon EC2 Systems Manager utiliza esta clave para cifrar el valor del parámetro durante el almacenamiento y descifrarlo durante la recuperación). Si usa la consola de CodeBuild para crear un parámetro, la consola asigna al principio del parámetro con `/CodeBuild/` ya que se está almacenando. Sin embargo, si utiliza la consola del almacén de parámetros de Amazon EC2 Systems Manager para crear un parámetro, el nombre del parámetro debe comenzar por `/CodeBuild/`, y debe establecer `TipoDe a Cadena segura`. Para obtener más información, consulte [AWS Systems Manager Parameter Store Tutorial: Crear y probar un parámetro String \(consola\)](#) en la Guía del usuario de Amazon EC2 Systems Manager.

Si el proyecto de compilación hace referencia a parámetros almacenados en el almacén de parámetros de Amazon EC2 Systems Manager, el rol de servicio del proyecto de compilación debe permitir `ssm:GetParameters`. Si eligió `Crear un nuevo rol de servicio` en la cuenta CodeBuild incluye automáticamente esta acción en el rol de servicio predeterminado del proyecto de compilación. Sin embargo, si eligió `Choose an existing service role from your account`, debe incluir esta acción en su rol de servicio por separado.

Las variables de entorno que defina reemplazan las variables de entorno existentes. Por ejemplo, si la imagen de Docker ya contiene una variable de entorno denominada `MY_VAR` con un valor de `my_value` y establece una variable de entorno denominada `MY_VAR` con un valor de `other_value`, `my_value` se reemplaza por `other_value`. Asimismo, si la imagen de Docker ya contiene una variable de entorno denominada `PATH` con un valor de `/usr/local/sbin:/usr/local/bin` y establece una variable de entorno denominada `PATH` con un valor de `$PATH:/usr/share/ant/bin`, `/usr/local/sbin:/usr/local/bin` se reemplaza por el valor literal `$PATH:/usr/share/ant/bin`.

No establezca variables de entorno con un nombre que empiece por `CODEBUILD_`. Este prefijo se reserva para uso interno.

Si se define una variable de entorno con el mismo nombre en varios lugares, el valor de la variable de entorno se determina de la siguiente manera:

- El valor de la llamada a la operación de inicio de la compilación tiene la máxima prioridad.
- El valor de la definición del proyecto de compilación es el siguiente en orden de prioridad.
- El valor en la declaración del archivo `buildspec` tiene la prioridad más baja.

Para obtener información sobre los valores válidos para estos marcadores de posición, consulte [Crear un proyecto de compilación \(AWS CLI\) \(p. 209\)](#). Para obtener una lista de las últimas opciones de configuración de un proyecto de compilación, consulte [Ver los detalles de un proyecto de compilación \(p. 227\)](#).

5. Cambie al directorio que contiene el archivo que acaba de guardar y ejecute de nuevo el comando `start-build`.

```
aws codebuild start-build --cli-input-json file://start-build.json
```

6. Si el comando se ejecuta correctamente, aparecerán datos similares a los que se describen en el procedimiento [Para ejecutar la compilación \(p. 24\)](#) en el resultado.

Para trabajar con información detallada acerca de esta compilación, anote el valor de `id` en la salida y después consulte [Ver detalles de las compilaciones \(AWS CLI\)](#) (p. 290).

Ejecutar una compilación por lotes (AWS CLI)

1. Ejecute el comando `start-build-batch` de una de las siguientes formas:

```
aws codebuild start-build-batch --project-name <project-name>
```

Utilice esta opción si desea ejecutar una compilación que utilice la última versión del artefacto de entrada de la compilación y la configuración existente del proyecto de compilación.

```
aws codebuild start-build-batch --generate-cli-skeleton > <json-file>
```

Utilice esta opción si desea ejecutar una versión de compilación con una versión anterior del artefacto de entrada de compilación o si desea anular la configuración de los artefactos de salida de compilación, variables de entorno, especificación de compilación o período de tiempo de espera de compilación predeterminado.

2. Si ejecutará la consola `start-build-batch` con el comando `--project-name` opción, reemplazar `<project-name>` Con el nombre del proyecto de compilación y vaya al paso 6 de este procedimiento. Para obtener una lista de proyectos de compilación, consulte [Ver una lista de nombres de proyectos de compilación](#) (p. 225).
3. Si ejecutará la consola `start-build-batch` con el comando `--idempotency-token` En, se incluye un identificador único que distingue entre mayúsculas y minúsculas, o token, con la opción `start-build-batch-request`. El token es válido durante 5 horas después de la solicitud. Si repite el `start-build-batch` Con el mismo token, pero cambia un parámetro, CodeBuild devuelve un error de coincidencia de parámetro.
4. Si ejecutará la consola `start-build-batch` con el comando `--generate-cli-skeleton`, los datos formateados JSON se envían a la `<json-file>` file. Este archivo es similar al skelton producido por el `start-build`, con la adición del siguiente objeto. Para obtener más información acerca de los objetos comunes, consulte [Ejecutar una compilación \(AWS CLI\)](#) (p. 282).

Modifique este archivo para agregar las modificaciones de compilación y guarde los resultados.

```
{
  "buildBatchConfigOverride": {
    "combineArtifacts": combineArtifacts,
    "restrictions": {
      "computeTypesAllowed": [
        allowedComputeTypes
      ],
      "maximumBuildsAllowed": maximumBuildsAllowed
    },
    "serviceRole": "batchServiceRole",
    "timeoutInMins": batchTimeout
  }
}
```

La `buildBatchConfigOverride` es un objeto `ProjectBuildBatchConfig` que contiene los overrides de configuración de compilación por lotes para esta compilación.

Combineartifacts

Valor booleano que especifica si los artefactos de compilación de la compilación por lotes se deben combinar en una única ubicación de artefacto.

Una matriz de cadenas que especifican los tipos de computación permitidos para la compilación por lotes. Consulte [Tipos de computación del entorno de compilación](#) para estos valores.

Especifica el número máximo de compilaciones permitidas.

Especifica el ARN del rol de servicio para el proyecto de compilación por lotes.

Especifica la cantidad máxima de tiempo, en minutos, en que se debe completar la compilación por lotes.

- ```
aws codebuild start-build-batch --cli-input-json file://start-build.json
```

- ## Iniciar la ejecución de compilaciones automáticamente (AWS CLI)

Ejecute el comando `create-webhook` de la siguiente manera:

```
aws codebuild create-webhook --project-name <project-name>
```

En GitHub, en la salida se muestra información similar a la siguiente:

```
{
 "webhook": {
 "url": "<url>"
 }
}
```

En GitHub Enterprise Server, en la salida se muestra información similar a la siguiente:

[illegible]



1. Copie la URL de carga y la clave secreta de la salida. Las necesitará para añadir un webhook en GitHub Enterprise Server.
2. En GitHub Enterprise Server, elija el repositorio en el que está almacenado su proyecto de CodeBuild. Elija Settings (Configuración), elija Hooks & services (Enlaces y servicios) y, a continuación, elija Add webhook (Añadir webhook).
3. Introduzca la URL de carga y la clave secreta, acepte los valores predeterminados para el resto de los campos y elija Add webhook.

## Detener la ejecución de compilaciones automáticamente (AWS CLI)

Si el código fuente está almacenado en un repositorio de GitHub o GitHub Enterprise, puede configurar webhooks de GitHub para que AWS CodeBuild vuelva a compilar el código fuente siempre que se inserte un cambio de código en el repositorio. Para obtener más información, consulte [Iniciar la ejecución de compilaciones automáticamente \(AWS CLI\)](#) (p. 288).

Si ha activado este comportamiento, puede desactivarlo ejecutando los comandos `delete-webhook` como se indica a continuación:

```
aws codebuild delete-webhook --project-name <project-name>
```

- where `<project-name>` Es el nombre del proyecto de compilación que contiene el código fuente que se va a volver a compilar.

Si este comando se ejecuta correctamente, no aparecerá información ni errores en el resultado.

### Note

Esto elimina el webhook únicamente del proyecto de CodeBuild. También debe eliminar el webhook del repositorio de GitHub o GitHub Enterprise Server.

## Ejecutar una compilación (SDK de AWS)

Para usar CodePipeline para ejecutar una compilación con AWS CodeBuild, omita estos pasos y siga las instrucciones de [Usar AWS CodePipeline con AWS CodeBuild para probar el código y ejecutar compilaciones](#) (p. 407) en su lugar.

Para obtener información sobre el uso de CodeBuild con el AWS SDK, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

## Ver detalles de las compilaciones de AWS CodeBuild

Puede utilizar el AWS CodeBuild Consola de: AWS CLI, o bien AWS para ver detalles sobre las compilaciones administradas por CodeBuild.

### Temas

- [Ver detalles de las compilaciones \(consola\)](#) (p. 290)
- [Ver detalles de las compilaciones \(AWS CLI\)](#) (p. 290)
- [Ver detalles de las compilaciones \(SDK de AWS\)](#) (p. 290)
- [Transiciones de fases de compilación](#) (p. 291)



## Ver detalles de las compilaciones (consola)

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Aplique alguna de las siguientes acciones:
  - En el panel de navegación, elija Build history. En la lista de compilaciones, en la columna Build run (Ejecución de compilación), elija el enlace de la compilación.
  - En el panel de navegación, elija Build projects. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el enlace del nombre del proyecto de compilación. A continuación, en la lista de compilaciones, en la columna Build run (Ejecución de compilación), elija el enlace de la compilación.

### Note

De forma predeterminada, solo se muestran las diez compilaciones o proyectos de compilación más recientes. Para ver más compilaciones o proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Builds per page (Compilaciones por página) o Projects per page (Proyectos por página) o seleccione las flechas atrás y adelante.

## Ver detalles de las compilaciones (AWS CLI)

Para obtener más información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos](#) (p. 402).

Ejecute el comando batch-get-builds:

```
aws codebuild batch-get-builds --ids ids
```

Reemplace el siguiente marcador de posición:

- *ids*: Cadena obligatoria. Uno o varios ID de compilación para los que desea ver los detalles. Para especificar más de un ID de compilación, separe cada ID de compilación con un espacio. Puede especificar hasta 100 ID de compilación. Para obtener una lista de identificadores de compilación, consulte los siguientes temas:
  - [Ver una lista de identificadores de compilación \(AWS CLI\)](#) (p. 292)
  - [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\)](#) (p. 295)

Por ejemplo, si ejecuta este comando:

```
aws codebuild batch-get-builds --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE my-other-project:813bb6c6-891b-426a-9dd7-6d8a3EXAMPLE
```

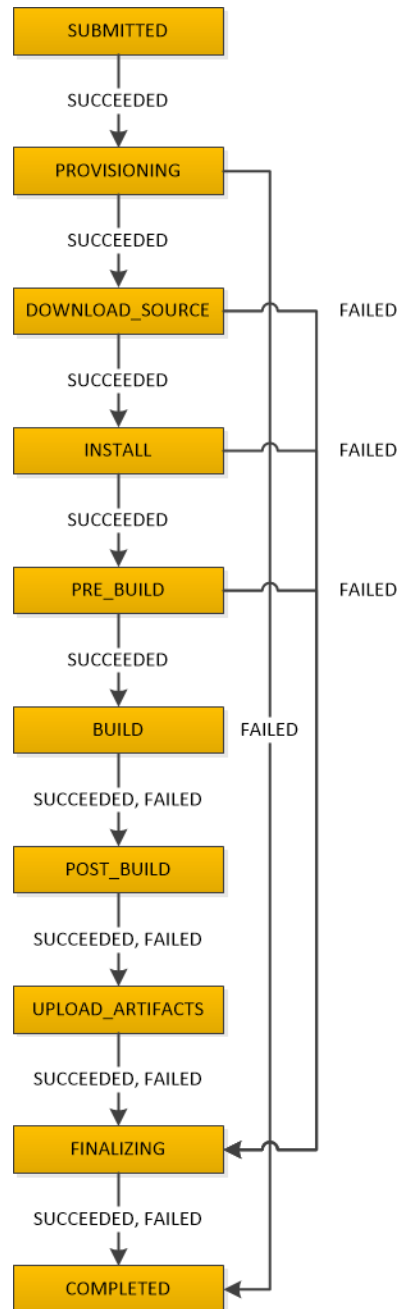
Si el comando tiene éxito, aparecen datos similares a los descritos en [Para ver la información resumida de la compilación](#) (p. 26) en la salida.

## Ver detalles de las compilaciones (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

## Transiciones de fases de compilación

Las compilaciones de AWS CodeBuild avanzan en fases:



### Important

La fase `UPLOAD_ARTIFACTS` se intenta siempre, aunque la fase `BUILD` produzca un error.

## Ver una lista de identificadores de compilación en AWS CodeBuild

Puede utilizar el [AWS CodeBuild Consola de: AWS CLI](#), o bien [AWS](#) Para ver una lista de identificadores de compilación de las compilaciones administradas por CodeBuild.

### Temas

- [Ver una lista de identificadores de compilación \(consola\)](#) (p. 292)
- [Ver una lista de identificadores de compilación \(AWS CLI\)](#) (p. 292)
- [Ver una lista de identificadores de compilación por lotes \(AWS CLI\)](#) (p. 293)
- [Ver una lista de identificadores de compilación \(SDK de AWS\)](#) (p. 294)

## Ver una lista de identificadores de compilación (consola)

1. Abra el icono [AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>](#).
2. En el panel de navegación, elija Build history.

### Note

De forma predeterminada, solo se muestran las diez compilaciones más recientes. Para ver más compilaciones, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Builds per page (Compilaciones por página) o use las flechas atrás y adelante.

## Ver una lista de identificadores de compilación (AWS CLI)

Para obtener más información acerca del uso de [AWS CLI](#) con CodeBuild, consulte la [Referencia de la línea de comandos](#) (p. 402).

- Ejecute el comando list-builds:

```
aws codebuild list-builds --sort-order sort-order --next-token next-token
```

En el comando anterior, sustituya los siguientes marcadores de posición:

- **orden de clasificación:** Cadena opcional que se utiliza para indicar cómo muestran los identificadores de compilación. Los valores válidos son ASCENDING y DESCENDING.
- **siguiente token:** cadena opcional. Durante una ejecución anterior, si hubiera más de 100 elementos en la lista, solo se devolverían los 100 primeros, junto con una única cadena denominada next token. Para obtener el siguiente lote de elementos de la lista, ejecute de nuevo este comando añadiendo el siguiente token a la llamada. Para obtener todos los elementos de la lista, siga ejecutando el comando con cada uno de los siguientes tokens hasta que no se devuelvan más tokens.

Por ejemplo, si ejecuta este comando:

```
aws codebuild list-builds --sort-order ASCENDING
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
 "nextToken": "4AEA6u7J...The full token has been omitted for brevity...MzY2OA==",
```

```
"ids": [
 "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE"
 "codebuild-demo-project:84a7f3d1-d40e-4956-b4cf-7a9d4EXAMPLE"
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:931d0b72-bf6f-4040-a472-5c707EXAMPLE"
]
}
```

Si ejecuta este comando de nuevo:

```
aws codebuild list-builds --sort-order ASCENDING --next-token 4AEA6u7J...The full token
has been omitted for brevity...MzY2OA==
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
 "ids": [
 "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
 "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
]
}
```

## Ver una lista de identificadores de compilación por lotes (AWS CLI)

Para obtener más información acerca del uso de AWS CLI con CodeBuild, consulte la [Referencia de la línea de comandos](#) (p. 402).

- Ejecute el comando `list-build-batches`:

```
aws codebuild list-build-batches --sort-order sort-order --next-token next-token
```

En el comando anterior, sustituya los siguientes marcadores de posición:

- **orden de clasificación**: Cadena opcional que se utiliza para indicar cómo muestran los identificadores de compilación por lotes. Los valores válidos son `ASCENDING` y `DESCENDING`.
- **siguiente token**: cadena opcional. Durante una ejecución anterior, si hubiera más de 100 elementos en la lista, solo se devolverían los 100 primeros, junto con una única cadena denominada `next token`. Para obtener el siguiente lote de elementos de la lista, ejecute de nuevo este comando añadiendo el siguiente token a la llamada. Para obtener todos los elementos de la lista, siga ejecutando el comando con cada uno de los siguientes tokens hasta que no se devuelvan más tokens.

Por ejemplo, si ejecuta este comando:

```
aws codebuild list-build-batches --sort-order ASCENDING
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
 "nextToken": "4AEA6u7J...The full token has been omitted for brevity...MzY2OA==",
 "ids": [
 "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
 "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
]
}
```

```
"codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE"
"codebuild-demo-project:84a7f3d1-d40e-4956-b4cf-7a9d4EXAMPLE"
... The full list of build IDs has been omitted for brevity ...
"codebuild-demo-project:931d0b72-bf6f-4040-a472-5c707EXAMPLE"
]
}
```

Si ejecuta este comando de nuevo:

```
aws codebuild list-build-batches --sort-order ASCENDING --next-token 4AEA6u7J...The
full token has been omitted for brevity...MzY2OA==
```

Un resultado similar al siguiente podría aparecer en la salida:

```
{
 "ids": [
 "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
 "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
]
}
```

## Ver una lista de identificadores de compilación (SDK de AWS)

Para obtener más información acerca del uso de CodeBuild con elAWSPara los SDK, consulte[Referencia de los SDK y las herramientas de AWS](#) (p. 403).

## Ver una lista de identificadores de compilación de un proyecto de compilación en AWS CodeBuild

Puede utilizar elAWS CodeBuildConsola de:AWS CLI, o bienAWSPara ver una lista de identificadores de compilación de un proyecto de compilación en CodeBuild.

### Temas

- [Ver una lista de identificadores de compilación de un proyecto de compilación \(consola\)](#) (p. 294)
- [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\)](#) (p. 295)
- [Ver una lista de identificadores de compilación por lotes de un proyecto de compilación \(AWS CLI\)](#) (p. 296)
- [Ver una lista de identificadores de compilación de un proyecto de compilación \(SDK de AWS\)](#) (p. 297)

## Ver una lista de identificadores de compilación de un proyecto de compilación (consola)

1. Abra la consola CodeBuild en<https://console.aws.amazon.com/codebuild/>.
2. En el panel de navegación, elija Build projects. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el proyecto de compilación.

## Note

De forma predeterminada, solo se muestran las 100 compilaciones o proyectos de compilación más recientes. Para ver más compilaciones o proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Builds per page (Compilaciones por página) o Projects per page (Proyectos por página) o seleccione las flechas atrás y adelante.

## Ver una lista de identificadores de compilación de un proyecto de compilación (AWS CLI)

Para obtener más información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos](#) (p. 402).

Ejecute el comando `list-builds-for-project` de la siguiente manera:

```
aws codebuild list-builds-for-project --project-name project-name --sort-order sort-order
--next-token next-token
```

En el comando anterior, sustituya los siguientes marcadores de posición:

- ***Project-name***: Cadena obligatoria que se utiliza para indicar el nombre del proyecto de compilación para el que desea mostrar los identificadores de compilación. Para obtener una lista de proyectos de compilación, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) (p. 225).
- ***orden de clasificación***: Cadena opcional que se utiliza para indicar cómo muestran los identificadores de compilación. Los valores válidos son `ASCENDING` y `DESCENDING`.
- ***siguiente token***: cadena opcional. Durante una ejecución anterior, si hubiera más de 100 elementos en la lista, solo se devolverían los 100 primeros, junto con una única cadena denominada next token. Para obtener el siguiente lote de elementos de la lista, ejecute de nuevo este comando añadiendo el siguiente token a la llamada. Para obtener todos los elementos de la lista, siga ejecutando el comando con cada uno de los siguientes tokens devueltos hasta que no se devuelvan más tokens.

Por ejemplo, si ejecuta un comando similar a este:

```
aws codebuild list-builds-for-project --project-name codebuild-demo-project --sort-order
ASCENDING
```

Aparecerá un resultado como el siguiente en la salida:

```
{
 "nextToken": "4AEA6u7J...The full token has been omitted for brevity...MzY2OA==",
 "ids": [
 "codebuild-demo-project:9b175d16-66fd-4e71-93a0-50a08EXAMPLE"
 "codebuild-demo-project:a9d1bd09-18a2-456b-8a36-7d65aEXAMPLE"
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:fe70d102-c04f-421a-9cfa-2dc15EXAMPLE"
]
}
```

Si ejecuta este comando de nuevo:

```
aws codebuild list-builds-for-project --project-name codebuild-demo-project --sort-order
ASCENDING --next-token 4AEA6u7J...The full token has been omitted for brevity...MzY2OA==
```

Podría aparecer un resultado como el siguiente en la salida:

```
{
 "ids": [
 "codebuild-demo-project:98253670-7a8a-4546-b908-dc890EXAMPLE"
 "codebuild-demo-project:ad5405b2-1ab3-44df-ae2d-fba84EXAMPLE"
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:f721a282-380f-4b08-850a-e0ac1EXAMPLE"
]
}
```

## Ver una lista de identificadores de compilación por lotes de un proyecto de compilación (AWS CLI)

Para obtener más información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos](#) (p. 402).

Ejecute el comando `list-build-batches-for-project` de la siguiente manera:

```
aws codebuild list-build-batches-for-project --project-name project-name --sort-order sort-order --next-token next-token
```

En el comando anterior, sustituya los siguientes marcadores de posición:

- ***Project-name***: Cadena obligatoria que se utiliza para indicar el nombre del proyecto de compilación para el que desea mostrar los identificadores de compilación. Para obtener una lista de proyectos de compilación, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) (p. 225).
- ***orden de clasificación***: Cadena opcional que se utiliza para indicar cómo muestran los identificadores de compilación. Los valores válidos son `ASCENDING` y `DESCENDING`.
- ***siguiente token***: cadena opcional. Durante una ejecución anterior, si hubiera más de 100 elementos en la lista, solo se devolverían los 100 primeros, junto con una única cadena denominada next token. Para obtener el siguiente lote de elementos de la lista, ejecute de nuevo este comando añadiendo el siguiente token a la llamada. Para obtener todos los elementos de la lista, siga ejecutando el comando con cada uno de los siguientes tokens devueltos hasta que no se devuelvan más tokens.

Por ejemplo, si ejecuta un comando similar a este:

```
aws codebuild list-build-batches-for-project --project-name codebuild-demo-project --sort-order ASCENDING
```

Aparecerá un resultado como el siguiente en la salida:

```
{
 "nextToken": "4AEA6u7J...The full token has been omitted for brevity...MzY2OA==",
 "ids": [
 "codebuild-demo-project:9b175d16-66fd-4e71-93a0-50a08EXAMPLE"
 "codebuild-demo-project:a9d1bd09-18a2-456b-8a36-7d65aEXAMPLE"
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:fe70d102-c04f-421a-9cfa-2dc15EXAMPLE"
]
}
```

Si ejecuta este comando de nuevo:

```
aws codebuild list-build-batches-for-project --project-name codebuild-demo-project
--sort-order ASCENDING --next-token 4AEA6u7J...The full token has been omitted for
brevity...MzY2OA==
```

Podría aparecer un resultado como el siguiente en la salida:

```
{
 "ids": [
 "codebuild-demo-project:98253670-7a8a-4546-b908-dc890EXAMPLE"
 "codebuild-demo-project:ad5405b2-1ab3-44df-ae2d-fba84EXAMPLE"
 ... The full list of build IDs has been omitted for brevity ...
 "codebuild-demo-project:f721a282-380f-4b08-850a-e0ac1EXAMPLE"
]
}
```

## Ver una lista de identificadores de compilación de un proyecto de compilación (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

## Detener una compilación en AWS CodeBuild

Puede usar la consola de AWS CodeBuild, la AWS CLI o los SDK de AWS para detener una compilación de AWS CodeBuild.

### Temas

- [Detener una compilación \(consola\)](#) (p. 297)
- [Detener una compilación \(AWS CLI\)](#) (p. 297)
- [Detener una compilación \(SDK de AWS\)](#) (p. 298)

## Detener una compilación (consola)

1. Abra el icono **AWS CodeBuild Consola** de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Aplique alguna de las siguientes acciones:
  - Si aparece la página **nombre-proyecto-compilación:ID-compilación**, elija **Stop build** (Detener compilación).
  - En el panel de navegación, elija **Build history**. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija **Stop build** (Detener compilación).
  - En el panel de navegación, elija **Build projects**. En la lista de proyectos de compilación, en la columna **Name** (Nombre), elija el enlace correspondiente al nombre del proyecto de compilación. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija **Stop build** (Detener compilación).

### Note

De forma predeterminada, solo se muestran las 100 compilaciones o proyectos de compilación más recientes. Para ver más compilaciones o proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para **Builds per page** (Compilaciones por página) o **Projects per page** (Proyectos por página) o seleccione las flechas atrás y adelante. Si AWS CodeBuild no puede detener correctamente una compilación (por ejemplo, porque el proceso ya se ha completado), el botón **Stop** (Detener) estará desactivado o podría no aparecer.

## Detener una compilación (AWS CLI)

- Ejecute el comando `stop-build`:



```
aws codebuild stop-build --id id
```

En el comando anterior, sustituya el siguiente marcador de posición:

- *id*: Cadena obligatoria. El ID de la compilación que desea detener. Para obtener una lista de identificadores de compilación, consulte los siguientes temas:
  - [Ver una lista de identificadores de compilación \(AWS CLI\) \(p. 292\)](#)
  - [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\) \(p. 295\)](#)

Si AWS CodeBuild detiene correctamente la compilación, el valor de `buildStatus` del objeto `build` de la salida será `STOPPED`.

Si CodeBuild no puede detener correctamente la compilación (por ejemplo, porque la compilación ya se ha completado), la opción `buildStatus` en el campo `build` en la salida es el estado de compilación final (por ejemplo, `SUCCEEDED`).

## Detener una compilación (SDK de AWS)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS \(p. 403\)](#).

## Detener una compilación por lotesAWS CodeBuild

Puede utilizar elAWS CodeBuildConsola de:AWS CLI, oAWSLos SDK de para detener una compilación por lotesAWS CodeBuild.

### Temas

- [Detener una compilación por lotes \(consola\) \(p. 298\)](#)
- [Detener una compilación por lotes \(AWS CLI\) \(p. 299\)](#)
- [Detener una compilación por lotes \(AWSSDK\) \(p. 299\)](#)

## Detener una compilación por lotes (consola)

1. Abra el iconoAWS CodeBuildConsola de:<https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Aplique alguna de las siguientes acciones:
  - Si aparece la página **nombre-proyecto-compilación:ID-compilación**, elija Stop build (Detener compilación).
  - En el panel de navegación, elija Build history. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija Stop build (Detener compilación).
  - En el panel de navegación, elija Build projects. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el enlace correspondiente al nombre del proyecto de compilación. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija Stop build (Detener compilación).

### Note

De forma predeterminada, solo se muestran las 100 compilaciones o proyectos de compilación más recientes. Para ver más compilaciones o proyectos de compilación, elija el icono de rueda

dentada y, a continuación, seleccione un valor diferente para Builds per page (Compilaciones por página) o Projects per page (Proyectos por página) o seleccione las flechas atrás y adelante. Si AWS CodeBuild no puede detener correctamente una compilación por lotes (por ejemplo, porque el proceso ya se ha completado), la opción Detener compilación está desactivado.

## Detener una compilación por lotes (AWS CLI)

- Ejecute el comando `stop-build-batch`:

```
aws codebuild stop-build-batch --id <batch-build-id>
```

En el comando anterior, sustituya el siguiente marcador de posición:

- **<batch-build-id>**: Cadena obligatoria. El identificador de la compilación por lotes que desea detener. Para obtener una lista de identificadores de compilación por lotes, consulte los siguientes temas:
  - [Ver una lista de identificadores de compilación por lotes \(AWS CLI\) \(p. 293\)](#)
  - [Ver una lista de identificadores de compilación por lotes de un proyecto de compilación \(AWS CLI\) \(p. 296\)](#)

Si AWS CodeBuild tiene correctamente la compilación por lotes, el `buildBatchStatus` en el campo `buildBatch` en la salida es `STOPPED`.

Si CodeBuild no puede detener correctamente la compilación por lotes (por ejemplo, porque la compilación por lotes ya se ha completado), el `buildBatchStatus` en el campo `buildBatch` en la salida es el estado de compilación final (por ejemplo, `SUCCEEDED`).

## Detener una compilación por lotes (AWS SDK)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS \(p. 403\)](#).

## Reintente una compilación en AWS CodeBuild

Puede utilizar el AWS CodeBuild Consola de: AWS CLI, o AWS SDK para reintentar una sola compilación o una compilación por lotes en AWS CodeBuild.

Temas

- [Reintentar una compilación \(consola\) \(p. 299\)](#)
- [Reintentar una compilación \(AWS CLI\) \(p. 300\)](#)
- [Reintentar una compilación \(AWS SDK\) \(p. 300\)](#)

## Reintentar una compilación (consola)

1. Abra el icono AWS CodeBuild Consola de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Aplique alguna de las siguientes acciones:
  - Si el archivo de **nombre-proyecto-construcción-proyecto:ID de compilación** en la página, haga clic Reintento de compilación.
  - En el panel de navegación, elija Build history. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija Reintento de compilación.

- En el panel de navegación, elija Build projects. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el enlace correspondiente al nombre del proyecto de compilación. En la lista de compilaciones, seleccione la casilla correspondiente a la compilación y después elija Reintentar de compilación.

#### Note

De forma predeterminada, solo se muestran las 100 compilaciones o proyectos de compilación más recientes. Para ver más compilaciones o proyectos de compilación, elija el icono de rueda dentada y, a continuación, seleccione un valor diferente para Builds per page (Compilaciones por página) o Projects per page (Proyectos por página) o seleccione las flechas atrás y adelante.

## Reintentar una compilación (AWS CLI)

- Ejecute el comando retry-build:

```
aws codebuild retry-build --id <build-id> --idempotency-token <idempotencyToken>
```

En el comando anterior, sustituya el siguiente marcador de posición:

- **<build-id>**: Cadena obligatoria. El ID de la compilación por lotes que desea reintentar. Para obtener una lista de identificadores de compilación, consulte los siguientes temas:
  - [Ver una lista de identificadores de compilación \(AWS CLI\) \(p. 292\)](#)
  - [Ver una lista de identificadores de compilación por lotes \(AWS CLI\) \(p. 293\)](#)
  - [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\) \(p. 295\)](#)
  - [Ver una lista de identificadores de compilación por lotes de un proyecto de compilación \(AWS CLI\) \(p. 296\)](#)
- **--idempotency-token**: opcional. Si ejecutará la consola `retry-build` con la opción, se incluye un identificador único que distingue entre mayúsculas y minúsculas, o token, con la opción `retry-buildrequest`. El token es válido durante 5 horas después de la solicitud. Si repite el `retry-build` con el mismo token, pero cambia un parámetro, CodeBuild devuelve un error de coincidencia de parámetro.

## Reintentar una compilación (AWS SDK)

Para obtener más información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS \(p. 403\)](#).

## Ver una compilación en ejecución en el Administrador de sesiones

En AWS CodeBuild, puede pausar una compilación en ejecución y luego usar AWS Systems Manager Administrador de sesiones para conectarse al contenedor de compilación y ver el estado del contenedor.

#### Note

Esta característica no está disponible en entornos Windows.

#### Temas

- [Prerequisites \(p. 301\)](#)

- [Pausar la compilación \(p. 302\)](#)
- [Iniciar la compilación \(p. 302\)](#)
- [Connect al contenedor de compilación \(p. 303\)](#)
- [Reanudar la compilación \(p. 303\)](#)

## Prerequisites

Para permitir que el Administrador de sesiones se utilice con la sesión de compilación, debe habilitar la conexión de sesión para la compilación. Hay dos requisitos previos:

- Las imágenes seleccionadas estándar de CodeBuild Linux ya tienen instalado el agente SSM y el agente SSM ContainerMode habilitado.

Si está utilizando una imagen personalizada para su compilación, haga lo siguiente:

1. Instalar SSM Agent de . Para obtener más información, consulte [Instalar manualmente el SSM Agent en instancias EC2 para Linux](#) en la AWS Systems Manager Guía del usuario de . La versión de SSM Agent debe ser 3.0.1295.0 o posterior.
  2. Copie el archivo <https://github.com/aws/aws-codebuild-docker-images/blob/master/ubuntu/standard/4.0/amazon-ssm-agent.json> a la /etc/amazon/ssm/ en su imagen. Esto habilita el modo contenedor en el agente SSM.
- El rol de servicio CodeBuild debe tener la siguiente política de SSM:

```
{
 "Effect": "Allow",
 "Action": [
 "ssmmessages:CreateControlChannel",
 "ssmmessages:CreateDataChannel",
 "ssmmessages:OpenControlChannel",
 "ssmmessages:OpenDataChannel"
],
 "Resource": "*"
}
```

Puede hacer que la consola de CodeBuild adjunte automáticamente esta directiva a su función de servicio cuando inicie la compilación. También puede adjuntar esta política al rol de servicio manualmente.

- Si tiene Auditoría y registro de la actividad de las sesiones habilitada en las preferencias de Systems Manager, el rol de servicio CodeBuild también debe tener permisos adicionales. Los permisos son diferentes, dependiendo de dónde se almacenan los registros.

Registros de CloudWatch

Si utiliza CloudWatch Logs para almacenar los registros, agregue el siguiente permiso a la función de servicio CodeBuild:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "logs:DescribeLogGroups",
 "Resource": "arn:aws:logs:<region-id>:<account-id>:log-group:*:*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "logs:CreateLogStream",

```

```
 "logs:PutLogEvents"
],
 "Resource": "arn:aws:logs:<region-id>:<account-id>:log-group:<log-group-
name>:*"
 }
]
}
```

### Amazon S3

Si utiliza Amazon S3 para almacenar los registros, añada el siguiente permiso al rol de servicio CodeBuild:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "s3:GetEncryptionConfiguration",
 "s3:PutObject"
],
 "Resource": [
 "arn:aws:s3:::<bucket-name>",
 "arn:aws:s3:::<bucket-name>/*"
]
 }
]
}
```

Para obtener más información, consulte [Auditoría y registro de la actividad de las sesiones](#) en la [AWS Systems Manager Guía del usuario](#) de.

## Pausar la compilación

Para pausar la compilación, inserte el `codebuild-breakpoint` en cualquiera de las fases de compilación de su archivo `buildspec`. La compilación se pausará en este punto, lo que le permite conectarse al contenedor de compilación y ver el contenedor en su estado actual.

Por ejemplo, añada lo siguiente a las fases de compilación del archivo `buildspec`.

```
phases:
 pre_build:
 commands:
 - echo Entered the pre_build phase...
 - echo "Hello World" > /tmp/hello-world
 - codebuild-breakpoint
```

Este código crea el `/tmp/hello-world`, a continuación, pausa la compilación en este punto.

## Iniciar la compilación

Para permitir que el Administrador de sesiones se utilice con la sesión de compilación, debe habilitar las conexiones de sesión para la compilación. Para ello, al iniciar la compilación, siga estos pasos:

1. Abra el icono **AWS CodeBuild Consola** de: <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija **Build projects**. Elija el proyecto de compilación y, a continuación, elija **Start build**.

3. Seleccione Advanced build overrides (Sustitución avanzada de compilaciones).
4. En el navegador Environment (Entorno):, haga clic en la [Habilitar conexión de sesión](#) opción. Si esta opción no está seleccionada, todos los `codebuild-breakpoint` y `codebuild-resume` comandos se omiten.
5. En el navegador Environment (Entorno):, haga clic en la [Permitir AWS CodeBuild](#) Para modificar este rol de servicio para poder usarlo con este proyecto de compilación para permitir que la consola de CodeBuild adjunte automáticamente la directiva del administrador de sesiones a su rol de servicio. Si ya ha agregado la directiva del administrador de sesiones a su rol, no es necesario seleccionar esta opción.
6. Realice cualquier otro cambio deseado y elija [Iniciar compilación](#).
7. Supervise el estado de compilación en la consola. Cuando la sesión está disponible, el [AWS Administrador de sesiones](#) aparece en el Estado de la compilación sección.

## Connect al contenedor de compilación

Puede conectarse al contenedor de compilación de una de las dos formas siguientes:

### CodeBuild

En un navegador web, abra el [AWS Administrador de sesiones](#) para conectarse al contenedor de compilación. Se abre una sesión de terminal que le permite examinar y controlar el contenedor de compilación.

### AWS CLI

#### Note

El equipo local debe tener instalado el complemento Administrador de sesiones para este procedimiento. Para obtener más información, consulte [Instale el complemento del Administrador de sesiones para el AWS CLI](#) en la [AWS Systems Manager Guía del usuario](#) de .

1. Llamada a `aws codebuild batch-get-builds` con el ID de compilación para obtener información sobre la compilación, incluido el identificador de destino de sesión. El nombre de la propiedad del identificador de destino de sesión varía según el tipo de salida de la `aws` comando. Es por eso que `--output json` Se agrega al comando.

```
aws codebuild batch-get-builds --ids <buildID> --region <region> --output json
```

2. Copie la consola `sessionTarget` Valor de la propiedad. La `sessionTarget` El nombre de propiedad puede variar en función del tipo de salida de la propiedad `aws` comando. Es por eso que `--output json` En el paso anterior se agrega al comando.
3. Utilice el siguiente comando para conectarse al contenedor de compilación.

```
aws ssm start-session --target <sessionTarget> --region <region>
```

Para este ejemplo, compruebe que `el/tmp/hello-world` existe y contiene el texto `Hello World`.

## Reanudar la compilación

Después de que termine de examinar el contenedor de compilación, ejecute el `codebuild-resume` desde el shell del contenedor.

```
$ codebuild-resume
```

## Eliminación de compilaciones en AWS CodeBuild

Puede utilizar AWS CLI o los SDK de AWS para eliminar compilaciones en AWS CodeBuild.

### Eliminación de compilaciones (AWS CLI)

Ejecute el comando `batch-delete-builds`:

```
aws codebuild batch-delete-builds --ids ids
```

En el comando anterior, sustituya el siguiente marcador de posición:

- *ids*: Cadena obligatoria. Los ID de las compilaciones que se van a eliminar. Para especificar varias compilaciones, separe cada ID de compilación con un espacio. Para obtener una lista de identificadores de compilación, consulte los siguientes temas:
  - [Ver una lista de identificadores de compilación \(AWS CLI\)](#) (p. 292)
  - [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\)](#) (p. 295)

En caso de éxito, aparece una matriz `buildsDeleted` en la salida, que contiene el Nombre de recurso de Amazon (ARN) de cada compilación eliminada correctamente. La información sobre las compilaciones que no se eliminaron correctamente aparece en la salida dentro de la matriz `buildsNotDeleted`.

Por ejemplo, si ejecuta este comando:

```
aws codebuild batch-delete-builds --ids my-demo-build-project:f8b888d2-5e1e-4032-8645-b115195648EX my-other-demo-build-project:a18bc6ee-e499-4887-b36a-8c90349c7eEX
```

En la salida se muestra información similar a la siguiente:

```
{
 "buildsNotDeleted": [
 {
 "id": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-build-project:f8b888d2-5e1e-4032-8645-b115195648EX",
 "statusCode": "BUILD_IN_PROGRESS"
 }
],
 "buildsDeleted": [
 "arn:aws:codebuild:us-west-2:123456789012:build/my-other-demo-build-project:a18bc6ee-e499-4887-b36a-8c90349c7eEX"
]
}
```

### Eliminación de compilaciones (SDK de AWS)

Para obtener información sobre cómo usar AWS CodeBuild con los SDK de AWS, consulte [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

# Uso de informes de pruebas en AWS CodeBuild

Puede crear informes en CodeBuild que contengan detalles sobre las pruebas que se ejecutan durante las compilaciones. Puede crear pruebas como pruebas unitarias, pruebas de configuración y pruebas funcionales.

Se admiten los siguientes formatos de archivo de informe de prueba:

- Pepino JSON (.json)
- XML JUnit (.xml)
- nUnit XML (.xml)
- XML de NUnit3 (.xml)
- XML de TestNG (.xml)
- Visual Studio TRX (.trx)

Cree sus casos de prueba con cualquier marco de pruebas que pueda crear archivos de informes en uno de esos formatos (por ejemplo, el complemento Surefire JUnit, TestNG o Cucumber).

Para crear un informe de pruebas, se añade un nombre de grupo de informes al archivo buildspec de un proyecto de compilación con información sobre los casos de prueba. Cuando ejecuta el proyecto de compilación, se ejecutan los casos de prueba y se crea un informe de pruebas. No es necesario crear un grupo de informes antes de ejecutar las pruebas. Si especifica un nombre de grupo de informes, CodeBuild creará un grupo de informes cuando ejecute los informes. Si desea utilizar un grupo de informes que ya exista, especifique su ARN en el archivo buildspec.

Puede utilizar un informe de pruebas para solucionar un problema durante una ejecución de compilación. Si tiene muchos informes de pruebas de varias compilaciones de un proyecto de compilación, puede utilizar los informes de pruebas para ver tendencias y tasas de pruebas y errores como ayuda para optimizar las compilaciones.

Un informe caduca 30 días después de su creación. No puede ver un informe de pruebas caducado. Si desea conservar los informes de pruebas durante más de 30 días, puede exportar los archivos de datos sin procesar de los resultados de prueba a un bucket de Amazon S3. Los archivos de prueba exportados no caducan. La información sobre el bucket de S3 se especifica al crear el grupo de informes.

## Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

## Temas

- [Crear un informe de pruebas \(p. 306\)](#)
- [Uso de grupos de informes \(p. 306\)](#)
- [Uso de informes \(p. 323\)](#)
- [Uso de permisos de informes de pruebas \(p. 323\)](#)
- [Consultar los informes de pruebas \(p. 326\)](#)
- [Generación de informes de pruebas con marcos de pruebas \(p. 327\)](#)
- [Informes de cobertura de códigos \(p. 331\)](#)



## Crear un informe de pruebas

Para crear un informe de pruebas, se ejecuta un proyecto de compilación que se configura con uno a cinco grupos de informes en su archivo buildspec. Se crea un informe de pruebas durante la ejecución. Contiene los resultados de los casos de prueba que se especifican para los grupos de informes. Se genera un nuevo informe de pruebas para cada compilación posterior que utilice el mismo archivo buildspec.

Para crear un informe de pruebas

1. Cree un proyecto de compilación. Para obtener información, consulte [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#).
2. Configure el archivo buildspec del proyecto con información de informe de pruebas
  - a. Adición de `unreports:` y especifique el ARN de un grupo de informes existente o el nombre de un grupo de informes existente.

Si especifica un ARN, CodeBuild utiliza ese grupo de informes.

Si especifica un nombre, CodeBuild creará un grupo de informes con el nombre del proyecto y el nombre especificado en el formato `<project-name>-<report-group-name>`. Si el grupo de informes con nombre ya existe, CodeBuild utiliza ese grupo de informes.
  - b. En el grupo de informes, especifique la ubicación de los archivos que contienen los resultados de prueba. Si utiliza más de un grupo de informes, especifique las ubicaciones de los archivos de resultados de prueba para cada uno de ellos. Cada vez que se ejecuta el proyecto de compilación, se crea un nuevo informe de pruebas. Para obtener más información, consulte [Especificar archivos de prueba \(p. 313\)](#).
  - c. En la sección `commands` de la secuencia `build` o `post_build`, especifique los comandos que ejecutan los casos de prueba que ha especificado para sus grupos de informes. Para obtener más información, consulte [Especificar comandos de prueba \(p. 313\)](#).

El siguiente es un ejemplo de un `buildspecreports` sección :

```
reports:
 php-reports:
 files:
 - "reports/php/*.xml"
 file-format: "JUNITXML"
 nunit-reports:
 files:
 - "reports/nunit/*.xml"
 file-format: "NUNITXML"
```

3. Ejecute una compilación del proyecto de compilación. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#).
4. Cuando la compilación se haya completado, elija la nueva ejecución de la compilación en Build history (Historial de compilación) en la página del proyecto. Seleccione Reports (Informes) para ver el informe de pruebas. Para obtener más información, consulte [Consultar los informes de pruebas de una compilación \(p. 326\)](#).

## Uso de grupos de informes

Un grupo de informes contiene informes de pruebas y especifica la configuración compartida. Utilice el archivo buildspec para especificar los casos de prueba que se van a ejecutar y los comandos para

ejecutarlos cuando se compila. Para cada grupo de informes configurado en un proyecto de compilación, una ejecución del proyecto de compilación crea un informe de pruebas. Varias ejecuciones de un proyecto de compilación configurado con un grupo de informes crean varios informes de pruebas en ese grupo de informes, cada uno con resultados de los mismos casos de prueba especificados para dicho grupo de informes.

Los casos de prueba se especifican para un grupo de informes en el archivo buildspec de un proyecto de compilación. Puede especificar hasta cinco grupos de informes en un proyecto de compilación. Cuando ejecuta una compilación, se ejecutan todos los casos de prueba. Se crea un nuevo informe de pruebas con los resultados de cada caso de prueba especificado para un grupo de informes. Cada vez que se ejecuta una nueva compilación, se ejecutan los casos de prueba y se crea un nuevo informe de pruebas con los nuevos resultados de prueba.

Los grupos de informes se pueden utilizar en varios proyectos de compilación. Todos los informes de pruebas creados con un grupo de informes comparten la misma configuración, como su opción de exportación y permisos, incluso si los informes de prueba se crean utilizando diferentes proyectos de compilación. Los informes de pruebas creados con un grupo de informes en varios proyectos de compilación pueden contener los resultados de la ejecución de diferentes conjuntos de casos de prueba (un conjunto de casos de prueba para cada proyecto de compilación). Esto se debe a que puede especificar diferentes archivos de casos de prueba para el grupo de informes en el archivo buildspec de cada proyecto. También puede cambiar los archivos de casos de prueba de un grupo de informes en un proyecto de compilación si edita su archivo buildspec. Las ejecuciones de compilación posteriores crean nuevos informes de pruebas que contienen los resultados de los archivos de casos de prueba en el archivo buildspec actualizado.

#### Temas

- [Crear un grupo de informes \(p. 307\)](#)
- [Actualizar un grupo de informes \(p. 311\)](#)
- [Especificar archivos de prueba \(p. 313\)](#)
- [Especificar comandos de prueba \(p. 313\)](#)
- [Nomenclatura de grupos de informes \(p. 314\)](#)
- [Etiquetado de grupos de informes en AWS CodeBuild \(p. 314\)](#)
- [Usar grupos de informes compartidos \(p. 319\)](#)

## Crear un grupo de informes

Puede usar la consola de CodeBuild, la consola de AWS CLI o un archivo buildspec para crear un grupo de informes. Su rol de IAM debe tener los permisos necesarios para crear un grupo de informes. Para obtener más información, consulte [Uso de permisos de informes de pruebas \(p. 323\)](#).

#### Temas

- [Crear un grupo de informes \(buildspec\) \(p. 307\)](#)
- [Crear un grupo de informes \(consola\) \(p. 308\)](#)
- [Crear un grupo de informes \(CLI\) \(p. 309\)](#)
- [Crear un grupo de informes \(AWS CloudFormation\) \(p. 310\)](#)

## Crear un grupo de informes (buildspec)

Un grupo de informes creado con buildspec no exporta archivos de resultados de prueba sin procesar. Puede ver el grupo de informes y especificar las opciones de exportación. Para obtener más información, consulte [Actualizar un grupo de informes \(p. 311\)](#).

Para crear un grupo de informes mediante un archivo buildspec

1. Elija un nombre de grupo de informes que no esté asociado a un grupo de informes de su cuenta de AWS.
2. Configure la sección `reports` del archivo buildspec con este nombre. En este ejemplo, el nombre del grupo de informes es `new-report-group` y los casos de prueba de uso se crean con el marco JUnit:

```
reports:
 new-report-group: #surefire junit reports
 files:
 - '**/*'
 base-directory: 'surefire/target/surefire-reports'
```

Para obtener más información, consulte [Especificar archivos de prueba \(p. 313\)](#) y [Reports syntax in the buildspec file](#).

3. En la sección `commands`, especifique el comando para ejecutar las pruebas. Para obtener más información, consulte [Especificar comandos de prueba \(p. 313\)](#).
4. Ejecute la compilación. Cuando se completa la compilación, se crea un nuevo grupo de informes con un nombre que utiliza el formato `project-name-report-group-name`. Para obtener más información, consulte [Nomenclatura de grupos de informes \(p. 314\)](#).

## Crear un grupo de informes (consola)

Para crear un informe de pruebas

1. Abra el iconoAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Report groups (Grupos de informes).
3. Elija Create report group (Crear grupo de informes).
4. En Report group name (Nombre del grupo de informes), escriba un nombre para el grupo de informes.
5. (Opcional) En Tags (Etiquetas), escriba el nombre y el valor de las etiquetas que desea que usen los servicios compatibles de AWS. Use Add row para añadir una etiqueta. Puede añadir hasta 50 etiquetas.
6. Si desea cargar los datos sin procesar de los resultados del informe de pruebas a un bucket de Amazon S3:
  - a. SelectExportar a Amazon S3.
  - b. En S3 bucket name (Nombre de bucket de S3), escriba el nombre del bucket de S3.
  - c. (Opcional) ParaPropietario del cucharón S3, introduzca la casillaAWSIdentificador de la cuenta propietaria del bucket de S3. Esto permite exportar los datos de los informes a un bucket de Amazon S3 que es propiedad de una cuenta distinta de la cuenta que ejecuta la compilación.
  - d. En Path prefix (Prefijo de ruta), introduzca la ruta en el bucket de S3 donde desea cargar los resultados de prueba.
  - e. Seleccione Compress test result data in a zip file (Comprimir los datos de los resultados de prueba en un archivo zip) para comprimir los archivos de datos de resultados de prueba sin procesar.
  - f. Expanda Additional configuration (Configuración adicional) para mostrar las opciones de cifrado. Elija una de las siguientes opciones:
    - Valor predeterminadoAWSclave administrada depara usar un Clave administrada por AWS para Amazon S3. Para obtener más información, consulte [CMK administradas por el cliente](#) en la Guía del usuario de AWS Key Management Service. Esta es la opción de cifrado predeterminada.

- Elegir una clave personalizada Para usar una clave administrada por el cliente que cree y configure. Para AWS KMS Clave de cifrado de, introduzca el ARN de su clave de cifrado. Su formato es el siguiente: `arn:aws:kms:<region-id>:<aws-account-id>:key/<key-id>`. Para obtener más información, consulte [Creación de claves de KMS](#) en la Guía del usuario de AWS Key Management Service.
- Disable artifact encryption (Desactivar el cifrado de artefactos) para desactivar el cifrado. Puede elegir esta opción si desea compartir los resultados de sus pruebas o publicarlos en un sitio web estático. (Un sitio web dinámico puede ejecutar código para descifrar los resultados de prueba).

Para obtener más información sobre el cifrado de datos en reposo, consulte [Cifrado de datos](#) (p. 352).

#### Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

7. Elija Create report group (Crear grupo de informes).

## Crear un grupo de informes (CLI)

Para crear un grupo de informes

1. Cree un archivo denominado `CreateReportGroup.json`.
2. Según sus requisitos, copie uno de los siguientes fragmentos de código JSON en `CreateReportGroup.json`:
  - Utilice el siguiente código JSON para especificar que su grupo de informes de pruebas exporta archivos de resultados de prueba sin procesar a un bucket de Amazon S3.

```
{
 "name": "<report-name>",
 "type": "TEST",
 "exportConfig": {
 "exportConfigType": "S3",
 "s3Destination": {
 "bucket": "<bucket-name>",
 "bucketOwner": "<bucket-owner>",
 "path": "<path>",
 "packaging": "NONE | ZIP",
 "encryptionDisabled": "false",
 "encryptionKey": "<your-key>"
 },
 "tags": [
 {
 "key": "tag-key",
 "value": "tag-value"
 }
]
 }
}
```

- Reemplazar `<bucket-name>` con el nombre del bucket de Amazon S3 y `<path>` con la ruta de acceso en el bucket de donde desea exportar los archivos.
- Si desea comprimir los archivos exportados, para `packaging`, especifique `ZIP`. De lo contrario, especifique `NONE`.

- `bucketOwner` Es opcional y solo es necesario si el bucket de Amazon S3 es propiedad de una cuenta distinta de la cuenta que ejecuta la compilación.
- Se utiliza `encryptionDisabled` para especificar si se deben cifrar los archivos exportados. Si cifra los archivos exportados, introduzca la clave administrada por el cliente. Para obtener más información, consulte [Actualizar un grupo de informes \(p. 311\)](#).
- Utilice el siguiente código JSON para especificar que el informe de pruebas no exporta archivos de prueba sin procesar:

```
{
 "name": "<report-name>",
 "type": "TEST",
 "exportConfig": {
 "exportConfigType": "NO_EXPORT"
 }
}
```

#### Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

3. Ejecute el comando siguiente:

```
aws codebuild create-report-group --cli-input-json file://CreateReportGroupInput.json
```

## Crear un grupo de informes (AWS CloudFormation)

Para crear un informe de pruebas con la plantilla de AWS CloudFormation

Puede utilizar un archivo de plantilla de AWS CloudFormation para crear y aprovisionar un grupo de informes. Si desea obtener más información, consulte la [Guía del usuario de AWS CloudFormation](#).

La siguiente plantilla YAML de AWS CloudFormation crea un grupo de informes que no exporta archivos de resultados de prueba sin procesar.

```
Resources:
 CodeBuildReportGroup:
 Type: AWS::CodeBuild::ReportGroup
 Properties:
 Name: my-report-group-name
 Type: TEST
 ExportConfig:
 ExportConfigType: NO_EXPORT
```

Los siguientes ejemplos de AWS CloudFormationLa plantilla YAML de crea un grupo de informes que exporta archivos de resultados de prueba sin procesar a un bucket de Amazon S3.

```
Resources:
 CodeBuildReportGroup:
 Type: AWS::CodeBuild::ReportGroup
 Properties:
 Name: my-report-group-name
 Type: TEST
 ExportConfig:
 ExportConfigType: S3
 S3Destination:
```

```
Bucket: my-s3-bucket-name
Path: path-to-folder-for-exported-files
Packaging: ZIP
EncryptionKey: my-KMS-encryption-key
EncryptionDisabled: false
```

#### Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

## Actualizar un grupo de informes

Cuando actualice un grupo de informes, puede especificar información acerca de si se deben exportar los datos de resultados de prueba sin procesar a archivos en un bucket de Amazon S3. Si decide exportar a un bucket de S3, puede especificar lo siguiente para su grupo de informes:

- Si los archivos de resultados de prueba sin procesar están comprimidos en un archivo ZIP.
- Si los archivos de resultados de prueba sin procesar están cifrados. Puede especificar el cifrado con una de las siguientes opciones:
  - Un registro Clave administrada por AWS para Amazon S3.
  - Clave administrada por el cliente que cree y configure.

Para obtener más información, consulte [Cifrado de datos \(p. 352\)](#).

Si utiliza la AWS CLI para actualizar un grupo de informes, también puede actualizar o agregar etiquetas. Para obtener más información, consulte [Etiquetado de grupos de informes en AWS CodeBuild \(p. 314\)](#).

#### Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

#### Temas

- [Actualizar un grupo de informes \(consola\) \(p. 311\)](#)
- [Actualizar un grupo de informes \(CLI\) \(p. 312\)](#)

## Actualizar un grupo de informes (consola)

Para actualizar un grupo de informes

1. Abra el iconoAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Report groups (Grupos de informes).
3. Elija el grupo de informes que desea actualizar.
4. Elija Edit (Edición).
5. Seleccione o desactiveCopia de seguridad en Amazon S3. Si ha seleccionado esta opción, especifique la configuración de exportación:
  - a. En S3 bucket name (Nombre de bucket de S3), escriba el nombre del bucket de S3.
  - b. En Path prefix (Prefijo de ruta), introduzca la ruta en el bucket de S3 donde desea cargar los resultados de prueba.
  - c. Seleccione Compress test result data in a zip file (Comprimir los datos de los resultados de prueba en un archivo zip) para comprimir los archivos de datos de resultados de prueba sin procesar.

- d. Expanda Additional configuration (Configuración adicional) para mostrar las opciones de cifrado. Elija una de las siguientes opciones:
- Valor predeterminado AWS clave administrada para usar una clave administrada por AWS para Amazon S3. Para obtener más información, consulte [CMK administradas por el cliente](#) en la Guía del usuario de AWS Key Management Service. Esta es la opción de cifrado predeterminada.
  - Elegir una clave personalizada Para usar una clave administrada por el cliente que cree y configure. Para AWS KMS clave de cifrado de, introduzca el ARN de su clave de cifrado. Su formato es el siguiente: `arn:aws:kms:<region-id>:<aws-account-id>:key/<key-id>`. Para obtener más información, consulte [Creación de claves de KMS](#) en la Guía del usuario de AWS Key Management Service.
  - Disable artifact encryption (Desactivar el cifrado de artefactos) para desactivar el cifrado. Puede elegir esta opción si desea compartir los resultados de sus pruebas o publicarlos en un sitio web estático. (Un sitio web dinámico puede ejecutar código para descifrar los resultados de prueba).

## Actualizar un grupo de informes (CLI)

Para actualizar un grupo de informes

1. Cree un archivo denominado `UpdateReportGroupInput.json`.
2. Copie lo siguiente en `UpdateReportGroupInput.json`:

```
{
 "arn": "",
 "exportConfig": {
 "exportConfigType": "S3",
 "s3Destination": {
 "bucket": "bucket-name",
 "path": "path",
 "packaging": "NONE | ZIP",
 "encryptionDisabled": "false",
 "encryptionKey": "your-key"
 }
 },
 "tags": [
 {
 "key": "tag-key",
 "value": "tag-value"
 }
]
}
```

3. Escriba el ARN de su grupo de informes en la línea de `arn` (por ejemplo, `"arn": "arn:aws:codebuild:region:123456789012:report-group/report-group-1"`).
4. Aplique a `UpdateReportGroupInput.json` las actualizaciones que desee implementar al grupo de informes.
  - Si desea actualizar el grupo de informes para exportar archivos de resultados de prueba sin procesar a un bucket de S3, actualice la sección `exportConfig`. Reemplace `bucket-name` por el nombre del bucket de S3 y `path` por la ruta de acceso del bucket de S3 a la que desea exportar los archivos. Si desea comprimir los archivos exportados, para `packaging`, especifique `ZIP`. De lo contrario, especifique `NONE`. Se utiliza `encryptionDisabled` para especificar si se deben cifrar los archivos exportados. Si cifra los archivos exportados, introduzca la clave administrada por el cliente.

- Si desea actualizar su grupo de informes para que no exporte archivos de resultados de prueba sin procesar a un bucket de S3, actualice la sección `exportConfig` con el siguiente JSON:

```
{
 "exportConfig": {
 "exportConfigType": "NO_EXPORT"
 }
}
```

- Si desea actualizar las etiquetas del grupo de informes, actualice la sección `tags`. Puede cambiar, agregar o quitar etiquetas. Si desea quitar todas las etiquetas, actualícelo con el siguiente JSON:

```
"tags": []
```

5. Ejecute el comando siguiente:

```
aws codebuild update-report-group \
--cli-input-json file://UpdateReportGroupInput.json
```

## Especificar archivos de prueba

Los archivos de resultados de prueba y su ubicación para cada grupo de informes se especifican en la sección `reports` del archivo `buildspec` del proyecto de compilación. Para obtener más información, consulte [Reports syntax in the buildspec file](#).

A continuación se muestra una sección `reports` de ejemplo que especifica dos grupos de informes para un proyecto de compilación. Uno se especifica con su ARN y el otro con un nombre. La sección `files` especifica los archivos que contienen los resultados del caso de prueba. La sección `base-directory` opcional especifica el directorio donde se encuentran los archivos de casos de prueba. La opción `discard-paths` se especifica si se descartan las rutas de los archivos de resultados de prueba cargados en un bucket de Amazon S3.

```
reports:
 arn:aws:codebuild:your-region:your-aws-account-id:report-group/report-group-name-1:
 #surefire junit reports
 files:
 - '**/*'
 base-directory: 'surefire/target/surefire-reports'
 discard-paths: false

sampleReportGroup: #Cucumber reports from json plugin
 files:
 - 'cucumber-json/target/cucumber-json-report.json'
 file-format: CUCUMBERJSON #Type of the report, defaults to JUNITXML
```

## Especificar comandos de prueba

Los comandos que ejecutan los casos de prueba se especifican en la sección `commands` del archivo `buildspec`. Estos comandos ejecutan los casos de prueba especificados para sus grupos de informes en la sección `reports` de su archivo `buildspec`. A continuación se muestra una sección `commands` de ejemplo que incluye comandos para ejecutar las pruebas en archivos de prueba:

```
commands:
 - echo Running tests for surefire junit
 - mvn test -f surefire/pom.xml -fn
 - echo
```



```
- echo Running tests for cucumber with json plugin
- mvn test -Dcucumber.options="--plugin json:target/cucumber-json-report.json" -f
cucumber-json/pom.xml -fn
```

## Nomenclatura de grupos de informes

Cuando utilice la AWS CLI o la consola de AWS CodeBuild para crear un grupo de informes, especifique un nombre para el grupo de informes. Si utiliza el archivo de especificación de compilación para crear un nuevo grupo de informes, se le asigna un nombre con el formato *project-name-report-group-name-specified-in-buildspec*. Todos los informes creados mediante la ejecución de compilaciones de ese proyecto de compilación pertenecen al nuevo grupo de informes que tiene el nuevo nombre.

Si no desea que CodeBuild cree un nuevo grupo de informes, especifique el ARN del grupo de informes en un archivo buildspec del proyecto de compilación. Puede especificar el ARN de un grupo de informes en varios proyectos de compilación. Después de que se ejecute cada proyecto de compilación, el grupo de informes contiene informes de pruebas creados por cada proyecto de compilación.

Por ejemplo, si crea un grupo de informes con el nombre `my-report-group` y, a continuación, utiliza su nombre en dos proyectos de compilación diferentes denominados `my-project-1` y `my-project-2`, y crea una compilación de ambos proyectos, se crean dos nuevos grupos de informes. El resultado son tres grupos de informes con los siguientes nombres:

- `my-report-group`: no tiene informes de pruebas.
- `my-project-1-my-report-group`: contiene informes con resultados de pruebas ejecutadas por el proyecto de compilación denominado `my-project-1`.
- `my-project-2-my-report-group`: contiene informes con resultados de pruebas ejecutadas por el proyecto de compilación denominado `my-project-2`.

Si utiliza el ARN del grupo de informes denominado `my-report-group` en ambos proyectos y, a continuación, ejecuta compilaciones de cada proyecto, seguirá teniendo un grupo de informes (`my-report-group`). Ese grupo de informes contiene informes de pruebas con resultados de pruebas ejecutadas por ambos proyectos de compilación.

Si elige un nombre de grupo de informes que no pertenece a un grupo de informes de su cuenta de AWS y, a continuación, utiliza ese nombre para un grupo de informes en un archivo buildspec y ejecuta una compilación de su proyecto de compilación, se creará un nuevo grupo de informes. El formato del nombre del nuevo grupo de informes es *project-name-new-group-name*. Por ejemplo, si no hay un grupo de informes en la cuenta de AWS con el nombre `new-report-group` y lo especifica en un proyecto de compilación denominado `test-project`, una ejecución de compilación crea un nuevo grupo de informes con el nombre `test-project-new-report-group`.

## Etiquetado de grupos de informes en AWS CodeBuild

Una etiqueta es un atributo personalizado que usted o AWS asignan a un recurso de AWS. Cada etiqueta de AWS tiene dos partes:

- Una clave de etiqueta (por ejemplo, `CostCenter`, `Environment`, `Project` o `Secret`). Las claves de etiqueta distinguen entre mayúsculas y minúsculas.
- Un campo opcional que se denomina valor de etiqueta (por ejemplo, `111122223333` o `Production` o el nombre de un equipo). Omitir el valor de etiqueta es lo mismo que usar una cadena vacía. Al igual que las claves de etiqueta, los valores de etiqueta distinguen entre mayúsculas y minúsculas.

En conjunto, se conocen como pares clave-valor. Para conocer los límites del número de etiquetas que puede tener en un grupo de informes y las restricciones sobre las claves y valores de las etiquetas, consulte [Tags](#) (p. 446).

Las etiquetas le ayudan a identificar y organizar los recursos de AWS. Muchos servicios de AWS admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, puede asignar la misma etiqueta a un grupo de informes de CodeBuild que asigne a un bucket de Amazon S3. Para obtener más información sobre el uso de etiquetas, consulte el documento técnico [Prácticas recomendadas de etiquetado](#).

En CodeBuild, los recursos principales son el grupo de informes y el proyecto. Puede usar la consola de CodeBuild, la consola de AWS CLI, API de CodeBuild, o AWS Para agregar, administrar y quitar etiquetas de un grupo de informes. Además de utilizar etiquetas para identificar, organizar y realizar el seguimiento del grupo de informes, puede utilizarlas en las políticas de IAM para ayudar a controlar quién puede consultar el grupo de informes o interactuar con él. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild](#) (p. 381).

#### Temas

- [Agregar una etiqueta a un grupo de informes](#) (p. 315)
- [Consultar etiquetas de un grupo de informes](#) (p. 316)
- [Editar etiquetas de un grupo de informes](#) (p. 317)
- [Quitar una etiqueta de un grupo de informes](#) (p. 318)

## Agregar una etiqueta a un grupo de informes

Agregar etiquetas a un grupo de informes puede ayudarle a identificar y organizar sus recursos de AWS y a administrar el acceso a ellos. En primer lugar, agregue una o varias etiquetas (pares de clave-valor) a un grupo de informes. Tenga en cuenta que hay límites en el número de etiquetas que puede tener en un grupo de informes. Existen restricciones sobre los caracteres que se pueden utilizar en los campos de clave y valor. Para obtener más información, consulte [Tags](#) (p. 446). Cuando tenga las etiquetas, puede crear políticas de IAM para administrar el acceso al grupo de informes en función de estas etiquetas. Puede usar la consola de CodeBuild o la consola de AWS CLI Para agregar etiquetas a un grupo de informes.

#### Important

Agregar etiquetas a un grupo de informes puede afectar al acceso a ese grupo de informes. Antes de agregar una etiqueta a un grupo de informes, asegúrese de revisar cualquier política de IAM que pueda utilizar etiquetas para controlar el acceso a recursos como los grupos de informes. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild](#) (p. 381).

Para obtener más información sobre cómo agregar etiquetas a un grupo de informes al crearlo, consulte [Crear un grupo de informes \(consola\)](#) (p. 308).

#### Temas

- [Agregar una etiqueta a un grupo de informes \(consola\)](#) (p. 315)
- [Agregar una etiqueta a un grupo de informes \(AWS CLI\)](#) (p. 316)

## Agregar una etiqueta a un grupo de informes (consola)

Puede usar la consola de CodeBuild de para agregar una o varias etiquetas a un grupo de informes de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Grupos de informes, elija el nombre del grupo de informes al que desea agregar etiquetas.
3. En el panel de navegación, seleccione Settings.

4. Si las etiquetas no se han agregado al grupo de informes, elija Agregar etiqueta. También puede elegir Editar y, a continuación, elegir Agregar etiqueta.
5. En Key (Clave), escriba un nombre para la etiqueta. Puede añadir un valor opcional para la etiqueta en Value (Valor).
6. (Opcional) Para añadir otra etiqueta, vuelva a elegir Add tag (Añadir etiqueta).
7. Cuando haya terminado de añadir etiquetas, seleccione Submit (Enviar).

## Agregar una etiqueta a un grupo de informes (AWS CLI)

Para agregar una etiqueta a un grupo de informes al crearlo, consulte [Crear un grupo de informes \(CLI\)](#) (p. 309). En `CreateReportGroup.json`, agregue sus etiquetas.

Para agregar etiquetas a un grupo de informes existente, consulte [Actualizar un grupo de informes \(CLI\)](#) (p. 312) y agregue las etiquetas en `UpdateReportGroupInput.json`.

En estos pasos, se presupone que ya ha instalado una versión reciente de la AWS CLI o que la ha actualizado a la versión actual. Para obtener más información, consulte [Instalación de la AWS Command Line Interface](#).

## Consultar etiquetas de un grupo de informes

Las etiquetas pueden ayudarle a identificar y organizar sus recursos de AWS y administrar el acceso a ellos. Para obtener más información sobre el uso de etiquetas, consulte el documento técnico [Prácticas recomendadas de etiquetado](#). Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Deny or allow actions on report groups based on resource tags](#).

### Consultar etiquetas de un grupo de informes (consola)

Puede usar la consola de CodeBuild de para consultar las etiquetas asociadas con un grupo de informes de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Grupos de informes, elija el nombre del grupo de informes en el que desea consultar las etiquetas.
3. En el panel de navegación, seleccione Settings.

### Consultar etiquetas de un grupo de informes (AWS CLI)

Siga estos pasos para utilizar la AWS CLI para consultar las etiquetas de AWS de un grupo de informes. Si no se han agregado etiquetas, la lista de etiquetas devueltas está vacía.

1. Utilice la consola o la AWS CLI para localizar el ARN del grupo de informes. Anótelo.

AWS CLI

Ejecute el siguiente comando de .

```
aws list-report-groups
```

Este comando devuelve información con formato JSON similar a la siguiente:

```
{
 "reportGroups": [
 "arn:aws:codebuild:region:123456789012:report-group/report-group-1",
 "arn:aws:codebuild:region:123456789012:report-group/report-group-2",
```

```
 "arn:aws:codebuild:region:123456789012:report-group/report-group-3"
]
}
```

El ARN del grupo de informes termina con su nombre, que puede utilizar para identificar el ARN del grupo de informes.

#### Console

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
  2. En Grupos de informes, elija el nombre del grupo de informes con las etiquetas que desea consultar.
  3. En Configuración localice el ARN del grupo de informes.
2. Ejecute el siguiente comando de . Utilice el ARN del que tomó nota para el parámetro `--report-group-arns`.

```
aws codebuild batch-get-report-groups --report-group-arns
arn:aws:codebuild:region:123456789012:report-group/report-group-name
```

Si tiene éxito, este comando devuelve información con formato JSON que contiene una sección `tags` similar a la siguiente:

```
{
 ...
 "tags": {
 "Status": "Secret",
 "Project": "TestBuild"
 }
 ...
}
```

## Editar etiquetas de un grupo de informes

Puede cambiar el valor de una etiqueta asociada con un grupo de informes. También puede cambiar el nombre de la clave, lo que equivale a eliminar la etiqueta actual y añadir otra distinta con el nuevo nombre y el mismo valor que la otra clave. Tenga en cuenta que hay restricciones en los caracteres que puede utilizar en los campos de clave y valor. Para obtener más información, consulte [Tags \(p. 446\)](#).

#### Important

La edición de etiquetas de un grupo de informes puede afectar al acceso a ese grupo de informes. Antes de editar el nombre (clave) o valor de una etiqueta de un grupo de informes, asegúrese de revisar cualquier política de IAM que pueda utilizar la clave o el valor de una etiqueta para controlar el acceso a recursos como los grupos de informes. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Deny or allow actions on report groups based on resource tags](#).

### Editar una etiqueta de un grupo de informes (consola)

Puede usar la consola de CodeBuild de para editar las etiquetas asociadas con un grupo de informes de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Grupos de informes, elija el nombre del grupo de informes en el que desea editar las etiquetas.
3. En el panel de navegación, seleccione Settings.
4. Elija Edit (Edición).

5. Aplique alguna de las siguientes acciones:
  - Para cambiar la etiqueta, escriba un nuevo nombre en Key (Clave). Cambiar el nombre de la etiqueta equivale a quitar una etiqueta y añadir otra nueva con el nuevo nombre de clave.
  - Para cambiar el valor de una etiqueta, escriba un nuevo valor. Si desea cambiar el valor por nada, elimine el valor actual y deje el campo en blanco.
6. Cuando haya terminado de editar las etiquetas, seleccione Submit (Enviar).

## Editar etiquetas de un grupo de informes (AWS CLI)

Para agregar, cambiar o eliminar etiquetas de un grupo de informes, consulte [Actualizar un grupo de informes \(CLI\)](#) (p. 312). Actualice las etiquetas en `UpdateReportGroupInput.json`.

## Quitar una etiqueta de un grupo de informes

Puede quitar una o varias etiquetas asociadas con un grupo de informes. La eliminación de una etiqueta no la elimina de otros recursos de AWS que están asociados con esa etiqueta.

### Important

Quitar etiquetas de un grupo de informes puede afectar al acceso a ese grupo de informes. Antes de quitar una etiqueta de un grupo de informes, asegúrese de revisar cualquier política de IAM que pueda utilizar la clave o el valor de una etiqueta para controlar el acceso a recursos como los grupos de informes. Para ver ejemplos de políticas de acceso basadas en etiquetas, consulte [Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild](#) (p. 381).

## Quitar una etiqueta de un grupo de informes (consola)

Puede usar la consola de CodeBuild de para eliminar la asociación entre una etiqueta y un grupo de informes de CodeBuild.

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En Grupos de informes, elija el nombre del grupo de informes en el que desea quitar etiquetas.
3. En el panel de navegación, seleccione Settings.
4. Elija Edit (Edición).
5. Busque la etiqueta que desea eliminar y, a continuación, seleccione Remove tag (Eliminar etiqueta).
6. Cuando haya terminado de eliminar las etiquetas, seleccione Submit (Enviar).

## Quitar una etiqueta de un grupo de informes (AWS CLI)

Siga estos pasos para utilizar laAWS CLIPara eliminar una etiqueta de un grupo de informes CodeBuild. Al quitar una etiqueta no la elimina, sino que simplemente elimina la asociación entre la etiqueta y el grupo de informes.

### Note

Si elimina un grupo de informes de CodeBuild, todas las asociaciones de etiquetas se quitarán del grupo de informes eliminado. No es necesario quitar las etiquetas antes de eliminar un grupo de informes.

Para eliminar una o varias etiquetas de un grupo de informes, consulte [Editar etiquetas de un grupo de informes \(AWS CLI\)](#) (p. 318). Actualice la sección `tags` de los datos con formato JSON con una lista actualizada de etiquetas que no contenga las que desea eliminar. Si desea eliminar todas las etiquetas, actualice la sección `tags` para:

```
"tags: []"
```

## Usar grupos de informes compartidos

El uso compartido de grupos de informes permite que varias cuentas o usuarios de AWS vean un grupo de informes, sus informes no caducados y los resultados de las pruebas de sus informes. En este modelo, la cuenta que posee el grupo de informes (propietario) comparte un grupo de informes con otras cuentas (consumidores). Un consumidor no puede editar un grupo de informes. Un informe caduca 30 días después de su creación.

### Contenido

- [Requisitos previos para compartir grupos de informes \(p. 319\)](#)
- [Requisitos previos para acceder a grupos de informes compartidos con usted \(p. 319\)](#)
- [Servicios de relacionados \(p. 319\)](#)
- [Uso compartido de un grupo de informes \(p. 320\)](#)
- [Dejar de compartir un grupo de informes compartidos \(p. 321\)](#)
- [Identificación de un grupo de informes compartido \(p. 321\)](#)
- [Permisos de un grupo de informes compartido \(p. 322\)](#)

## Requisitos previos para compartir grupos de informes

Para compartir un grupo de informes, su cuenta de AWS debe ser la propietaria. No puede compartir un grupo de informes que se haya compartido con usted.

## Requisitos previos para acceder a grupos de informes compartidos con usted

Para tener acceso a un grupo de informes compartido, el rol de IAM de un consumidor requiere el permiso `BatchGetReportGroups`. Puede adjuntar la siguiente política a su rol de IAM:

```
{
 "Effect": "Allow",
 "Resource": [
 "*"
],
 "Action": [
 "codebuild:BatchGetReportGroups"
]
}
```

Para obtener más información, consulte [Uso de políticas basadas en identidades para AWS CodeBuild \(p. 358\)](#).

## Servicios de relacionados

El uso compartido de grupos de informes se integra con AWS Resource Access Manager (AWS RAM), un servicio que le permite compartir sus recursos de AWS con cualquier cuenta de AWS o a través de AWS Organizations. Con AWS RAM, comparte los recursos que posee mediante la creación de un recurso compartido que especifica los recursos y los consumidores con los que los comparte. Los consumidores pueden ser cuentas de AWS individuales, unidades organizativas en AWS Organizations o toda una organización de AWS Organizations.

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

## Uso compartido de un grupo de informes

Cuando comparte un grupo de informes, al consumidor se le concede acceso de solo lectura al grupo de informes y a sus informes. El consumidor puede usar la AWS CLI para ver el grupo de informes, sus informes y los resultados del caso de prueba para cada informe. El consumidor no puede:

- Ver un grupo de informes compartido o sus informes en la consola de CodeBuild.
- Editar un grupo de informes compartido.
- Utilice el ARN del grupo de informes compartido en un proyecto para ejecutar un informe. Se produce un error en una compilación de proyecto que especifica un grupo de informes compartido.

Puede usar la consola de CodeBuild de para agregar un grupo de informes a un recurso compartido existente. Si desea añadir el grupo de informes a un nuevo recurso compartido, primero debe crearlo en la [consola de AWS RAM](#).

Para compartir un grupo de informes con unidades organizativas o con toda la organización, debe permitir el uso compartido con AWS Organizations. Para obtener más información, consulte [Habilitar el uso compartido con AWS Organizations](#) en la Guía del usuario de AWS RAM.

Puede usar la consola de CodeBuild, AWS RAM Consola de, o AWS CLI Para compartir grupos de informes que posee.

Para compartir un grupo de informes que posee (consola de CodeBuild)

1. Abra el icono AWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Report groups (Grupos de informes).
3. Elija el proyecto que desea compartir y, a continuación, elija Share (Compartir). Para obtener más información, consulte [Crear un recurso compartido](#) en la Guía del usuario de AWS RAM.

Para compartir grupos de informes que posee (consola de AWS RAM)

Consulte [Crear un recurso compartido](#) en la AWS RAM Guía del usuario de.

Para compartir grupos de informes que posee (comando AWS RAM)

Utilice el comando [create-resource-share](#).

Para compartir un grupo de informes que posee (comando CodeBuild)

Utilice el comando [put-resource-policy](#):

1. Cree un archivo llamado `policy.json` y copie lo siguiente en él.

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Principal": {
 "AWS": "consumer-aws-account-id-or-user"
 },
 "Action": [
 "codebuild:BatchGetReportGroups",
 "codebuild:BatchGetReports",
 "codebuild:ListReportsForReportGroup",
 "codebuild:DescribeTestCases"
],
 "Resource": "arn-of-report-group-to-share"
 }]
}
```

2. Actualice `policy.json` con el ARN del grupo de informes y los identificadores para compartirlo. En el siguiente ejemplo se concede acceso de solo lectura al grupo de informes con el ARN `arn:aws:codebuild:us-west-2:123456789012:report-group/my-report-group` a Alice y al usuario `root` para el AWS cuenta identificada por `123456789012`.

```
{
 "Version": "2012-10-17",
 "Statement": [{
 "Effect": "Allow",
 "Principal": {
 "AWS": [
 "arn:aws:iam::123456789012:user/Alice",
 "123456789012"
]
 },
 "Action": [
 "codebuild:BatchGetReportGroups",
 "codebuild:BatchGetReports",
 "codebuild:ListReportsForReportGroup",
 "codebuild:DescribeTestCases"
],
 "Resource": "arn:aws:codebuild:us-west-2:123456789012:report-group/my-report-group"
 }]
}
```

3. Ejecute el siguiente comando de .

```
aws codebuild put-resource-policy --resource-arn report-group-arn --policy file://
policy.json
```

## Dejar de compartir un grupo de informes compartidos

Solo el propietario puede obtener acceso a un grupo de informes no compartido, incluidos sus informes y los resultados de los casos de prueba. Si deja de compartir un grupo de informes, cualquier cuenta de AWS o usuario con el que lo haya compartido anteriormente no podrá obtener acceso al grupo de informes, a sus informes ni a los resultados de los casos de prueba de los informes.

Para dejar de compartir un grupo de informes compartido que posee, debe quitarlo del recurso compartido. Puede usar la consola de AWS RAM o AWS CLI para realizar esta operación.

Para dejar de compartir un grupo de informes compartido que posee (consola de AWS RAM)

Consulte [Actualización de un recurso compartido](#) en la Guía del usuario de AWS RAM.

Para dejar de compartir un grupo de informes compartido que posee (comando AWS RAM)

Utilice el comando `disassociate-resource-share`.

Para dejar de compartir el grupo de informes que posee (comando CodeBuild)

Ejecute el comando `delete-resource-policy` y especifique el ARN del grupo de informes que desea dejar de compartir:

```
aws codebuild delete-resource-policy --resource-arn report-group-arn
```

## Identificación de un grupo de informes compartido

Los propietarios y los consumidores pueden utilizar la AWS CLI para identificar grupos de informes compartidos.



Para identificar y obtener información acerca de un grupo de informes compartido y sus informes, utilice los siguientes comandos:

- Para ver los ARN de los grupos de informes compartidos con usted, ejecute `list-shared-report-groups`:

```
aws codebuild list-shared-report-groups
```

- Para ver los ARN de los informes de un grupo de informes, ejecute `list-reports-for-report-group` con el ARN del grupo de informes:

```
aws codebuild list-reports-for-report-group --report-group-arn report-group-arn
```

- Para ver la información sobre los casos de prueba en un informe, ejecute `describe-test-cases` con el ARN de informe:

```
aws codebuild describe-test-cases --report-arn report-arn
```

El resultado es similar al siguiente:

```
{
 "testCases": [
 {
 "status": "FAILED",
 "name": "Test case 1",
 "expired": 1575916770.0,
 "reportArn": "report-arn",
 "prefix": "Cucumber tests for agent",
 "message": "A test message",
 "durationInNanoSeconds": 1540540,
 "testRawDataPath": "path-to-output-report-files"
 },
 {
 "status": "SUCCEEDED",
 "name": "Test case 2",
 "expired": 1575916770.0,
 "reportArn": "report-arn",
 "prefix": "Cucumber tests for agent",
 "message": "A test message",
 "durationInNanoSeconds": 1540540,
 "testRawDataPath": "path-to-output-report-files"
 }
]
}
```

## Permisos de un grupo de informes compartido

### Permisos de los propietarios

Un propietario de un grupo de informes puede editarlo y especificarlo en un proyecto para ejecutar informes.

### Permisos de los consumidores

Un consumidor de grupo de informes puede ver un grupo de informes, sus informes y los resultados de los casos de prueba de sus informes. Un consumidor no puede editar un grupo de informes ni sus informes y no puede usarlo para crear informes.

## Uso de informes

Un informe contiene los resultados de los casos de prueba especificados para un grupo de informes. Se crea un informe de pruebas durante la ejecución de un proyecto de compilación. Se especifican un grupo de informes, los archivos de casos de prueba y los comandos para ejecutar los casos de prueba en su archivo buildspec. Cada vez que se ejecutan los casos de prueba, se crea un nuevo informe de pruebas en el grupo de informes.

Un informe de pruebas caduca 30 días después de su creación. No puede ver un informe de pruebas caducado, pero puede exportar los resultados de pruebas a archivos de resultados de pruebas sin procesar a un bucket de S3. Los archivos de pruebas sin procesar exportados no caducan. Para obtener más información, consulte [Actualizar un grupo de informes \(p. 311\)](#).

El estado de un informe de pruebas puede ser uno de los siguientes:

- **GENERATING**: la ejecución de los casos de prueba todavía está en curso.
- **DELETING**: se está eliminando el informe de pruebas. Cuando se elimina un informe de pruebas, también se eliminan sus casos de prueba. Los archivos de datos de resultados de pruebas sin procesar exportados a un bucket de S3 no se eliminan.
- **INCOMPLETE**: el informe de pruebas no se ha completado. Este estado se puede devolver por uno de los siguientes motivos:
  - Un problema con la configuración del grupo de informes que especifica los casos de prueba de este informe. Por ejemplo, la ruta de los casos de prueba en el grupo de informes del archivo buildspec puede ser incorrecta.
  - El usuario de IAM que ha ejecutado la compilación no tiene permisos para ejecutar pruebas. Para obtener más información, consulte [Uso de permisos de informes de pruebas \(p. 323\)](#).
  - La compilación no se ha completado debido a un error que no está relacionado con las pruebas.
- **SUCCEEDED**: todos los casos de prueba se realizaron correctamente.
- **FAILED**: algunos de los casos de prueba no se realizaron correctamente.

Cada caso de prueba devuelve un estado. El estado de un caso de prueba puede ser uno de los siguientes:

- **SUCCEEDED**: se ha superado el caso de prueba.
- **FAILED**: se ha producido un error en el caso de prueba.
- **ERROR**: el caso de prueba ha producido un error inesperado.
- **SKIPPED**: el caso de prueba no se ha ejecutado.
- **UNKNOWN**: el caso de prueba ha devuelto un estado distinto de **SUCCEEDED**, **FAILED**, **ERROR** o **SKIPPED**.

Un informe de pruebas puede tener un máximo de 500 resultados de casos de prueba. Si se ejecutan más de 500 casos de prueba, CodeBuild prioriza las pruebas con el estado **FAILED** y trunca los resultados de casos de prueba.

## Uso de permisos de informes de pruebas

En este tema se describe información importante acerca de los permisos relacionados con los informes de pruebas.

### Temas

- [Crear un rol para los informes de pruebas \(p. 324\)](#)
- [Permisos para las operaciones de informes de pruebas \(p. 325\)](#)

- [Ejemplos de permisos de informes de pruebas \(p. 325\)](#)

## Crear un rol para los informes de pruebas

Para ejecutar un informe de pruebas y actualizar un proyecto para incluir informes de pruebas, el rol de IAM requiere los siguientes permisos. Estos permisos se incluyen en las políticas administradas por AWS predefinidas. Si desea añadir un informe de pruebas a un proyecto de compilación existente, debe añadir estos permisos.

- `CreateReportGroup`
- `CreateReport`
- `UpdateReport`
- `BatchPutTestCases`

Para ejecutar un informe de cobertura de código, su rol de IAM también debe incluir el `BatchPutCodeCoverages` permiso.

### Note

`BatchPutTestCases`, `CreateReport`, `UpdateReport`, y `BatchPutCodeCoverages` No son permisos públicos. No puede llamar a un comando de la AWS CLI o a método del SDK correspondiente para estos permisos.

Para asegurarse de que tiene estos permisos, puede asociar la siguiente política a su rol de IAM:

```
{
 "Effect": "Allow",
 "Resource": [
 "*"
],
 "Action": [
 "codebuild:CreateReportGroup",
 "codebuild:CreateReport",
 "codebuild:UpdateReport",
 "codebuild:BatchPutTestCases",
 "codebuild:BatchPutCodeCoverages"
]
}
```

Le recomendamos que restrinja esta política únicamente a los grupos de informes que debe utilizar. Lo siguiente restringe los permisos solo a los grupos de informes con los dos ARN en la política:

```
{
 "Effect": "Allow",
 "Resource": [
 "arn:aws:codebuild:your-region:your-aws-account-id:report-group/report-group-name-1",
 "arn:aws:codebuild:your-region:your-aws-account-id:report-group/report-group-name-2"
],
 "Action": [
 "codebuild:CreateReportGroup",
 "codebuild:CreateReport",
 "codebuild:UpdateReport",
 "codebuild:BatchPutTestCases",
 "codebuild:BatchPutCodeCoverages"
]
}
```

Lo siguiente restringe los permisos solo a los grupos de informes creados por la ejecución de compilaciones de un proyecto llamado `my-project`:

```
{
 "Effect": "Allow",
 "Resource": [
 "arn:aws:codebuild:your-region:your-aws-account-id:report-group/my-project-*"
],
 "Action": [
 "codebuild:CreateReportGroup",
 "codebuild:CreateReport",
 "codebuild:UpdateReport",
 "codebuild:BatchPutTestCases",
 "codebuild:BatchPutCodeCoverages"
]
}
```

#### Note

El rol de servicio de CodeBuild especificado en el proyecto se utiliza para los permisos para cargar en el bucket de S3.

## Permisos para las operaciones de informes de pruebas

Puede especificar permisos para las siguientes operaciones de la API de CodeBuild de informes de pruebas:

- `BatchGetReportGroups`
- `BatchGetReports`
- `CreateReportGroup`
- `DeleteReportGroup`
- `DeleteReport`
- `DescribeTestCases`
- `ListReportGroups`
- `ListReports`
- `ListReportsForReportGroup`
- `UpdateReportGroup`

Para obtener más información, consulte [Referencia de permisos de AWS CodeBuild \(p. 376\)](#).

## Ejemplos de permisos de informes de pruebas

Para obtener información acerca de las directivas de ejemplo relacionadas con los informes de pruebas, consulte lo siguiente:

- [Permitir que un usuario pueda cambiar un grupo de informes \(p. 372\)](#)
- [Permitir que un usuario pueda crear un grupo de informes \(p. 370\)](#)
- [Permitir que un usuario pueda eliminar un informe \(p. 371\)](#)
- [Permitir que un usuario pueda eliminar un grupo de informes \(p. 370\)](#)
- [Permitir que un usuario obtenga información sobre los grupos de informes \(p. 369\)](#)
- [Permitir que un usuario obtenga información sobre los informes \(p. 370\)](#)

- Permitir que a un usuario pueda obtener una lista de grupos de informes (p. 373)
- Permitir que un usuario pueda obtener una lista de informes (p. 373)
- Permitir que un usuario pueda obtener una lista con los informes de un grupo de informes (p. 374)
- Permitir que un usuario pueda obtener una lista con los casos de prueba de un informe (p. 374)

## Consultar los informes de pruebas

Puede ver detalles sobre un informe de pruebas, como información sobre sus casos de prueba, los números de realizaciones correctas e incorrectas, y el tiempo que tardó en ejecutarse. Puede ver los informes de pruebas agrupados por ejecución de compilación, grupo de informes o cuenta de AWS. Elija un informe de pruebas en la consola para ver los detalles y los resultados de sus casos de prueba.

Puede ver los informes de pruebas que no están caducados. Los informes de pruebas caducan 30 días después de su creación. No puede ver un informe caducado en CodeBuild.

### Temas

- Consultar los informes de pruebas de una compilación (p. 326)
- Consultar los informes de pruebas de un grupo de informes (p. 326)
- Consultar los informes de pruebas de una cuenta de AWS (p. 327)

## Consultar los informes de pruebas de una compilación

Para ver los informes de pruebas de una compilación

1. Abra el iconoAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. Localice la compilación que desea ver. Si conoce el proyecto que ejecutó la compilación que creó el informe de pruebas:
  1. En el panel de navegación, elija Build projects (Proyectos de compilación) y, a continuación, elija el proyecto con la compilación que ejecutó el informe de pruebas que desea ver.
  2. Elija Build history (Historial de compilación) y, a continuación, elija la compilación ejecutada que creó los informes que desea ver.

También puede localizar la compilación en el historial de compilación de su cuenta de AWS:

1. En el panel de navegación, elija Build history (Historial de compilación) y, a continuación, elija la compilación que creó los informes de pruebas que desea ver.
3. En la página de compilación, elija Reports (Informes) y, a continuación, elija un informe de pruebas para ver sus detalles.

## Consultar los informes de pruebas de un grupo de informes

Para ver informes de pruebas de un grupo de informes

1. Abra el iconoAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Report groups (Grupos de informes).
3. Elija el grupo de informes que contiene los informes de pruebas que desea ver.
4. Elija un informe de pruebas para ver sus detalles.

## Consultar los informes de pruebas de una cuenta de AWS

Para ver informes de pruebas en su cuenta de AWS

1. Abra el icono AWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Report history (Historial de informes).
3. Elija un informe de pruebas para ver sus detalles.

## Generación de informes de pruebas con marcos de pruebas

En los temas de esta sección se muestra cómo configurar informes de pruebas en AWS CodeBuild para varios marcos de pruebas.

Temas

- [Configurar generación de informes de pruebas con Jasmine \(p. 327\)](#)
- [Configurar generación de informes de pruebas con Jest \(p. 329\)](#)
- [Configurar generación de informes de pruebas con pytest \(p. 330\)](#)
- [Configurar generación de informes de pruebas con RSpec \(p. 330\)](#)

## Configurar generación de informes de pruebas con Jasmine

El siguiente procedimiento muestra cómo configurar generación de informes de prueba en AWS CodeBuild con el [marco de pruebas JasmineBDD](#).

El procedimiento requiere los siguientes requisitos previos:

- Debe tener un proyecto de CodeBuild existente.
- Su proyecto es un proyecto Node.js que está configurado para usar el marco de pruebas de Jasmine.

Agregue el paquete [jasmine-reporters](#) a la sección `devDependencies` del archivo `package.json` de su proyecto. Este paquete tiene una colección de clases de generador de informes de JavaScript que se pueden usar con Jasmine.

```
npm install --save-dev jasmine-reporters
```

Si todavía no está presente, agregue el script `test` al archivo `package.json` de su proyecto. LatestEl script asegura que se llama a Jasmine cuando `npm test` se ejecuta.

```
{
 "scripts": {
 "test": "npm test"
 }
}
```

CodeBuild es compatible con los siguientes generadores de informes de pruebas de Jasmine:

#### JUnitXmlReporter

Se utiliza para generar informes en el formato JunitXml.

#### NUnitXmlReporter

Se utiliza para generar informes en el formato NunitXml.

Un proyecto Node.js con Jasmine tendrá, de forma predeterminada, un subdirectorio `spec`, que contiene los scripts de configuración y prueba de Jasmine.

Para configurar Jasmine para generar informes en el formato JunitXML, cree una instancia para el generador de informes JUnitXmlReporter agregando el siguiente código a las pruebas.

```
var reporters = require('jasmine-reporters');

var junitReporter = new reporters.JUnitXmlReporter({
 savePath: <test report directory>,
 filePrefix: <report filename>,
 consolidateAll: true
});

jasmine.getEnv().addReporter(junitReporter);
```

Para configurar Jasmine para generar informes en el formato NunitXML, cree una instancia para el generador de informes NUnitXmlReporter agregando el siguiente código a las pruebas.

```
var reporters = require('jasmine-reporters');

var nunitReporter = new reporters.NUnitXmlReporter({
 savePath: <test report directory>,
 filePrefix: <report filename>,
 consolidateAll: true
});

jasmine.getEnv().addReporter(nunitReporter)
```

Los informes de pruebas se exportan al archivo especificado por `<directorio de informes de pruebas>/<nombre de archivo de pruebas>`.

En su archivo `buildspec.yml`, agregue y actualice las siguientes secciones.

```
version: 0.2

phases:
 pre_build:
 commands:
 - npm install
 build:
 commands:
 - npm build
 - npm test

reports:
 jasmine_reports:
 files:
 - <report filename>
 file-format: JUNITXML
 base-directory: <test report directory>
```

Si utiliza el formato de informe NunitXml, cambie el valor `file-format` al siguiente.

```
file-format: NUNITXML
```

## Configurar generación de informes de pruebas con Jest

El siguiente procedimiento muestra cómo configurar generación de informes de pruebas en AWS CodeBuild con el [marco de pruebas de Jest](#).

El procedimiento requiere los siguientes requisitos previos:

- Debe tener un proyecto de CodeBuild existente.
- Su proyecto es un proyecto Node.js que está configurado para usar el marco de pruebas de Jest.

Agregue el paquete `jest-junit` a la sección `devDependencies` del archivo `package.json` de su proyecto. CodeBuild utiliza este paquete para generar informes en el `JunitXml` formato.

```
npm install --save-dev jest-junit
```

Si todavía no está presente, agregue el script `test` al archivo `package.json` de su proyecto. LatestEl script asegura que se llama a Jest cuando `npm test` se ejecuta.

```
{
 "scripts": {
 "test": "jest"
 }
}
```

Configure Jest para usar el generador de informes `JunitXml` agregando lo siguiente al archivo de configuración de Jest. Si su proyecto no tiene un archivo de configuración de Jest, cree un archivo denominado `jest.config.js` en la raíz del proyecto y agregue lo siguiente. Los informes de pruebas se exportan al archivo especificado por `<directorio de informes de pruebas>/<nombre de archivo de pruebas>`.

```
module.exports = {
 reporters: [
 'default',
 ['jest-junit', {
 outputDirectory: <test report directory>,
 outputName: <report filename>,
 }]
]
};
```

En su archivo `buildspec.yml`, agregue y actualice las siguientes secciones.

```
version: 0.2

phases:
 pre_build:
 commands:
 - npm install
 build:
```



```
commands:
 - npm build
 - npm test

reports:
 jest_reports:
 files:
 - <report filename>
 file-format: JUNITXML
 base-directory: <test report directory>
```

## Configurar generación de informes de pruebas con pytest

El siguiente procedimiento muestra cómo configurar informes de pruebas en AWS CodeBuild con el [marco de pruebas de pytest](#).

El procedimiento requiere los siguientes requisitos previos:

- Debe tener un proyecto de CodeBuild existente.
- Su proyecto es un proyecto de Python que está configurado para usar el marco de pruebas de pytest.

Agregue la siguiente entrada a la fase `build` o `post_build` del archivo `buildspec.yml`. Este código detecta automáticamente las pruebas en el directorio actual y exporta los informes de pruebas al archivo especificado por `<directorio de informes de pruebas>/<nombre de archivo de informes>`. El informe utiliza el formato JunitXml.

```
- python -m pytest --junitxml=<test report directory>/<report filename>
```

En su archivo `buildspec.yml`, agregue y actualice las siguientes secciones.

```
version: 0.2

phases:
 install:
 runtime-versions:
 python: 3.7
 commands:
 - pip3 install pytest
 build:
 commands:
 - python -m pytest --junitxml=<test report directory>/<report filename>

reports:
 pytest_reports:
 files:
 - <report filename>
 base-directory: <test report directory>
 file-format: JUNITXML
```

## Configurar generación de informes de pruebas con RSpec

El siguiente procedimiento muestra cómo configurar generación de informes de pruebas en AWS CodeBuild con el [marco de pruebas de RSpec](#).

El procedimiento requiere los siguientes requisitos previos:

- Debe tener un proyecto de CodeBuild existente.
- Su proyecto es un proyecto Ruby que está configurado para usar el marco de pruebas de RSpec.

Agregue y actualice lo siguiente en su archivo `buildspec.yml`. Este código ejecuta las pruebas en el directorio `<directorio de fuente de pruebas>` y exporta los informes de pruebas al archivo especificado por `<directorio de informes de pruebas>/<nombre de archivo del informe>`. El informe utiliza el formato JunitXml.

```
version: 0.2

phases:
 install:
 runtime-versions:
 ruby: 2.6
 pre_build:
 commands:
 - gem install rspec
 - gem install rspec_junit_formatter
 build:
 commands:
 - rspec <test source directory>/* --format RspecJunitFormatter --out <test report
 directory>/<report filename>
 reports:
 rspec_reports:
 files:
 - <report filename>
 base-directory: <test report directory>
 file-format: JUNITXML
```

## Informes de cobertura de códigos

CodeBuild le permite generar informes de cobertura de código para sus pruebas. Se proporcionan los siguientes informes de cobertura de código:

### coverage de líneas

La cobertura de línea mide cuántas declaraciones cubren sus pruebas. Una declaración es una sola instrucción, sin incluir comentarios ni condicionales.

$$\text{line coverage} = (\text{total lines covered}) / (\text{total number of lines})$$

### coverage de sucursales

La cobertura de sucursales mide cuántas ramas cubren sus pruebas de cada rama posible de una estructura de control, como `unif` o `biencasestatement`.

$$\text{branch coverage} = (\text{total branches covered}) / (\text{total number of branches})$$

Se admiten los siguientes formatos de archivo de informe de cobertura de código:

- jaCoco XML
- SimpleCov JSON<sup>1</sup>
- Trébol XML
- coverage XML

\* CodeBuild acepta informes de cobertura de código JSON generados por [simplecov](#), [nosimplecov](#) o [nosimplecov-json](#).

## Crear un informe de cobertura de código

Para crear un informe de cobertura de código, se ejecuta un proyecto de compilación que se configura con al menos un grupo de informes de cobertura de código en su archivo `buildspec`. CodeBuild interpretará los resultados de cobertura de código y proporcionará un informe de cobertura de código para la ejecución. Se genera un nuevo informe de pruebas para cada compilación posterior que utilice el mismo archivo `buildspec`.

Para crear un informe de pruebas

1. Cree un proyecto de compilación. Para obtener información, consulte [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#).
2. Configure el archivo `buildspec` del proyecto con información de informe de pruebas:
  - a. Añada `unreports:` y especifique el nombre del grupo de informes. CodeBuild crea un grupo de informes utilizando el nombre del proyecto y el nombre especificado con el formato `project-name-report-group-name-in-buildspec`. Si ya tiene un grupo de informes que desee utilizar, especifique su ARN. Si utiliza el nombre en lugar de ARN, CodeBuild crea un nuevo grupo de informes. Para obtener más información, consulte [Reports syntax in the buildspec file](#).
  - b. En el grupo de informes, especifique la ubicación de los archivos que contienen los resultados de cobertura de código. Si utiliza más de un grupo de informes, especifique las ubicaciones de los archivos de resultados para cada grupo de informes. Cada vez que se ejecuta el proyecto de compilación, se crea un nuevo informe de cobertura de código. Para obtener más información, consulte [Especificar archivos de prueba \(p. 313\)](#).

Este es un ejemplo que genera un informe de cobertura de código para un archivo de resultados XML de JaCoco ubicado en `test-results/jacoco-coverage-report.xml`.

```
reports:
 jacoco-report:
 files:
 - 'test-results/jacoco-coverage-report.xml'
 file-format: 'JACOCOXML'
```

- c. En la `commands` Sección sobre de la `build` bien `post_build`, especifique los comandos que ejecutan el análisis de cobertura de código. Para obtener más información, consulte [Especificar comandos de prueba \(p. 313\)](#).
3. Ejecute una compilación del proyecto de compilación. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#).
  4. Cuando la compilación se haya completado, elija la nueva ejecución de la compilación en Build history (Historial de compilación) en la página del proyecto. Haga clic en `.Informes` para ver el informe de cobertura de código. Para obtener más información, consulte [Consultar los informes de pruebas de una compilación \(p. 326\)](#).

# Registro y monitoreo en AWS CodeBuild

La monitorización es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de AWS CodeBuild y sus soluciones de AWS. Debe recopilar datos de monitorización de todas las partes de su AWS para que le resulte más sencillo depurar un error que se produce en distintas partes del código, en caso de que ocurra. AWS proporciona las siguientes herramientas para supervisar sus recursos y compilaciones de CodeBuild y responder a posibles incidentes.

## Temas

- [Registrar llamadas a la API de AWS CodeBuild con AWS CloudTrail \(p. 333\)](#)
- [Monitorización de AWS CodeBuild \(p. 335\)](#)

## Registrar llamadas a la API de AWS CodeBuild con AWS CloudTrail

AWS CodeBuild está integrado con AWS CloudTrail, un servicio que proporciona un registro de las medidas de adoptadas por un usuario, un rol o un AWS en CodeBuild. CloudTrail captura todas las llamadas a la API de CodeBuild como eventos, incluidas las llamadas procedentes de la consola de CodeBuild y las llamadas de código a las API de CodeBuild. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de S3, incluidos los eventos de CodeBuild. Si no configura un registro de seguimiento, puede ver los eventos más recientes de la consola de CloudTrail en el Event history (Historial de eventos). Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a CodeBuild, la dirección IP desde la que se realizó, quién la realizó y cuándo, etc.

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

## AWS CodeBuild Información en CloudTrail

CloudTrail se habilita en su cuenta de AWS cuando la crea. Cuando se produce una actividad en CodeBuild, dicha actividad se registra en un evento de CloudTrail junto con los demás AWS eventos de servicios de en Historial de eventos. Puede ver, buscar y descargar los últimos eventos de la cuenta de AWS. Para obtener más información, consulte [Ver eventos con el historial de eventos de CloudTrail](#) en la AWS CloudTrail Guía del usuario.

Para mantener un registro continuo de los eventos de la AWS, incluidos los eventos de CodeBuild, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail que pueda enviar archivos de registro a un bucket de S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de S3 especificado. Puede configurar otros AWS para analizar y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para obtener más información, consulte:

- [Introducción a la creación de registros de seguimiento](#)
- [Servicios e integraciones compatibles con CloudTrail](#)

- [Configuración de notificaciones de Amazon SNS para CloudTrail](#)
- [Recibir archivos de registro de de de de varias regiones](#) y [Recepción de archivos de registro de de de de varias cuentas](#)

CloudTrail registra todas las acciones de CodeBuild y están documentadas en la [Referencia de la API de CodeBuild](#). Por ejemplo, las llamadas a `CreateProject` (en el AWS CLI, `create-project`), `StartBuild` (en el AWS CLI, `start-project`), y `UpdateProject` (en el AWS CLI, `update-project`) generan entradas en los archivos de registro de CloudTrail.

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

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

Para obtener más información, consulte la [CloudTrail userIdentity de elemento](#) en la AWS CloudTrail Guía del usuario.

## Descripción de las entradas de los archivos de registro de AWS CodeBuild

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de log al bucket de S3 que se especifique. Los archivos log de CloudTrail pueden contener una o varias entradas de log. Un evento representa una solicitud específica realizada desde un origen y contiene información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no son un rastro de la stack ordenado de las llamadas a las API públicas, por lo que no aparecen en ningún orden específico.

### Note

Para proteger la información confidencial, lo siguiente está oculto en los registros de CodeBuild:

- ID de clave de acceso de AWS. Para obtener más información, consulte [Administración de las claves de acceso de los usuarios de IAM](#) en la AWS Identity and Access Management Guía del usuario.
- Cadenas especificadas mediante el almacén de parámetros. Para obtener más información, consulte [Almacén de parámetros de Systems Manager](#) y [Descripción de la consola de almacén de parámetros de Systems Manager](#) en la Guía del usuario de Amazon EC2 Systems Manager.
- Cadenas especificadas mediante AWS Secrets Manager. Para obtener más información, consulte [Administración de claves \(p. 353\)](#).

En el siguiente ejemplo, se muestra una entrada de registro de CloudTrail que muestra la creación de un proyecto de compilación en CodeBuild.

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "type": "FederatedUser",
 "principalId": "account-ID:user-name",
 "arn": "arn:aws:sts::account-ID:federated-user/user-name",
 "accountId": "account-ID",
 "accessKeyId": "access-key-ID",
 "sessionContext": {
```

```

 "attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2016-09-06T17:59:10Z"
 },
 "sessionIssuer": {
 "type": "IAMUser",
 "principalId": "access-key-ID",
 "arn": "arn:aws:iam:account-ID:user/user-name",
 "accountId": "account-ID",
 "userName": "user-name"
 }
 },
 "eventTime": "2016-09-06T17:59:11Z",
 "eventSource": "codebuild.amazonaws.com",
 "eventName": "CreateProject",
 "awsRegion": "region-ID",
 "sourceIPAddress": "127.0.0.1",
 "userAgent": "user-agent",
 "requestParameters": {
 "awsActId": "account-ID"
 },
 "responseElements": {
 "project": {
 "environment": {
 "image": "image-ID",
 "computeType": "BUILD_GENERAL1_SMALL",
 "type": "LINUX_CONTAINER",
 "environmentVariables": []
 },
 "name": "codebuild-demo-project",
 "description": "This is my demo project",
 "arn": "arn:aws:codebuild:region-ID:account-ID:project/codebuild-demo-project:project-ID",
 "encryptionKey": "arn:aws:kms:region-ID:key-ID",
 "timeoutInMinutes": 10,
 "artifacts": {
 "location": "arn:aws:s3:::codebuild-region-ID-account-ID-output-bucket",
 "type": "S3",
 "packaging": "ZIP",
 "outputName": "MyOutputArtifact.zip"
 },
 "serviceRole": "arn:aws:iam:account-ID:role/CodeBuildServiceRole",
 "lastModified": "Sep 6, 2016 10:59:11 AM",
 "source": {
 "type": "GITHUB",
 "location": "https://github.com/my-repo.git"
 },
 "created": "Sep 6, 2016 10:59:11 AM"
 }
 },
 "requestID": "9d32b228-745b-11e6-98bb-23b67EXAMPLE",
 "eventID": "581f7dd1-8d2e-40b0-aeae-0dbf7EXAMPLE",
 "eventType": "AwsApiCall",
 "recipientAccountId": "account-ID"
}

```

## Monitorización de AWS CodeBuild

Puede utilizar Amazon CloudWatch para vigilar sus compilaciones, informar cuando algo no funcione y realizar acciones automáticas cuando proceda. Puede monitorear sus compilaciones en dos niveles:

#### Nivel de proyecto

Estas métricas son para todas las compilaciones en el proyecto especificado. Para ver las métricas de un proyecto, especifique `ProjectName` para la dimensión en CloudWatch.

#### AWS nivel de cuenta

Estas métricas son para todas las compilaciones de una cuenta. Para ver las métricas en el AWS, no especifica una dimensión en CloudWatch. Las métricas de utilización de recursos de compilación no están disponibles en el AWS nivel de cuenta.

Las métricas de CloudWatch muestran el comportamiento de sus compilaciones con el paso del tiempo. Por ejemplo, puede monitorear:

- Cuántas compilaciones se intentaron en un proyecto de compilación o en una cuenta de AWS con el paso del tiempo.
- Cuántas compilaciones se intentaron con éxito en un proyecto de compilación o en una cuenta de AWS con el paso del tiempo.
- Cuántas compilaciones mostraron errores en un proyecto de compilación o en una cuenta de AWS con el paso del tiempo.
- CodeBuild invirtió en la ejecución de compilaciones en un proyecto de compilación o en una AWS Cuenta con el paso del tiempo.
- Utilización de recursos de compilación para una compilación o un proyecto de compilación completo. Las métricas de utilización de recursos de compilación incluyen métricas como CPU, memoria y uso del almacenamiento de información.

Para obtener más información, consulte [Monitorización de métricas CodeBuild \(p. 340\)](#).

## Métricas de CodeBuild CloudWatch

Se puede realizar un seguimiento de las siguientes métricas por AWS Cuenta o proyecto de compilación.

#### BuildDuration

Mide la duración de la fase `BUILD` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### Compilaciones

Mide el número de compilaciones activadas.

Unidades: Recuento

Estadísticas de CloudWatch válí: Sum

#### DescargarSourceDuration

Mide la duración de la fase `DOWNLOAD_SOURCE` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### Duración

Mide la duración de todas las compilaciones con el paso del tiempo.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### FailedBuilds

Mide el número de compilaciones con errores debido al error del cliente o un tiempo de espera agotado.

Unidades: Recuento

Estadísticas de CloudWatch válí: Sum

#### FinalizarDuration

Mide la duración de la fase `FINALIZING` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### InstallDuration

Mide la duración de la fase `INSTALL` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### PostBuildDuration

Mide la duración de la fase `POST_BUILD` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### PrebuildDuration

Mide la duración de la fase `PRE_BUILD` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### ProvisioningDuration

Mide la duración de la fase `PROVISIONING` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### QueuedDuration

Mide la duración de la fase `QUEUED` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

#### SubmittedDuration

Mide la duración de la fase `SUBMITTED` de la compilación.



Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

SucceededBuilds

Mide el número de compilaciones correctas.

Unidades: Recuento

Estadísticas de CloudWatch válí: Sum

UploadArtifactsDuration

Mide la duración de la fase `UPLOAD_ARTIFACTS` de la compilación.

Unidades: Segundos

Estadísticas de CloudWatch válí: Average (recomendado)

## Métricas de utilización de recursos de CodeBuild CloudWatch

Se puede realizar un seguimiento de las siguientes métricas de utilización de recursos.

### Note

Las métricas de utilización de recursos de CodeBuild solo están disponibles en las siguientes regiones:

- Asia Pacific (Tokyo) Region
- Asia Pacific (Seoul) Region
- Asia Pacific (Mumbai) Region
- Asia Pacific (Singapore) Region
- Asia Pacific (Sydney) Region
- Canada (Central) Region
- Europe (Frankfurt) Region
- Europe (Ireland) Region
- Europe (London) Region
- Región de Europa (París)
- South America (São Paulo) Region
- US East (N. Virginia) Region
- US East (Ohio) Region
- US West (N. California) Region
- US West (Oregon) Region

CPUUtilizado

El número de unidades de CPU por el contenedor de compilación.

Unidades: Unidades de CPU

Estadísticas de CloudWatch válidas: Average (recomendado)

#### CPUUtilizedPorcentaje

El porcentaje de procesamiento asignado utilizado por el contenedor de compilación.

Unidades: Porcentaje

Estadísticas de CloudWatch válidas: Average (recomendado)

#### MemoryUtilization

El número de megabytes de memoria utilizados por el contenedor de compilación.

Unidades: Megabytes

Estadísticas de CloudWatch válidas: Average (recomendado)

#### MemoryUtilizedPercent

Porcentaje de memoria asignada que utiliza el contenedor de compilación.

Unidades: Porcentaje

Estadísticas de CloudWatch válidas: Average (recomendado)

#### StorageReadBytes

La velocidad de lectura de almacenamiento utilizada por el contenedor de compilación.

Unidades: Bytes/segundo

Estadísticas de CloudWatch válidas: Average (recomendado)

#### StorageWriteBytes

La velocidad de escritura de almacenamiento utilizada por el contenedor de compilación.

Unidades: Bytes/segundo

Estadísticas de CloudWatch válidas: Average (recomendado)

## Dimensiones de CodeBuild CloudWatch

CodeBuild proporciona las siguientes dimensiones métricas de CloudWatch. Si no se especifica ninguno de estos, las métricas son para elAWSaccount.

#### BuildId, BuildNumber, ProjectName

Se proporcionan métricas para un identificador de compilación, número de compilación y nombre de proyecto.

#### ProjectName

Para un nombre de proyecto se proporcionan las métricas.

## Alarmas de CodeBuild CloudWatch

Puede utilizar la consola de CloudWatch para crear alarmas en función de las métricas de CodeBuild para que pueda reaccionar si hay algún problema con sus compilaciones. Las dos métricas que son más útiles con alarmas son:

- **FailedBuild.** Puede crear una alarma que se active cuando se detecte un determinado número de compilaciones con errores en un número de segundos predeterminado. En CloudWatch, especifique el número de segundos y de compilaciones con errores que activarán una alarma.
- **Duration.** Puede crear una alarma que se activa cuando una compilación tarda más de lo esperado. Especifique la cantidad de segundos que debe transcurrir después de que se inicie una compilación y antes de que se complete una compilación para activar una alarma.

Para obtener información acerca de cómo crear alarmas para las métricas de CodeBuild, consulte [Monitorización de compilaciones con alarmas de CloudWatch \(p. 350\)](#). Para obtener más información sobre las alarmas, consulte [Creación de alarmas de Amazon CloudWatch](#) en la Guía del usuario de Amazon CloudWatch.

## Monitorización de métricas CodeBuild

AWS CodeBuild monitorea funciones en su nombre e informa sobre las métricas a través de Amazon CloudWatch. Estas métricas incluyen el número de compilaciones totales, compilaciones con errores, compilaciones correctas y la duración de las compilaciones.

Puede utilizar la consola de CodeBuild o la consola de CloudWatch para monitorear métricas para CodeBuild. Los procedimientos siguientes muestran cómo aceptar las métricas.

### Temas

- [Métricas de compilación de acceso \(consola de CodeBuild\) \(p. 340\)](#)
- [Métricas de compilación de acceso \(consola de Amazon CloudWatch\) \(p. 341\)](#)

## Métricas de compilación de acceso (consola de CodeBuild)

### Note

No puede personalizar las métricas o los gráficos utilizados para mostrarlas en la consola de CodeBuild. Para personalizar la pantalla, utilice la consola de Amazon CloudWatch para ver las métricas de compilación.

### Métricas de nivel de cuenta

Para obtener acceso a AWS Métricas de nivel de cuenta

1. Inicie sesión en AWS Management Console y abra AWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Account metrics (Métricas de la cuenta).

### Métricas de nivel de proyecto

Para obtener acceso a las métricas de nivel de proyecto

1. Inicie sesión en AWS Management Console y abra AWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. En la lista de proyectos de compilación, en la columna Name (Nombre), elija el proyecto en el que desee ver las métricas.
4. Elija la pestaña Metrics (Métricas).

## Métricas de compilación de acceso (consola de Amazon CloudWatch)

Puede personalizar las métricas y los gráficos utilizados para mostrarlas con la consola de CloudWatch.

### Métricas de nivel de cuenta

Para obtener acceso a las métricas de nivel de cuenta

1. Inicie sesión en la AWS Management Console y abra la consola de CloudWatch en <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, seleccione Metrics.
3. En la pestaña All metrics (Todas las métricas), elija CodeBuild.

# Metrics

Favorites

 Add a dashboard

4. Elija Account Metrics (Métricas de cuenta).
5. Elija uno o varios proyectos y métricas. Por cada proyecto, puede elegir las métricas SucceededBuilds, FailedBuilds, Builds y Duration. Todas las combinaciones de métricas y proyectos seleccionados se muestran en el gráfico de la página.

## Métricas de nivel de proyecto

Para obtener acceso a las métricas de nivel de proyecto

1. Inicie sesión en la AWS Management Console y abra la consola de CloudWatch en <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, seleccione Metrics.
3. En la pestaña All metrics (Todas las métricas), elija CodeBuild.

# Metrics

Favorites

 Add a dashboard

4. Elija By Project (Por proyecto).
5. Elija uno o varias combinaciones de proyectos y métricas. Por cada proyecto, puede elegir las métricas SucceededBuilds, FailedBuilds, Builds y Duration. Todas las combinaciones de métricas y proyectos seleccionados se muestran en el gráfico de la página.
6. (Opcional) Puede personalizar las métricas y los gráficos. Por ejemplo, en la lista desplegable en la columna Estadística, puede elegir otra estadística que mostrar. O bien, en el menú desplegable en la columna Period (Período), puede elegir un periodo de tiempo diferente que se va a utilizar para monitorear las métricas.

Para obtener más información, consulte [Representación gráfica de métricas](#) y [Ver métricas disponibles](#) en la Guía del usuario de Amazon CloudWatch.

## Supervisión de métricas de utilización de recursos de CodeBuild

AWS CodeBuild monitorea la utilización de recursos en su nombre e informa sobre las métricas a través de Amazon CloudWatch. Entre ellas se incluyen métricas como CPU, memoria y uso del almacenamiento.

Puede utilizar la consola CodeBuild o la consola de CloudWatch para supervisar las métricas de utilización de recursos para CodeBuild. Los procedimientos siguientes muestran cómo aceptar las métricas de utilización de recursos.

### Note

Las métricas de utilización de recursos de CodeBuild solo están disponibles en las siguientes regiones:

- Asia Pacific (Tokyo) Region
- Asia Pacific (Seoul) Region
- Asia Pacific (Mumbai) Region
- Asia Pacific (Singapore) Region
- Asia Pacific (Sydney) Region
- Canada (Central) Region
- Europe (Frankfurt) Region
- Europe (Ireland) Region
- Europe (London) Region
- Región de Europa (París)
- South America (São Paulo) Region
- US East (N. Virginia) Region
- US East (Ohio) Region
- US West (N. California) Region
- US West (Oregon) Region

### Temas

- [Acceder a métricas de utilización de recursos \(consola de CodeBuild\)](#) (p. 346)
- [Acceder a métricas de utilización de recursos \(consola de Amazon CloudWatch\)](#) (p. 346)



## Acceder a métricas de utilización de recursos (consola de CodeBuild)

### Note

No puede personalizar las métricas o los gráficos utilizados para mostrarlas en la consola de CodeBuild. Para personalizar la pantalla, utilice la consola de Amazon CloudWatch para ver las métricas de compilación.

### Métricas de utilización de recursos en el nivel de proyecto

Para obtener acceso a las métricas de utilización de recursos de nivel de proyecto

1. Inicie sesión enAWS Management Console y abraAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build projects.
3. En la lista de proyectos de compilación, en la columnaNombreElija el proyecto cuyas métricas de utilización quiera ver.
4. Elija la pestaña Metrics (Métricas). Las métricas de utilización de recursos se muestran en el cuadro de diálogoMétricas de utilización de recursossección.
5. Para ver las métricas de utilización de recursos a nivel de proyecto en la consola de CloudWatch, elijaVer en CloudWatch en laMétricas de utilización de recursossección.

### Métricas de utilización de recursos en el nivel de compilación

Para obtener acceso a las métricas de utilización de recursos de nivel de compilación

1. Inicie sesión enAWS Management Console y abraAWS CodeBuild en <https://console.aws.amazon.com/codesuite/codebuild/home>.
2. En el panel de navegación, elija Build history.
3. En la lista de compilaciones, en elEjecutar compilaciónElija la compilación para la que desea ver las métricas de utilización.
4. Elija el iconoUtilización de recursosPestaña.
5. Para ver las métricas de utilización de recursos en el nivel de compilación en la consola de CloudWatch, elijaVer en CloudWatch en laMétricas de utilización de recursossección.

## Acceder a métricas de utilización de recursos (consola de Amazon CloudWatch)

La consola de Amazon CloudWatch se puede utilizar para acceder a las métricas de utilización de recursos de CodeBuild.

### Métricas de utilización de recursos en el nivel de proyecto

Para obtener acceso a las métricas de utilización de recursos de nivel de proyecto

1. Inicie sesión en la AWS Management Console y abra la consola de CloudWatch en <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, seleccione Metrics.
3. En la pestaña All metrics (Todas las métricas), elija CodeBuild.

 Metrics

Favorites

 Add a dashboard

4. Elija By Project (Por proyecto).
5. Elija uno o varias combinaciones de proyectos y métricas para añadirlas al gráfico. Todas las combinaciones de métricas y proyectos seleccionados se muestran en el gráfico de la página.
6. (Opcional) Puede personalizar las métricas y los gráficos en la **Métricas de Pestaña**. Por ejemplo, en la lista desplegable en la columna **Estadística**, puede elegir otra estadística que mostrar. O bien, en el menú desplegable en la columna **Period (Período)**, puede elegir un periodo de tiempo diferente que se va a utilizar para monitorear las métricas.

Para obtener más información, consulte [Representación gráfica de métricas](#) y [Ver métricas disponibles](#) en la Guía del usuario de Amazon CloudWatch.

## Métricas de utilización de recursos en el nivel de compilación

Para obtener acceso a las métricas de utilización de recursos de nivel de compilación

1. Inicie sesión en la AWS Management Console y abra la consola de CloudWatch en <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, seleccione Metrics.
3. En la pestaña All metrics (Todas las métricas), elija CodeBuild.

 Metrics

Favorites

 Add a dashboard

4. SeleccionarBuildId, BuildNumber, ProjectName.
5. Elija una o varias combinaciones de compilaciones y métricas para añadirlas al gráfico. Todas las combinaciones de métricas y compilaciones seleccionadas se muestran en el gráfico de la página.
6. (Opcional) Puede personalizar las métricas y los gráficos en la **Métricas de Pestaña**. Por ejemplo, en la lista desplegable en la columna **Estadística**, puede elegir otra estadística que mostrar. O bien, en el menú desplegable en la columna **Period (Período)**, puede elegir un periodo de tiempo diferente que se va a utilizar para monitorear las métricas.

Para obtener más información, consulte [Representación gráfica de métricas](#) y [Ver métricas disponibles](#) en la Guía del usuario de Amazon CloudWatch.

## Monitorización de compilaciones con alarmas de CloudWatch

Puede crear una alarma de CloudWatch para sus compilaciones. Una alarma vigila una única métrica durante el periodo de tiempo especificado y realiza una o varias acciones en función del valor de la métrica en relación con un determinado umbral durante una serie de periodos de tiempo. Con la funcionalidad de alarma de CloudWatch nativa, puede especificar cualquiera de las acciones admitidas por CloudWatch cuando se supera un umbral. Puede especificar, por ejemplo, que se envíe una notificación de Amazon SNS cuando más de tres compilaciones en su cuenta muestren errores en un plazo de quince minutos.

Para crear una alarma de CloudWatch para una métrica de CodeBuild

1. Inicie sesión en la AWS Management Console y abra la consola de CloudWatch en <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, elija Alarms.
3. Elija Create Alarm (Crear alarma).
4. En CloudWatch Metrics by Category (Métricas de CloudWatch por categoría), elija la categoría CodeBuild Metrics (Métricas CodeBuild). Si sabe que desea solo las métricas en el nivel de proyecto, elija By Project (Por proyecto). Si sabe que desea solo las métricas en el nivel de cuenta, elija Account Metrics (Métricas de cuenta).
5. En Create Alarm (Crear alarma) si aún no está seleccionada, elija Select Metric (Seleccionar métrica).
6. Seleccione una métrica para la que desee crear una alarma. Las opciones son By Project (Por proyecto) o Account Metrics (Métricas de cuenta).
7. Elija Next (Siguiente) o Define Alarm (Definir alarma) y, a continuación, cree la alarma. Para obtener más información, consulte [Creación de alarmas de Amazon CloudWatch](#) en la Guía del usuario de Amazon CloudWatch. Para obtener más información sobre la configuración de las notificaciones de Amazon SNS cuando se desencadena una alarma, consulte [Configurar notificaciones de Amazon SNS](#) en la Guía para desarrolladores de Amazon SNS.
8. Elija Create Alarm.

# Seguridad en AWS CodeBuild

La seguridad en la nube de AWS es la mayor prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

Los asuntos relacionados con la seguridad y la conformidad son una responsabilidad entre AWS y usted. Este modelo compartido puede ayudarle a liberar su carga operativa, ya que AWS opera, administra y controla los componentes, desde el sistema operativo host y la capa de virtualización hasta la seguridad física en las instalaciones del servicio. Usted asume la responsabilidad y la administración del sistema operativo invitado (incluidas las actualizaciones y los parches de seguridad) y de cualquier otro software de aplicaciones asociadas. Es también responsable de la configuración del firewall del grupo de seguridad proporcionado por AWS. Sus responsabilidades variarán en función de los servicios que utilice, la integración de estos con su entorno de TI y las leyes y regulaciones aplicables. Por lo tanto, debe prestar atención cuando valore los servicios que utiliza su organización. Para obtener más información, consulte el [Modelo de responsabilidad compartida](#).

Para aprender a proteger los recursos de CodeBuild, consulte los siguientes temas.

## Temas

- [Protección de los datos en AWS CodeBuild](#) (p. 351)
- [Administración de identidades y accesos en AWS CodeBuild](#) (p. 353)
- [Validación de la conformidad enAWSCodeBuild](#) (p. 384)
- [Resiliencia en AWS CodeBuild](#) (p. 385)
- [Seguridad de la infraestructura enAWSCodeBuild](#) (p. 385)
- [Acceda a su proveedor de código fuente en CodeBuild](#) (p. 385)

## Protección de los datos en AWS CodeBuild

El [modelo de responsabilidad compartida](#) de AWS se aplica a la protección de datos de AWS CodeBuild. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta toda la Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Este contenido incluye la configuración de seguridad y las tareas de administración de los servicios de AWS que usted utiliza. Para obtener más información sobre la privacidad de los datos, consulte las [Preguntas frecuentes sobre la privacidad de datos](#). Para obtener información sobre la protección de datos en Europa, consulte la publicación de blog [The AWS Shared Responsibility Model and GDPR](#) en el Blog de seguridad de AWS.

Para fines de protección de datos, recomendamos proteger las credenciales de Cuenta de AWS y configurar cuentas de usuario individuales con AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir con sus obligaciones laborales. También recomendamos proteger sus datos de las siguientes formas:

- Utilice Multi-Factor Authentication (MFA) con cada cuenta.
- Utilice SSL/TLS para comunicarse con los recursos de AWS. Recomendamos TLS 1.2 o una versión posterior.
- Configure la API y el registro de actividad del usuario con AWS CloudTrail.
- Utilice las soluciones de cifrado de AWS, junto con todos los controles de seguridad predeterminados dentro de los servicios de AWS.

- Utilice avanzados servicios de seguridad administrados, como Amazon Macie, que lo ayuden a detectar y proteger los datos personales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados FIPS 140-2 al acceder a AWS a través de una interfaz de línea de comandos o una API, utilice un punto de enlace de FIPS. Para obtener más información sobre los puntos de enlace de FIPS disponibles, consulte [Estándar de procesamiento de la información federal \(FIPS\) 140-2](#).

Recomendamos encarecidamente que nunca introduzca información de identificación confidencial, como, por ejemplo, direcciones de email de sus clientes, en etiquetas o en los campos de formato libre, como el campo Name (Nombre). Esto incluye cuando trabaje con CodeBuild u otros AWS servicios que utilizan la consola, API, AWS CLI, o bien AWS SDK. Los datos que ingresa en etiquetas o campos de formato libre utilizados para los nombres se pueden utilizar 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.

Para proteger la información confidencial, lo siguiente está oculto en los registros de CodeBuild:

- ID de clave de acceso de AWS. Para obtener más información, consulte [Administración de las claves de acceso de los usuarios de IAM](#) en el [AWS Identity and Access Management Guía del usuario](#) de.
- Cadenas especificadas mediante el almacén de parámetros. Para obtener más información, consulte [Almacén de parámetros de Administrador de sistemas](#) y [Tutorial de la consola del almacén de parámetros de Systems Manager](#) en la [Guía del usuario de Amazon EC2 Systems Manager](#).
- Cadenas especificadas mediante AWS Secrets Manager. Para obtener más información, consulte [Administración de claves \(p. 353\)](#).

Para obtener más información sobre la protección de datos, consulte la entrada de blog relativa al [modelo de responsabilidad compartida de AWS y GDPR](#) en el blog de seguridad de AWS.

#### Temas

- [Cifrado de datos \(p. 352\)](#)
- [Administración de claves \(p. 353\)](#)
- [Privacidad de tráfico \(p. 353\)](#)

## Cifrado de datos

El cifrado es una parte importante de la seguridad de CodeBuild. Algunos cifrados, por ejemplo, para el cifrado de datos en tránsito se proporcionan de forma predeterminada y no es necesario que haga nada. Otros, por ejemplo, para el cifrado de datos en reposo, se pueden configurar cuando cree su proyecto o compilación.

- Cifrado de datos en reposo- Los artefactos de compilación, como, por ejemplo, una memoria caché, los registros, los archivos de datos de informes de pruebas sin formato exportados y los resultados de compilación se cifran de forma predeterminada con Claves administradas por AWS. Si no desea utilizar estas claves de KMS, debe crear y configurar una clave administrada por el cliente. Para obtener más información, consulte [Creación de claves de KMS](#) y [Conceptos de AWS Key Management Service](#) en la Guía del usuario de AWS Key Management Service.
- Puede almacenar el identificador de la AWS clave de KMS que CodeBuild utiliza para cifrar el artefacto de salida de la compilación en el `CODEBUILD_KMS_KEY_ID` variable de entorno. Para obtener más información, consulte [Variables de entorno en los entornos de compilación \(p. 172\)](#)
- Puede especificar una clave administrada por el cliente cuando cree un proyecto de compilación. Para obtener más información, consulte [Set the Encryption Key Using the Console](#) y [Establecer la clave de cifrado mediante la CLI](#).

Los volúmenes de Amazon Elastic Block Store de su flota de compilación se cifran de forma predeterminada con Claves administradas por AWS .

- Cifrado de datos en tránsito- Todas las comunicaciones entre los clientes y CodeBuild y entre CodeBuild y sus dependencias posteriores están protegidas con conexiones TLS que se firman mediante el proceso de firma de Signature Version 4. Todos los puntos de enlace de CodeBuild utilizan certificados SHA-256 administrados por AWS Certificate Manager Private Certificate Authority. Para obtener más información, consulte [Proceso de firma de Signature Version 4](#) y [¿Qué es PCA de ACM?](#).
- Construir cifrado de artefacto- El rol de servicio CodeBuild asociado al proyecto de compilación requiere acceso a una clave KMS para cifrar sus artefactos de salida de compilación. De forma predeterminada, CodeBuild utiliza un Clave administrada por AWS para Amazon S3 en su AWS account. Si no desea usar esta Clave administrada por AWS , debe crear y configurar una clave administrada por el cliente. Para obtener más información, consulte [Creación de una clave administrada por el cliente \(p. 400\)](#) y [Crear claves](#) en la AWS KMS Guía para desarrolladores.

## Administración de claves

Puede proteger su contenido del uso no autorizado mediante el cifrado. Almacena tus claves de cifrado en AWS Secrets Manager, a continuación, asigne a la función de servicio CodeBuild asociada al proyecto de compilación permiso para obtener las claves de cifrado de su cuenta de Secrets Manager. Para obtener más información, consulte [Crear y configurar una clave administrada por el cliente para CodeBuild \(p. 400\)](#), [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#), [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#), y [Tutorial: Almacenar y recuperar un secreto](#).

Usar `CODEBUILD_KMS_KEY_ID` variable de entorno en un comando de compilación para obtener la AWS KMS identificador de clave. Para obtener más información, consulte [Variables de entorno en los entornos de compilación \(p. 172\)](#).

Puede utilizar Secrets Manager para proteger credenciales en un registro privado que almacena una imagen de Docker utilizada para su entorno en tiempo de ejecución. Para obtener más información, consulte [Ejemplo de un registro privado con AWS Secrets Manager](#) [ejemplo de CodeBuild \(p. 128\)](#).

## Privacidad de tráfico

Puede mejorar la seguridad de sus compilaciones configurando CodeBuild para que utilice un punto de enlace de VPC de interfaz. Para ello, no necesita una gateway de Internet, ni un dispositivo NAT, ni una gateway privada virtual. Tampoco es necesaria para configurar PrivateLink, aunque se recomienda. Para obtener más información, consulte [Uso de puntos de enlace de la VPC \(p. 182\)](#). Para obtener más información sobre PrivateLink y los puntos de enlace de la VPC, consulte [AWS PrivateLink](#) y [Acceso a AWS servicios a través de PrivateLink](#).

# Administración de identidades y accesos en AWS CodeBuild

El acceso a AWS CodeBuild requiere credenciales. Estas credenciales deben tener permisos para obtener acceso a los recursos de, como, por ejemplo, almacenar y recuperar artefactos de compilación en buckets de S3 y consultar Amazon CloudWatch Logs para compilaciones. En las secciones siguientes se describe cómo se puede utilizar [AWS Identity and Access Management \(IAM\)](#) y CodeBuild para ayudar a proteger el acceso a sus recursos:

- [Authentication \(p. 354\)](#)
- [Control de acceso \(p. 355\)](#)



## Authentication

Puede obtener acceso a AWS con los siguientes tipos de identidades:

- **Usuario raíz de la cuenta de AWS:** cuando se registra en AWS, proporciona una dirección de correo electrónico y la contraseña asociada a su cuenta de AWS. Estas son las credenciales raíz y proporcionan acceso completo a todos los recursos de AWS.

### Important

Por motivos de seguridad, le recomendamos que utilice las credenciales raíz solo para crear un usuario administrador, que es un usuario de IAM con permiso total para administrar su AWS account. A continuación, puede utilizar este usuario administrador para crear otros usuarios y roles de IAM; con permisos limitados. Para obtener más información, consulte [Prácticas recomendadas de IAM](#) y [Creación de un grupo y usuario administrador](#) en la Guía del usuario de IAM.

- **Usuario de IAM— Un [Usuario de IAM](#)** es simplemente una identidad en tu AWS cuenta que tiene permisos personalizados (por ejemplo, permiso para crear proyectos de compilación en CodeBuild). Puede utilizar un nombre de usuario y una contraseña de IAM para iniciar sesión en páginas web seguras de AWS tales como [AWS Management Console](#), [foros de discusión de AWS](#) o el [Centro de AWS Support](#).

Además de un nombre de usuario y una contraseña, también puede generar [claves de acceso](#) para cada usuario. Puede utilizar estas claves cuando obtenga acceso a los servicios de AWS mediante programación, ya sea a través de [uno de los SDK de AWS](#) o mediante la [AWS Command Line Interface \(AWS CLI\)](#). La AWS CLI y los SDK de AWS usan las claves de acceso para firmar criptográficamente la solicitud. Si no utiliza las herramientas de AWS, debe firmar usted mismo la solicitud. CodeBuild es compatible con Signature Version 4, un protocolo para autenticar solicitudes de API de entrada. Para obtener más información acerca de la autenticación de solicitudes, consulte la sección de [Proceso de firma Signature Version 4](#) en la [AWS Referencia general de](#).

- **Rol de IAM— Un [Rol de IAM](#)** es similar a un usuario de IAM, pero no está asociado a una determinada persona. Un rol de IAM lo habilita a obtener claves de acceso temporales que se pueden utilizar para acceder a los servicios y recursos de AWS. Los roles de IAM con credenciales temporales son útiles en las siguientes situaciones:
  - **Acceso de usuarios federados:** en lugar de crear un usuario de IAM, puede usar identidades de usuario preexistentes de AWS Directory Service, el directorio de usuarios de la empresa o un proveedor de identidad web. A estas identidades se les llama usuarios federados. AWS asigna una función a un usuario federado cuando se solicita acceso a través de un [proveedor de identidades](#). Para obtener más información acerca de los usuarios federados, consulte [Usuarios federados y roles](#) en la Guía del usuario de IAM.
  - **Acceso entre cuentas:** puede utilizar un rol de IAM en su cuenta para conceder permisos a otra cuenta de AWS con el fin de que tenga acceso a los recursos de su cuenta. Para ver un ejemplo, consulte [Tutorial: Delegar el acceso entre AWS Cuentas que utilizan roles de IAM](#) en la IAM User Guide.
  - **AWS Acceso a servicios de :—** Puede utilizar un rol de IAM de su cuenta para conceder permisos a un AWS servicio para acceder a los recursos de tu cuenta. Por ejemplo, puede crear una función que permita a Amazon Redshift obtener acceso a un bucket de S3 en su nombre y, a continuación, cargar los datos almacenados en el bucket de S3 en un clúster de Amazon Redshift. Con el fin de obtener más información, consulte [Creación de un rol para delegar permisos a un servicio de AWS](#) en la Guía del usuario de IAM.
- **Aplicaciones que se ejecutan en Amazon EC2—** En lugar de almacenar claves de acceso en la instancia Amazon EC2 para que lo usen las aplicaciones que se ejecutan en la instancia y crean AWS Solicitudes de API de, puede usar un rol de IAM para administrar credenciales temporales para estas aplicaciones. Para asignar un AWS rol en una instancia de Amazon EC2 y ponerla a disposición de todas las aplicaciones, cree un perfil asociado a dicha instancia. Un perfil de instancia contiene el rol y permite a los programas que se encuentran en ejecución en la instancia Amazon EC2 obtener credenciales temporales. Para obtener más información, consulte [Uso de roles para aplicaciones en Amazon EC2](#) en la Guía del usuario de IAM.

## Control de acceso

Puede tener credenciales válidas para autenticar las solicitudes, pero a menos que tenga permisos no podrá crear ni tener acceso a los recursos de AWS CodeBuild. Por ejemplo, debe disponer de permisos para crear, ver o eliminar proyectos de compilación y para iniciar, detener o consultar compilaciones.

En las secciones siguientes se describe cómo administrar los permisos de CodeBuild. Le recomendamos que lea primero la información general.

- [Información general sobre la administración de los permisos de acceso a los recursos de AWS CodeBuild](#) (p. 355)
- [Uso de políticas basadas en identidades para AWS CodeBuild](#) (p. 358)
- [Referencia de permisos de AWS CodeBuild](#) (p. 376)
- [Visualización de recursos en la consola](#) (p. 384)

## Información general sobre la administración de los permisos de acceso a los recursos de AWS CodeBuild

Cada recurso de AWS es propiedad de una cuenta de AWS, y los permisos para crear o tener acceso a un recurso se rigen por las políticas de permisos. Un administrador de cuentas puede asociar políticas de permisos a identidades de IAM (es decir, usuarios, grupos y funciones).

### Note

Un administrador de la cuenta (o usuario administrador) es un usuario con privilegios de administrador. Para obtener más información, consulte [Prácticas recomendadas de IAM](#) en la Guía del usuario de IAM.

Cuando concede permisos, decide quién obtiene los permisos, los recursos a los que se tiene acceso y las acciones que se pueden realizar en esos recursos.

### Temas

- [Recursos y operaciones de AWS CodeBuild](#) (p. 355)
- [Titularidad de los recursos](#) (p. 356)
- [Administración del acceso a los recursos](#) (p. 357)
- [Especificación de elementos de política: Acciones, efectos y entidades de seguridad](#) (p. 358)

## Recursos y operaciones de AWS CodeBuild

En AWS CodeBuild, el recurso principal es un proyecto de compilación. En una política, se usa un nombre de recurso de Amazon (ARN) para identificar el recurso al que se aplica la política. Las compilaciones son también recursos y tienen ARN asociados. Para obtener más información, consulte [Nombres de recursos de Amazon \(ARN\) y AWS Espacios de nombres de servicios](#) en la Referencia general de Amazon Web Services.

| Tipo de recurso         | Formato de ARN                                                                                |
|-------------------------|-----------------------------------------------------------------------------------------------|
| Proyecto de compilación | <code>arn:aws:codebuild:<i>region-ID</i>:<i>account-ID</i>:project/<i>project-name</i></code> |

| Tipo de recurso                                                                       | Formato de ARN                                                                     |
|---------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| Compilación                                                                           | <code>arn:aws:codebuild:region-ID:account-ID:build/build-ID</code>                 |
| Grupo de informes                                                                     | <code>arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name</code> |
| Informe                                                                               | <code>arn:aws:codebuild:region-ID:account-ID:report/report-ID</code>               |
| Todos los recursos CodeBuild                                                          | <code>arn:aws:codebuild:*</code>                                                   |
| Todos los recursos de CodeBuild de propiedad de la cuenta especificada en elAWSRegión | <code>arn:aws:codebuild:region-ID:account-ID:*</code>                              |

#### Note

La mayoría de los servicios de AWS tratan el carácter de dos puntos (:) o la barra inclinada (/) como el mismo carácter en los ARN. Sin embargo, CodeBuild utiliza una coincidencia exacta en las reglas y los patrones de los recursos. Asegúrese de utilizar los caracteres correctos cuando cree patrones de eventos para que coincidan con la sintaxis de ARN en el recurso.

Por ejemplo, puede indicar un proyecto de compilación específico (*myBuildProject*) en su instrucción utilizando su ARN del modo siguiente:

```
"Resource": "arn:aws:codebuild:us-east-2:123456789012:project/myBuildProject"
```

Para especificar todos los recursos, o si una acción de API no admite ARN, utilice el carácter comodín (\*) en el elemento Resource de la siguiente manera:

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

Algunas acciones de la API de CodeBuild aceptan varios recursos (por ejemplo, `BatchGetProjects`). Para especificar varios recursos en una única instrucción, separe sus ARN con comas, tal y como se indica a continuación:

```
"Resource": [
 "arn:aws:codebuild:us-east-2:123456789012:project/myBuildProject",
 "arn:aws:codebuild:us-east-2:123456789012:project/myOtherBuildProject"
]
```

CodeBuild proporciona un conjunto de operaciones para trabajar con los recursos de CodeBuild. Para ver una lista, consulte [Referencia de permisos de AWS CodeBuild](#) (p. 376).

## Titularidad de los recursos

La cuenta de AWS es la propietaria de los recursos que se crean en ella, independientemente de quién los haya creado. En concreto, el propietario de los recursos es la cuenta de AWS de la [entidad principal](#) (es decir, la cuenta raíz, un usuario de IAM o un rol de IAM) que autentica la solicitud de creación de recursos. Los siguientes ejemplos ilustran cómo funciona:

- Si utiliza las credenciales de la cuenta raíz deAWS para crear una regla, suAWS es el propietario del recurso CodeBuild.

- Si crea un usuario de IAM en su AWS cuenta y concede permisos para crear recursos de CodeBuild a ese usuario, el usuario puede crear recursos de CodeBuild. Sin embargo, tu AWS, a la que pertenece el usuario, será la propietaria de los recursos de CodeBuild.
- Si crea un rol de IAM en su AWS cuenta con permisos para crear recursos de CodeBuild, cualquier persona que pueda asumir el rol podrá crear recursos de CodeBuild. Su AWS, a la que pertenece el rol, será la propietaria de los recursos de CodeBuild.

## Administración del acceso a los recursos

Una política de permisos describe quién tiene acceso a qué recursos.

### Note

En esta sección se describe el uso de IAM en AWS CodeBuild. No se proporciona información detallada sobre el servicio de IAM. Para ver la documentación completa de IAM, consulte [¿Qué es IAM?](#) en la IAM User Guide. Para obtener información acerca de la sintaxis y descripciones de la política de IAM, consulte [AWS Referencia de políticas de IAM](#) en la IAM User Guide.

Las políticas que se asocian a una identidad de IAM se denominan políticas basadas en identidades (o políticas de IAM). Las políticas que se adjuntan a un recurso se denominan políticas basadas en recursos. CodeBuild es compatible con las políticas basadas en identidad y las políticas basadas en recursos para ciertas API de solo lectura con el fin de compartir recursos entre cuentas.

## Políticas con base en identidad

Puede asociar políticas a identidades de IAM.

- Asociar una política de permisos a un usuario o un grupo de su cuenta: para conceder permisos a un usuario para ver proyectos de compilación y otros recursos de AWS CodeBuild en la consola de AWS CodeBuild, puede asociar una política de permisos a un usuario o grupo al que pertenezca el usuario.
- Asociar una política de permisos a un rol (conceder permisos entre cuentas): puede asociar una política de permisos basada en identidad a un rol de IAM para conceder permisos entre cuentas. Por ejemplo, el administrador de la Cuenta A puede crear una función para conceder permisos entre cuentas a otra cuenta de AWS (por ejemplo, a la Cuenta B) o a un servicio de AWS, tal y como se indica a continuación:
  1. El administrador de la Cuenta A crea un rol de IAM y asocia una política de permisos a dicho rol, que concede permisos sobre los recursos de la Cuenta A.
  2. El administrador de la Cuenta A asocia una política de confianza al rol que identifica la Cuenta B como la entidad principal que puede asumir el rol.
  3. A continuación, el administrador de la Cuenta B puede delegar a cualquier usuario de la Cuenta B permisos para adoptar la función. De este modo, los usuarios de la Cuenta B podrán crear recursos y tener acceso a ellos en la Cuenta A. Si desea conceder a un servicio de AWS permisos para adoptar la función, la entidad principal de la política de confianza deberá ser también la entidad principal de un servicio de AWS.

A fin de obtener más información sobre el uso de IAM para delegar permisos, consulte [Administración de accesos](#) en la Guía del usuario de IAM.

En CodeBuild, las políticas basadas en identidad se utilizan para administrar los permisos a los recursos relacionados con el proceso de implementación. Por ejemplo, puede controlar el acceso a proyectos de compilación.

Puede crear políticas de IAM para restringir las llamadas y los recursos a los que los usuarios de su cuenta tienen acceso y, a continuación, asociar esas políticas a usuarios de IAM. Para obtener más información acerca de cómo crear funciones de IAM y para ver instrucciones de políticas de IAM de para CodeBuild,

consulte [Información general sobre la administración de los permisos de acceso a los recursos de AWS CodeBuild](#) (p. 355).

## Acceso seguro a los buckets de S3

Le recomendamos encarecidamente que incluya los siguientes permisos en el rol de IAM para verificar que el bucket de S3 asociado al proyecto de CodeBuild es de su propiedad o es propiedad de una persona de confianza. Estos permisos no se incluyen en los roles y las políticas administradas de AWS. Debe agregarlos usted.

- `s3:GetBucketAcl`
- `s3:GetBucketLocation`

Si cambia el propietario de un bucket de S3 que se utiliza en el proyecto, debe verificar que siga siendo el propietario del bucket y actualizar los permisos en el rol de IAM de si no es así. Para obtener más información, consulte [Agregar permisos de acceso de CodeBuild a un grupo de IAM o usuario de IAM](#) (p. 391) y [Cree un rol de servicio de CodeBuild](#) (p. 396).

## Especificación de elementos de política: Acciones, efectos y entidades de seguridad

Para cada recurso de AWS CodeBuild, el servicio define un conjunto de operaciones de API. Para conceder permisos para estas operaciones de API, CodeBuild define un conjunto de acciones que usted puede especificar en una política. Algunas operaciones de API pueden requerir permisos para más de una acción para poder realizar la operación de API. Para obtener más información, consulte [Recursos y operaciones de AWS CodeBuild](#) (p. 355) y [Referencia de permisos de AWS CodeBuild](#) (p. 376).

A continuación, se indican los elementos básicos de la política:

- **Recurso:** use un Nombre de recurso de Amazon (ARN) para identificar el recurso al que se aplica la política.
- **Acción:** use palabras de clave de acción para identificar las operaciones del recurso que desea permitir o denegar. Por ejemplo, el permiso `codebuild:CreateProject` concede a los usuarios permiso para realizar la operación `CreateProject`.
- **Efecto:** especifique el efecto (permitir o denegar) cuando el usuario solicite la acción específica. Si no concede acceso de forma explícita (permitir) a un recurso, el acceso se deniega implícitamente. También puede denegar explícitamente el acceso a un recurso. Puede hacer esto para asegurarse de que un usuario no pueda tener acceso a un recurso, aunque otra política le otorgue acceso.
- **Principal:** en las políticas basadas en identidad (políticas de IAM), el usuario al que se asocia esta política es la entidad principal implícita. Para las políticas basadas en recursos, debe especificar el usuario, cuenta, servicio u otra entidad que desee que reciba permisos.

Para obtener más información sobre la sintaxis y descripciones de las políticas de IAM consulte [Referencia de la política de IAM de AWS](#) de la Guía del usuario de IAM.

Para ver una tabla con todas las acciones de API de CodeBuild y los recursos a los que se aplican, consulte [la Referencia de permisos de AWS CodeBuild](#) (p. 376).

## Uso de políticas basadas en identidades para AWS CodeBuild

En este tema se ofrecen ejemplos de políticas basadas en identidad que muestran cómo un administrador de la cuenta puede asociar políticas de permisos a identidades de IAM (es decir, usuarios, grupos y roles) y, de ese modo, conceder permisos para realizar operaciones en recursos de AWS CodeBuild.

## Important

Le recomendamos que consulte primero los temas de introducción en los que se explican los conceptos básicos y las opciones disponibles para administrar el acceso a sus recursos de CodeBuild. Para obtener más información, consulte [Información general sobre la administración de los permisos de acceso a los recursos de AWS CodeBuild](#) (p. 355).

## Temas

- [Permisos necesarios para usar la consola de AWS CodeBuild](#) (p. 359)
- [Permisos necesarios para que la consola de AWS CodeBuild se conecte a los proveedores de código fuente](#) (p. 360)
- [Políticas administradas \(predefinidas\) por AWS para AWS CodeBuild](#) (p. 360)
- [CodeBuild administradas y notificaciones de](#) (p. 366)
- [Ejemplos de políticas administradas por los clientes](#) (p. 368)

A continuación se muestra un ejemplo de una política de permisos que permite a un usuario obtener información acerca de los proyectos de compilación únicamente en la región `us-east-2` de la cuenta `123456789012` para cualquier proyecto de compilación que empiece con el nombre `my`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchGetProjects",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permisos necesarios para usar la consola de AWS CodeBuild

Un usuario que utilice la consola de AWS CodeBuild debe tener un conjunto mínimo de permisos que le permita describir otros recursos de AWS en la cuenta de AWS. Debe tener permisos de los siguientes servicios:

- AWS CodeBuild
- Amazon CloudWatch
- CodeCommit (si va a almacenar el código fuente en un AWS CodeCommit(repository))
- Amazon Elastic Container Registry (Amazon ECR) (si va a utilizar un entorno de compilación basado en una imagen de Docker de un repositorio de Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS) (si va a utilizar un entorno de compilación basado en una imagen de Docker de un repositorio de Amazon ECR)
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- Amazon Simple Storage Service (Amazon S3)

Si crea una política de IAM que sea más restrictiva que los permisos mínimos necesarios, la consola no funcionará según lo previsto.

## Permisos necesarios para que la consola de AWS CodeBuild se conecte a los proveedores de código fuente

La consola de AWS CodeBuild usa las siguientes acciones API para conectar con los proveedores de código fuente (como los repositorios de GitHub).

- `codebuild:ListConnectedOAuthAccounts`
- `codebuild:ListRepositories`
- `codebuild:PersistOAuthToken`
- `codebuild:ImportSourceCredentials`

Puede asociar proveedores de código fuente (como repositorios de GitHub) a sus proyectos de compilación a través de la consola de AWS CodeBuild. Para ello, primero debe añadir las acciones API anteriores a las políticas de acceso de IAM asociadas al usuario de IAM que utiliza para obtener acceso a la AWS CodeBuild consola de .

Las acciones API `ListConnectedOAuthAccounts`, `ListRepositories` y `PersistOAuthToken` no están diseñadas para que puedan invocarse desde el código. Por lo tanto, estas acciones de API no se incluyen en la AWS CLI ni en los SDK de AWS.

## Políticas administradas (predefinidas) por AWS para AWS CodeBuild

AWS aborda muchos casos de uso comunes proporcionando políticas de IAM independientes creadas y administradas por AWS. Estas políticas administradas por AWS conceden los permisos necesarios para casos de uso comunes, lo que le evita tener que investigar los permisos que se necesitan. Las políticas administradas de CodeBuild también proporcionan permisos para llevar a cabo operaciones en otros servicios, como IAM, AWS CodeCommit, Amazon EC2, Amazon ECR, Amazon SNS y Amazon CloudWatch Events, según sea necesario para las responsabilidades de los usuarios a los que se conceda la política en cuestión. Por ejemplo, `AWSCodeBuildAdminAccesses` una política de usuario de nivel administrativo que permite a los usuarios crear y administrar reglas de eventos de CloudWatch para compilaciones de proyectos y temas de Amazon SNS para notificaciones sobre eventos relacionados con proyectos (temas cuyos nombres llevan el prefijo `arn:aws:codebuild:`), así como administrar proyectos y grupos de informes en CodeBuild. Para más información, consulte [Políticas administradas de AWS](#) en la Guía del usuario de IAM.

Las siguientes políticas administradas por AWS, que pueden asociarse a los usuarios de la cuenta, son específicas de AWS CodeBuild:

### Acceso a Admin de compilación de código AWS

Ofrece acceso completo a CodeBuild, incluidos los permisos para administrar proyectos de compilación de CodeBuild.

### Acceso para desarrolladores de AWS Code Build

Proporciona acceso a CodeBuild, pero no permite la administración de proyectos de compilación.

### Acceso de solo lectura de AWS Code Build

Proporciona acceso de solo lectura a CodeBuild.

Para tener acceso a los artefactos de salida de compilación que crea CodeBuild, debe asociar también el AWSdirectiva administrada denominada `AmazonS3ReadOnlyAccess`.

Para crear y administrar las funciones de servicio de CodeBuild, debe asociar también el AWSdirectiva administrada denominada `IAMFullAccess`.



También puede crear sus propias políticas de IAM personalizadas para conceder permisos a las acciones y recursos de CodeBuild. Puede asociar estas políticas personalizadas a los usuarios o grupos de IAM que requieran esos permisos.

#### Temas

- [AWSCodeBuildAdminAccess](#) (p. 361)
- [AWSCodeBuildDeveloperAccess](#) (p. 363)
- [AWSCodeBuildReadOnlyAccess](#) (p. 365)

## AWSCodeBuildAdminAccess

La `AWSCodeBuildAdminAccess` proporciona acceso completo a CodeBuild, incluidos los permisos para administrar proyectos de compilación de CodeBuild. Aplique esta política únicamente a los usuarios de nivel administrativo para concederles control completo sobre los proyectos de CodeBuild, grupos de informes y recursos relacionados en el AWS cuenta, incluida la capacidad de eliminar proyectos y grupos de informes.

La política `AWSCodeBuildAdminAccess` contiene la siguiente instrucción:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": [
 "codebuild:*",
 "codecommit:GetBranch",
 "codecommit:GetCommit",
 "codecommit:GetRepository",
 "codecommit:ListBranches",
 "codecommit:ListRepositories",
 "cloudwatch:GetMetricStatistics",
 "ec2:DescribeVpcs",
 "ec2:DescribeSecurityGroups",
 "ec2:DescribeSubnets",
 "ecr:DescribeRepositories",
 "ecr:ListImages",
 "elasticfilesystem:DescribeFileSystems",
 "events:DeleteRule",
 "events:DescribeRule",
 "events:DisableRule",
 "events:EnableRule",
 "events:ListTargetsByRule",
 "events:ListRuleNamesByTarget",
 "events:PutRule",
 "events:PutTargets",
 "events:RemoveTargets",
 "logs:GetLogEvents",
 "s3:GetBucketLocation",
 "s3:ListAllMyBuckets"
],
 "Effect": "Allow",
 "Resource": "*"
 },
 {
 "Action": [
 "logs>DeleteLogGroup"
],
 "Effect": "Allow",
 "Resource": "arn:aws:logs:*:*:log-group:/aws/codebuild/*:log-stream:*"
 }
]
}
```



```
"Effect": "Allow",
"Action": [
 "ssm:PutParameter"
],
"Resource": "arn:aws:ssm:*:*:parameter/CodeBuild/*"
},
{
 "Effect": "Allow",
 "Action": [
 "ssm:StartSession"
],
 "Resource": "arn:aws:ecs:*:*:task/*/*"
},
{
 "Sid": "CodeStarConnectionsReadWriteAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-connections:CreateConnection",
 "codestar-connections:DeleteConnection",
 "codestar-connections:UpdateConnectionInstallation",
 "codestar-connections:TagResource",
 "codestar-connections:UntagResource",
 "codestar-connections:ListConnections",
 "codestar-connections:ListInstallationTargets",
 "codestar-connections:ListTagsForResource",
 "codestar-connections:GetConnection",
 "codestar-connections:GetIndividualAccessToken",
 "codestar-connections:GetInstallationUrl",
 "codestar-connections:PassConnection",
 "codestar-connections:StartOAuthHandshake",
 "codestar-connections:UseConnection"
],
 "Resource": "arn:aws:codestar-connections:*:*:connection/*"
},
{
 "Sid": "CodeStarNotificationsReadWriteAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:CreateNotificationRule",
 "codestar-notifications:DescribeNotificationRule",
 "codestar-notifications:UpdateNotificationRule",
 "codestar-notifications:DeleteNotificationRule",
 "codestar-notifications:Subscribe",
 "codestar-notifications:Unsubscribe"
],
 "Resource": "*",
 "Condition": {
 "StringLike": {
 "codestar-notifications:NotificationsForResource": "arn:aws:codebuild:*"
 }
 }
},
{
 "Sid": "CodeStarNotificationsListAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:ListNotificationRules",
 "codestar-notifications:ListEventTypes",
 "codestar-notifications:ListTargets",
 "codestar-notifications:ListTagsforResource"
],
 "Resource": "*"
},
{
 "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
 "Effect": "Allow",
```

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

## AWSCodeBuildDeveloperAccess

La `AWSCodeBuildDeveloperAccess` permite acceder a toda la funcionalidad de CodeBuild y a los recursos relacionados con el proyecto y el grupo de informes. Esta política no permite a los usuarios eliminar proyectos de CodeBuild ni grupos de informes de, ni recursos relacionados en otros AWS servicios, como CloudWatch Events. Le recomendamos que aplique esta política a la mayoría de los usuarios.

La política `AWSCodeBuildDeveloperAccess` contiene la siguiente instrucción:

```
{
 "Statement": [
 {
 "Action": [
 "codebuild:StartBuild",
 "codebuild:StopBuild",
 "codebuild:StartBuildBatch",
 "codebuild:StopBuildBatch",
 "codebuild:RetryBuild",
 "codebuild:RetryBuildBatch",
 "codebuild:BatchGet*",
 "codebuild:GetResourcePolicy",
 "codebuild:DescribeTestCases",
 "codebuild:DescribeCodeCoverages",
 "codebuild:List*",
 "codecommit:GetBranch",
 "codecommit:GetCommit",
 "codecommit:GetRepository",
 "codecommit:ListBranches",
 "cloudwatch:GetMetricStatistics",
 "events:DescribeRule",
 "events:ListTargetsByRule",
 "events:ListRuleNamesByTarget",
 "logs:GetLogEvents",
 "s3:GetBucketLocation",
 "s3:ListAllMyBuckets"
],
 "Effect": "Allow",
 "Resource": "*"
 }
]
}
```

```
 },
 {
 "Effect": "Allow",
 "Action": [
 "ssm:PutParameter"
],
 "Resource": "arn:aws:ssm:*:*:parameter/CodeBuild/*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ssm:StartSession"
],
 "Resource": "arn:aws:ecs:*:*:task/*/*"
 },
 {
 "Sid": "CodeStarConnectionsUserAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-connections:ListConnections",
 "codestar-connections:GetConnection"
],
 "Resource": "arn:aws:codestar-connections:*:*:connection/*"
 },
 {
 "Sid": "CodeStarNotificationsReadWriteAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:CreateNotificationRule",
 "codestar-notifications:DescribeNotificationRule",
 "codestar-notifications:UpdateNotificationRule",
 "codestar-notifications:Subscribe",
 "codestar-notifications:Unsubscribe"
],
 "Resource": "*",
 "Condition": {
 "StringLike": {
 "codestar-notifications:NotificationsForResource": "arn:aws:codebuild:*"
 }
 }
 },
 {
 "Sid": "CodeStarNotificationsListAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:ListNotificationRules",
 "codestar-notifications:ListEventTypes",
 "codestar-notifications:ListTargets",
 "codestar-notifications:ListTagsForResource"
],
 "Resource": "*"
 },
 {
 "Sid": "SNSTopicListAccess",
 "Effect": "Allow",
 "Action": [
 "sns:ListTopics",
 "sns:GetTopicAttributes"
],
 "Resource": "*"
 },
 {
 "Sid": "CodeStarNotificationsChatbotAccess",
 "Effect": "Allow",
 "Action": [
 "chatbot:DescribeSlackChannelConfigurations"
]
 }
]
}
```

```
],
 "Resource": "*"
 }
],
"Version": "2012-10-17"
}
```

## AWSCodeBuildReadOnlyAccess

La `AWSCodeBuildReadOnlyAccess` concede acceso de solo lectura a CodeBuild y recursos relacionados en otros AWS Servicios de . Aplique esta política a los usuarios que pueden consultar y ejecutar compilaciones, consultar proyectos y consultar grupos de informes, pero que no pueden realizar ningún cambio en ellos.

La política `AWSCodeBuildReadOnlyAccess` contiene la siguiente instrucción:

```
{
 "Statement": [
 {
 "Action": [
 "codebuild:BatchGet*",
 "codebuild:GetResourcePolicy",
 "codebuild:List*",
 "codebuild:DescribeTestCases",
 "codebuild:DescribeCodeCoverages",
 "codecommit:GetBranch",
 "codecommit:GetCommit",
 "codecommit:GetRepository",
 "cloudwatch:GetMetricStatistics",
 "events:DescribeRule",
 "events:ListTargetsByRule",
 "events:ListRuleNamesByTarget",
 "logs:GetLogEvents"
],
 "Effect": "Allow",
 "Resource": "*"
 },
 {
 "Sid": "CodeStarConnectionsUserAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-connections:ListConnections",
 "codestar-connections:GetConnection"
],
 "Resource": "arn:aws:codestar-connections:*:*:connection/*"
 },
 {
 "Sid": "CodeStarNotificationsPowerUserAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:DescribeNotificationRule"
],
 "Resource": "*",
 "Condition": {
 "StringLike": {
 "codestar-notifications:NotificationsForResource": "arn:aws:codebuild:*"
 }
 }
 },
 {
 "Sid": "CodeStarNotificationsListAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:ListNotificationRules",

```

```
 "codestar-notifications:ListEventTypes",
 "codestar-notifications:ListTargets"
],
 "Resource": "*"
 }
],
"Version": "2012-10-17"
}
```

## CodeBuild administradas y notificaciones de

CodeBuild admite notificaciones, que pueden notificar a los usuarios cambios importantes en los proyectos de compilación de. Las políticas administradas de CodeBuild incluyen instrucciones de política para la funcionalidad de notificación. Para obtener más información, consulte [¿Qué son las notificaciones?](#)

### Permisos relacionados con las notificaciones en políticas administradas de acceso total

La política administrada `AWSCodeBuildFullAccess` incluye las siguientes instrucciones para permitir el acceso completo a las notificaciones. Los usuarios con esta política administrada aplicada también pueden crear y administrar temas de Amazon SNS para notificaciones, suscribirse y cancelar la suscripción a los temas, mostrar los temas que se pueden elegir como destinos para las reglas de notificación y mostrar AWS Chatbot clientes configurados para Slack.

```
{
 "Sid": "CodeStarNotificationsReadWriteAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:CreateNotificationRule",
 "codestar-notifications:DescribeNotificationRule",
 "codestar-notifications:UpdateNotificationRule",
 "codestar-notifications>DeleteNotificationRule",
 "codestar-notifications:Subscribe",
 "codestar-notifications:Unsubscribe"
],
 "Resource": "*",
 "Condition": {
 "StringLike": { "codestar-notifications:NotificationsForResource" :
"arn:aws:codebuild:*" }
 }
},
{
 "Sid": "CodeStarNotificationsListAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:ListNotificationRules",
 "codestar-notifications:ListTargets",
 "codestar-notifications:ListTagsForResource",
 "codestar-notifications:ListEventTypes"
],
 "Resource": "*"
},
{
 "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
 "Effect": "Allow",
 "Action": [
 "sns:CreateTopic",
 "sns:SetTopicAttributes"
],
 "Resource": "arn:aws:sns:*:*:codestar-notifications*"
},
{
```

```
 "Sid": "SNSTopicListAccess",
 "Effect": "Allow",
 "Action": [
 "sns:ListTopics"
],
 "Resource": "*"
},
{
 "Sid": "CodeStarNotificationsChatbotAccess",
 "Effect": "Allow",
 "Action": [
 "chatbot:DescribeSlackChannelConfigurations"
],
 "Resource": "*"
}
```

## Permisos relacionados con las notificaciones en políticas administradas de solo lectura

La política administrada `AWSCodeBuildReadOnlyAccess` incluye las siguientes instrucciones para permitir el acceso de solo lectura a las notificaciones. Los usuarios con esta política administrada aplicada pueden ver notificaciones de recursos, pero no pueden crearlas, administrarlas ni suscribirse a ellas.

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

## Permisos relacionados con las notificaciones en otras políticas administradas

La política `AWSCodeBuildDeveloperAccess` administrada incluye las siguientes instrucciones para que los usuarios puedan crear notificaciones, editarlas y suscribirse a ellas. Los usuarios no pueden eliminar reglas de notificación ni administrar etiquetas para recursos.

```
{
 "Sid": "CodeStarNotificationsReadWriteAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:CreateNotificationRule",
 "codestar-notifications:DescribeNotificationRule",
 "codestar-notifications:UpdateNotificationRule",
 "codestar-notifications:Subscribe",
 "codestar-notifications:Unsubscribe"
]
}
```

```
],
 "Resource": "*",
 "Condition": {
 "StringLike": {"codestar-notifications:NotificationsForResource" :
"arn:aws:codebuild*"}
 }
},
{
 "Sid": "CodeStarNotificationsListAccess",
 "Effect": "Allow",
 "Action": [
 "codestar-notifications:ListNotificationRules",
 "codestar-notifications:ListTargets",
 "codestar-notifications:ListTagsForResource",
 "codestar-notifications:ListEventTypes"
],
 "Resource": "*"
},
{
 "Sid": "SNSTopicListAccess",
 "Effect": "Allow",
 "Action": [
 "sns:ListTopics"
],
 "Resource": "*"
},
{
 "Sid": "CodeStarNotificationsChatbotAccess",
 "Effect": "Allow",
 "Action": [
 "chatbot:DescribeSlackChannelConfigurations"
],
 "Resource": "*"
}
```

Para obtener más información acerca de IAM y las notificaciones, consulte [Identity and Access Management en AWS CodeStar Notifications](#).

## Ejemplos de políticas administradas por los clientes

En esta sección, encontrará ejemplos de políticas de usuario que conceden permisos para acciones de AWS CodeBuild. Estas políticas funcionan cuando se utiliza la API de CodeBuild, AWS SDK, o AWS CLI. Cuando se utiliza la consola, se deben conceder permisos adicionales específicos de la consola. Para obtener información, consulte [Permisos necesarios para usar la consola de AWS CodeBuild \(p. 359\)](#).

Puede utilizar las siguientes políticas de IAM de para limitar el acceso a CodeBuild para sus usuarios y roles de IAM.

### Temas

- [Permitir que un usuario obtenga información sobre los proyectos de compilación \(p. 369\)](#)
- [Permitir que un usuario obtenga información sobre los grupos de informes \(p. 369\)](#)
- [Permitir que un usuario obtenga información sobre los informes \(p. 370\)](#)
- [Permitir que un usuario pueda crear proyectos de compilación \(p. 370\)](#)
- [Permitir que un usuario pueda crear un grupo de informes \(p. 370\)](#)
- [Permitir que un usuario pueda eliminar un grupo de informes \(p. 370\)](#)
- [Permitir que un usuario pueda eliminar un informe \(p. 371\)](#)
- [Permitir que un usuario pueda eliminar proyectos de compilación \(p. 371\)](#)
- [Permitir que un usuario pueda obtener una lista con los nombres de los proyectos de compilación \(p. 371\)](#)

- [Permitir que un usuario pueda cambiar la información de los proyectos de compilación \(p. 372\)](#)
- [Permitir que un usuario pueda cambiar un grupo de informes \(p. 372\)](#)
- [Permitir que un usuario obtenga información sobre las compilaciones \(p. 372\)](#)
- [Permitir un usuario pueda obtener una lista con los identificadores de compilación de un proyecto de compilación \(p. 373\)](#)
- [Permitir que un usuario pueda obtener una lista de identificadores de compilación \(p. 373\)](#)
- [Permitir que a un usuario pueda obtener una lista de grupos de informes \(p. 373\)](#)
- [Permitir que un usuario pueda obtener una lista de informes \(p. 373\)](#)
- [Permitir que un usuario pueda obtener una lista con los informes de un grupo de informes \(p. 374\)](#)
- [Permitir que un usuario pueda obtener una lista con los casos de prueba de un informe \(p. 374\)](#)
- [Permitir que un usuario pueda comenzar a ejecutar compilaciones \(p. 374\)](#)
- [Permitir que un usuario pueda intentar detener las compilaciones \(p. 375\)](#)
- [Permitir que un usuario pueda intentar eliminar compilaciones \(p. 375\)](#)
- [Permitir que usuario obtenga información sobre las imágenes de Docker administradas por CodeBuild \(p. 375\)](#)
- [Permitir acceso a CodeBuild aAWSservicios necesarios para crear una interfaz de red de VPC \(p. 375\)](#)
- [Utilizar una instrucción de denegación para evitar que AWS CodeBuild se desconecte de los proveedores de código fuente \(p. 376\)](#)

## Permitir que un usuario obtenga información sobre los proyectos de compilación

A continuación se muestra una instrucción de política de ejemplo que permite a un usuario obtener información acerca de los proyectos de compilación en la región `us-east-2` de la cuenta `123456789012` para cualquier proyecto de compilación que empiece con el nombre `my`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchGetProjects",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permitir que un usuario obtenga información sobre los grupos de informes

El ejemplo siguiente de instrucción de política permite a un usuario obtener información sobre grupos de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchGetReportGroups",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```



## Permitir que un usuario obtenga información sobre los informes

El ejemplo siguiente de instrucción de política permite a un usuario obtener información sobre informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchGetReports",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda crear proyectos de compilación

La siguiente instrucción de política de ejemplo permite a un usuario crear proyectos de compilación con cualquier nombre, pero solo en `us-east-2` Región para cuenta `123456789012` y solo utilizando el rol de servicio CodeBuild especificado:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:CreateProject",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::123456789012:role/CodeBuildServiceRole"
 }
]
}
```

## Permitir que un usuario pueda crear un grupo de informes

El ejemplo siguiente de la instrucción de una política permite a un usuario crear un grupo de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:CreateReportGroup",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda eliminar un grupo de informes

El ejemplo siguiente de instrucción de política permite a un usuario eliminar un grupo de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:DeleteReportGroup",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda eliminar un informe

El ejemplo siguiente de instrucción de política permite a un usuario eliminar un informe en la región us-east-2 para la cuenta 123456789012:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:DeleteReport",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda eliminar proyectos de compilación

A continuación se muestra una instrucción de política de ejemplo que permite a un usuario eliminar proyectos de compilación en la región us-east-2 de la cuenta 123456789012 para cualquier proyecto de compilación que empiece con el nombre my:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:DeleteProject",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permitir que un usuario pueda obtener una lista con los nombres de los proyectos de compilación

La siguiente instrucción de política de ejemplo permite a un usuario obtener una lista de nombres de proyectos de compilación de la misma cuenta:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListProjects",
 "Resource": "*"
 }
]
}
```

```
}
```

## Permitir que un usuario pueda cambiar la información de los proyectos de compilación

La siguiente instrucción de política de ejemplo permite a un usuario cambiar la información de los proyectos de compilación con cualquier nombre, pero solo en la región `us-east-2` de la cuenta `123456789012` y solo mediante el rol de servicio de AWS CodeBuild especificado:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:UpdateProject",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/*"
 },
 {
 "Effect": "Allow",
 "Action": "iam:PassRole",
 "Resource": "arn:aws:iam::123456789012:role/CodeBuildServiceRole"
 }
]
}
```

## Permitir que un usuario pueda cambiar un grupo de informes

El ejemplo siguiente de instrucción de política permite a un usuario cambiar un grupo de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:UpdateReportGroup",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario obtenga información sobre las compilaciones

La siguiente instrucción de política de ejemplo permite a un usuario obtener información acerca de las compilaciones en la región `us-east-2` de la cuenta `123456789012` para los proyectos de compilación denominados `my-build-project` y `my-other-build-project`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchGetBuilds",
 "Resource": [
 "arn:aws:codebuild:us-east-2:123456789012:project/my-build-project",
 "arn:aws:codebuild:us-east-2:123456789012:project/my-other-build-project"
]
 }
]
}
```

## Permitir un usuario pueda obtener una lista con los identificadores de compilación de un proyecto de compilación

La siguiente instrucción de política de ejemplo permite a un usuario obtener una lista de identificadores de compilación en la región `us-east-2` de la cuenta `123456789012` para los proyectos de compilación denominados `my-build-project` y `my-other-build-project`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListBuildsForProject",
 "Resource": [
 "arn:aws:codebuild:us-east-2:123456789012:project/my-build-project",
 "arn:aws:codebuild:us-east-2:123456789012:project/my-other-build-project"
]
 }
]
}
```

## Permitir que un usuario pueda obtener una lista de identificadores de compilación

La siguiente instrucción de política de ejemplo permite a un usuario obtener una lista de todos los identificadores de compilación de la misma cuenta:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListBuilds",
 "Resource": "*"
 }
]
}
```

## Permitir que a un usuario pueda obtener una lista de grupos de informes

El ejemplo siguiente de instrucción de política permite a un usuario obtener una lista de los grupos de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListReportGroups",
 "Resource": "*"
 }
]
}
```

## Permitir que un usuario pueda obtener una lista de informes

El ejemplo siguiente de instrucción de política permite a un usuario obtener una lista de los informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListReports",
 "Resource": "*"
 }
]
```

## Permitir que un usuario pueda obtener una lista con los informes de un grupo de informes

El ejemplo siguiente de instrucción de política permite a un usuario obtener una lista de informes de un grupos de informes en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListReportsForReportGroup",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda obtener una lista con los casos de prueba de un informe

La siguiente instrucción de política de ejemplo permite a un usuario obtener una lista de casos de prueba para un informe en la región `us-east-2` para la cuenta `123456789012`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:DescribeTestCases",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:report-group/*"
 }
]
}
```

## Permitir que un usuario pueda comenzar a ejecutar compilaciones

La siguiente instrucción de política de ejemplo permite a un usuario ejecutar compilaciones en la región `us-east-2` de la cuenta `123456789012` para proyectos de compilación que empiezan con el nombre `my`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:StartBuild",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permitir que un usuario pueda intentar detener las compilaciones

A continuación se muestra una instrucción de política de ejemplo que permite a un usuario intentar detener la ejecución de compilaciones únicamente en la región `us-east-2` de la cuenta `123456789012` para cualquier proyecto de compilación que empiece con el nombre `my`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:StopBuild",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permitir que un usuario pueda intentar eliminar compilaciones

La siguiente instrucción de la política de ejemplo permite que un usuario pueda intentar eliminar compilaciones únicamente en la región `us-east-2` de la cuenta `123456789012` para cualquier proyecto de compilación que empiece con el nombre `my`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:BatchDeleteBuilds",
 "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/my*"
 }
]
}
```

## Permitir que usuario obtenga información sobre las imágenes de Docker administradas por CodeBuild

La siguiente instrucción de política de ejemplo permite a un usuario obtener información acerca de todas las imágenes de Docker administradas por CodeBuild:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": "codebuild:ListCuratedEnvironmentImages",
 "Resource": "*"
 }
]
}
```

## Permitir acceso a CodeBuild a AWS servicios necesarios para crear una interfaz de red de VPC

La siguiente instrucción de política de ejemplo concede permiso a AWS CodeBuild para crear una interfaz de red en una VPC con dos subredes:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateNetworkInterface",
 "ec2:DescribeDhcpOptions",
 "ec2:DescribeNetworkInterfaces",
 "ec2>DeleteNetworkInterface",
 "ec2:DescribeSubnets",
 "ec2:DescribeSecurityGroups",
 "ec2:DescribeVpcs"
],
 "Resource": "*"
 },
 {
 "Effect": "Allow",
 "Action": [
 "ec2:CreateNetworkInterfacePermission"
],
 "Resource": "arn:aws:ec2:region:account-id:network-interface/*",
 "Condition": {
 "StringEquals": {
 "ec2:AuthorizedService": "codebuild.amazonaws.com"
 },
 "ArnEquals": {
 "ec2:Subnet": [
 "arn:aws:ec2:region:account-id:subnet/subnet-id-1",
 "arn:aws:ec2:region:account-id:subnet/subnet-id-2"
]
 }
 }
 }
]
```

## Utilizar una instrucción de denegación para evitar que AWS CodeBuild se desconecte de los proveedores de código fuente

La siguiente instrucción de política de ejemplo utiliza una instrucción de denegación para evitar que AWS CodeBuild se desconecte de los proveedores de código fuente. Utiliza `codebuild:DeleteOAuthToken`, que es lo contrario de `codebuild:PersistOAuthToken` y `codebuild:ImportSourceCredentials`, para establecer conexión con los proveedores de código fuente. Para obtener más información, consulte [Permisos necesarios para que la consola de AWS CodeBuild se conecte a los proveedores de código fuente](#) (p. 360).

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "codebuild:DeleteOAuthToken",
 "Resource": "*"
 }
]
}
```

## Referencia de permisos de AWS CodeBuild

Puede utilizar la siguiente tabla como referencia cuando configure [Control de acceso](#) (p. 355) y políticas de permisos de escritura que puede adjuntar a una identidad de IAM (políticas basadas en identidad).

Puede utilizar claves de condiciones generales de AWS en sus políticas de AWS CodeBuild para expresar condiciones. Para ver una lista, consulte [Claves disponibles en la IAM User Guide](#).

Las acciones se especifican en el campo `Action` de la política. Para especificar una acción, use el prefijo `codebuild:` seguido del nombre de operación de API (por ejemplo, `codebuild:CreateProject` y `codebuild:StartBuild`). Para especificar varias acciones en una única instrucción, sepárelas con comas (por ejemplo, `"Action": [ "codebuild:CreateProject", "codebuild:StartBuild" ]`).

#### Uso de caracteres comodín

Debe especificar un ARN, con o sin un carácter comodín (\*), como el valor del recurso en el campo `Resource` de la política. Puede utilizar un carácter comodín para especificar varias acciones o recursos. Por ejemplo, `codebuild:*` especifica todas las acciones de CodeBuild y `codebuild:Batch*` especifica todas las acciones CodeBuild que empiezan por la palabra `Batch`. El siguiente ejemplo otorga acceso a todos los proyectos de compilación con nombres que comienzan por `my`:

```
arn:aws:codebuild:us-east-2:123456789012:project/my*
```

### Operaciones de la API de CodeBuild y permisos necesarios para las acciones

#### BatchDeleteBuilds

Acción: `codebuild:BatchDeleteBuilds`

Necesario para eliminar compilaciones.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### Obtención de compilaciones por lotes

Acción: `codebuild:BatchGetBuilds`

Necesario para obtener información acerca de las compilaciones.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### BatchGetProjects

Acción: `codebuild:BatchGetProjects`

Necesario para obtener información acerca de los proyectos de compilación.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### Grupos de informes de obtención por lotes

Acción: `codebuild:BatchGetReportGroups`

Necesario para obtener información sobre grupos de informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### BatchGetReports

Acción: `codebuild:BatchGetReports`

Necesario para obtener información sobre informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### BatchPutTestCases<sup>1</sup>

Acción: `codebuild:BatchPutTestCases`

Necesario para crear o actualizar un informe de prueba.



Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`  
CreateProject

Acciones: `codebuild:CreateProject`, `iam:PassRole`

Necesario para crear proyectos de compilación.

Recursos:

- `arn:aws:codebuild:region-ID:account-ID:project/project-name`
- `arn:aws:iam::account-ID:role/role-name`

CreateReport <sup>1</sup>

Acción: `codebuild:CreateReport`

Necesario para crear un informe de prueba.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`  
Crear grupo de informes

Acción: `codebuild:CreateReportGroup`

Necesario para crear un grupo de informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`  
CreateWebhook

Acción: `codebuild:CreateWebhook`

Se necesita para crear un webhook.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`  
DeleteProject

Acción: `codebuild>DeleteProject`

Necesario para eliminar un proyecto de CodeBuild.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`  
DeleteReport

Acción: `codebuild>DeleteReport`

Necesario para eliminar un informe.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`  
Eliminar grupo de informes

Acción: `codebuild>DeleteReportGroup`

Necesario para eliminar un grupo de informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`  
Eliminar credenciales de origen

Acción: `codebuild>DeleteSourceCredentials`

Necesario para eliminar un conjunto de objetos `SourceCredentialsInfo` que contienen información sobre las credenciales de un repositorio de GitHub, GitHub Enterprise Server o Bitbucket.

Recurso: \*

#### DeleteWebhook

Acción: `codebuild:DeleteWebhook`

Se necesita para crear un webhook.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### Describir casos de prueba

Acción: `codebuild:DescribeTestCases`

Necesario para devolver una lista paginada de casos de prueba.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### Importar credenciales de origen

Acción: `codebuild:ImportSourceCredentials`

Necesario para importar un conjunto de objetos `SourceCredentialsInfo` que contienen información sobre las credenciales de un repositorio de GitHub, GitHub Enterprise o Bitbucket.

Recurso: \*

#### Invaldar la caché de proyectos

Acción: `codebuild:InvalidateProjectCache`

Necesario para restablecer la caché de un proyecto.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### Lista de lotes de compilación

Acción: `codebuild:ListBuildBatches`

Necesario para obtener una lista de identificadores de lote de compilación.

Recurso: \*

#### ListBuild Lotes para proyectos

Acción: `codebuild:ListBuildBatchesForProjects`

Necesario para obtener una lista de identificadores de lotes de compilaciones de un proyecto específico.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### ListBuilds

Acción: `codebuild:ListBuilds`

Necesario para obtener una lista de identificadores de compilación.

Recurso: \*

#### Compilaciones de listas para proyecto

Acción: `codebuild:ListBuildsForProject`

Necesario para obtener una lista de identificadores de compilaciones de un proyecto de compilación.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### Listar imágenes de entorno seleccionadas

Acción: `codebuild:ListCuratedEnvironmentImages`

Necesario para obtener información sobre todas las imágenes de Docker administradas por AWS CodeBuild.

Recurso: \* (es necesario, pero no hace referencia a un recurso de AWS atribuible)

#### ListProjects

Acción: `codebuild:ListProjects`

Necesario para obtener una lista de nombres de proyectos de compilación.

Recurso: \*

#### Lista de grupos de informes

Acción: `codebuild:ListReportGroups`

Necesario para obtener una lista de grupos de informes.

Recurso: \*

#### Informes de lista

Acción: `codebuild:ListReports`

Necesario para obtener una lista de informes.

Recurso: \*

#### Lista de informes para el grupo de informes

Acción: `codebuild:ListReportsForReportGroup`

Necesario para obtener una lista de informes de un grupo de informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### Volver a intentar construir

Acción: `codebuild:RetryBuild`

Necesario para volver a intentar compilaciones.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### StartBuild

Acción: `codebuild:StartBuild`

Necesario para empezar a ejecutar compilaciones.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### StopBuild

Acción: `codebuild:StopBuild`

Necesario para intentar detener la ejecución de compilaciones.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

#### UpdateProject

Acciones: `codebuild:UpdateProject`, `iam:PassRole`

Necesario para cambiar la información sobre las compilaciones.

Recursos:

- `arn:aws:codebuild:region-ID:account-ID:project/project-name`
- `arn:aws:iam::account-ID:role/role-name`

#### Actualizar visibilidad del proyecto

Acciones: `codebuild:UpdateProjectVisibility`, `iam:PassRole`

Necesario para cambiar la visibilidad pública de las compilaciones de un proyecto de.

Recursos:

- `arn:aws:codebuild:region-ID:account-ID:project/project-name`
- `arn:aws:iam::account-ID:role/role-name`

#### UpdateReport <sup>1</sup>

Acción: `codebuild:UpdateReport`

Necesario para crear o actualizar un informe de prueba.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### Grupo de informes de actualización

Acción: `codebuild:UpdateReportGroup`

Necesario para actualizar un grupo de informes.

Recurso: `arn:aws:codebuild:region-ID:account-ID:report-group/report-group-name`

#### Actualizar WebHook

Acción: `codebuild:UpdateWebhook`

Se necesita para actualizar un webhook.

Recurso: `arn:aws:codebuild:region-ID:account-ID:project/project-name`

<sup>1</sup> Se utiliza solo para permisos. No hay API para esta acción.

## Uso de etiquetas para controlar el acceso a los recursos de AWS CodeBuild

Las condiciones de las declaraciones de política de IAM forman parte de la sintaxis que puede utilizar para especificar permisos para acciones basadas en proyectos de CodeBuild. Puede crear una política que permita o deniegue acciones sobre los proyectos en función de las etiquetas asociadas a dichos proyectos y aplicar después esas políticas a los grupos de IAM que configure para administrar los usuarios de IAM. Para obtener información acerca de cómo aplicar etiquetas a un proyecto mediante la consola o AWS CLI, consulte [Creación de un proyecto de compilación en AWS CodeBuild \(p. 197\)](#). Para obtener información acerca de cómo aplicar etiquetas a través del SDK de CodeBuild, consulte [CreateProjectyEtiquetas](#) en la Referencia de la API de Code. Para obtener más información sobre cómo usar etiquetas para controlar el acceso a AWS recursos, consulte [Control del acceso a AWS Recursos que utilizan etiquetas de recursos](#) en la IAM User Guide.

#### Example Ejemplo 1: Limitar las acciones del proyecto de CodeBuild en función de etiquetas

En el ejemplo siguiente, se deniegan todas las acciones `BatchGetProjects` en los proyectos etiquetados con la clave `Environment` y el valor de clave `Production`. El administrador de un usuario debe asociar esta política de IAM, junto con la política de usuario administrada, a los usuarios de IAM no autorizados. La clave de condición `aws:ResourceTag` se utiliza para controlar el acceso a los recursos en función de sus etiquetas.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": "BatchGetProjects",
 "Resource": "arn:aws:codebuild:region-ID:account-ID:project/*",
 "Condition": {"
 "aws:ResourceTag": {"
 "key": "Environment",
 "value": "Production"
 }
 }
 }
]
}
```

```
"Statement": [
 {
 "Effect": "Deny",
 "Action": [
 "codebuild:BatchGetProjects"
],
 "Resource": "*",
 "Condition": {
 "ForAnyValue:StringEquals": {
 "aws:ResourceTag/Environment": "Production"
 }
 }
 }
]
```

#### Example Ejemplo 2: Limitar las acciones del proyecto de CodeBuild en función de etiquetas

La siguiente política deniega a los usuarios el permiso para la acción `CreateProject` si la solicitud contiene una etiqueta con la clave `Environment` y el valor de clave `Production`. Además, la política impide que estos usuarios no autorizados puedan modificar los proyectos utilizando la clave de condición `aws:TagKeys` y no permite `UpdateProject` si la solicitud contiene una etiqueta con la clave `Environment`. Un administrador debe asociar esta política de IAM, junto con la política de usuario administrada, a los usuarios que no estén autorizados para realizar estas acciones. La clave de condición `aws:RequestTag` se utiliza para controlar qué etiquetas se pueden pasar en una solicitud de IAM.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": [
 "codebuild:CreateProject"
],
 "Resource": "*",
 "Condition": {
 "ForAnyValue:StringEquals": {
 "aws:RequestTag/Environment": "Production"
 }
 }
 },
 {
 "Effect": "Deny",
 "Action": [
 "codebuild:UpdateProject"
],
 "Resource": "*",
 "Condition": {
 "ForAnyValue:StringEquals": {
 "aws:TagKeys": ["Environment"]
 }
 }
 }
]
}
```

#### Example Ejemplo 3: Denegar o permitir acciones en grupos de informes en función de etiquetas de recursos

Puede crear una política que permita o deniegue acciones sobre los recursos de CodeBuild (proyectos y grupos de informes) en función de las etiquetas asociadas con esos recursos y aplicar después esas

políticas a los grupos de IAM que configure para administrar los usuarios de IAM. Por ejemplo, puede crear una política de que deniegue todas las acciones de CodeBuild en cualquier grupo de informes con `AWSClave de etiquetaStatus` el valor clave de `Secret`, a continuación, aplique esa política al grupo de IAM que creó para desarrolladores generales (*Desarrolladores*). Tendrá que asegurarse de que los desarrolladores que trabajan en esos grupos de informes etiquetados no sean miembros de ese general *Desarrolladores* grupo, sino que pertenecen a otro grupo de IAM que no tiene aplicada la política restrictiva (`SecretDevelopers`).

En el ejemplo siguiente, se deniegan todas las acciones de CodeBuild en grupos de informes etiquetados con la clave `Status` el valor clave de `Secret`:

```
{
 "Version": "2012-10-17",
 "Statement" : [
 {
 "Effect" : "Deny",
 "Action" : [
 "codebuild:BatchGetReportGroups",
 "codebuild:CreateReportGroup",
 "codebuild>DeleteReportGroup",
 "codebuild>ListReportGroups",
 "codebuild>ListReportsForReportGroup",
 "codebuild:UpdateReportGroup"
],
 "Resource" : "*",
 "Condition" : {
 "StringEquals" : "aws:ResourceTag/Status": "Secret"
 }
 }
]
}
```

#### Example Ejemplo 4: Limitar acciones de CodeBuild a `AWSCodeBuildDeveloperAccess` en función de etiquetas de recursos

Puede crear políticas que permitan acciones de CodeBuild en todos los grupos de informes y proyectos que no estén etiquetados con etiquetas específicas. Por ejemplo, la siguiente política permite el equivalente de permisos `AWSCodeBuildDeveloperAccess` (p. 363) para todos los grupos de informes y proyectos excepto los etiquetados con las etiquetas especificadas:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "codebuild:StartBuild",
 "codebuild:StopBuild",
 "codebuild:BatchGet*",
 "codebuild:GetResourcePolicy",
 "codebuild:DescribeTestCases",
 "codebuild>List*",
 "codecommit:GetBranch",
 "codecommit:GetCommit",
 "codecommit:GetRepository",
 "codecommit>ListBranches",
 "cloudwatch:GetMetricStatistics",
 "events:DescribeRule",
 "events>ListTargetsByRule",
 "events>ListRuleNamesByTarget",
 "logs:GetLogEvents",
 "s3:GetBucketLocation",

```

```
 "s3:ListAllMyBuckets"
],
 "Resource": "*",
 "Condition": {
 "StringNotEquals": {
 "aws:ResourceTag/Status": "Secret",
 "aws:ResourceTag/Team": "Saanvi"
 }
 }
}
```

## Visualización de recursos en la consola

La consola de AWS CodeBuild requiere el permiso `ListRepositories` para mostrar una lista de repositorios para la cuenta de AWS en la región de AWS en la que se ha iniciado la sesión. La consola también incluye una función `Go to resource` (Ir al recurso) para realizar una búsqueda rápida de recursos sin distinción entre mayúsculas y minúsculas. Esta búsqueda se realiza en su cuenta de AWS en la región de AWS en la que se ha iniciado sesión. Los siguientes recursos se muestran en los siguientes servicios:

- AWS CodeBuild: proyectos de compilación
- AWS CodeCommit: repositorios
- AWS CodeDeploy: aplicaciones
- AWS CodePipeline: canalizaciones

Para realizar esta búsqueda en los recursos de todos los servicios, debe contar con los siguientes permisos:

- CodeBuild: `ListProjects`
- CodeCommit: `ListRepositories`
- CodeDeploy: `ListApplications`
- CodePipeline: `ListPipelines`

Los resultados de los recursos de un servicio no se devuelven si no tiene permisos para ese servicio. Aunque tenga permisos para ver los recursos, algunos recursos no se devolverán si hay un permiso `Deny` explícito para ver esos recursos.

## Validación de la conformidad enAWSCodeBuild

Audidores externos evalúan la seguridad y la conformidad deAWSCodeBuild como parte de variosAWSprogramas de conformidad. Estos incluyen SOC, PCI, FedRAMP, HIPAA y otros.

Para obtener una lista de AWS servicios en el ámbito de programas de cumplimiento específicos, consulte los [AWS servicios en ámbito por programa de cumplimiento](#). Para obtener información general, consulte [Programas de conformidad de AWS](#).

Puede descargar los informes de auditoría de terceros con AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#).

Su responsabilidad de conformidad al utilizar CodeBuild se determina en función de la confidencialidad de los datos, los objetivos de conformidad de su empresa, así como de la legislación y los reglamentos aplicables. Si su uso de CodeBuild está sujeto a conformidad con normas tales como HIPAA, PCI o FedRAMP,AWSproporciona recursos para ayudar con:

- [Guías de inicio rápido de seguridad y conformidad](#): estas guías de implementación tratan consideraciones sobre arquitectura y ofrecen pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- [Documento técnico sobre arquitectura para seguridad y conformidad de HIPAA](#) : en este documento técnico, se describe cómo las empresas pueden utilizar AWS para crear aplicaciones conformes con HIPAA.
- [Recursos de conformidad de AWS](#): este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- [AWS Config](#): este servicio de AWS evalúa en qué medida las configuraciones de los recursos cumplen las prácticas internas, las directrices del sector y la normativa.
- [AWS Security Hub](#): este servicio de AWS proporciona una vista integral de su estado de seguridad en AWS que lo ayuda a verificar la conformidad con los estándares y las prácticas recomendadas del sector de seguridad.

## Resiliencia en AWS CodeBuild

La infraestructura global de AWS se compone de regiones y zonas de disponibilidad de AWS. AWS Las regiones proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre zonas de disponibilidad sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre las zonas de disponibilidad y las regiones de AWS, consulte [Infraestructura global de AWS](#).

## Seguridad de la infraestructura en AWS CodeBuild

Como servicio administrado, AWS CodeBuild está protegido por el AWS procedimientos de seguridad de red globales de que se describen en la CLI de [Amazon Web Services: Información general sobre los procesos de seguridad](#) white paper.

Usa AWS llamadas a la API publicadas en para obtener acceso a CodeBuild a través de Los clientes deben ser compatibles con Transport Layer Security (TLS) 1.0 o una versión posterior. Le recomendamos TLS 1.2 o una versión posterior. Los clientes también deben ser compatibles con conjuntos de cifrado con confidencialidad directa total (PFS) tales como Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

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 puede utilizar [AWS Security Token Service](#) (AWS STS) para generar credenciales de seguridad temporales a fin de firmar solicitudes.

## Acceda a su proveedor de código fuente en CodeBuild

En GitHub o GitHub Enterprise Server, utilice un token de acceso personal para acceder al proveedor de origen. Para Bitbucket, utilice una contraseña de aplicación para tener acceso al proveedor de código fuente.



#### Temas

- [Token de acceso a GitHub Enterprise Server \(p. 386\)](#)
- [La contraseña de la aplicación Bitbucket \(p. 388\)](#)

## Token de acceso a GitHub Enterprise Server

### Requisitos previos de los tokens de acceso

Antes de comenzar, debe añadir los ámbitos de permiso adecuados para su token de acceso de GitHub.

Para GitHub, su token de acceso personal debe tener los siguientes ámbitos.

- repositorio: Concede a control total de repositorios privados.
- repo:status: Otorga acceso de lectura/escritura a los estados de confirmación de repositorios públicos y privados.
- admin:repo\_hook: Concede control total de enlaces de repositorio. Este ámbito no es necesario si su token tiene el ámbito repo.

Para obtener más información, consulte [Understanding Scopes for OAuth Apps](#) en el sitio web de GitHub.

### Connect GitHub con un token de acceso (consola)

Para utilizar la consola para conectar el proyecto a GitHub con un token de acceso, haga lo siguiente al crear un proyecto. Para obtener información, consulte [Creación de un proyecto de compilación \(consola\) \(p. 198\)](#).

1. En Source provider (Proveedor de código fuente), elija GitHub.
2. En Repository (Repositorio), seleccione Connect with a GitHub personal access token (Conectar con un token de acceso personal de GitHub).
3. En GitHub personal access token (Token de acceso personal de GitHub), escriba su token de acceso personal de GitHub.
4. Seleccione Save token (Guardar token).

### Connect GitHub con un token de acceso (CLI)

Siga estos pasos para utilizar laAWS CLIpara conectar tu proyecto a GitHub mediante un token de acceso. Para obtener información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos \(p. 402\)](#).

1. Ejecute el comando import-source-credentials:

```
aws codebuild import-source-credentials --generate-cli-skeleton
```

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

```
{
 "serverType": "server-type",
 "authType": "auth-type",
 "shouldOverwrite": "should-overwrite",
```

```
{
 "token": "token",
 "username": "username"
}
```

Sustituya lo siguiente:

- **tipo de servidor**: Valor obligatorio. El proveedor de código fuente utilizado para esta credencial. Los valores válidos son GITHUB o GITHUB\_ENTERPRISE.
  - **auth-type**: Valor obligatorio. El tipo de autenticación que se utiliza para conectarse a un repositorio de GitHub o GitHub Enterprise Server. Los valores válidos incluyen PERSONAL\_ACCESS\_TOKEN y BASIC\_AUTH. No se puede utilizar la API de CodeBuild para crear una conexión OAUTH. Debe usar la consola de CodeBuild en su lugar.
  - **debe sobrescribir**: Valor opcional. Establézcalo en `false` para impedir que se sobrescriban las credenciales de origen del repositorio. Establézcalo en `true` para sobrescribir las credenciales de origen del repositorio. El valor predeterminado es `true`.
  - **token**: Valor obligatorio. Para GitHub o GitHub Enterprise Server, se trata del token de acceso personal.
  - **nombreDeUsuario**: Valor opcional. Este parámetro se ignora para los proveedores de origen de GitHub y GitHub Enterprise Server.
2. Para conectar su cuenta con un token de acceso, cambie al directorio que contiene el archivo `import-source-credentials.json` que guardó en el paso 1 y vuelva a ejecutar el comando `import-source-credentials`.

```
aws codebuild import-source-credentials --cli-input-json file://import-source-credentials.json
```

Los datos con formato JSON aparecen en el resultado con un nombre de recurso de Amazon (ARN).

```
{
 "arn": "arn:aws:codebuild:region:account-id:token/server-type"
}
```

#### Note

Si ejecuta el comando `import-source-credentials` con el mismo tipo de servidor y tipo de autorización una segunda vez, se actualiza el token de acceso almacenado.

Una vez que su cuenta se conecta con un token de acceso, puede utilizar `create-project` para crear su proyecto de CodeBuild. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

3. Para ver los tokens de acceso conectados, ejecute el comando `list-source-credentials`.

```
aws codebuild list-source-credentials
```

Un objeto `sourceCredentialsInfos` con formato JSON aparece en el resultado:

```
{
 "sourceCredentialsInfos": [
 {
 "authType": "auth-type",
 "serverType": "server-type",
 "arn": "arn"
 }
]
}
```

El `sourceCredentialsObject` contiene una lista de información de credenciales de código fuente conectado:

- El `authType` es el tipo de autenticación que utilizan las credenciales. Este valor puede ser `OAUTH`, `BASIC_AUTH` o `PERSONAL_ACCESS_TOKEN`.
  - El `serverType` es el tipo de proveedor de código fuente. Este valor puede ser `GITHUB`, `GITHUB_ENTERPRISE` o `BITBUCKET`.
  - El `arn` es el ARN del token.
4. Para desconectarse de un proveedor de código fuente y eliminar sus tokens de acceso, ejecute el comando `delete-source-credentials` con su ARN.

```
aws codebuild delete-source-credentials --arn arn-of-your-credentials
```

Los datos con formato JSON se devuelven con un ARN de las credenciales eliminadas.

```
{
 "arn": "arn:aws:codebuild:region:account-id:token/server-type"
}
```

## La contraseña de la aplicación Bitbucket

### Requisitos previos para contraseñas de aplicaciones

Antes de comenzar, debe añadir los ámbitos de permiso adecuados para su contraseña de aplicación Bitbucket.

Para Bitbucket, la contraseña de aplicación debe tener los siguientes ámbitos.

- `repositorio:leer`: Concede acceso de lectura a todos los repositorios a los que tiene acceso el usuario de autorización.
- `solicitud de extracción: leer`: Concede acceso de lectura a las solicitudes de extracción. Si su proyecto tiene un webhook de Bitbucket, entonces la contraseña de su aplicación debe tener este ámbito.
- `webhook`: Concede acceso a webhooks. Si su proyecto tiene una operación de webhook, entonces la contraseña de su aplicación debe tener este ámbito.

Para obtener más información, consulte [Scopes for Bitbucket Cloud REST API](#) y [OAuth on Bitbucket Cloud](#) en el sitio web de Bitbucket.

### Connect Bitbucket con una contraseña de aplicación (consola)

Para utilizar la consola para conectar el proyecto a Bitbucket con una contraseña de aplicación, haga lo siguiente al crear un proyecto. Para obtener información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).

1. En Source provider (Proveedor de código fuente), elija Bitbucket.

#### Note

CodeBuild no admite Bitbucket Server.

2. En Repository (Repositorio), elija Connect with a Bitbucket app password (Conectar con una contraseña de aplicación de Bitbucket).
3. En Bitbucket username (Nombre de usuario de Bitbucket), escriba su nombre de usuario de Bitbucket.

4. En Bitbucket app password (Contraseña de aplicación de Bitbucket), escriba su contraseña de aplicación de Bitbucket.
5. Seleccione Save Bitbucket credentials (Guardar credenciales de Bitbucket).

## Connect Bitbucket con una contraseña de aplicación (CLI)

Siga estos pasos para utilizar la AWS CLI para conectar tu proyecto a Bitbucket mediante una contraseña de aplicación. Para obtener información sobre cómo usar la AWS CLI con AWS CodeBuild, consulte [Referencia de la línea de comandos \(p. 402\)](#).

1. Ejecute el comando `import-source-credentials`:

```
aws codebuild import-source-credentials --generate-cli-skeleton
```

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

```
{
 "serverType": "BITBUCKET",
 "authType": "auth-type",
 "shouldOverwrite": "should-overwrite",
 "token": "token",
 "username": "username"
}
```

Sustituya lo siguiente:

- **auth-type**: Valor obligatorio. El tipo de autenticación que se utiliza para conectarse a un repositorio de Bitbucket. Los valores válidos incluyen `PERSONAL_ACCESS_TOKEN` y `BASIC_AUTH`. No se puede utilizar la API de CodeBuild para crear una conexión `OAuth`. Debe usar la consola de CodeBuild en su lugar.
  - **debe sobrescribir**: Valor opcional. Establézcalo en `false` para impedir que se sobrescriban las credenciales de origen del repositorio. Establézcalo en `true` para sobrescribir las credenciales de origen del repositorio. El valor predeterminado es `true`.
  - **token**: Valor obligatorio. Para Bitbucket, es la contraseña de la aplicación.
  - **nombreDeUsuario**: Valor opcional. El nombre de usuario de Bitbucket cuando `authType` es `BASIC_AUTH`. Este parámetro se omite para otros tipos de proveedores de código fuente o conexiones.
2. Para conectar su cuenta con una contraseña de aplicación, cambie al directorio que contiene el `import-source-credentials.json` que guardó en el paso 1 y ejecute `import-source-credentials` de nuevo.

```
aws codebuild import-source-credentials --cli-input-json file://import-source-credentials.json
```

Los datos con formato JSON aparecen en el resultado con un nombre de recurso de Amazon (ARN).

```
{
 "arn": "arn:aws:codebuild:region:account-id:token/server-type"
}
```

## Note

Si ejecuta el comando `import-source-credentials` con el mismo tipo de servidor y tipo de autorización una segunda vez, se actualiza el token de acceso almacenado.

Una vez que su cuenta se conecta con una contraseña de aplicación, puede utilizar `create-project` para crear su proyecto de CodeBuild. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

3. Para ver las contraseñas de la aplicación conectada, ejecute el comando `list-source-credentials`.

```
aws codebuild list-source-credentials
```

Un objeto `sourceCredentialsInfos` con formato JSON aparece en el resultado:

```
{
 "sourceCredentialsInfos": [
 {
 "authType": "auth-type",
 "serverType": "BITBUCKET",
 "arn": "arn"
 }
]
}
```

El `sourceCredentialsObject` contiene una lista de información de credenciales de código fuente conectado:

- El `authType` es el tipo de autenticación que utilizan las credenciales. Este valor puede ser `OAUTH`, `BASIC_AUTH` o `PERSONAL_ACCESS_TOKEN`.
  - El `arn` es el ARN del token.
4. Para desconectarse de un proveedor de código fuente y eliminar su contraseña de aplicación, ejecute el comando `delete-source-credentials` con su ARN.

```
aws codebuild delete-source-credentials --arn arn-of-your-credentials
```

Los datos con formato JSON se devuelven con un ARN de las credenciales eliminadas.

```
{
 "arn": "arn:aws:codebuild:region:account-id:token/server-type"
}
```

# Temas avanzados

Esta sección contiene varios temas avanzados que son útiles para los usuarios de AWS CodeBuild con más experiencia.

## Temas

- [Configuración avanzada](#) (p. 391)
- [Referencia de la línea de comandos de AWS CodeBuild](#) (p. 402)
- [Referencia de los SDK y las herramientas de AWS para AWS CodeBuild](#) (p. 403)
- [Especificación del punto de enlace de AWS CodeBuild](#) (p. 404)
- [Ejecutar AWS CodeBuild directamente](#) (p. 406)
- [Usar AWS CodePipeline con AWS CodeBuild para probar el código y ejecutar compilaciones](#) (p. 407)
- [Uso de AWS CodeBuild con Jenkins](#) (p. 420)
- [Uso de AWS CodeBuild con Codecov](#) (p. 422)
- [Usar AWS CodeBuild con aplicaciones sin servidor](#) (p. 425)

## Configuración avanzada

Si sigue los pasos de [Primeros pasos con la consola](#) (p. 5) para acceder a AWS CodeBuild por primera vez, lo más probable es que no necesite la información de este tema. Sin embargo, si sigue usando CodeBuild, es posible que quiera realizar algunas acciones, como conceder a los grupos de IAM y usuarios de su organización acceso a CodeBuild, modificar las funciones de servicio existentes en IAM o AWS KMS keys para acceder a CodeBuild, o configure el AWS CLI en las estaciones de trabajo de su organización para acceder a CodeBuild. En este tema se describe cómo realizar los pasos de configuración relacionados.

En él se presupone que ya tiene una cuenta de AWS. No obstante, si todavía no tiene una, vaya a <http://aws.amazon.com>, elija iniciar sesión en la consola y siga las instrucciones online.

## Temas

- [Agregar permisos de acceso de CodeBuild a un grupo de IAM o usuario de IAM](#) (p. 391)
- [Cree un rol de servicio de CodeBuild](#) (p. 396)
- [Crear y configurar una clave administrada por el cliente para CodeBuild](#) (p. 400)
- [Instalación y configuración de AWS CLI](#) (p. 402)

## Agregar permisos de acceso de CodeBuild a un grupo de IAM o usuario de IAM

Para obtener acceso a AWS CodeBuild con un grupo de IAM o un usuario de IAM, debe agregar permisos de acceso. En esta sección se describe cómo hacer esto con la consola de IAM o la AWS CLI.

Si accederá a CodeBuild con su AWS (no recomendado) o un usuario administrador de IAM en su AWS. Sin embargo, no es necesario que siga estas instrucciones.

Para obtener más información sobre AWS cuentas raíz y usuarios de IAM administrador, consulte [El usuario raíz de la cuenta de AWS](#) y [Creación del primer grupo y usuario administrador de IAM](#) en la IAM User Guide.

Para añadir permisos de acceso de CodeBuild a un grupo de IAM o a un usuario de IAM (consola)

1. Abra la consola de IAM en <https://console.aws.amazon.com/iam/>.

Debe haber iniciado sesión en la AWS Management Console con alguna de las entidades siguientes:

- Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [El usuario raíz de la cuenta de la IAM User Guide](#).
- Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la IAM User Guide.
- Un usuario de IAM en su AWS con permiso para realizar el siguiente conjunto mínimo de acciones:

```
iam:AttachGroupPolicy
iam:AttachUserPolicy
iam:CreatePolicy
iam:ListAttachedGroupPolicies
iam:ListAttachedUserPolicies
iam:ListGroups
iam:ListPolicies
iam:ListUsers
```

Para obtener más información, consulte [Información general sobre las políticas de IAM](#) en la IAM User Guide.

2. En el panel de navegación, seleccione Políticas.
3. Para añadir un conjunto personalizado de AWS CodeBuild para obtener permisos de acceso a un grupo de IAM o a un usuario de IAM, vaya al paso 4 de este procedimiento.

Para añadir un conjunto predeterminado de permisos de acceso de CodeBuild a un grupo de IAM o a un usuario de IAM, elija Tipo de política, AWS Administrador y luego haga lo siguiente:

- Para añadir permisos de acceso completo a CodeBuild, seleccione la casilla denominada `awscodebuildAdminAccess`, elija Acciones de política elija y luego seleccione `Attach`. Active la casilla situada junto al grupo de IAM o usuario de IAM de destino y, a continuación, elija Asociación de políticas. Repita esta operación para las políticas denominadas `AmazonS3ReadOnlyAccess` e `IAMFullAccess`.
- Para añadir permisos de acceso a CodeBuild para todo salvo para administrar proyectos de compilación, seleccione la casilla denominada `AWSCodeBuildDeveloperAccess`, elija Acciones de política elija y luego seleccione `Attach`. Active la casilla situada junto al grupo de IAM o usuario de IAM de destino y, a continuación, elija Asociación de políticas. Repita esta operación para la política `AmazonS3ReadOnlyAccess`.
- Para añadir permisos de solo lectura a CodeBuild, seleccione las casillas `awscodebuildReadOnlyAccess`. Active la casilla situada junto al grupo de IAM o usuario de IAM de destino y, a continuación, elija Asociación de políticas. Repita esta operación para la política `AmazonS3ReadOnlyAccess`.

Habrá añadido un conjunto predeterminado de permisos de acceso de CodeBuild a un grupo de IAM o a un usuario de IAM. Omita el resto de los pasos de este procedimiento.

4. Elija Create Policy (Crear política).
5. En la página Create Policy, junto a Create Your Own Policy, elija Select.
6. En la página Review Policy (Revisar política), en Policy Name (Nombre de política), escriba un nombre para la política (por ejemplo, **CodeBuildAccessPolicy**). Si elige otro nombre, no olvide utilizarlo durante todo este procedimiento.
7. En Policy Document (Documento de la política), escriba lo siguiente y elija Create Policy (Crear política).

```
{
 "Version": "2012-10-17",
 "Statement": [
```

```
{
 "Sid": "CodeBuildAccessPolicy",
 "Effect": "Allow",
 "Action": [
 "codebuild:*"
],
 "Resource": "*"
},
{
 "Sid": "CodeBuildRolePolicy",
 "Effect": "Allow",
 "Action": [
 "iam:PassRole"
],
 "Resource": "arn:aws:iam::account-ID:role/role-name"
},
{
 "Sid": "CloudWatchLogsAccessPolicy",
 "Effect": "Allow",
 "Action": [
 "logs:FilterLogEvents",
 "logs:GetLogEvents"
],
 "Resource": "*"
},
{
 "Sid": "S3AccessPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:CreateBucket",
 "s3:GetObject",
 "s3:List*",
 "s3:PutObject"
],
 "Resource": "*"
},
{
 "Sid": "S3BucketIdentity",
 "Effect": "Allow",
 "Action": [
 "s3:GetBucketAcl",
 "s3:GetBucketLocation"
],
 "Resource": "*"
}
]
```

#### Note

Esta política permite el acceso a todas las acciones de CodeBuild y a un número potencialmente grande de AWS de AWS. Para restringir los permisos a acciones de CodeBuild específicas, cambie el valor de `codebuild:*` en la instrucción de política de CodeBuild. Para obtener más información, consulte [Identity and Access Management \(p. 353\)](#). Para restringir el acceso a recursos de AWS específicos, cambie el valor del objeto `Resource`. Para obtener más información, consulte [Identity and Access Management \(p. 353\)](#). La `CodeBuildRolePolicy` para permitir que se cree o modifique un proyecto de compilación.

8. En el panel de navegación, elija Groups o Users.
9. En la lista de grupos o usuarios, elija el nombre del grupo de IAM o usuario de IAM al que desea añadir permisos de acceso de CodeBuild.



10. Si se trata de un grupo, en la página de configuración del grupo, en la pestaña Permissions (Permisos), expanda Managed Policies (Políticas administradas) y elija Attach Policy (Asociar política).

Para un usuario, en la página de configuración del usuario, en la pestaña Permissions, seleccione Add permissions.

11. En el caso de un grupo, en la Asociación de políticasPágina, seleccioneCodeBuildAccessPolicyy elija y luego seleccioneAsociación de políticas.

Para un usuario, en laAgregar permisosEn la página, elijaAsociación de políticas existentes directamente. SelectCodeBuildAccessPolicy, elijaSiguiente: Review (Revisar)elija y luego seleccioneAgregar permisos.

Para agregar permisos de acceso de CodeBuild a un grupo de IAM o usuario de IAM (AWS CLI)

1. Asegúrese de que ha configurado laAWS CLI conAWS Clave de acceso yAWSUna clave de acceso secreta de correspondientes a alguna de las entidades de IAM, tal y como se ha descrito en el procedimiento anterior. Para obtener más información, consulte [Configuración inicial de la AWS Command Line Interface](#) en la Guía del usuario de AWS Command Line Interface.
2. Para añadir un conjunto personalizado deAWS CodeBuildPara obtener permisos de acceso a un grupo de IAM o a un usuario de IAM, vaya al paso 3 de este procedimiento.

Para añadir un conjunto predeterminado de permisos de acceso de CodeBuild a un grupo de IAM o a un usuario de IAM, proceda del modo siguiente:

Ejecute uno de los siguientes comandos, en función de si desea añadir permisos a un usuario de IAM o a un grupo de IAM:

```
aws iam attach-group-policy --group-name group-name --policy-arn policy-arn
aws iam attach-user-policy --user-name user-name --policy-arn policy-arn
```

Debe ejecutar el comando tres veces, reemplazando*group name*o*user name*con el nombre de grupo de IAM o el nombre de usuario de IAM, y reemplazar*policy-arn*Una vez para cada una de las siguientes ARN de política:

- Para añadir permisos de acceso completo a CodeBuild, use los siguientes ARN de política:
  - `arn:aws:iam::aws:policy/AWSCodeBuildAdminAccess`
  - `arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess`
  - `arn:aws:iam::aws:policy/IAMFullAccess`
- Para añadir permisos de acceso a CodeBuild para todo salvo para administrar proyectos de compilación, use los siguientes ARN de política:
  - `arn:aws:iam::aws:policy/AWSCodeBuildDeveloperAccess`
  - `arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess`
- Para añadir permisos de solo lectura a CodeBuild, use los siguientes ARN de política:
  - `arn:aws:iam::aws:policy/AWSCodeBuildReadOnlyAccess`
  - `arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess`

Habrá añadido un conjunto predeterminado de permisos de acceso de CodeBuild a un grupo de IAM o a un usuario de IAM. Omita el resto de los pasos de este procedimiento.

3. En un directorio vacío de la estación de trabajo o la instancia local en la que esté instalada la AWS CLI, cree un archivo denominado `put-group-policy.json` o `put-user-policy.json`. Si elige otro nombre de archivo, no olvide utilizarlo durante todo este procedimiento.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "CodeBuildAccessPolicy",
 "Effect": "Allow",
 "Action": [
 "codebuild:*"
],
 "Resource": "*"
 },
 {
 "Sid": "CodeBuildRolePolicy",
 "Effect": "Allow",
 "Action": [
 "iam:PassRole"
],
 "Resource": "arn:aws:iam::account-ID:role/role-name"
 },
 {
 "Sid": "CloudWatchLogsAccessPolicy",
 "Effect": "Allow",
 "Action": [
 "logs:FilterLogEvents",
 "logs:GetLogEvents"
],
 "Resource": "*"
 },
 {
 "Sid": "S3AccessPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:CreateBucket",
 "s3:GetObject",
 "s3:List*",
 "s3:PutObject"
],
 "Resource": "*"
 },
 {
 "Sid": "S3BucketIdentity",
 "Effect": "Allow",
 "Action": [
 "s3:GetBucketAcl",
 "s3:GetBucketLocation"
],
 "Resource": "*"
 }
]
}
```

#### Note

Esta política permite el acceso a todas las acciones de CodeBuild y a un número potencialmente grande de AWS de AWS. Para restringir los permisos a acciones de CodeBuild específicas, cambie el valor de `codebuild:*` en la instrucción de política de CodeBuild. Para obtener más información, consulte [Identity and Access Management \(p. 353\)](#). Para restringir el acceso a recursos de AWS específicos, cambie el valor del objeto `Resource` relacionado. Para obtener más información, consulte [Identity and Access Management \(p. 353\)](#) o la documentación de seguridad del servicio de AWS específico. La `CodeBuildRolePolicy` para permitir que se cree o modifique un proyecto de compilación.

4. Cambie al directorio donde ha guardado el archivo y, a continuación, ejecute uno de los siguientes comandos. Puede utilizar diferentes valores para `CodeBuildGroupAccessPolicy` y `CodeBuildUserAccessPolicy`. Si emplea valores diferentes, asegúrese de usarlos aquí.

Para un grupo de IAM:

```
aws iam put-group-policy --group-name group-name --policy-name
CodeBuildGroupAccessPolicy --policy-document file://put-group-policy.json
```

Para un usuario de IAM:

```
aws iam put-user-policy --user-name user-name --policy-name CodeBuildUserAccessPolicy
--policy-document file://put-user-policy.json
```

En los comandos anteriores, reemplace *group name* o *user name* con el nombre del grupo de IAM de destino o del usuario de IAM.

## Cree un rol de servicio de CodeBuild

Necesita un AWS CodeBuild para que CodeBuild pueda interactuar con AWS Servicios de en su nombre. Puede crear un rol de servicio CodeBuild mediante el comando CodeBuild o AWS CodePipeline Consola de: Para obtener información, consulte:

- [Creación de un proyecto de compilación \(consola\) \(p. 198\)](#)
- [Crear una canalización que use CodeBuild \(consola de CodePipeline\) \(p. 409\)](#)
- [Agregar una acción de compilación de CodeBuild a una canalización \(consola de CodePipeline\) \(p. 415\)](#)
- [Cambiar la configuración de un proyecto de compilación \(consola\) \(p. 254\)](#)

Si no tiene intención de utilizar estas consolas, en esta sección se describe cómo crear un rol de servicio de CodeBuild con la consola de IAM o la AWS CLI.

### Note

El rol de servicio que se describe en esta página contiene una política que concede los permisos mínimos necesarios para utilizar CodeBuild. Es posible que tenga que añadir permisos adicionales en función del caso de uso. Por ejemplo, si desea utilizar CodeBuild con Amazon Virtual Private Cloud, el rol de servicio que va a crear requiere los permisos de la siguiente política: [Cree un rol de servicio de CodeBuild \(p. 396\)](#).

Para crear un rol de servicio de CodeBuild (consola)

1. Abra la consola de IAM en <https://console.aws.amazon.com/iam/>.

Debe haber iniciado sesión en la consola con alguna de las entidades siguientes:

- Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [El usuario raíz de la cuenta de en la IAM User Guide](#).
- Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM en la IAM User Guide](#).
- Un usuario de IAM en su AWS con permiso para realizar el siguiente conjunto mínimo de acciones:

```
iam:AddRoleToInstanceProfile
iam:AttachRolePolicy
```

```
iam:CreateInstanceProfile
iam:CreatePolicy
iam:CreateRole
iam:GetRole
iam:ListAttachedRolePolicies
iam:ListPolicies
iam:ListRoles
iam:PassRole
iam:PutRolePolicy
iam:UpdateAssumeRolePolicy
```

Para obtener más información, consulte [Información general sobre las políticas de IAM](#) en la IAM User Guide.

2. En el panel de navegación, seleccione Políticas.
3. Elija Create Policy (Crear política).
4. En la página Create Policy, elija JSON.
5. En la política JSON, escriba lo siguiente y elija Review Policy (Consultar política):

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "CloudWatchLogsPolicy",
 "Effect": "Allow",
 "Action": [
 "logs:CreateLogGroup",
 "logs:CreateLogStream",
 "logs:PutLogEvents"
],
 "Resource": "*"
 },
 {
 "Sid": "CodeCommitPolicy",
 "Effect": "Allow",
 "Action": [
 "codecommit:GitPull"
],
 "Resource": "*"
 },
 {
 "Sid": "S3GetObjectPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:GetObject",
 "s3:GetObjectVersion"
],
 "Resource": "*"
 },
 {
 "Sid": "S3PutObjectPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:PutObject"
],
 "Resource": "*"
 },
 {
 "Sid": "ECRPullPolicy",
 "Effect": "Allow",
 "Action": [
 "ecr:BatchCheckLayerAvailability",
 "ecr:GetDownloadUrlForLayer",
```

```
 "ecr:BatchGetImage"
],
 "Resource": "*"
 },
 {
 "Sid": "ECRAuthPolicy",
 "Effect": "Allow",
 "Action": [
 "ecr:GetAuthorizationToken"
],
 "Resource": "*"
 },
 {
 "Sid": "S3BucketIdentity",
 "Effect": "Allow",
 "Action": [
 "s3:GetBucketAcl",
 "s3:GetBucketLocation"
],
 "Resource": "*"
 }
]
}
```

#### Note

Esta política contiene instrucciones que permiten tener acceso a número potencialmente grande de recursos de AWS. Para restringir el acceso de AWS CodeBuild a recursos de AWS específicos, cambie el valor de la matriz `Resource`. Para obtener más información, consulte la documentación de seguridad del servicio de AWS.

6. En la página Review Policy (Consultar política), en Policy Name (Nombre de la política), escriba un nombre para la política (por ejemplo, **CodeBuildServiceRolePolicy**) y elija Create policy (Crear política).

#### Note

Si elige otro nombre, no olvide utilizarlo durante todo este procedimiento.

7. Seleccione Roles en el panel de navegación.
8. Elija Create role.
9. En la página Creación de rol, en `AWSService` (Servicio) ya seleccionado, elija `CodeBuild` y luego seleccione `Permissions`.
10. En la página Asociación de políticas de permisos, seleccione `CodeBuildServiceRolePolicy` y luego seleccione `Consulte`.
11. En la página Create role and review (Crear función y revisar), en Role name (Nombre del rol), escriba un nombre para el rol (por ejemplo, **CodeBuildServiceRole**) y elija Create role (Crear rol).

#### Para crear un rol de servicio de CodeBuild (AWS CLI)

1. Asegúrese de que ha configurado la AWS CLI con una clave de acceso y una clave de acceso secreta de correspondientes a alguna de las entidades de IAM, tal y como se ha descrito en el procedimiento anterior. Para obtener más información, consulte [Configuración inicial de la AWS Command Line Interface](#) en la Guía del usuario de AWS Command Line Interface.
2. En un directorio vacío de la estación de trabajo o la instancia local donde se ha instalado la AWS CLI, cree dos archivos llamados `create-role.json` y `put-role-policy.json`. Si elige nombres de archivo diferentes, no olvide utilizarlos durante todo este procedimiento.

`create-role.json`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Principal": {
 "Service": "codebuild.amazonaws.com"
 },
 "Action": "sts:AssumeRole"
 }
]
}
```

put-role-policy.json:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "CloudWatchLogsPolicy",
 "Effect": "Allow",
 "Action": [
 "logs:CreateLogGroup",
 "logs:CreateLogStream",
 "logs:PutLogEvents"
],
 "Resource": "*"
 },
 {
 "Sid": "CodeCommitPolicy",
 "Effect": "Allow",
 "Action": [
 "codecommit:GitPull"
],
 "Resource": "*"
 },
 {
 "Sid": "S3GetObjectPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:GetObject",
 "s3:GetObjectVersion"
],
 "Resource": "*"
 },
 {
 "Sid": "S3PutObjectPolicy",
 "Effect": "Allow",
 "Action": [
 "s3:PutObject"
],
 "Resource": "*"
 },
 {
 "Sid": "S3BucketIdentity",
 "Effect": "Allow",
 "Action": [
 "s3:GetBucketAcl",
 "s3:GetBucketLocation"
],
 "Resource": "*"
 }
]
}
```

### Note

Esta política contiene instrucciones que permiten tener acceso a número potencialmente grande de recursos de AWS. Para restringir el acceso de AWS CodeBuild a recursos de AWS específicos, cambie el valor de la matriz `Resource`. Para obtener más información, consulte la documentación de seguridad del servicio de AWS.

3. Cambie al directorio donde guardó los archivos anteriores y, a continuación, ejecute los dos comandos siguientes, uno cada vez, en este orden. Puede utilizar valores diferentes para `CodeBuildServiceRole` y `CodeBuildServiceRolePolicy`, pero no olvide emplearlos aquí.

```
aws iam create-role --role-name CodeBuildServiceRole --assume-role-policy-document
file://create-role.json
```

```
aws iam put-role-policy --role-name CodeBuildServiceRole --policy-name
CodeBuildServiceRolePolicy --policy-document file://put-role-policy.json
```

## Crear y configurar una clave administrada por el cliente para CodeBuild

Para `AWS CodeBuild` para que cifre sus artefactos de salida de la compilación, necesita acceso a una clave KMS. De forma predeterminada, `CodeBuild` utiliza la Clave administrada por `AWS` para `Amazon S3` en su `AWS account`.

Si no desea utilizar la Clave administrada por AWS , debe crear y configurar usted mismo una clave administrada por el cliente. En esta sección se describe cómo hacer esto con la consola de IAM.

Para obtener información sobre las claves administradas por el cliente, consulte [AWS Key Management ServiceConceptos](#) y [Creación de claves](#) en la [AWS KMSGuía para desarrolladores](#).

Para configurar una clave administrada por el cliente para que la use CodeBuild, siga las instrucciones que se indican en la sección «Cómo modificar una política de claves» del tema [Modificación de una política de claves](#) en la AWS KMS Guía para desarrolladores. A continuación, añada las siguientes instrucciones (entre **### BEGIN ADDING STATEMENTS HERE ###** y **### END ADDING STATEMENTS HERE ###**) a la política de la clave. Los puntos suspensivos ( . . . ) se usan por motivos de brevedad y para ayudarle a encontrar el lugar donde debe añadir las instrucciones. No elimine ninguna instrucción ni incluya estos puntos suspensivos en la política de la clave.

```
{
 "Version": "2012-10-17",
 "Id": "...",
 "Statement": [
 ### BEGIN ADDING STATEMENTS HERE ###
 {
 "Sid": "Allow access through Amazon S3 for all principals in the account that are
authorized to use Amazon S3",
 "Effect": "Allow",
 "Principal": {
 "AWS": "*"
 },
 "Action": [
 "kms:Encrypt",
 "kms:Decrypt",
 "kms:ReEncrypt*",
 "kms:GenerateDataKey*",

```

```
 "kms:DescribeKey"
],
 "Resource": "*",
 "Condition": {
 "StringEquals": {
 "kms:ViaService": "s3.region-ID.amazonaws.com",
 "kms:CallerAccount": "account-ID"
 }
 }
},
{
 "Effect": "Allow",
 "Principal": {
 "AWS": "arn:aws:iam::account-ID:role/CodeBuild-service-role"
 },
 "Action": [
 "kms:Encrypt",
 "kms:Decrypt",
 "kms:ReEncrypt*",
 "kms:GenerateDataKey*",
 "kms:DescribeKey"
],
 "Resource": "*"
},
END ADDING STATEMENTS HERE
{
 "Sid": "Enable IAM User Permissions",
 ...
},
{
 "Sid": "Allow access for Key Administrators",
 ...
},
{
 "Sid": "Allow use of the key",
 ...
},
{
 "Sid": "Allow attachment of persistent resources",
 ...
}
]
```

- *Id. de región* representa el ID de la AWS en la región de donde se encuentran los buckets de Amazon S3 asociados a CodeBuild (por ejemplo, *us-east-1*).
- *ID de la cuenta de* representa el ID de la AWS Cuenta de que es propietaria de la clave administrada por el cliente.
- *Función de servicio de CodeBuild*—representa el nombre de la función de servicio de CodeBuild que creó o identificó anteriormente en este tema.

#### Note

Para crear o configurar una clave administrada por el cliente a través de la consola de IAM, primero debe iniciar sesión en la AWS Management Console con una de las operaciones siguientes:

- Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [El usuario raíz de la cuenta de](#) en la IAM User Guide.
- Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la IAM User Guide.



- Un usuario de IAM en su AWS con permiso para crear o modificar la clave administrada por el cliente. Para obtener más información, consulte [Permisos necesarios para usar la consola de AWS KMS](#) en la Guía para desarrolladores de AWS KMS.

## Instalación y configuración de AWS CLI

Para obtener acceso a AWS CodeBuild, puede utilizar la AWS CLI (o en lugar de) la consola de CodeBuild, la consola de CodePipeline o la AWS SDK de. Para instalar y configurar la AWS CLI, consulte [Configuración inicial de AWS Command Line Interface](#) en la AWS Command Line Interface Guía del usuario de.

1. Ejecute el siguiente comando para confirmar si la instalación de la AWS CLI es compatible con CodeBuild:

```
aws codebuild list-builds
```

Si el comando se ejecuta correctamente, aparecerá información similar a la siguiente en el resultado:

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

Los corchetes vacíos indican que aún no ha ejecutado ninguna compilación.

2. Si se produce un error, debe desinstalar la versión actual de la AWS CLI y, a continuación, instalar la versión más reciente. Para obtener más información, consulte el tema acerca de [cómo desinstalar la AWS CLI](#) y [cómo instalar la AWS Command Line Interface](#) en la Guía del usuario de AWS Command Line Interface.

## Referencia de la línea de comandos de AWS CodeBuild

La AWS CLI dispone de comandos para automatizar AWS CodeBuild. Utilice la información de este tema como complemento de la [AWS Command Line Interface Guía del usuario](#) y la [AWS CLI Referencia de la para AWS CodeBuild](#).

¿No es lo que está buscando? Si desea usar la AWS SDK para llamar a CodeBuild, consulte la [Referencia de los SDK y las herramientas de AWS](#) (p. 403).

Para utilizar la información de este tema, debe haber instalado la AWS CLI y lo configuró para su uso con CodeBuild, como se describe en [Instalación y configuración de AWS CLI](#) (p. 402).

Para utilizar la AWS CLI para especificar el punto de enlace de CodeBuild, consulte [Especificación del punto de enlace de AWS CodeBuild \(AWS CLI\)](#) (p. 404).

Ejecute este comando para obtener una lista de comandos de CodeBuild.

```
aws codebuild help
```

Ejecute este comando para obtener información sobre un comando de CodeBuild, donde *nombre-comando* es el nombre del comando.

```
aws codebuild command-name help
```

Los comandos son los siguientes:

- `batch-delete-builds`: elimina una o varias compilaciones de CodeBuild. Para obtener más información, consulte [Eliminación de compilaciones \(AWS CLI\)](#) (p. 304).
- `batch-get-builds`: obtiene información sobre varias compilaciones de CodeBuild. Para obtener más información, consulte [Ver detalles de las compilaciones \(AWS CLI\)](#) (p. 290).
- `batch-get-projects`: obtiene información acerca de uno o varios proyectos de compilación especificados. Para obtener más información, consulte [Ver los detalles de un proyecto de compilación \(AWS CLI\)](#) (p. 227).
- `create-project`: crea un proyecto de compilación. Para obtener más información, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).
- `delete-project`: elimina un proyecto de compilación. Para obtener más información, consulte [Eliminar un proyecto de compilación \(AWS CLI\)](#) (p. 269).
- `list-builds`: muestra los nombres de recursos de Amazon (ARN) de las compilaciones de CodeBuild. Para obtener más información, consulte [Ver una lista de identificadores de compilación \(AWS CLI\)](#) (p. 292).
- `list-builds-for-project`: obtiene una lista de los identificadores de compilación asociados con un proyecto de compilación especificado. Para obtener más información, consulte [Ver una lista de identificadores de compilación de un proyecto de compilación \(AWS CLI\)](#) (p. 295).
- `list-curated-environment-images`: obtiene una lista de imágenes de Docker administradas por CodeBuild que puede utilizar para sus compilaciones. Para obtener más información, consulte [Imágenes de Docker proporcionadas por CodeBuild](#) (p. 161).
- `list-projects`: obtiene una lista de nombres de proyecto de compilación. Para obtener más información, consulte [Ver una lista de nombres de proyectos de compilación \(AWS CLI\)](#) (p. 225).
- `start-build`: empieza a ejecutar una compilación. Para obtener más información, consulte [Ejecutar una compilación \(AWS CLI\)](#) (p. 282).
- `stop-build`: intenta detener la ejecución de la compilación especificada. Para obtener más información, consulte [Detener una compilación \(AWS CLI\)](#) (p. 297).
- `update-project`: cambia la información sobre el proyecto de compilación especificado. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(AWS CLI\)](#) (p. 267).

## Referencia de los SDK y las herramientas de AWS para AWS CodeBuild

Si desea utilizar uno de los SDK o las herramientas de AWS para automatizar AWS CodeBuild, consulte los siguientes recursos.

Si desea usar la AWS CLI para ejecutar CodeBuild, consulte la [Referencia de la línea de comandos](#) (p. 402).

## SDK y herramientas de AWS compatibles con AWS CodeBuild

Los siguientes ejemplos de AWS SDK y las herramientas de admiten CodeBuild:

- [SDK de AWS para C++](#). Para obtener más información, consulte la sección del espacio de nombres `Aws::CodeBuild` en la Referencia del SDK de AWS para la API de C++.
- [SDK de AWS para Go](#). Para obtener más información, consulte la [codebuild](#) Sección sobre de la AWS Referencia de la API de SDK for Go de.

- [SDK de AWS para Java](#). Para obtener más información, consulte las secciones `com.amazonaws.services.codebuild` y `com.amazonaws.services.codebuild.model` de la [Referencia del SDK de AWS para la API de Java](#).
- [SDK de AWS para JavaScript en el navegador](#) y [SDK de AWS para JavaScript en Node.js](#). Para obtener más información, consulte la [Clase: AWS.CodeBuild](#) Sección sobre de la [AWSReferencia](#) de la API del SDK para JavaScript.
- [SDK de AWS para .NET](#). Para obtener más información, consulte la [Amazon.codeBuild](#) y [Amazon.codebuild.model](#) secciones de espacio de nombres de la [AWSReferencia](#) de la API de SDK for .NET.
- [SDK de AWS para PHP](#). Para obtener más información, consulte la [Espacio de nombres Aws\CodeBuild](#) Sección sobre de la [AWSReferencia](#) de la API SDK for PHP.
- [SDK de AWS para Python \(Boto3\)](#). Para obtener más información, consulte la [CodeBuild](#) Sección sobre de la [Documentación de Boto 3](#).
- [SDK de AWS para Ruby](#). Para obtener más información, consulte la [Módulo de: Aws# CodeBuild](#) Sección sobre de la [AWSReferencia](#) de la API del SDK for Ruby de.
- [Herramientas de AWS para PowerShell](#). Para obtener más información, consulte la sección [AWS CodeBuild](#) de la [Referencia de herramientas de AWS](#) para los cmdlet de PowerShell.

## Especificación del punto de enlace de AWS CodeBuild

Puede utilizar la AWS Command Line Interface (AWS CLI) o uno de los SDK de AWS para especificar el punto de enlace que se utiliza en AWS CodeBuild. Hay un punto de enlace para cada región en la que CodeBuild está disponible. Además de un punto de enlace regional, hay cuatro regiones que también tienen un punto de enlace que cumple el Estándar federal de procesamiento de información (FIPS). Para obtener más información sobre los puntos de enlace de FIPS, consulte [Información general sobre FIPS 140-2](#).

La especificación de un punto de enlace es opcional. Si no indica explícitamente a CodeBuild qué punto de enlace debe utilizar, el servicio usará el punto de enlace asociado a la región que utiliza la [AWS](#) utiliza la cuenta. CodeBuild nunca utiliza de forma predeterminada un punto de enlace de FIPS. Si desea utilizar un punto de enlace de FIPS, debe asociarlo a mediante uno de los métodos siguientes.

### Note

Puede utilizar un alias o un nombre de región para especificar un punto de enlace que usa un SDK de AWS. Si utiliza la AWS CLI, debe usar el nombre completo del punto de enlace.

Para obtener información sobre los puntos de enlace que se pueden utilizar con CodeBuild, consulte [Regiones y puntos de enlace de CodeBuild](#).

### Temas

- [Especificación del punto de enlace de AWS CodeBuild \(AWS CLI\) \(p. 404\)](#)
- [Especificación del punto de enlace de AWS CodeBuild \(SDK de AWS\) \(p. 405\)](#)

## Especificación del punto de enlace de AWS CodeBuild (AWS CLI)

Puede utilizar el [AWS CLI](#) para especificar el punto de enlace a través del que [AWS CodeBuild](#) se accede mediante el `--endpoint-url` en cualquier comando CodeBuild. Por ejemplo, ejecute este comando para

obtener una lista de los nombres de proyectos de compilación que utilizan el punto de enlace que cumple el Estándar federal de procesamiento de información (FIPS) en la región EE. UU. Este (Norte de Virginia):

```
aws codebuild list-projects --endpoint-url https://codebuild-fips.us-east-1.amazonaws.com
```

Incluya el código `https://` al principio del punto de enlace.

El argumento `--endpoint-url` de la AWS CLI está disponible para todos los servicios de AWS. Para obtener más información sobre este y otros AWS CLI, consulte [AWS CLI Referencia de los comandos de la](#).

## Especificación del punto de enlace de AWS CodeBuild (SDK de AWS)

Puede utilizar un SDK de AWS para especificar el punto de enlace a través del que se obtiene acceso a AWS CodeBuild. Aunque en este ejemplo se utiliza el [SDK de AWS para Java](#), puede especificar el punto de enlace con los demás SDK de AWS.

Utilice el método `withEndpointConfiguration` cuando cree el cliente `AWSCodeBuild`. Utilice el siguiente formato:

```
AWSCodeBuild awsCodeBuild = AWSCodeBuildClientBuilder.standard().
 withEndpointConfiguration(new AwsClientBuilder.EndpointConfiguration("endpoint",
 "region")).
 withCredentials(new AWSStaticCredentialsProvider(sessionCredentials)).
 build();
```

Para obtener información sobre `AWSCodeBuildClientBuilder`, consulte [Clase AWSCodeBuildClientBuilder](#).

Las credenciales utilizadas en `withCredentials` deben ser del tipo `AWSCredentialsProvider`. Para obtener más información, consulte [Uso de credenciales de AWS](#).

No incluya el código `https://` al principio del punto de enlace.

Si desea especificar un punto de enlace que no sea de FIPS, puede utilizar la región en lugar del punto de enlace propiamente dicho. Por ejemplo, para especificar el punto de enlace en la región EE. UU. Este (Norte de Virginia), puede utilizar `us-east-1` en lugar del nombre completo del punto de enlace, `codebuild.us-east-1.amazonaws.com`.

Si desea especificar un punto de enlace de FIPS, puede utilizar un alias para simplificar el código. Solo los puntos de enlace de FIPS disponen de un alias. El resto de puntos de enlace se deben especificar utilizando la región o el nombre completo.

En la tabla siguiente se muestra el alias para cada uno de los cuatro puntos de enlace de FIPS disponibles:

| Nombre de la región             | Región    | punto de enlace                        | Alias          |
|---------------------------------|-----------|----------------------------------------|----------------|
| EE.UU. Este (Norte de Virginia) | us-east-1 | codebuild-fips.us-east-1.amazonaws.com | us-east-1-fips |
| US East (Ohio)                  | us-east-2 | codebuild-fips.us-east-2.amazonaws.com | us-east-2-fips |

| Nombre de la región                | Región    | punto de enlace                        | Alias          |
|------------------------------------|-----------|----------------------------------------|----------------|
| EE.UU. Oeste (Norte de California) | us-west-1 | codebuild-fips.us-west-1.amazonaws.com | us-west-1-fips |
| EE.UU. Oeste (Oregón)              | us-west-2 | codebuild-fips.us-west-2.amazonaws.com | us-west-2-fips |

Para especificar el uso del punto de enlace de FIPS en la región de EE. UU. Oeste (Oregón) mediante un alias:

```
AWSCodeBuild awsCodeBuild = AWSCodeBuildClientBuilder.standard().
 withEndpointConfiguration(new AwsClientBuilder.EndpointConfiguration("us-west-2-fips",
 "us-west-2")).
 withCredentials(new AWSStaticCredentialsProvider(sessionCredentials)).
 build();
```

Para especificar el uso del punto de enlace que no es de FIPS de la región EE. UU. Este (Norte de Virginia):

```
AWSCodeBuild awsCodeBuild = AWSCodeBuildClientBuilder.standard().
 withEndpointConfiguration(new AwsClientBuilder.EndpointConfiguration("us-east-1", "us-
 east-1")).
 withCredentials(new AWSStaticCredentialsProvider(sessionCredentials)).
 build();
```

Para especificar el uso del punto de enlace que no es de FIPS de la región Asia-Pacífico (Mumbai):

```
AWSCodeBuild awsCodeBuild = AWSCodeBuildClientBuilder.standard().
 withEndpointConfiguration(new AwsClientBuilder.EndpointConfiguration("ap-south-1", "ap-
 south-1")).
 withCredentials(new AWSStaticCredentialsProvider(sessionCredentials)).
 build();
```

## Ejecutar AWS CodeBuild directamente

Puede utilizar AWS CodeBuild consola de, AWS CLI, o bien AWS Para configurar, ejecutar y monitorear las compilaciones directamente con CodeBuild.

¿No es lo que está buscando? Para utilizar AWS CodePipeline para ejecutar CodeBuild, consulte [Usar CodePipeline con CodeBuild \(p. 407\)](#).

### Temas

- [Prerequisites \(p. 407\)](#)
- [Ejecutar AWS CodeBuild directamente \(p. 407\)](#)

## Prerequisites

Responda a las preguntas de [Planificación de una compilación](#) (p. 135).

## Ejecutar AWS CodeBuild directamente

1. Cree el proyecto de compilación. Para utilizar la consola, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198). Para utilizar la AWS CLI, consulte [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).
2. Ejecute la compilación. Para utilizar la consola, consulte [Ejecutar una compilación \(consola\)](#) (p. 282). Para utilizar la AWS CLI, consulte [Ejecutar una compilación \(AWS CLI\)](#) (p. 282).
3. Obtenga información acerca de la compilación. Para utilizar la consola, consulte [Ver detalles de las compilaciones \(consola\)](#) (p. 290). Para utilizar la AWS CLI, consulte [Ver detalles de las compilaciones \(AWS CLI\)](#) (p. 290).

## Usar AWS CodePipeline con AWS CodeBuild para probar el código y ejecutar compilaciones

Puede automatizar el proceso de lanzamiento utilizando AWS CodePipeline para probar el código y ejecutar las compilaciones con AWS CodeBuild.

En la siguiente tabla se muestran las tareas y los métodos disponibles que se pueden realizar. El uso de los SDK de AWS para realizar estas tareas queda fuera del ámbito de este tema.

| Tarea                                                                                                           | Enfoques disponibles                                                                                   | Enfoques descritos en este tema                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Cree una canalización de entrega continua (CD) con CodePipeline que automatice las compilaciones con CodeBuild  | <ul style="list-style-type: none"> <li>• CodePipeline</li> <li>• AWS CLI</li> <li>• AWS SDK</li> </ul> | <ul style="list-style-type: none"> <li>• <a href="#">Utilizar la consola CodePipeline</a> (p. 409)</li> <li>• <a href="#">Use la AWS CLI</a> (p. 412)</li> <li>• Puede adaptar la información de este tema para utilizar los SDK de AWS. Para obtener más información, consulte <code>create_pipeline</code> en la documentación de acciones de su lenguaje de programación en la <a href="#">SDK de Herramientas para Amazon Web Services</a> consulte <a href="#">CreatePipeline</a> en la <a href="#">AWS CodePipeline Referencia de la API</a>.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                            |
| Agregar la automatización de pruebas y compilaciones con CodeBuild a una canalización existente en CodePipeline | <ul style="list-style-type: none"> <li>• CodePipeline</li> <li>• AWS CLI</li> <li>• AWS SDK</li> </ul> | <ul style="list-style-type: none"> <li>• <a href="#">Utilice la consola de CodePipeline para añadir la automatización de compilaciones</a> (p. 415)</li> <li>• <a href="#">Uso de la consola de CodePipeline para añadir la automatización de pruebas</a> (p. 418)</li> <li>• Para el <a href="#">AWS CLI</a> puede adaptar la información de este tema para crear una canalización que contenga una acción de compilación o de prueba de CodeBuild. Para obtener más información, consulte <a href="#">Editar una canalización (AWS CLI)</a> y la <a href="#">Referencia de la estructura de la canalización de de en la AWS CodePipeline Guía del usuario</a>.</li> <li>• Puede adaptar la información de este tema para utilizar la canalización de los SDK de AWS. Para obtener más información, consulte la documentación de la acción de actualización de canalizaciones para su lenguaje de programación en la <a href="#">SDK de Herramientas para</a></li> </ul> |

| Tarea | Enfoques disponibles | Enfoques descritos en este tema                                                                          |
|-------|----------------------|----------------------------------------------------------------------------------------------------------|
|       |                      | Amazon Web Services consulte <a href="#">UpdatePipeline</a> en la AWS CodePipeline Referencia de la API. |

## Prerequisites

1. Responda a las preguntas de [Planificación de una compilación](#) (p. 135).
2. Si está utilizando un usuario de IAM para acceder a CodePipeline en lugar de un AWS o un usuario administrador de IAM, adjunte la directiva administrada denominada `AWSCodePipelineFullAccess` al usuario (o al grupo IAM al que pertenece el usuario de). No se recomienda usar una cuenta raíz de AWS. Esta política concede al usuario permiso para crear la canalización en CodePipeline. Para obtener más información, consulte [Adhesión de políticas administradas](#) en la Guía del usuario de IAM.

### Note

La entidad de IAM que asocia la política al usuario (o al grupo de IAM al que pertenece el usuario) debe tener permiso en IAM para asociar políticas. Para obtener más información, consulte [Delegación de permisos para administrar usuarios de IAM, grupos y credenciales de IAM](#) en la Guía del usuario de IAM.

3. Cree un rol de servicio de CodePipeline si aún no dispone de uno en su AWS account. CodePipeline utiliza este rol de servicio para interactuar con otros AWS servicios, incluyendo AWS CodeBuild. En su nombre. Por ejemplo, para utilizar AWS CLI para crear un rol de servicio CodePipeline, ejecute el `iam create-role` Comando de la :

Para Linux, macOS o Unix:

```
aws iam create-role --role-name AWS-CodePipeline-CodeBuild-Service-Role --assume-role-policy-document '{"Version":"2012-10-17","Statement":[{"Effect":"Allow","Principal":{"Service":"codepipeline.amazonaws.com"},"Action":["sts:AssumeRole"]}]}'
```

Para Windows:

```
aws iam create-role --role-name AWS-CodePipeline-CodeBuild-Service-Role --assume-role-policy-document '{"Version":"2012-10-17","Statement":[{"Effect":"Allow","Principal":{"Service":"codepipeline.amazonaws.com"},"Action":["sts:AssumeRole"]}]}'
```

### Note

La entidad de IAM que crea este rol de servicio de CodePipeline debe tener permiso en IAM para crear roles de servicio.

4. Después de crear un rol de servicio de CodePipeline o identificar uno existente, debe agregar la política de roles de servicio de CodePipeline predeterminada al rol de servicio, tal y como se describe en [Revisar la directiva de rol de servicio CodePipeline predeterminada](#) en la AWS CodePipeline Guía del usuario. Si aún no forma parte de la política de la función.

### Note

La entidad de IAM que añade esta política de roles de servicio de CodePipeline debe tener permiso en IAM para añadir políticas de roles de servicio a roles de servicio.

5. Cree y cargue el código fuente de un tipo de repositorio compatible con CodeBuild y CodePipeline, como CodeCommit, Amazon S3 o GitHub (CodePipeline no admite actualmente Bitbucket). El código



fuente debe contener un archivo buildspec, pero puede declarar uno cuando defina un proyecto de compilación más adelante en este tema. Para obtener más información, consulte [Referencia de la especificación de compilación \(p. 136\)](#).

#### Important

Si tiene previsto utilizar la canalización para implementar el código fuente compilado, el artefacto de salida de la compilación debe ser compatible con el sistema de implementación que utilice.

- Para CodeDeploy, consulte la [Ejemplo de AWS CodeDeploy \(p. 54\)](#) en esta guía y [Preparar una revisión para CodeDeploy](#) en la AWS CodeDeploy Guía del usuario.
- En el caso de AWS Elastic Beanstalk, consulte [Ejemplo de AWS Elastic Beanstalk \(p. 66\)](#) en esta guía y este artículo acerca de [cómo crear un paquete de código fuente de una aplicación](#) en la Guía para desarrolladores de AWS Elastic Beanstalk.
- Para AWS OpsWorks consulte [Origen de lasy Usar CodePipeline AWS OpsWorks](#) en la AWS OpsWorks Guía del usuario.

#### Temas

- [Crear una canalización que use CodeBuild \(consola de CodePipeline\) \(p. 409\)](#)
- [Cree una canalización que use CodeBuild \(AWS CLI\) \(p. 412\)](#)
- [Agregar una acción de compilación de CodeBuild a una canalización \(consola de CodePipeline\) \(p. 415\)](#)
- [Agregar una acción de prueba de CodeBuild a una canalización \(consola de CodePipeline\) \(p. 418\)](#)

## Crear una canalización que use CodeBuild (consola de CodePipeline)

Utilice el siguiente procedimiento para crear una canalización que utilice CodeBuild para compilar e implementar el código fuente.

Para crear una canalización que solo pruebe el código fuente:

- Utilice el siguiente procedimiento para crear la canalización y, a continuación, elimine las fases Build y Beta de la compilación. A continuación, utilice [Agregar una acción de prueba de CodeBuild a una canalización \(consola de CodePipeline\) \(p. 418\)](#) en este tema para añadir a la canalización una acción de prueba que use CodeBuild.
- Utilice uno de los otros procedimientos de este tema para crear la canalización y, a continuación, use la [Agregar una acción de prueba de CodeBuild a una canalización \(consola de CodePipeline\) \(p. 418\)](#) en este tema para añadir a la canalización una acción de prueba que use CodeBuild.

Para utilizar el asistente para crear canalizaciones de CodePipeline para crear una canalización que use CodeBuild

1. Inicie sesión en AWS Management Console utilizando:

- Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [Usuario de la cuenta raíz](#) en la Guía del usuario de IAM.
- Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la Guía del usuario de IAM.
- Un usuario de IAM en su AWS con permiso para utilizar el siguiente conjunto mínimo de acciones:



```
codepipeline:*
iam:ListRoles
iam:PassRole
s3:CreateBucket
s3:GetBucketPolicy
s3:GetObject
s3:ListAllMyBuckets
s3:ListBucket
s3:PutBucketPolicy
codecommit:ListBranches
codecommit:ListRepositories
codedeploy:GetApplication
codedeploy:GetDeploymentGroup
codedeploy:ListApplications
codedeploy:ListDeploymentGroups
elasticbeanstalk:DescribeApplications
elasticbeanstalk:DescribeEnvironments
lambda:GetFunctionConfiguration
lambda:ListFunctions
opsworks:DescribeStacks
opsworks:DescribeApps
opsworks:DescribeLayers
```

2. Abra el iconoAWS CodePipelineen la consola de<https://console.aws.amazon.com/codesuite/codepipeline/home>.
3. En el navegadorAWSSeleccionador de región, elija la opciónAWSRegión donde su proyecto de construcciónAWSse encuentran los recursos. Deben ser unaAWSRegión en la que se admite CodeBuild. Para obtener más información, consulteAWS CodeBuilden laReferencia general de Amazon Web Services.
4. Cree una canalización. Si aparece una página con información de CodePipeline, elijaCrear canalización. Si aparece la página Pipelines (Canalizaciones), seleccione Create pipeline (Crear canalización).
5. En la páginaPaso 1: Elija la configuración de canalizaciónPágina, paraNombre de canalización, escriba un nombre para la canalización (por ejemplo, **CodeBuildDemoPipeline**). Si elige otro nombre, asegúrese de utilizarlo durante todo este procedimiento.
6. En Role name (Nombre de la función), realice una de las operaciones siguientes:

Seleccione New service role (Nueva función de servicio) y, en Role Name (Nombre de la función), especifique un nombre para la nueva función de servicio.

SeleccionarRol de servicio existenteElija y elija después el rol de servicio de CodePipeline que creó o identificó como parte de los requisitos previos de este tema.

7. En Artifact store (Almacén de artefactos), haga lo siguiente:
  - SeleccionarUbicación predeterminadaPara utilizar el almacén de artefactos predeterminado (por ejemplo, el bucket de artefactos de S3 designado como predeterminado) con la canalización en laAWSRegión que ha seleccionado para su canalización.
  - SeleccionarUbicación personalizadaSi ya dispone de un almacén de artefactos existente, como un bucket de artefactos de S3, en la mismaAWSRegión como su gasoducto.

#### Note

Esto no es el bucket de código fuente del código fuente de la canalización. Este es el almacén de artefactos de la canalización. Se necesita un almacén de artefactos diferente (por ejemplo, un bucket de S3) para cada canalización que esté en la mismaAWSRegión como la tubería.

8. Elija Next (Siguiendo).

9. En la página Paso 2: Agregar fase de origenPágina, paraProveedor de origenEn, realice una de las siguientes operaciones:
- Si el código fuente está almacenado en un bucket de S3, elijaAmazon S3. En Bucket, seleccione el bucket de S3 que contiene el código fuente. En S3 object key (Clave de objeto de S3), escriba el nombre del archivo que contiene el código fuente (por ejemplo, *file-name.zip*). Elija Next (Siguiente).
  - Si el código fuente está almacenado en unaAWS CodeCommitrepositorio, elijaCodeCommit. En Repository name, elija el nombre del repositorio que contiene el código fuente. En Branch name (Nombre de la ramificación), elija el nombre de la ramificación que contiene la versión del código fuente que desea compilar. Elija Next (Siguiente).
  - Si el código fuente está almacenado en un repositorio GitHub, elija GitHub. Elija Connect to GitHub y siga las instrucciones para autenticarse con GitHub. En Repository, elija el nombre del repositorio que contiene el código fuente. En Branch (Ramificación), elija el nombre de la ramificación que contiene la versión del código fuente que desea compilar.

Elija Next (Siguiente).

10. En la página Paso 3: Agregar fase de compilaciónPágina, paraProveedor de creación, elijaCodeBuild.
11. Si ya tiene un proyecto de compilación que desee utilizar, paraNombre de proyectoEn, seleccione el nombre del proyecto de compilación y vaya al siguiente paso de este procedimiento.

Si necesita crear un nuevo proyecto de compilación de CodeBuild, siga las instrucciones de[Creación de un proyecto de compilación \(consola\) \(p. 198\)](#)y volver a este procedimiento.

Si elige un proyecto de compilación existente, debe haber definido ya la configuración del artefacto de salida de la compilación (aunque CodePipeline la invalide). Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\) \(p. 254\)](#).

#### Important

Si habilita webhooks en un proyecto de CodeBuild y el proyecto se usa como un paso de compilación en CodePipeline, se crearán dos compilaciones idénticas para cada confirmación. Una compilación se activa a través de los webhooks y la otra a través de CodePipeline. Como se factura por compilación, se le cobrarán ambas compilaciones. Por lo tanto, si utiliza CodePipeline, le recomendamos que deshabilite los webhooks en CodeBuild. En la consola de AWS CodeBuild, desactive la casilla Webhook. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\) \(p. 254\)](#).

12. En la página Paso 4: Agregar fase de implementaciónEn, realice una de las siguientes operaciones:
- Si no desea implementar el artefacto de salida de la compilación, elija Skip (Omitir) y confirme esta opción cuando se lo soliciten.
  - Si desea implementar el artefacto de salida de la compilación, en Deploy provider (Proveedor de implementación), elija un proveedor de implementación y especifique la configuración cuando se lo soliciten.

Elija Next (Siguiente).

13. En la página Review (Revisar), revise las opciones y seleccione Create pipeline (Crear canalización).
14. Una vez que la canalización se ejecute correctamente, podrá obtener el artefacto de salida de la compilación. Con la canalización mostrada en la consola de CodePipeline, en la ventana deBuild, elija la información sobre herramientas. Anote el valor de Output artifact (por ejemplo, MyAppBuild).

#### Note

También puede obtener el artefacto de salida de la compilación eligiendo la opción **Construir artefactos** en la página de detalles de compilación de la consola de CodeBuild. Para ir a esta página, omita el resto de los pasos de este procedimiento y consulte [Ver detalles de las compilaciones \(consola\)](#) (p. 290).

15. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.
16. En la lista de buckets, abra el bucket utilizado por la canalización. El nombre del bucket debe seguir el formato `codepipeline-region-ID-random-number`. Puede utilizar AWS CLI para ejecutar `CodePipeline get-pipeline` para obtener el nombre del bucket, donde *mi-nombre de canalización* es el nombre para mostrar de la canalización:

```
aws codepipeline get-pipeline --name my-pipeline-name
```

En el resultado, el objeto `pipeline` contiene un objeto `artifactStore`, que a su vez contiene un valor `location` con el nombre del bucket.

17. Abra la carpeta que tiene el mismo nombre que la canalización (en función de la longitud del nombre de la canalización, el nombre de la carpeta podría aparecer truncado) y después abra la carpeta que tiene el mismo nombre que el valor de `Output artifact` (Artefacto de salida) que anotó anteriormente.
18. Extraiga el contenido del archivo. Si hay varios archivos en esa carpeta, extraiga el contenido del archivo con la última marca temporal `Last Modified`. (Es posible que tenga que asignar al archivo la extensión `.zip` para que pueda trabajar con él en la utilidad ZIP del sistema). El artefacto de salida de la compilación está en el contenido extraído del archivo.
19. Si indicó a CodePipeline que implementara el artefacto de salida de la compilación, utilice las instrucciones del proveedor de implementación para obtener el artefacto de salida de la compilación en los destinos de la implementación.

## Cree una canalización que use CodeBuild (AWS CLI)

Utilice el siguiente procedimiento para crear una canalización que utilice CodeBuild para compilar el código fuente.

Para utilizar AWS CLI para crear una canalización que implemente el código fuente compilado que solo pruebe el código fuente, puede adaptar las instrucciones en [Editar una canalización \(AWS CLI\)](#) y la [Referencia de la estructura de la canalización de de](#) en la AWS CodePipeline Guía del usuario.

1. Crear o identificar un proyecto de compilación en CodeBuild. Para obtener más información, consulte [Creación de un proyecto de compilación](#) (p. 197).

#### Important

El proyecto de compilación debe definir la configuración del artefacto de salida de la compilación (aunque CodePipeline la invalide). Para obtener más información, consulte la descripción de `artifacts` en [Crear un proyecto de compilación \(AWS CLI\)](#) (p. 209).

2. Asegúrese de que ha configurado AWS CLI con AWS clave de acceso AWS. La clave de acceso secreta que corresponden a una de las entidades de IAM descritas en este tema. Para obtener más información, consulte [Configuración inicial de la AWS Command Line Interface](#) en la Guía del usuario de AWS Command Line Interface.
3. Crear un archivo con formato JSON que represente la estructura de la canalización. Asigne al archivo el nombre `create-pipeline.json` o uno similar. Por ejemplo, esta estructura con formato JSON crea una canalización con una acción de origen que hace referencia a un bucket de entrada de S3 y una acción de compilación que utiliza CodeBuild:

```
{
```

```

"pipeline": {
 "roleArn": "arn:aws:iam::<account-id>:role/<AWS-CodePipeline-service-role-name>",
 "stages": [
 {
 "name": "Source",
 "actions": [
 {
 "inputArtifacts": [],
 "name": "Source",
 "actionTypeId": {
 "category": "Source",
 "owner": "AWS",
 "version": "1",
 "provider": "S3"
 },
 "outputArtifacts": [
 {
 "name": "MyApp"
 }
],
 "configuration": {
 "S3Bucket": "<bucket-name>",
 "S3ObjectKey": "<source-code-file-name.zip>"
 },
 "runOrder": 1
 }
]
 },
 {
 "name": "Build",
 "actions": [
 {
 "inputArtifacts": [
 {
 "name": "MyApp"
 }
],
 "name": "Build",
 "actionTypeId": {
 "category": "Build",
 "owner": "AWS",
 "version": "1",
 "provider": "CodeBuild"
 },
 "outputArtifacts": [
 {
 "name": "default"
 }
],
 "configuration": {
 "ProjectName": "<build-project-name>"
 },
 "runOrder": 1
 }
]
 }
],
 "artifactStore": {
 "type": "S3",
 "location": "<CodePipeline-internal-bucket-name>"
 },
 "name": "<my-pipeline-name>",
 "version": 1
}

```

En estos datos con formato JSON:

- El valor de `roleArn` debe coincidir con el ARN del rol de servicio de CodePipeline que creó o identificó como parte de los requisitos previos.
- Los valores de `S3Bucket` y `S3ObjectKey` en `configuration` presuponen que el código fuente está almacenado en un bucket de S3. Para obtener información sobre la configuración de otros tipos de repositorios de código fuente, consulte [Referencia de la estructura de la canalización de AWS CodePipeline](#) en la AWS CodePipeline Guía del usuario.
- El valor de `projectName` es el nombre del proyecto de compilación de CodeBuild que creó anteriormente en este procedimiento.
- El valor de `location` es el nombre del bucket de S3 que se utiliza en esta canalización. Para obtener más información, consulte [Crear una política para un bucket de S3 para usarlo como almacén de artefactos de CodePipeline](#) en la AWS CodePipeline Guía del usuario.
- El valor de `name` es el nombre de esta canalización. Todos los nombres de canalización deben ser únicos en su cuenta.

Aunque estos datos describen únicamente una acción de origen y una acción de compilación, puede agregar acciones para actividades relacionadas con las pruebas, la implementación del artefacto de salida de la compilación, las llamadas a funciones de AWS Lambda, etc. Para obtener más información, consulte [Referencia de la estructura de la canalización de AWS CodePipeline](#) en la Guía del usuario de AWS CodePipeline.

4. Cambie a la carpeta que contiene el archivo JSON y, a continuación, ejecute `CodePipeline create-pipeline` Comando, especificando el nombre de archivo:

```
aws codepipeline create-pipeline --cli-input-json file://create-pipeline.json
```

#### Note

Debe crear la canalización en una AWS Región en la que se admite CodeBuild. Para obtener más información, consulte [AWS CodeBuild](#) en la Referencia general de Amazon Web Services.

Los datos con formato JSON aparecen en la salida y CodePipeline crea la canalización.

5. Para obtener información sobre el estado de la canalización, ejecute `CodePipeline get-pipeline-state` Comando, especificando el nombre de la canalización:

```
aws codepipeline get-pipeline-state --name <my-pipeline-name>
```

En el resultado, busque información que confirme que la compilación se ha realizado correctamente. Los puntos suspensivos ( `...` ) se utilizan para mostrar los datos que se han omitido por motivos de brevedad.

```
{
 ...
 "stageStates": [
 ...
 {
 "actionStates": [
 {
 "actionName": "CodeBuild",
 "latestExecution": {
 "status": "SUCCEEDED",
 ...
 },
 ...
 },
 ...
]
 }
]
}
```

```
]
 }
]
}
```

Si ejecuta este comando demasiado pronto, es posible que no vea ninguna información sobre la acción de compilación. Tal vez tenga que ejecutar este comando varias veces hasta que la canalización termine de ejecutar la acción de compilación.

- Una vez que la compilación se realice correctamente, siga estas instrucciones para obtener el artefacto de salida de la compilación. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.

#### Note

También puede obtener el artefacto de salida de la compilación eligiendo la opción **Construir artefactos** en la página de detalles de compilación relacionada en la consola de CodeBuild.

Para ir a esta página, omita el resto de los pasos de este procedimiento y consulte [Ver detalles de las compilaciones \(consola\)](#) (p. 290).

- En la lista de buckets, abra el bucket utilizado por la canalización. El nombre del bucket debe seguir el formato `codepipeline-<region-ID>-<random-number>`. Puede obtener el nombre de bucket en `create-pipeline.json` o puede ejecutar el archivo `CodePipelineget-pipeline` para obtener el nombre del depósito.

```
aws codepipeline get-pipeline --name <pipeline-name>
```

En el resultado, el objeto `pipeline` contiene un objeto `artifactStore`, que a su vez contiene un valor `location` con el nombre del bucket.

- Abra la carpeta que coincide con el nombre de la canalización (por ejemplo, **<pipeline-name>**).
- En esa carpeta, abra la carpeta con el nombre `default`.
- Extraiga el contenido del archivo. Si hay varios archivos en esa carpeta, extraiga el contenido del archivo con la última marca temporal `Last Modified`. (Es posible que tenga que asignar al archivo una extensión `.zip` para que pueda trabajar con él en la utilidad ZIP del sistema). El artefacto de salida de la compilación está en el contenido extraído del archivo.

## Agregar una acción de compilación de CodeBuild a una canalización (consola de CodePipeline)

- Inicie sesión en AWS Management Console utilizando:
  - Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [Usuario de la cuenta raíz](#) en la Guía del usuario de IAM.
  - Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la Guía del usuario de IAM.
  - Un usuario de IAM en su AWS con permiso para realizar el siguiente conjunto mínimo de acciones:

```
codepipeline:*
iam:ListRoles
iam:PassRole
s3:CreateBucket
s3:GetBucketPolicy
s3:GetObject
s3:ListAllMyBuckets
s3:ListBucket
s3:PutBucketPolicy
```

```
codecommit:ListBranches
codecommit:ListRepositories
codedeploy:GetApplication
codedeploy:GetDeploymentGroup
codedeploy:ListApplications
codedeploy:ListDeploymentGroups
elasticbeanstalk:DescribeApplications
elasticbeanstalk:DescribeEnvironments
lambda:GetFunctionConfiguration
lambda:ListFunctions
opsworks:DescribeStacks
opsworks:DescribeApps
opsworks:DescribeLayers
```

2. Abra la consola de CodePipeline <https://console.aws.amazon.com/codesuite/codepipeline/home>.
3. En el navegador AWS Seleccione la AWS Región en la que se encuentra la canalización. Debe ser una región en la que se admita CodeBuild. Para obtener más información, consulte [CodeBuild](#) en la Referencia general de Amazon Web Services.
4. En la página Pipelines (Canalizaciones), seleccione el nombre de la canalización.
5. En la página de detalles de la canalización, en la acción Source (Origen), elija la ayuda contextual. Anote el valor de Output artifact (Artefacto de salida) (por ejemplo, MyApp).

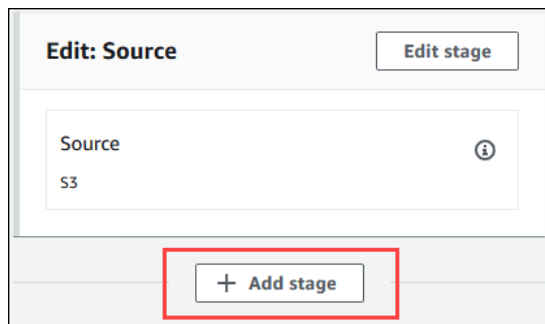
#### Note

En este procedimiento se indica cómo añadir una acción de compilación en una fase de compilación entre las fases Source y Beta. Si desea añadir la acción de compilación en otro lugar, elija la ayuda contextual sobre la acción justo antes del lugar en el que desea añadir la acción de compilación y anote el valor de Output artifact (Artefacto de salida).

6. Elija Edit.
7. Entre las etapas Source (Origen) y Beta, seleccione Add stage (Añadir fase).

#### Note

En este procedimiento, se explica cómo se agrega una etapa de compilación entre las etapas Source y Beta. Para añadir una acción de compilación a una fase existente, seleccione Edit stage (Editar fase) en la fase y vaya al paso 8 de este procedimiento. Para añadir la fase de compilación en otra parte, seleccione Add stage (Añadir fase) en el lugar que desee.



8. En Stage name (Nombre de fase), escriba el nombre de la fase de compilación (por ejemplo, **Build**). Si elige otro nombre, úselo en todo este procedimiento.
9. Dentro de la fase seleccionada, seleccione Add action (Añadir acción).

#### Note

En este procedimiento se muestra cómo agregar la acción de compilación dentro de una fase de compilación. Para añadir la acción de compilación en otra parte, seleccione Add action

(Añadir acción) en el lugar que desee. Es posible que primero tenga que seleccionar Edit stage (Editar fase) en la fase existente en la que desea añadir la acción de compilación.

10. En Edit action (Editar acción), en Action name (Nombre de acción), escriba un nombre para la acción (por ejemplo, **CodeBuild**). Si elige otro nombre, úselo en todo este procedimiento.
11. ParaProveedor de acción, elijaCodeBuild.
12. Si ya tiene un proyecto de compilación que desee utilizar, paraNombre de proyectoEn, seleccione el nombre del proyecto de compilación y vaya al siguiente paso de este procedimiento.

Si necesita crear un nuevo proyecto de compilación de CodeBuild, siga las instrucciones de [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y volver a este procedimiento.

Si elige un proyecto de compilación existente, debe haber definido ya la configuración del artefacto de salida de la compilación (aunque CodePipeline la invalide). Para obtener más información, consulte la descripción de artefactos en [Creación de un proyecto de compilación \(consola\)](#) (p. 198) o [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).

### Important

Si habilita webhooks en un proyecto de CodeBuild y el proyecto se usa como un paso de compilación en CodePipeline, se crearán dos compilaciones idénticas para cada confirmación. Una compilación se activa a través de los webhooks y la otra a través de CodePipeline. Como se factura por compilación, se le cobrarán ambas compilaciones. Por lo tanto, si utiliza CodePipeline, le recomendamos que deshabilite los webhooks en CodeBuild. En la consola de CodeBuild, desactive la casillaWebhook. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254)

13. En Input artifacts (Artefactos de entrada), elija el artefacto de salida que anotó anteriormente en este procedimiento.
14. En Output artifacts (Artefactos de salida), escriba un nombre para el artefacto de salida (por ejemplo, **MyAppBuild**).
15. Elija Añadir acción.
16. Elija Save (Guardar) y después otra vez Save (Guardar) para guardar los cambios realizados en la canalización.
17. Elija Release change.
18. Una vez que la canalización se ejecute correctamente, podrá obtener el artefacto de salida de la compilación. Con la canalización mostrada en la consola de CodePipeline, en la ventana deBuild, elija la información sobre herramientas. Anote el valor de Output artifact (por ejemplo, MyAppBuild).

### Note

También puede obtener el artefacto de salida de la compilación eligiendo la opciónConstruir artefactosEn la página de detalles de compilación de la consola de CodeBuild. Para tener acceso a esta página, consulte [Ver detalles de las compilaciones \(consola\)](#) (p. 290) y vaya al paso 31 de este procedimiento.

19. Abra la consola de Amazon S3 en <https://console.aws.amazon.com/s3>.
20. En la lista de buckets, abra el bucket utilizado por la canalización. El nombre del bucket debe seguir el formato `codepipeline-region-ID-random-number`. Puede utilizarAWS CLIpara ejecutar `CodePipelineget-pipeline`Para obtener el nombre del bucket:

```
aws codepipeline get-pipeline --name my-pipeline-name
```

En el resultado, el objeto pipeline contiene un objeto artifactStore, que a su vez contiene un valor location con el nombre del bucket.

21. Abra la carpeta que tiene el mismo nombre que la canalización (en función de la longitud del nombre de la canalización, el nombre de la carpeta podría aparecer truncado) y después abra la carpeta que



tiene el mismo nombre que el valor de Output artifact (Artefacto de salida) que anotó anteriormente en este procedimiento.

22. Extraiga el contenido del archivo. Si hay varios archivos en esa carpeta, extraiga el contenido del archivo con la última marca temporal Last Modified. (Es posible que tenga que asignar al archivo la extensión .zip para que pueda trabajar con él en la utilidad ZIP del sistema). El artefacto de salida de la compilación está en el contenido extraído del archivo.
23. Si indicó a CodePipeline que implementara el artefacto de salida de la compilación, utilice las instrucciones del proveedor de implementación para obtener el artefacto de salida de la compilación en los destinos de la implementación.

## Agregar una acción de prueba de CodeBuild a una canalización (consola de CodePipeline)

1. Inicie sesión en AWS Management Console utilizando:
  - Su cuenta raíz de AWS. No se recomienda. Para obtener más información, consulte [Usuario de la cuenta raíz](#) en la Guía del usuario de IAM.
  - Un usuario administrador de IAM en su AWS account. Para obtener más información, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la Guía del usuario de IAM.
  - Un usuario de IAM en su AWS con permiso para realizar el siguiente conjunto mínimo de acciones:

```
codepipeline:*
iam:ListRoles
iam:PassRole
s3:CreateBucket
s3:GetBucketPolicy
s3:GetObject
s3:ListAllMyBuckets
s3:ListBucket
s3:PutBucketPolicy
codecommit:ListBranches
codecommit:ListRepositories
codedeploy:GetApplication
codedeploy:GetDeploymentGroup
codedeploy:ListApplications
codedeploy:ListDeploymentGroups
elasticbeanstalk:DescribeApplications
elasticbeanstalk:DescribeEnvironments
lambda:GetFunctionConfiguration
lambda:ListFunctions
opsworks:DescribeStacks
opsworks:DescribeApps
opsworks:DescribeLayers
```

2. Abra la consola de CodePipeline <https://console.aws.amazon.com/codesuite/codepipeline/home>.
3. En el navegador AWS Seleccione la AWS Región en la que se encuentra la canalización. Deben ser una AWS Región en la que se admite CodeBuild. Para obtener más información, consulte [AWS CodeBuild](#) en la Referencia general de Amazon Web Services.
4. En la página Pipelines (Canalizaciones), seleccione el nombre de la canalización.
5. En la página de detalles de la canalización, en la acción Source (Origen), elija la ayuda contextual. Anote el valor de Output artifact (Artefacto de salida) (por ejemplo, MyApp).

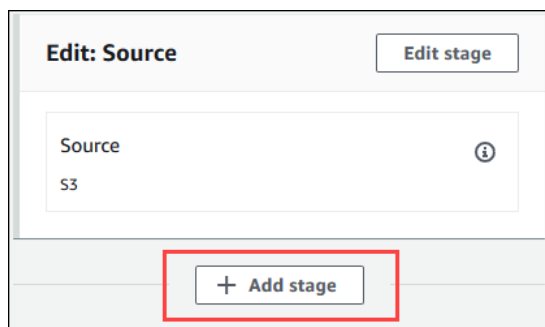
#### Note

En este procedimiento se indica cómo añadir una acción de prueba a una fase de prueba entre las fases Source y Beta. Si desea añadir la acción de prueba en otro lugar, coloque el puntero del ratón sobre la acción que aparece justo delante y anote el valor de Output artifact.

6. Elija Edit.
7. Inmediatamente después de la fase Source (Origen), seleccione Add stage (Añadir fase).

#### Note

En este procedimiento, se muestra también como se agrega a la canalización una etapa de prueba inmediatamente después de la fase Source. Para añadir una acción de prueba a una fase existente, seleccione Edit stage (Editar fase) en la fase y vaya al paso 8 de este procedimiento. Para añadir la fase de prueba en otra parte, seleccione Add stage (Añadir fase) en el lugar que desee.



8. En Stage name (Nombre de fase), escriba el nombre de la fase de prueba (por ejemplo, **Test**). Si elige otro nombre, úselo en todo este procedimiento.
9. En la fase seleccionada, elija Add action (Añadir acción).

#### Note

En este procedimiento se indica cómo añadir la acción de prueba en una fase de prueba. Para añadir la acción de prueba en otra parte, seleccione Add action (Añadir acción) en el lugar que desee. Es posible que primero tenga que seleccionar Edit (Editar) en la fase existente en la que desea añadir la acción de prueba.

10. En Edit action (Editar acción), en Action name (Nombre de acción), escriba un nombre para la acción (por ejemplo, **Test**). Si elige otro nombre, úselo en todo este procedimiento.
11. Para Proveedor de acción, elija CodeBuild.
12. Si ya tiene un proyecto de compilación que desee utilizar, para Nombre de proyecto, seleccione el nombre del proyecto de compilación y vaya al siguiente paso de este procedimiento.

Si necesita crear un nuevo proyecto de compilación de CodeBuild, siga las instrucciones de [Creación de un proyecto de compilación \(consola\)](#) (p. 198) y volver a este procedimiento.

#### Important

Si habilita webhooks en un proyecto de CodeBuild y el proyecto se usa como un paso de compilación en CodePipeline, se crearán dos compilaciones idénticas para cada confirmación. Una compilación se activa a través de los webhooks y la otra a través de CodePipeline. Como se factura por compilación, se le cobrarán ambas compilaciones. Por lo tanto, si utiliza CodePipeline, le recomendamos que deshabilite los webhooks en CodeBuild. En la consola de CodeBuild, desactive la casilla Webhook. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).

13. En Input artifacts (Artefactos de entrada), seleccione el valor de Output artifact (Artefacto de salida) que anotó anteriormente en este procedimiento.
14. (Opcional) Si desea que la acción de prueba produzca un artefacto de salida y configura la especificación de compilación en consecuencia, entonces en Output artifact (Artefacto de salida), escriba el valor que desea asignar al artefacto de salida.
15. Seleccione Save.
16. Elija Release change.
17. Una vez que la canalización se ejecute correctamente, obtendrá los resultados de prueba. En el navegador Pruebas de la canalización, elija la opción CodeBuild Para abrir la página del proyecto de compilación relacionada en la consola de CodeBuild.
18. En la página del proyecto de compilación, en el área Build history (Historial de compilaciones), elija el hipervínculo Build run (Ejecución de compilación) relacionado.
19. En la página de ejecución de compilación, en Crear registros, elija la opción Ver registro completo para abrir el registro de compilación en la consola de Amazon CloudWatch.
20. Desplácese por el log de compilación para ver los resultados de la prueba.

## Uso de AWS CodeBuild con Jenkins

Puede usar el complemento Jenkins de AWS CodeBuild Integración de CodeBuild con sus trabajos de compilación de Jenkins. En lugar de enviar sus trabajos de compilación a los nodos de compilación de Jenkins, puede usar el complemento para enviar sus trabajos de compilación a CodeBuild. De esta manera, ya no tendrá que aprovisionar, configurar ni administrar nodos de compilación de Jenkins.

### Configuración de Jenkins

Para obtener información acerca de la configuración de Jenkins con el AWS CodeBuild, y para descargar el código fuente del complemento, consulte <https://github.com/aws-labs/aws-codebuild-jenkins-plugin>.

### Instalación del complemento

Si ya ha configurado el servidor de Jenkins y solo desea instalar el complemento de AWS CodeBuild, en su instancia de Jenkins, en el administrador de complementos, busque **CodeBuild Plugin for Jenkins**.

### Uso del complemento

Para utilizar AWS CodeBuild con orígenes que se encuentren fuera de una VPC

1. Cree un proyecto en la consola de CodeBuild. Para obtener más información, consulte [Creación de un proyecto de compilación \(consola\)](#) (p. 198).
  - Elija el icono AWS Región en la que desea ejecutar la compilación.
  - (Opcional) Establezca la configuración de Amazon VPC para permitir que el contenedor de compilaciones de CodeBuild obtenga acceso a los recursos de la VPC.
  - Anote el nombre de su proyecto. Lo necesitará en el paso 3.
  - (Opcional) Si CodeBuild no admite de forma nativa el repositorio de origen, puede configurar Amazon S3 como el tipo de origen de entrada del proyecto.
2. En la consola de IAM Console, cree el usuario de IAM que va a usar el complemento Jenkins.
  - Cuando cree las credenciales del usuario, elija Programmatic Access.
  - Cree una política similar a la siguiente y, a continuación, asocie la política al usuario.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Resource": ["arn:aws:logs:{region}:::{awsAccountId}:log-group:/aws/codebuild/{projectName}:*"],
 "Action": ["logs:GetLogEvents"]
 },
 {
 "Effect": "Allow",
 "Resource": ["arn:aws:s3:::{inputBucket}"],
 "Action": ["s3:GetBucketVersioning"]
 },
 {
 "Effect": "Allow",
 "Resource": ["arn:aws:s3:::{inputBucket}/{inputObject}"],
 "Action": ["s3:PutObject"]
 },
 {
 "Effect": "Allow",
 "Resource": ["arn:aws:s3:::{outputBucket}/*"],
 "Action": ["s3:GetObject"]
 },
 {
 "Effect": "Allow",
 "Resource": ["arn:aws:codebuild:{region}:::{awsAccountId}:project/{projectName}"],
 "Action": ["codebuild:StartBuild",
 "codebuild:BatchGetBuilds",
 "codebuild:BatchGetProjects"]
 }
]
}
```

3. Cree un proyecto de estilo libre en Jenkins.
  - En la página Configuración elija la página Agregar paso de compilación Haga clic en y luego en Ejecutar compilación en CodeBuild.
  - Configure el paso de compilación.
    - Proporcione valores para Region, Credentials y Project Name.
    - Elija Use Project source.
    - Guarde la configuración y ejecute una compilación desde Jenkins.
4. En Source Code Management, elija cómo desea recuperar el origen. Es posible que necesite instalar el complemento GitHub (o el complemento Jenkins del proveedor del repositorio de origen) en su servidor de Jenkins.
  - En la página Configure, elija Add build step y después Run build on AWS CodeBuild.
  - Configure el paso de compilación.
    - Proporcione valores para Region, Credentials y Project Name.
    - Elija Use Jenkins source.
    - Guarde la configuración y ejecute una compilación desde Jenkins.

Para utilizar el complemento de AWS CodeBuild con el complemento Pipeline de Jenkins

- En la página del proyecto de canalizaciones de Jenkins, utilice el generador de fragmentos para generar un script de canalización que añada CodeBuild como un paso de la canalización. Debe generar un script similar a este:

```
awsCodeBuild projectName: 'project', credentialsType: 'keys', region: 'us-west-2',
sourceControlType: 'jenkins'
```

## Uso de AWS CodeBuild con Codecov

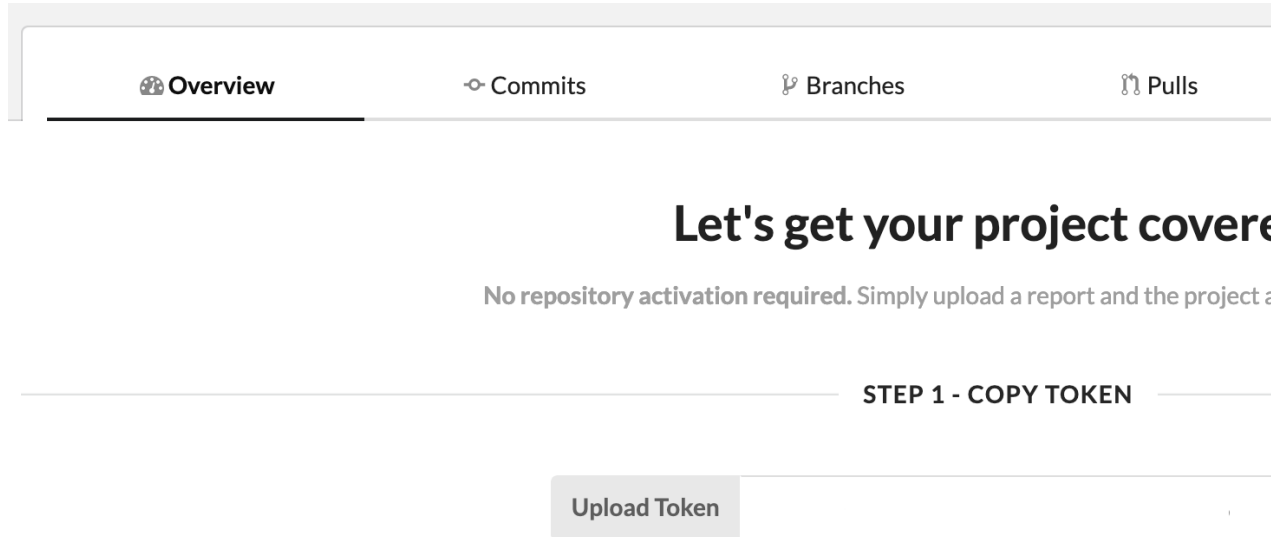
Codecov es una herramienta que mide la cobertura de las pruebas del código. Codecov identifica qué métodos e instrucciones del código no se han comprobado. Use los resultados para determinar dónde deben escribirse pruebas para mejorar la calidad del código. CodeCov está disponible en tres de los repositorios de origen admitidos por CodeBuild: GitHub, GitHub Enterprise Server y Bitbucket. Si el proyecto de compilación usa GitHub Enterprise Server, debe utilizar Codecov Enterprise.

Cuando se ejecuta una compilación de un proyecto de CodeBuild v que está integrado con Codecov, los informes de Codecov que analiza el código del repositorio se cargan en Codecov. Los registros de compilación incluyen un enlace a los informes. En este ejemplo, se muestra cómo se integra un proyecto de compilación de Python y Java con Codecov. Para ver una lista de los idiomas admitidos en Codecov, consulte [Codecov Supported Languages](#) en el sitio web de Codecov.

## Integrar Codecov en un proyecto de compilación

Para integrar Codecov en un proyecto de compilación

1. Vaya a <https://codecov.io/signup> y regístrese en un repositorio de origen de GitHub o Bitbucket. Si usa GitHub Enterprise, consulte [Codecov Enterprise](#) en el sitio web de Codecov.
2. En Codecov, agregue el repositorio para el que desea cobertura.
3. Cuando se muestre la información del token, elija Copy (Copiar).



4. Agregue el token copiado como una variable de entorno llamada CODECOV\_TOKEN al proyecto de compilación. Para obtener más información, consulte [Cambiar la configuración de un proyecto de compilación \(consola\)](#) (p. 254).
5. Cree un archivo de texto llamado `my_script.sh` en el repositorio. Copie lo siguiente en el archivo:

```
#!/bin/bash
bash <(curl -s https://codecov.io/bash) -t $CODECOV_TOKEN
```

6. Elija la pestaña Python o Java según corresponda en función de los usos del proyecto de compilación y siga estos pasos.

#### Java

1. Agregue el siguiente complemento de JacoCo a `pom.xml` en el repositorio.

```
<build>
 <plugins>
 <plugin>
 <groupId>org.jacoco</groupId>
 <artifactId>jacoco-maven-plugin</artifactId>
 <version>0.8.2</version>
 <executions>
 <execution>
 <goals>
 <goal>prepare-agent</goal>
 </goals>
 </execution>
 <execution>
 <id>report</id>
 <phase>test</phase>
 <goals>
 <goal>report</goal>
 </goals>
 </execution>
 </executions>
 </plugin>
 </plugins>
</build>
```

2. Especifique los siguientes comandos en el archivo `buildspec`. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

```
build:
 - mvn test -f pom.xml -fn
postbuild:
 - echo 'Connect to CodeCov'
 - bash my_script.sh
```

#### Python

Especifique los siguientes comandos en el archivo `buildspec`. Para obtener más información, consulte [Sintaxis de buildspec \(p. 137\)](#).

```
build:
 - pip install coverage
 - coverage run -m unittest discover
postbuild:
 - echo 'Connect to CodeCov'
 - bash my_script.sh
```

7. Ejecute una compilación del proyecto. En los registros de compilación aparece un enlace a los informes de Codecov generados para el proyecto. Utilice este enlace para ver los informes de Codecov. Para obtener más información, consulte [Ejecutar una compilación en AWS CodeBuild \(p. 281\)](#) y [Registrar llamadas a la API de AWS CodeBuild con AWS CloudTrail \(p. 333\)](#). La información de Codecov en los registros de compilación tiene el siguiente aspecto:

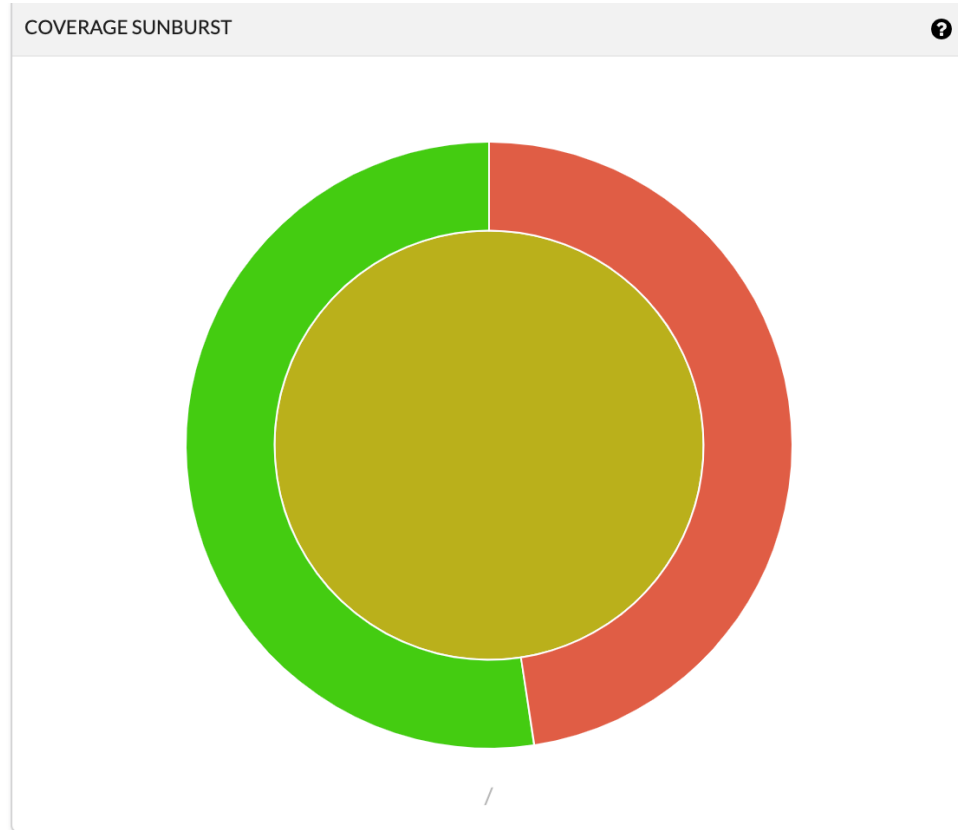
```
[Container] 2020/03/09 16:31:04 Running command bash my_script.sh
```

```
Codecov
Bash-20200303-bc4d7e6

•[0;90m==>•[0m AWS CodeBuild detected.
... The full list of Codecov log entries has been omitted for brevity ...
.
•[0;32m->•[0m View reports at •[0;36mhttps://codecov.io/github/user/test_py/commit/commit-id•[0m

[Container] 2020/03/09 16:31:07 Phase complete: POST_BUILD State: SUCCEEDED
```

Los informes tienen el siguiente aspecto:



Files	
<code>code.py</code>	10
<code>tests.py</code>	11
<b>Project Totals</b> (2 files)	21

## UsarAWS CodeBuildCon aplicaciones sin servidor

LaAWS Serverless Application Model(AWS SAM) es un marco de código abierto que permite crear aplicaciones sin servidor. Para obtener más información, consulte el repositorio del [modelo de aplicaciones sin servidor de AWS](#) en GitHub.

Puede utilizarAWS CodeBuildPara empaquetar e implementar aplicaciones sin servidor que sigan laAWS SAMEstándar. Para el paso de implementación, CodeBuild puede utilizarAWS CloudFormation. Para automatizar la creación e implementación de aplicaciones sin servidor con CodeBuild yAWS CloudFormationEn el caso de, puede utilizarAWS CodePipeline.



Para obtener más información, consulte [Implementación de aplicaciones sin servidores](#) en la AWS Serverless Application Model Guía para desarrolladores.

## Recursos relacionados

- Para obtener información acerca de cómo empezar a trabajar con AWS CodeBuild, consulte [Empezar a trabajar con AWS CodeBuild utilizando la consola \(p. 5\)](#).
- Para obtener información acerca de cómo solucionar problemas en CodeBuild, consulte [Solución de problemas de AWS CodeBuild \(p. 427\)](#).
- Para obtener información acerca de las cuotas de CodeBuild, consulte, consulte [Cuotas para AWS CodeBuild \(p. 444\)](#).

# Solución de problemas de AWS CodeBuild

Utilice la información de este tema como ayuda para identificar, diagnosticar y resolver problemas. Para obtener información sobre cómo registrar y monitorizar compilaciones de CodeBuild para solucionar problemas, consulte [Registro y monitorización](#) (p. 333).

## Temas

- [Apache Maven crea artefactos de referencia en un repositorio incorrecto](#) (p. 428)
- [De forma predeterminada, los comandos de compilación se ejecutan como usuario raíz](#) (p. 429)
- [Las compilaciones podrían registrar errores si los nombres de archivo tienen caracteres que no pertenecen al inglés \(Estados Unidos\)](#) (p. 429)
- [Las compilaciones pueden registrar errores al obtener parámetros del almacén de parámetros de Amazon EC2](#) (p. 430)
- [No se puede obtener acceso al filtro de ramificaciones en la consola CodeBuild](#) (p. 431)
- [No se puede ver si la compilación se ha realizado correctamente o no](#) (p. 431)
- [Estado de compilación no notificado al proveedor de origen](#) (p. 431)
- [No se puede encontrar ni seleccionar la imagen base de la plataforma Windows Server Core 2019](#) (p. 432)
- [Comandos anteriores de los archivos buildspec no reconocidos por comandos más recientes](#) (p. 432)
- [Error: «Access denied \(Acceso denegado\)» al intentar descargar la caché](#) (p. 432)
- [Error: «BUILD\\_CONTAINER\\_UNABLE\\_TO\\_PULL\\_IMAGE» al usar una imagen de compilación personalizada](#) (p. 433)
- [Error: Error: «Build container found dead before complete the build. build container died by it was out of memory, or the Docker image is not supported \(Se ha detectado que el contenedor de compilaciones está inactivo antes ErrorCode: 500"»](#) (p. 434)
- [Error: «No se puede conectar al demonio de Docker» al ejecutar una compilación](#) (p. 434)
- [Error: «CodeBuild is not authorized to perform: sts:AssumeRole» al crear o actualizar un proyecto de compilación](#) (p. 435)
- [Error: «Error al llamar a GetBucketAcl: El propietario del bucket ha cambiado o el rol de servicio ya no tiene permiso para llamar a s3:GetBucketAcl»](#) (p. 435)
- [Error: «Error al cargar artefactos: Arn» no válido al ejecutar una compilación](#) (p. 436)
- [Error: «El clon de Git falló: No se puede acceder'your-repository-URL': Problema de certificado SSL: Certificado autofirmado»](#) (p. 436)
- [Error: «The bucket you are are try to access must be address using the specified endpoint» al ejecutar una compilación](#) (p. 436)
- [Error: «The policy's default version was not created by enhanced zero click role creation or not the most recent version created by enhanced zero click role creation».](#) (p. 437)
- [Error: «Esta imagen de compilación requiere seleccionar al menos una versión de tiempo de ejecución».](#) (p. 437)
- [Error: «EN COLA: INSUFFICIENT\\_SUSNET» cuando se produce un error en una compilación de una cola de compilaciones](#) (p. 438)

- Error: «Unable to download cache: RequestError: Error de envío de solicitud causado por: x509: Error al cargar las raíces del sistema y no se han proporcionado raíces» (p. 439)
- Error: «Unable to download certificate from S3. AccessDenied" (p. 439)
- Error: «Unable to locate credentials» (p. 439)
- Error de tiempo de espera agotado de RequestError al ejecutar CodeBuild en un servidor proxy (p. 440)
- El shell de Bourne (sh) debe existir en las imágenes de compilación (p. 441)
- Advertencia: Al ejecutar una compilación, «Skipping install of runtimes. runtime version selection is not supported by this build image» (Si se omite la instalación de los entornos de ejecución (p. 442)
- Error: «No se puede verificar la identidad de JobWorker» al abrir la consola de CodeBuild (p. 442)
- No se ha podido iniciar la compilación (p. 442)
- Acceder a los metadatos de GitHub en compilaciones almacenadas en caché localmente (p. 442)
- AccessDenied: El propietario del bucket del grupo de informes no coincide con el propietario del bucket de S3... (p. 443)

## Apache Maven crea artefactos de referencia en un repositorio incorrecto

Problema: Cuando usa Maven con unAWS CodeBuildEl entorno de compilación de Java proporcionado por, Maven obtiene las dependencias de la compilación y los complementos del repositorio central seguro de Maven ubicado en<https://repo1.maven.org/maven2>. Esto es así aunque el archivo pom.xml del proyecto de compilación declare explícitamente otras ubicaciones para usar en su lugar.

Causa posible: Los entornos de compilación de Java proporcionados por Codebuild incluyen un archivo denominadosettings.xmlque está preinstalado en el entorno de compilación/root/.m2directorio. Este archivo settings.xml contiene las siguientes declaraciones, que indican a Maven que obtenga siempre las dependencias de la compilación y los complementos del repositorio central seguro de Maven disponible en <https://repo1.maven.org/maven2>.

```
<settings>
 <activeProfiles>
 <activeProfile>securecentral</activeProfile>
 </activeProfiles>
 <profiles>
 <profile>
 <id>securecentral</id>
 <repositories>
 <repository>
 <id>central</id>
 <url>https://repo1.maven.org/maven2</url>
 <releases>
 <enabled>true</enabled>
 </releases>
 </repository>
 </repositories>
 <pluginRepositories>
 <pluginRepository>
 <id>central</id>
 <url>https://repo1.maven.org/maven2</url>
 <releases>
 <enabled>true</enabled>
 </releases>
 </pluginRepository>
 </pluginRepositories>
 </profile>
 </profiles>
</settings>
```

```
</pluginRepositories>
</profile>
</profiles>
</settings>
```

Solución recomendada: Haga lo siguiente:

1. Añada un archivo `settings.xml` al código fuente.
2. En este archivo `settings.xml`, use el formato de `settings.xml` anterior como guía para declarar los repositorios de los que desee que Maven obtenga las dependencias de la compilación y los complementos en su lugar.
3. En el navegador `install` haga una fase de su proyecto de compilación, indique a CodeBuild que copie `settings.xml` al archivo `/root/.m2` directorio. Por ejemplo, considere el siguiente fragmento de un archivo `buildspec.yml` que ilustra este comportamiento.

```
version 0.2

phases:
 install:
 commands:
 - cp ./settings.xml /root/.m2/settings.xml
```

## De forma predeterminada, los comandos de compilación se ejecutan como usuario raíz

Problema: AWS CodeBuild ejecuta los comandos de compilación como usuario raíz. Esto ocurre incluso si el `Dockerfile` de la imagen de compilación correspondiente establece la instrucción `USER` en otro usuario.

Causa posible: De forma predeterminada, CodeBuild ejecuta todos los comandos de compilación como usuario raíz.

Solución recomendada: Ninguno.

## Las compilaciones podrían registrar errores si los nombres de archivo tienen caracteres que no pertenecen al inglés (Estados Unidos)

Problema: Cuando ejecuta una compilación que utiliza archivos con nombres de archivo que contienen caracteres que no pertenecen al inglés de Estados Unidos (por ejemplo, caracteres chinos), la compilación no se realiza correctamente.

Causa posible: Crear entornos proporcionados por AWS CodeBuild tienen su configuración regional predeterminada establecida en `POSIX`. La configuración de localización es menos compatible con CodeBuild y con los nombres de archivo que contienen caracteres del inglés de Estados Unidos, lo que puede provocar que las compilaciones relacionadas no se realicen correctamente.

Solución recomendada: Añada los siguientes comandos al `pre_build` en la sección `buildspec`. Estos comandos permiten que el entorno de compilación use US English UTF-8 para sus ajustes de localización,

lo que es más compatible con CodeBuild y con los nombres de archivo que no contienen caracteres caracteres que no pertenecen al inglés (Estados Unidos)

Para los entornos de compilación basados en Ubuntu:

```
pre_build:
 commands:
 - export LC_ALL="en_US.UTF-8"
 - locale-gen en_US en_US.UTF-8
 - dpkg-reconfigure locales
```

Para los entornos de compilación basados en Amazon Linux:

```
pre_build:
 commands:
 - export LC_ALL="en_US.utf8"
```

## Las compilaciones pueden registrar errores al obtener parámetros del almacén de parámetros de Amazon EC2

**Problema:** Cuando una compilación intenta obtener el valor de uno o varios parámetros almacenados en el almacén de parámetros de Amazon EC2, la compilación produce un error en la propiedad `DOWNLOAD_SOURCE` fase con el error `Parameter does not exist`.

**Causa posible:** El rol de servicio en el que se basa el proyecto de compilación no tiene permiso para llamar al método `ssm:GetParameters` Acción o el proyecto de compilación utiliza un rol de servicio generado por AWS CodeBuild y permite llamar al método `ssm:GetParameters`, pero los parámetros tienen nombres que no comienzan con `/CodeBuild/`.

**Soluciones recomendadas:**

- Si CodeBuild no ha generado el rol de servicio, actualice su definición para permitir que CodeBuild llame al método `ssm:GetParameters` acción. Por ejemplo, la siguiente instrucción de política permite llamar a la acción `ssm:GetParameters` para obtener parámetros con nombres que empiecen por `/CodeBuild/`:

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": "ssm:GetParameters",
 "Effect": "Allow",
 "Resource": "arn:aws:ssm:REGION_ID:ACCOUNT_ID:parameter/CodeBuild/*"
 }
]
}
```

- Si CodeBuild ha generado el rol de servicio, actualice su definición para permitir que CodeBuild tenga acceso a los parámetros del almacén de parámetros de Amazon EC2 con nombres que no empiecen por `/CodeBuild/`. Por ejemplo, la siguiente instrucción de política permite llamar a la acción `ssm:GetParameters` para obtener parámetros con el nombre especificado:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
 {
 "Action": "ssm:GetParameters",
 "Effect": "Allow",
 "Resource": "arn:aws:ssm:REGION_ID:ACCOUNT_ID:parameter/PARAMETER_NAME"
 }
]
```

## No se puede obtener acceso al filtro de ramificaciones en la consola CodeBuild

**Problema:** La opción del filtro de ramificaciones no está disponible en la consola al crear o actualizar una AWS CodeBuild Proyecto.

**Causa posible:** La opción del filtro de ramificaciones ha quedado en desuso. Se ha sustituido por grupos de filtros de webhook, que proporcionan más control sobre los eventos de webhook que desencadenan una nueva compilación en CodeBuild.

**Solución recomendada:** Para migrar un filtro de ramificaciones que creó antes de que aparecieran los filtros de webhook, cree un grupo de filtros de webhook con una `HEAD_REF` filtrar con la expresión regular `^refs/heads/branchName$`. Por ejemplo, si la expresión regular del filtro de ramificaciones era `^branchName$`, la expresión regular actualizada que debe insertar en el filtro `HEAD_REF` es `^refs/heads/branchName$`. Para obtener más información, consulte [Eventos de webhooks de Bitbucket \(p. 238\)](#) y [Filtrar eventos de webhooks en GitHub \(consola\) \(p. 247\)](#).

## No se puede ver si la compilación se ha realizado correctamente o no

**Problema:** No puede ver el éxito o el error de una compilación reprobada.

**Causa posible:** La opción para notificar el estado de la compilación no está habilitada.

**Soluciones recomendadas:** Habilitar `Informar el estado de compilación` al crear o actualizar un proyecto de CodeBuild. Esta opción indica a CodeBuild que le informe del estado cuando active una compilación. Para obtener más información, consulte [reportBuildStatus](#) en la Referencia de la API de AWS CodeBuild.

## Estado de compilación no notificado al proveedor de origen

**Problema:** Después de permitir la generación de informes de estado a un proveedor de origen, como GitHub o Bitbucket, el estado de compilación no se actualiza.

**Causa posible:** El usuario asociado con el proveedor de origen no tiene acceso de escritura al repositorio.

**Soluciones recomendadas:** Para poder informar del estado de compilación al proveedor de origen, el usuario asociado con el proveedor de origen debe tener acceso de escritura al repositorio. Si el usuario no

tiene acceso de escritura, el estado de compilación no se puede actualizar. Para obtener más información, consulte [Acceso al proveedor de origen \(p. 385\)](#).

## No se puede encontrar ni seleccionar la imagen base de la plataforma Windows Server Core 2019

**Problema:** No se puede encontrar ni seleccionar la imagen base de la plataforma Windows Server Core 2019.

**Causa posible:** Está utilizando unAWSRegión que no es compatible con esta imagen.

**Soluciones recomendadas:** Utilice una de las operaciones siguientesAWSRegiones compatibles con la imagen base de la plataforma Windows Server Core 2019:

- US East (N. Virginia)
- US East (Ohio)
- US West (Oregon)
- Europe (Ireland)

## Comandos anteriores de los archivos buildspec no reconocidos por comandos más recientes

**Problema:** Los resultados de uno o varios comandos del archivo buildspec no los reconocen los comandos posteriores del mismo archivo buildspec. Por ejemplo, un comando podría establecer una variable de entorno local y un comando ejecutado más tarde podría no ser capaz de obtener el valor de esa variable de entorno local.

**Causa posible:** En el archivo buildspec versión 0.1, AWS CodeBuild ejecuta cada comando en una instancia del shell predeterminado distinta en el entorno de compilación. Esto significa que cada comando se ejecuta con independencia de los demás. De forma predeterminada, no se puede ejecutar un comando que se base en el estado de un comando anterior.

**Soluciones recomendadas:** Le recomendamos que utilice la versión de especificación de compilación 0.2, que soluciona este problema. Si necesita utilizar la versión de especificación de compilación 0.1, le recomendamos que utilice el operador de encadenamiento de comandos del shell (por ejemplo, && en Linux) para combinar varios comandos en uno solo. También puede incluir un script del shell en el código fuente que contenga varios comandos y después llamar a dicho script desde un solo comando en el archivo buildspec. Para obtener más información, consulte [Shells y comandos de los entornos de compilación \(p. 171\)](#) y [Variables de entorno en los entornos de compilación \(p. 172\)](#).

## Error: «Access denied (Acceso denegado)» al intentar descargar la caché

**Problema:** Al intentar descargar la caché en un proyecto de compilación que tiene la caché habilitada, recibe unaAccess deniedERROR

**Causas posibles:**

- Ha configurado la caché como parte de su proyecto de compilación.
- La caché se ha invalidado recientemente mediante la API `InvalidateProjectCache`.
- El rol de servicio que se va a usar en CodeBuild no tiene los permisos `s3:GetObject` y `s3:PutObject` para el bucket de S3 en el que se aloja la caché.

Solución recomendada: Para el primer uso, es normal que aparezca este error inmediatamente después de actualizar la configuración de la caché. Si el error continúa, debe comprobar si el rol de servicio tiene los permisos `s3:GetObject` y `s3:PutObject` en el bucket de S3 donde se aloja la caché. Para obtener más información, consulte [Especificación de permisos de S3](#) en la Guía para desarrolladores de Amazon S3.

## Error: «BUILD\_CONTAINER\_UNABLE\_TO\_PULL\_IMAGE» al usar una imagen de compilación personalizada

Problema: Cuando intenta ejecutar una compilación que utiliza una imagen de compilación personalizada, la compilación produce el error `BUILD_CONTAINER_UNABLE_TO_PULL_IMAGE`.

Causa posible: El tamaño total sin comprimir de la imagen de compilación es mayor que el espacio en disco disponible del tipo de computación del entorno de compilación. Para comprobar el tamaño de la imagen de compilación, use Docker para ejecutar el comando `docker images REPOSITORY:TAG`. Para obtener una lista del espacio en disco disponible por tipo de computación, consulte [Tipos de computación del entorno de compilación](#) (p. 169).

Solución recomendada: Utilice un tipo de computación mayor con más espacio en disco o reduzca el tamaño de la imagen de compilación personalizada.

Causa posible: AWS CodeBuild no tiene permiso para extraer la imagen de compilación de Amazon Elastic Container Registry (Amazon ECR).

Solución recomendada: Actualice los permisos del repositorio de Amazon ECR para que CodeBuild pueda extraer la imagen de compilación personalizada en el entorno de compilación. Para obtener más información, consulte la [Ejemplo de Amazon ECR](#) (p. 46).

Causa posible: La imagen de Amazon ECR solicitada no está disponible en el AWS Región en la que AWS está usando.

Solución recomendada: Utilizar una imagen de Amazon ECR que esté en el mismo AWS Región como la que su AWS está usando.

Causa posible: Está utilizando un registro privado de una VPC que no tiene acceso público a Internet. CodeBuild no puede extraer ninguna imagen de una dirección IP privada de una VPC. Para obtener más información, consulte [Ejemplo de un registro privado con AWS Secrets Manager ejemplo de CodeBuild](#) (p. 128).

Solución recomendada: Si utiliza un registro privado de una VPC, asegúrese de que la VPC tiene acceso público a Internet.

Causa posible: Si el mensaje de error contiene «toomanyrequests», y la imagen se obtiene de Docker Hub, este error significa que se ha alcanzado el límite de extracción de Docker Hub.

Solución recomendada: Utilice un registro privado de Docker Hub u obtenga su imagen de Amazon ECR. Para obtener más información acerca del uso de un registro privado, consulte [Ejemplo de un registro privado con AWS Secrets Manager ejemplo de CodeBuild](#) (p. 128). Para obtener más información acerca del uso de Amazon ECR, consulte [Ejemplo de Amazon ECR para CodeBuild](#) (p. 46).



AWS CodeBuild Guía del usuario  
Error: Error: «Build container found dead before complete the build. build container died by it was out of memory, or the Docker image is not supported (Se ha detectado que el contenedor de compilaciones está inactivo antes Error Code: 500)

## Error: Error: «Build container found dead before complete the build. build container died by it was out of memory, or the Docker image is not supported (Se ha detectado que el contenedor de compilaciones está inactivo antes Error Code: 500"

Problema: Cuando intenta utilizar un contenedor de Microsoft Windows o Linux en AWS CodeBuild, este error se produce durante la fase de PROVISION.

Causas posibles:

- CodeBuild no admite la versión del sistema operativo del contenedor.
- Se ha especificado `HTTP_PROXY`, `HTTPS_PROXY` o ambos en el contenedor.

Soluciones recomendadas:

- En Microsoft Windows, utilice un contenedor de Windows con un sistema operativo de contenedor que tenga la versión `microsoft/windowsservercore:10.0.x`. Por ejemplo, `microsoft/windowsservercore:10.0.14393.2125`.
- En Linux, desactive las opciones `HTTPS_PROXY` y `HTTP_PROXY` en la imagen de Docker o especifique la configuración de la VPC del proyecto de compilación.

## Error: «No se puede conectar al demonio de Docker» al ejecutar una compilación

Problema: Error en la compilación y recibe un error similar `aCannot connect to the Docker daemon at unix:/var/run/docker.sock. Is the docker daemon running?` en el registro de compilación.

Causa posible: No está ejecutando la compilación en un modo privilegiado.

Solución recomendada: Siga estos pasos para ejecutar la compilación en un modo privilegiado:

1. Abra la consola de CodeBuild en <https://console.aws.amazon.com/codebuild/>.
2. En el panel de navegación, elija Build projects (Proyectos de compilación) y, a continuación, elija el proyecto de compilación.
3. En Edit (Editar), seleccione Environment (Entorno).
4. Elija Override images (Anular imágenes) y, a continuación, seleccione Environment (Entorno).
5. Especifique la imagen de entorno, el sistema operativo, el tiempo de ejecución y la imagen. Esta configuración debe coincidir con la configuración de la compilación que no se realizó correctamente.
6. Seleccione Privileged (Con privilegios).

Note

De forma predeterminada, los contenedores Docker no permiten el acceso a ningún dispositivo. El modo privilegiado otorga acceso al contenedor Docker de un proyecto de

compilación a todos los dispositivos. Para obtener más información, consulte la sección sobre [privilegios en tiempo de ejecución y capacidades de Linux](#) en el sitio web de Docker Docs.

7. Seleccione Update environment (Actualizar entorno).
8. Seleccione Start build (Comenzar compilación) para volver a intentar crear la compilación.

## Error: «CodeBuild is not authorized to perform: sts:AssumeRole» al crear o actualizar un proyecto de compilación

Problema: Al intentar crear o actualizar un proyecto de compilación, aparece el `errorCode:InvalidInputException, Message:CodeBuild is not authorized to perform: sts:AssumeRole on arn:aws:iam::account-ID:role/service-role-name`.

Causas posibles:

- AWS Security Token Service (AWS STS) se ha desactivado para la región de AWS en la que intenta crear o actualizar el proyecto de compilación.
- La AWS CodeBuild el rol de servicio de asociado con el proyecto de compilación no existe o no tiene permisos suficientes para confiar en CodeBuild.

Soluciones recomendadas:

- Asegúrese de que AWS STS está activado para la región de AWS en la que intenta crear o actualizar el proyecto de compilación. Para obtener más información, consulte [Activación y desactivación de AWS STS en una región de AWS](#) en la Guía del usuario de IAM.
- Asegúrese de que el rol de servicio CodeBuild de destino existe en el AWS account. Si no utiliza la consola, asegúrese de que no haya escrito incorrectamente el nombre de recurso de Amazon (ARN) del rol de servicio cuando creó o actualizó el proyecto de compilación.
- Asegúrese de que el rol de servicio de CodeBuild de destino tiene permisos suficientes para confiar en CodeBuild. Para obtener más información, consulte la instrucción de relación de confianza de políticas en [Cree un rol de servicio de CodeBuild \(p. 396\)](#).

## Error: «Error al llamar a GetBucketAcl: El propietario del bucket ha cambiado o el rol de servicio ya no tiene permiso para llamar a s3:GetBucketAcl»

Problema: Cuando ejecuta una compilación, recibe un error acerca de un cambio en la propiedad de un bucket de S3 y `GetBucketAcl` permisos.

Causa posible: Añadió `s3:GetBucketAcl` y `s3:GetBucketLocation` permisos a su rol de IAM. Estos permisos protegen el bucket de S3 del proyecto y garantizan que solo usted tenga acceso a él. Después de agregar estos permisos, el propietario del bucket de S3 cambió.

Solución recomendada: Compruebe que usted es el propietario del bucket de S3 y, a continuación, vuelva a agregar permisos a su rol de IAM. Para obtener más información, consulte [Acceso seguro a los buckets de S3 \(p. 358\)](#).

## Error: «Error al cargar artefactos: Arn» no válido al ejecutar una compilación

Problema: Cuando ejecuta una compilación, el `UPLOAD_ARTIFACTS` falla la fase de compilación con el error `Failed to upload artifacts: Invalid arn`.

Causa posible: Su cubo de salida S3 (el cubo donde AWS CodeBuild almacena su salida de la compilación) está en una AWS Región diferente del proyecto de compilación CodeBuild.

Solución recomendada: Actualice la configuración del proyecto de compilación para que apunte a un bucket que esté en la misma AWS Región como proyecto de compilación.

## Error: «El clon de Git falló: No se puede acceder 'your-repository-URL': Problema de certificado SSL: Certificado autofirmado»

Problema: Cuando intenta ejecutar un proyecto de compilación, se produce este error en la compilación.

Causa posible: El repositorio de código fuente tiene un certificado autofirmado, pero no ha elegido la opción para instalar el certificado desde el bucket de S3 como parte del proyecto de compilación.

Soluciones recomendadas:

- Edite el proyecto. En Certificate, elija `Install certificate from S3`. En `Bucket of certificate`, elija el bucket de S3 en el que esté almacenado el certificado SSL. En `Object key of certificate` (Clave de objeto del certificado), escriba el nombre de la clave de objeto de S3.
- Edite el proyecto. Seleccione `Insecure SSL` para omitir las advertencias de SSL al conectarse al repositorio del proyecto de GitHub Enterprise Server.

### Note

Le recomendamos que utilice `Insecure SSL` únicamente para pruebas. No debe utilizarse en un entorno de producción.

## Error: «The bucket you are are try to access must be address using the specified endpoint» al ejecutar una compilación

Problema: Cuando ejecuta una compilación, el `DOWNLOAD_SOURCE` falla la fase de compilación con el error `The bucket you are attempting to access must be addressed using the specified endpoint. Please send all future requests to this endpoint`.

Causa posible: El código fuente precompilado está almacenado en un bucket de S3 y ese bucket está en una AWS Región diferente de la AWS CodeBuild Proyecto de compilación.

Solución recomendada: Actualice la configuración del proyecto de compilación para que apunte a un bucket que contenga el código fuente precompilado. Asegúrese de que el bucket esté en la misma AWS Región como proyecto de compilación.

## Error: «The policy's default version was not created by enhanced zero click role creation or not the most recent version created by enhanced zero click role creation».

Problema: Al intentar actualizar un proyecto de la consola, se produce el siguiente error:

Causas posibles:

- Ha actualizado las políticas asociadas al rol de servicio de AWS CodeBuild de destino.
- Ha seleccionado una versión anterior de una política asociada al rol de servicio de CodeBuild de destino.

Soluciones recomendadas:

- Edite su proyecto CodeBuild y desactive la casilla Permitir que CodeBuild modifique este rol de servicio para poder usarlo con este proyecto de compilación. Compruebe que el rol de servicio de CodeBuild que está utilizando tiene permisos suficientes. Si vuelve a editar el proyecto de CodeBuild, debe desactivar esta casilla de verificación de nuevo. Para obtener más información, consulte [Cree un rol de servicio de CodeBuild \(p. 396\)](#).
- Siga estos pasos para editar el proyecto de CodeBuild y utilizar un nuevo rol de servicio:
  1. Abra la consola de IAM y cree un nuevo rol de servicio. Para obtener más información, consulte [Cree un rol de servicio de CodeBuild \(p. 396\)](#).
  2. Abra el icono AWS CodeBuild Consola de <https://console.aws.amazon.com/codesuite/codebuild/home>.
  3. En el panel de navegación, elija Build projects.
  4. Elija el botón situado junto al proyecto de compilación, elija Edit (Editar) y, a continuación, elija Environment (Entorno).
  5. En Service role (Rol de servicio), elija el rol que ha creado.
  6. Seleccione Update environment (Actualizar entorno).

## Error: «Esta imagen de compilación requiere seleccionar al menos una versión de tiempo de ejecución».

Problema: Cuando ejecuta una compilación, el `DOWNLOAD_SOURCE` falla la fase de compilación con el error `YAML_FILE_ERROR: This build image requires selecting at least one runtime version`.

Causa posible: La compilación utiliza la versión 1.0 o posterior de la imagen estándar de Amazon Linux 2 (AL2) o la versión 2.0 o posterior de la imagen estándar de Ubuntu y no se especifica un runtime en el archivo `buildspec`.

Solución recomendada: Si utiliza `elaws/codebuild/standard:2.0` en la imagen administrada por CodeBuild, debe especificar una versión de tiempo de ejecución en `runtime-versions` en el archivo `buildspec`. Por ejemplo, puede usar el siguiente archivo `buildspec` para un proyecto que usa PHP:

```
version: 0.2

phases:
 install:
 runtime-versions:
 php: 7.3
 build:
 commands:
 - php --version
artifacts:
 files:
 - README.md
```

#### Note

Si especifica `runtime-versions` la compilación emite la advertencia: «Skipping install of runtimes. Runtime version selection is not supported by this build image.»

Para obtener más información, consulte [Specify runtime versions in the buildspec file](#).

## Error: «EN COLA: INSUFFICIENT\_SUSNET» cuando se produce un error en una compilación de una cola de compilaciones

Problema: Error en una compilación de una cola de compilaciones: `QUEUED: INSUFFICIENT_SUBNET`.

Causas posibles: El bloque de CIDR de IPv4 especificado para la VPC utiliza una dirección IP reservada. Las cuatro primeras direcciones IP y la última dirección IP de cada bloque de CIDR de las subredes no se pueden utilizar y no se pueden asignar a ninguna instancia. Por ejemplo, en una subred con el bloque de CIDR `10.0.0.0/24`, estarán reservadas las cinco direcciones IP siguientes:

- `10.0.0.0`: dirección de red.
- `10.0.0.1`: reservada por AWS para el router de la VPC.
- `10.0.0.2`: reservada por AWS. La dirección IP del servidor DNS es siempre la base del intervalo de red VPC más dos; sin embargo, también se reserva la base de cada intervalo de subred más dos. En el caso de las VPC con varios bloques de CIDR, la dirección IP del servidor DNS se encuentra en el CIDR principal. Para obtener más información, consulte [Servidor DNS de Amazon](#) en la Guía del usuario de Amazon VPC.
- `10.0.0.3`: reservada por AWS para futura utilización.
- `10.0.0.255`: dirección de difusión de red. No admitimos la difusión en las VPC. Esta dirección está reservada.

Soluciones recomendadas: Compruebe si la VPC utiliza una dirección IP reservada. Reemplace las direcciones IP reservadas por otras que no estén reservadas. Para obtener más información, consulte [Tamaño de subred y VPC](#) en la Guía del usuario de Amazon VPC.

## Error: «Unable to download cache: RequestError: Error de envío de solicitud causado por: x509: Error al cargar las raíces del sistema y no se han proporcionado raíces»

Problema: Cuando intenta ejecutar un proyecto de compilación, se produce este error en la compilación.

Causa posible: Ha configurado el almacenamiento en caché como parte del proyecto de compilación y está utilizando una imagen de Docker antigua que incluye un certificado raíz caducado.

Solución recomendada: Actualice la imagen de Docker que se utiliza en su proyecto de AWS CodeBuild. Para obtener más información, consulte [Imágenes de Docker proporcionadas por CodeBuild \(p. 161\)](#).

## Error: «Unable to download certificate from S3. AccessDenied»

Problema: Cuando intenta ejecutar un proyecto de compilación, se produce este error en la compilación.

Causas posibles:

- Ha elegido el bucket de S3 erróneo para el certificado.
- Ha introducido la clave de objeto errónea para el certificado.

Soluciones recomendadas:

- Edite el proyecto. En Bucket of certificate, elija el bucket de S3 en el que esté almacenado el certificado SSL.
- Edite el proyecto. En Object key of certificate (Clave de objeto del certificado), escriba el nombre de la clave de objeto de S3.

## Error: «Unable to locate credentials»

Problema: Cuando intenta ejecutar el comando AWS CLI, use un AWS SDK, o llame a otro componente similar como parte de una compilación, pueden surgir errores de compilación relacionados directamente con la AWS CLI, AWS SDK o componente. Por ejemplo, puede obtener un error de compilación como `Unable to locate credentials`.

Causas posibles:

- La versión de la AWS CLI, el SDK de AWS o el componente del entorno de compilación no son compatibles con AWS CodeBuild.
- Está ejecutando un contenedor Docker en un entorno de compilación que usa Docker y, de forma predeterminada, ese contenedor no tiene acceso a las credenciales de AWS.

Soluciones recomendadas:

- Asegúrese de que el entorno de compilación tenga la siguiente versión o una versión superior de la AWS CLI, el SDK de AWS o el componente.
  - AWS CLI: 1.10.47
  - AWSSDK for C++: 0.2.19
  - AWSSDK for Go: 1.2.5
  - AWSSDK for Java: 1.11.16
  - AWSSDK para JavaScript: 2.4.7
  - AWSSDK for PHP: 3.18.28
  - AWSSDK for Python (Boto3): 1.4.0
  - AWSSDK for Ruby: 2.3.22
  - Botocore: 1.4.37
  - CoreCLR: 3.2.6-beta
  - Node.js: 2.4.7
- Si necesita ejecutar un contenedor Docker en un entorno de compilación y dicho contenedor necesita credenciales de AWS, debe pasar las credenciales del entorno de compilación al contenedor. En el archivo buildspec, incluya un comando `run` de Docker como el siguiente. En este ejemplo se utiliza el comando `aws s3 ls` para mostrar los buckets de S3 disponibles. La opción `-e` se pasa a través de las variables de entorno para que el contenedor tenga acceso a las credenciales de AWS.

```
docker run -e AWS_DEFAULT_REGION -e AWS_CONTAINER_CREDENTIALS_RELATIVE_URI your-image-tag
aws s3 ls
```

- Si está construyendo una imagen Docker y la compilación requiere AWS (por ejemplo, para descargar un archivo de Amazon S3), debe pasar las credenciales del entorno de compilación al proceso de compilación de Docker de la siguiente manera.
  1. En el Dockerfile del código fuente para la imagen de Docker, especifique las siguientes instrucciones ARG.

```
ARG AWS_DEFAULT_REGION
ARG AWS_CONTAINER_CREDENTIALS_RELATIVE_URI
```

2. En el archivo buildspec, incluya un comando `build` de Docker como el siguiente. Las opciones de `--build-arg` establecen las variables de entorno necesarias para que el proceso de compilación de Docker tenga acceso a las credenciales de AWS.

```
docker build --build-arg AWS_DEFAULT_REGION=$AWS_DEFAULT_REGION --build-arg
AWS_CONTAINER_CREDENTIALS_RELATIVE_URI=$AWS_CONTAINER_CREDENTIALS_RELATIVE_URI -
t your-image-tag .
```

## Error de tiempo de espera agotado de RequestError al ejecutar CodeBuild en un servidor proxy

Problema: Recibirá un `RequestError` similar a uno de los siguientes:

- `RequestError: send request failed caused by: Post https://logs.<your-region>.amazonaws.com/: dial tcp 52.46.158.105:443: i/o timeout` Logs de CloudWatch.
- `Error uploading artifacts: RequestError: send request failed caused by: Put https://<your-bucket>.s3.<your-aws-region>.amazonaws.com/*: dial tcp 52.219.96.208:443: connect: connection refused` Amazon S3.

Causas posibles:

- `ssl-bump` no está configurado correctamente.
- La política de seguridad de la organización no permite utilizar `ssl_bump`.
- El archivo `buildspec` no tiene una configuración de proxy especificada con un elemento `proxy`.

Soluciones recomendadas:

- Asegúrese de que `ssl-bump` esté configurado correctamente. Si utiliza Squid con el servidor proxy, consulte [Configuración de Squid como un servidor proxy explícito \(p. 191\)](#).
- Siga estos pasos para usar puntos de enlace privados para los registros de Amazon S3 y CloudWatch:
  1. En la tabla de ruteo de la subred privada, quite la regla que añadió y que dirige el tráfico destinado a Internet al servidor proxy. Para obtener información, consulte [Crear una subred en la VPC](#) en la Amazon VPC User Guide.
  2. Cree un punto de enlace de Amazon S3 privado y un punto de enlace de CloudWatch Logs y asócielos con la subred privada de su Amazon VPC. Para obtener información, consulte [Servicios de punto de conexión de la VPC](#) en la Amazon VPC User Guide.
  3. Confirmar/Habilitar nombre de DNS privado en su Amazon VPC está seleccionada. Para obtener más información, consulte [Creación de un punto de enlace de interfaz](#) en la Guía del usuario de Amazon VPC.
- Si no utiliza `ssl-bump` para un servidor de proxy explícito, añada una configuración de proxy al archivo `buildspec` con un elemento `proxy`. Para obtener más información, consulte [Ejecución de CodeBuild en un servidor proxy explícito \(p. 191\)](#) y [Sintaxis de buildspec \(p. 137\)](#).

```
version: 0.2
proxy:
 upload-artifacts: yes
 logs: yes
phases:
 build:
 commands:
```

## El shell de Bourne (sh) debe existir en las imágenes de compilación

Problema: Está utilizando una imagen de compilación que no la ha proporcionado AWS CodeBuild, y sus compilaciones fallan con el mensaje `Build container found dead before completing the build`.

Causa posible: El caparazón Bourne (`sh`) no está incluido en la imagen de compilación. Necesidades de CodeBuild `sh` Para ejecutar comandos de compilación y scripts.

Solución recomendada: `Ssh` No esté presente en la imagen de compilación, asegúrese de incluirlo antes de iniciar nuevas compilaciones que usen la imagen. (CodeBuild ya incluye `sh` en sus imágenes de compilación).



AWS CodeBuild Guía del usuario  
Advertencia: Al ejecutar una compilación, «Skipping  
install of runtimes. runtime version selection is  
not supported by this build image» (Si se omite  
la instalación de los entornos de ejecución

## Advertencia: Al ejecutar una compilación, «Skipping install of runtimes. runtime version selection is not supported by this build image» (Si se omite la instalación de los entornos de ejecución

Problema: Cuando ejecuta una compilación, el registro de compilación contiene esta advertencia.

Causa posible: La compilación no utiliza la versión 1.0 o posterior de la imagen estándar de Amazon Linux 2 (AL2) o la versión 2.0 o posterior de la imagen estándar de Ubuntu y se especifica un runtime en `unruntime-versions`En el archivo `buildspec`.

Solución recomendada: Asegúrese de que el archivo `buildspec` no contenga `unruntime-versions`sección. La `runtime-versions`La sección solo es necesaria si utiliza la imagen estándar de Amazon Linux 2 (AL2) o posterior o la versión 2.0 o posterior de la imagen estándar de Ubuntu.

## Error: «No se puede verificar la identidad de JobWorker» al abrir la consola de CodeBuild

Problema: Al abrir la consola de CodeBuild, aparece un mensaje de error «No se puede verificar la identidad de JobWorker».

Causa posible: El rol de IAM que se utiliza para el acceso a la consola tiene una etiqueta con `jobId` como la clave. Esta clave de etiqueta está reservada para CodeBuild y causará este error si está presente.

Solución recomendada: Cambiar las etiquetas de rol de IAM personalizadas que tengan la clave `jobId` para tener una clave diferente, como `jobIdentifier`.

## No se ha podido iniciar la compilación

Problema: Al iniciar una compilación, recibirá unNo se ha podido iniciar la compilaciónMensaje de error.

Causa posible: Se ha alcanzado el número de compilaciones concurrentes.

Soluciones recomendadas: Espere hasta que se completen otras compilaciones, o aumente el límite de compilación concurrente para el proyecto y vuelva a iniciar la compilación. Para obtener más información, consulte [Configuración de proyectos \(p. 198\)](#).

## Acceder a los metadatos de GitHub en compilaciones almacenadas en caché localmente

Problema: En algunos casos, el directorio `.git` de una compilación almacenada en caché es un archivo de texto y no un directorio.

Causas posibles: Cuando el almacenamiento en caché de origen local está habilitado para una compilación, CodeBuild crea un enlace `gitlink` para el `.git` directorio. Esto significa que la `.git`El directorio es en realidad un archivo de texto que contiene la ruta del directorio.

Soluciones recomendadas: En todos los casos, utilice el siguiente comando para obtener el directorio de metadatos de Git. Este comando funcionará sin importar el formato de .git:

```
git rev-parse --git-dir
```

## AccessDenied: El propietario del bucket del grupo de informes no coincide con el propietario del bucket de S3...

Problema: Al cargar datos de prueba en un bucket de Amazon S3, CodeBuild no puede escribir los datos de prueba en el bucket.

Causas posibles:

- La cuenta especificada para el propietario del bucket del grupo de informes no coincide con el propietario del bucket de Amazon S3.
- El rol de servicio no tiene acceso de escritura al bucket.

Soluciones recomendadas:

- Cambie el propietario del bucket del grupo de informes para que coincida con el propietario del bucket de Amazon S3.
- Modifique el rol de servicio para permitir acceso de escritura al bucket de Amazon S3.

# Cuotas para AWS CodeBuild

En las siguientes tablas, se indican las cuotas actuales de AWS CodeBuild. Estas cuotas se aplican a todas las regiones de AWS admitidas de cada cuenta de AWS, a menos que se especifique lo contrario.

## Cuotas de servicio

A continuación, se indican las cuotas predeterminadas para el AWS CodeBuild servicio.

Nombre	Valor predeterminado	Ajustable
Etiquetas asociadas por proyecto	Todas las regiones compatibles: 50	No
Proyectos de compilación	Todas las regiones compatibles: 5 000	Sí
Tiempo de espera de compilaciones en minutos	Todas las regiones compatibles: 480	No
Solicitud simultánea de información sobre compilaciones	Todas las regiones compatibles: 100	No
Solicitudes simultáneas de información sobre proyectos de compilación	Todas las regiones compatibles: 100	No
Compilaciones en ejecución simultáneas	Todas las regiones compatibles: 60	Sí
Período mínimo de tiempo de espera de compilación en minutos	Todas las regiones compatibles: 5	No
Grupos de seguridad en configuración de VPC	Todas las regiones compatibles: 5	No
Subredes bajo configuración de VPC	Todas las regiones compatibles: 16	No

Las cuotas del número máximo de compilaciones en ejecución simultáneas varían en función del tipo de computación. Para algunos tipos de computación y plataformas, el valor predeterminado es 20. Para una cuenta nueva, la cuota puede ser de tan solo 5. Para solicitar un incremento de la cuota de compilaciones simultáneas o si recibe el error «No se pueden tener más de X compilaciones activas para la cuenta», utilice el enlace anterior para realizar la solicitud.

## Otros límites

### Proyectos de compilación

Recurso	Valor predeterminado
Caracteres permitidos en una descripción de proyecto de compilación	Cualquiera
Caracteres permitidos en un nombre de proyecto de compilación	Las letras A–Z y a–z, los números 0–9 y los caracteres especiales – y _
Longitud de un nombre de proyecto de compilación	De 2 a 255 caracteres, inclusive
Longitud máxima de la descripción de un proyecto de compilación	255 caracteres
Número máximo de informes que puede añadir a un proyecto	5
Número de minutos que puede especificar en un proyecto de compilación para el tiempo de espera de la compilación de todas las compilaciones relacionadas	De 5 a 480 (8 horas)

### Builds

Recurso	Valor predeterminado
Tiempo máximo que se conserva el historial de una compilación	1 año
Número de minutos que puede especificar para el tiempo de espera de compilación de una sola compilación	De 5 a 480 (8 horas)

### Reports

Recurso	Valor predeterminado
La duración máxima de un informe de pruebas está disponible después de su creación	30 días
Número máximo de grupos de informes por cuenta de AWS	1 000
Número máximo de casos de prueba por informe	500

## Tags

Los límites de etiquetas se aplican a las etiquetas en los proyectos de compilación de CodeBuild y en los recursos del grupo de informes

Recurso	Valor predeterminado
Nombres de claves de etiquetas de recursos	<p>Cualquier combinación de letras, números, espacios y caracteres permitidos en UTF-8 con una longitud de entre 1 y 127 caracteres. Los caracteres permitidos son + - = . _ : / @</p> <p>Los nombres de clave de etiqueta deben ser únicos y cada clave solo puede tener un valor. Un nombre de clave de etiqueta no puede:</p> <ul style="list-style-type: none"> <li>• comenzar por aws :</li> <li>• contener únicamente espacios</li> <li>• terminar con un espacio</li> <li>• contener emojis ni ninguno de los siguientes caracteres: ? ^ * [ \ ~ ! # \$ % &amp; * ( ) &gt; &lt;   " ' ` [ ] { } ;</li> </ul>
Valores de las etiquetas de recursos	<p>Cualquier combinación de letras, números, espacios y caracteres permitidos en UTF-8 con una longitud de entre 0 y 255 caracteres. Los caracteres permitidos son + - = . _ : / @</p> <p>Una clave solo puede tener un valor, pero muchas claves pueden tener el mismo valor. Un valor de clave de etiqueta no puede contener emojis ni ninguno de los siguientes caracteres: ? ^ * [ \ ~ ! # \$ % &amp; * ( ) &gt; &lt;   " ' ` [ ] { } ;</p>

# Avisos de terceros de AWS CodeBuild para Windows

Cuando se utilizan compilaciones de CodeBuild para Windows, tiene la opción de utilizar algunos paquetes y módulos de terceros para hacer posible que su aplicación integrada se ejecute en sistemas operativos Microsoft Windows y para interoperar con algunos productos de terceros. La lista siguiente contiene los términos legales de terceros aplicables que rigen el uso de los paquetes y módulos de terceros especificados.

## Temas

- 1) imagen base Docker: windowsservercore (p. 447)
- 2) Imagen de Docker base de Windows - choco (p. 448)
- 3) Imagen de Docker base de Windows - git —versión 2.16.2 (p. 448)
- 4) Imagen de Docker base de Windows - microsoft-build-tools —versión 15.0.26320.2 (p. 449)
- 5) Imagen de Docker base de Windows - nuget.commandline —version 4.5.1 (p. 451)
- 7) Imagen de Docker base de Windows - netfx-4.6.2-devpack (p. 452)
- 8) Imagen de Docker base de Windows - visualsharpertools, v 4.0 (p. 453)
- 9) Imagen de Docker base de Windows - netfx-pcl-reference-assemblies-4.6 (p. 453)
- 10) Imagen de Docker base de Windows - visualcppbuildtools, v 14.0.25420.1 (p. 456)
- 11) Imagen de Docker base de Windows - microsoft-windows-netfx3-ondemand-package.cab (p. 458)
- 12) Imagen de Docker base de Windows - dotnet-sdk (p. 459)

## 1) imagen base Docker: windowsservercore

(términos de licencia disponibles en: <https://hub.docker.com/r/microsoft/windowsservercore/>)

Licencia: Al solicitar y utilizar esta imagen de SO de contenedor para contenedores de Windows, usted acepta, comprende y consiente los siguientes Términos de licencia complementarios:

### TÉRMINOS DE LICENCIA COMPLEMENTARIOS DEL SOFTWARE DE MICROSOFT

#### IMAGEN DE SO DE CONTENEDOR

Microsoft Corporation (o, según dónde viva, una de sus filiales) (a la que se hace referencia como "nosotros", "nos" o "Microsoft") le otorga la licencia de este complemento de imagen de sistema operativo de contenedor ("Complemento"). Se le concede la licencia para usar este Complemento junto con el software de sistema operativo host subyacente ("Software del host") exclusivamente para ayudar a ejecutar la característica de contenedores en el Software del host. Estos términos de licencia del Software del host se aplican al uso que usted realice del Complemento. No puede usarlo si no tiene una licencia para el Software del host. Puede usar este Complemento con cada copia con licencia válida del Software del host.

#### REQUISITOS Y/O DERECHOS DE USO DE LICENCIAS ADICIONALES

El uso del Complemento como se especifica en el párrafo anterior podría resultar en la creación o modificación de una imagen de contenedor ("Imagen de contenedor") que incluye algunos componentes del Complemento. Para mayor claridad, una Imagen de contenedor es independiente y distinta de una máquina virtual o de una imagen de aplicación virtual. En virtud de estos términos de licencia, te otorgamos un derecho restringido para redistribuir dichos componentes del Complemento en las siguientes condiciones:

- (i) puede usar los componentes del Complemento solo cuando se usan en y como parte de la Imagen de contenedor,
- (ii) puede usar esos componentes del complemento en la Imagen de contenedor siempre que tenga funcionalidad principal relevante en la Imagen de contenedor que sea materialmente independiente y distinta del Complemento; y
- (iii) acepta incluir estos términos de licencia (o términos similares requeridos por nosotros o un proveedor de servicios de hosting) con la Imagen de contenedor para licenciar correctamente el posible uso de los componentes del Complemento por parte de los usuarios finales.

Nos reservamos todos los demás derechos no otorgados expresamente en este documento.

Al hacer uso de este Complemento, acepta estos términos. Si no los acepta, no utilice este Complemento.

Como parte de los términos de licencia adicionales para esta imagen de sistema operativo de contenedor para los contenedores de Windows, también está sujeto a los términos de licencia subyacentes de software del host de Windows Server, que se encuentran en <https://www.microsoft.com/en-us/useterms>.

## 2) Imagen de Docker base de Windows - choco

(términos de licencia disponibles en: <https://github.com/chocolatey/choco/blob/master/LICENSE>)

Copyright 2011 - Presente RealDimensions Software, LLC

Con licencia bajo la licencia de Apache, versión 2.0 (la "Licencia"); no puede usar estos archivos excepto de conformidad con la Licencia. Puede obtener una copia de la Licencia en

<http://www.apache.org/licenses/LICENSE-2.0>

A menos que lo exija la legislación vigente o que se acuerde por escrito, el software distribuido bajo la licencia se distribuye "TAL CUAL", SIN GARANTÍAS NI CONDICIONES DE NINGÚN TIPO, ya sean explícitas o implícitas. Consulte la Licencia para conocer los permisos y limitaciones específicos en virtud de la Licencia.

## 3) Imagen de Docker base de Windows - git — versión 2.16.2

(términos de licencia disponibles en: <https://chocolatey.org/packages/git/2.16.2>)

Autorizado en virtud de GNU General Public License, versión 2, disponible en: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html>.

## 4) Imagen de Docker base de Windows - microsoft-build-tools —versión 15.0.26320.2

(términos de licencia disponibles en: <https://www.visualstudio.com/license-terms/mt171552/>)

EXTENSIONES DE MICROSOFT VISUAL STUDIO 2015, SHELL DE VISUAL STUDIO y C++  
REDISTRIBUTABLE

-----

Los presentes términos de licencia constituyen un acuerdo entre Microsoft Corporation (o, en función de donde resida, una de sus filiales) y usted. Se aplican al software citado anteriormente. Los términos también se aplican a cualquier servicio y actualización de Microsoft para el software, excepto en la medida que tengan términos diferentes.

-----

SI CUMPLE ESTOS TÉRMINOS DE LICENCIA, TIENE LOS DERECHOS QUE SE INDICAN A CONTINUACIÓN.

1. DERECHOS DE USO Y DE INSTALACIÓN. Puede instalar y utilizar cualquier número de copias del software.
2. TÉRMINOS PARA COMPONENTES ESPECÍFICOS.
  - a. Utilidades. El software podría contener algunos elementos en la Lista de utilidades en <https://docs.microsoft.com/en-us/visualstudio/productinfo/2015-redistribution-vs>. Puede copiar e instalar dichos elementos, si se incluyen con el software, en sus equipos o en los de terceros, para depurar e implementar sus aplicaciones y las bases de datos que ha desarrollado con el software. Tenga en cuenta que las Utilidades se han diseñado para uso temporal, que Microsoft podría no ser capaz de aplicar parches o actualizar las Utilidades por separado del resto del software y que algunas Utilidades por su naturaleza podrían hacer posible a otras personas acceder a los equipos en los que están instalados. En consecuencia, debería eliminar todas las Utilidades que haya instalado después de terminar de depurar o de implementar sus aplicaciones y bases de datos. Microsoft no es responsable del uso por parte de terceros o del acceso a las Utilidades que instale en cualquier equipo.
  - b. Plataformas de Microsoft. El software podría incluir componentes de Microsoft Windows; Microsoft Windows Server; Microsoft SQL Server; Microsoft Exchange; Microsoft Office y Microsoft SharePoint. Estos componentes se rigen por acuerdos independientes y por sus propias políticas de soporte de productos, tal y como se describe en los términos de licencia que se encuentran en el directorio de instalación de dicho componente o en la carpeta "Licencias" que acompaña al software.
  - c. Componentes de terceros. El software podría incluir componentes de terceros con avisos legales independientes o que se rigen por otros acuerdos, tal y como se describe en el archivo ThirdPartyNotices que acompaña al software. Incluso si tales componentes se rigen por otros acuerdos, también se aplican los descargos de responsabilidad y las limitaciones y exclusiones de daños siguientes. El software también podría incluir componentes sujetos a licencias de código abierto con obligaciones de disponibilidad de código fuente. Las copias de dichas licencias, si procede, se incluyen en el archivo ThirdPartyNotices. Puede obtener este código fuente correspondiente completo de Microsoft, si se requiere y tal como se requiera en virtud de las licencias de código abierto pertinente enviando un giro postal o cheque por valor de 5,00 USD a: Equipo de cumplimiento de código fuente, Microsoft Corporation, 1 Microsoft Way, Redmond, WA 98052. Escriba código fuente de uno o varios de los componentes indicados a continuación en la línea de memorando de su pago:
    - Herramientas remotas para Visual Studio 2015;
    - Generador de perfiles independiente para Visual Studio 2015;
    - IntelliTraceCollector para Visual Studio 2015;



- Microsoft VC++ redistribuible 2015;
- Biblioteca Multibyte MFC para Visual Studio 2015;
- Microsoft Build Tools 2015;
- Feedback Client;
- Shell integrado de Visual Studio 2015; o
- Shell aislado de Visual Studio 2015.

También podemos realizar una copia del código fuente disponible en <http://thirdpartysource.microsoft.com>.

3. **DATOS.** El software podría recopilar información acerca de usted y de su uso del software y enviarlo a Microsoft. Microsoft podría utilizar esta información para proporcionar servicios y mejorar nuestros productos y servicios. Es posible darse de baja en muchas de estas situaciones, pero no en todas, tal y como se describe en la documentación del producto. También hay algunas características en el software que le permitirían recopilar datos de los usuarios de sus aplicaciones. Si utiliza estas características para permitir la recopilación de datos en sus aplicaciones, debe cumplir con la legislación aplicable, que incluye la entrega de notificaciones pertinentes a los usuarios de sus aplicaciones. Puede obtener más información sobre la recopilación y el uso de datos en la documentación de ayuda y la declaración de privacidad en <https://privacy.microsoft.com/en-us/privacystatement>. El uso que hace del software funciona como su consentimiento para estas prácticas.
4. **ÁMBITO DE LA LICENCIA.** El software se cede bajo licencia y no es objeto de venta. Este acuerdo solo le otorga algunos derechos de uso del software. Microsoft se reserva todos los demás derechos. A menos que la legislación aplicable le otorgue más derechos a pesar de esta limitación, solo podrá utilizar el software tal como se permite expresamente en este acuerdo. Al hacerlo, deberá ajustarse a las limitaciones técnicas del software que solo permiten utilizarlo de determinadas formas. No podrá
  - eludir las limitaciones técnicas del software;
  - utilizar técnicas de ingeniería inversa, descompilar o desensamblar el software, así como tampoco intentar hacerlo, excepto y en la medida que lo exijan los términos de licencia de terceros que rigen el uso de ciertos componentes de código abierto que se podrían incluir con el software;
  - eliminar, minimizar, bloquear o modificar ninguna notificación de Microsoft o sus proveedores en el software;
  - utilizar el software de ninguna manera que esté en contra de la ley; o
  - compartir, publicar, alquilar o dar el software en préstamo, ni entregarlo como solución hospedada independiente para que otros lo utilicen.
5. **RESTRICCIONES EN MATERIA DE EXPORTACIÓN.** Debe cumplir con todas las leyes y reglamentos, nacionales e internacionales, en materia de exportación que sean de aplicación al software, lo que incluye restricciones en cuanto a destino, usuarios finales y uso final. Para obtener más información sobre las restricciones en materia de exportación, visite ([aka.ms/exporting](http://aka.ms/exporting)).
6. **SERVICIOS DE SOPORTE TÉCNICO.** Dado que este software se suministra "tal cual", es posible que no se proporcionen servicios de soporte técnico para el mismo.
7. **ACUERDO COMPLETO.** Este acuerdo y los términos aplicables a complementos, actualizaciones, servicios basados en Internet y servicios de soporte técnico que utilice constituyen el acuerdo completo para el software y los servicios de soporte técnico.
8. **LEGISLACIÓN APLICABLE.** Si adquirió el software en los Estados Unidos, la legislación del estado de Washington regirá la interpretación de este acuerdo, así como las reclamaciones por incumplimiento, y la legislación del estado donde usted resida regirá todas las demás reclamaciones. Si adquiere el software en otro país, se aplicará la legislación de dicho país.
9. **DERECHOS DEL CONSUMIDOR, VARIACIONES REGIONALES.** En este acuerdo se describen determinados derechos legales. Puede que usted tenga otros derechos, incluidos derechos del consumidor, según la legislación de su estado o país. Por separado e independientemente de su relación con Microsoft, también puede tener derechos con respecto a la parte de la que obtuvo el software. Este acuerdo no modifica los otros derechos de los que dispone en virtud de la legislación de su estado, país o región si dicha legislación no lo permite. Por ejemplo, si adquirió el software en una

de las regiones que figuran a continuación, o si se aplican leyes obligatorias del país o la región, las siguientes disposiciones se aplican a su caso:

- a. Australia. Tiene garantías legales previstas según la Ley de Consumo australiana y ninguna parte de este acuerdo pretende afectar a dichos derechos.
- b. Canadá. Si adquirió este software en Canadá, para dejar de recibir actualizaciones, puede desactivar la función de actualizaciones automáticas, desconectar su dispositivo de Internet (sin embargo, cuando vuelva a conectarse a Internet el software volverá a comprobar e instalar actualizaciones) o desinstalar el software. La documentación del producto, de haberla, también puede especificar cómo desactivar las actualizaciones para su dispositivo o software específico.
- c. Alemania y Austria.
  - i. Garantía. El software que disponga de la debida licencia funcionará esencialmente como se describe en la documentación de Microsoft que lo acompaña. Sin embargo, Microsoft no otorga ninguna garantía contractual en relación con el software licenciado.
  - ii. Limitación de responsabilidad. En caso de comportamiento intencional, negligencia grave, reclamaciones basadas en la Ley de Responsabilidad por Productos, así como en caso de muerte, daños corporales o lesiones físicas, Microsoft es responsable conforme a la legislación. Sujeto a la cláusula anterior (ii), Microsoft solo será responsable de negligencia leve si Microsoft incumple las obligaciones contractuales materiales, cuyo cumplimiento permite la debida ejecución de este acuerdo, y cuya infracción haría peligrar el propósito de este acuerdo y la confianza de una parte en su cumplimiento (denominadas "obligaciones cardinales"). En otros casos de negligencia leve, Microsoft no será responsable de dicha negligencia.

10 DESCARGO DE RESPONSABILIDAD DE LA GARANTÍA. EL SOFTWARE SE CONCEDE BAJO LICENCIA "TAL CUAL". USTED ASUME EL RIESGO DE UTILIZARLO. MICROSOFT NO PROPORCIONA NINGUNA GARANTÍA EXPRESA, ASÍ COMO TAMPOCO AVALES O CONDICIONES. EN LA MEDIDA QUE LO PERMITE LA LEGISLACIÓN LOCAL, MICROSOFT EXCLUYE LAS GARANTÍAS IMPLÍCITAS DE COMERCIABILIDAD, IDONEIDAD PARA UN DETERMINADO FIN Y NO INFRACCIÓN.

11 LIMITACIÓN Y EXCLUSIÓN DE LOS DAÑOS Y PERJUICIOS. PUEDE RECUPERAR DE MICROSOFT Y SUS PROVEEDORES ONLY DAÑOS DIRECTOS EE. UU. HASTA 5,00 USD. USTED NO PUEDE RECUPERAR NINGÚN OTRO DAÑO, INCLUYENDO CONSECUENTES, LUCRO CESANTE, DAÑOS ESPECIALES, INDIRECTOS O INCIDENTALES. Esta limitación se aplica a (a) nada relacionado con el software, servicios, contenido, (incluido código) en sitios de Internet o aplicaciones de terceros; y (b) reclamaciones por incumplimiento del contrato, incumplimiento de garantía o condición, responsabilidad estricta, negligencia u otros perjuicios en la medida permitida por la ley vigente.

También se aplicará incluso si Microsoft conocía o debería haber conocido la posibilidad de que se produjesen dichos daños. La limitación o exclusión precedente podría no aplicarse a su caso porque el país podría no admitir la exclusión o limitación de daños accidentales, resultantes o de otra índole.

ID EULA: VS2015\_Update3\_ShellsRedist\_ <ENU>

## 5) Imagen de Docker base de Windows - nuget.commandline —version 4.5.1

(términos de licencia disponibles en: <https://github.com/NuGet/Home/blob/dev/LICENSE.txt>)

Copyright (c) .NET Foundation. Todos los derechos reservados.

Con licencia bajo la licencia de Apache, versión 2.0 (la "Licencia"); no puede usar estos archivos excepto de conformidad con la Licencia. Puede obtener una copia de la Licencia en

<http://www.apache.org/licenses/LICENSE-2.0>

A menos que lo exija la legislación vigente o que se acuerde por escrito, el software distribuido bajo la licencia se distribuye "TAL CUAL", SIN GARANTÍAS NI CONDICIONES DE NINGÚN TIPO, ya sean explícitas o implícitas. Consulte la Licencia para conocer los permisos y limitaciones específicos en virtud de la Licencia.

## 7) Imagen de Docker base de Windows - netfx-4.6.2-devpack

### TÉRMINOS DE LICENCIA COMPLEMENTARIOS DEL SOFTWARE DE MICROSOFT

#### .NET FRAMEWORK 4.5 Y PAQUETES DE IDIOMAS ASOCIADOS PARA SISTEMA OPERATIVO MICROSOFT WINDOWS

-----

Microsoft Corporation (o, en función de donde resida, una de sus filiales) le otorga licencia para este complemento. Si posee licencias para utilizar software de sistema operativo Windows (el "software"), puede utilizar este complemento. No puede utilizarlo si no tiene una licencia para el software. Puede utilizar este complemento con cada copia de licencia válida del software.

Los siguientes términos de licencia describen términos de uso adicionales para este complemento. Los presentes términos y los términos de licencia para el software se aplican al uso que usted haga del complemento. Si hay un conflicto, se aplican estos términos de licencia complementarios.

AL HACER USO DE ESTE COMPLEMENTO, ACEPTA ESTOS TÉRMINOS. SI NO LOS ACEPTA, NO UTILICE ESTE COMPLEMENTO.

-----

Si cumple estos términos de licencia, tiene los derechos que se indican a continuación.

1. CÓDIGO DISTRIBUIBLE. El complemento se compone de Código distribuible. El "Código distribuible" es código que puede distribuir en los programas que desarrolle siempre que cumpla con los términos siguientes.
  - a. Derecho a Utilizar y Distribuir.
    - Puede copiar y distribuir el código objeto del complemento.
    - Distribución de Terceros. Puede permitir a los distribuidores de sus programas que copien y distribuyan el Código distribuible como parte de esos programas.
  - b. Requisitos de distribución. Para cualquier Código distribuible que distribuya, debe
    - agregarle una funcionalidad primaria significativa en sus programas;
    - para cualquier Código distribuible que tenga una extensión de nombre de archivo .lib, distribuya solo los resultados de ejecutar dicho Código distribuible a través de un vinculador con su programa;
    - distribuir Código distribuible incluido en un programa de instalación como parte de ese programa de instalación sin modificarlo;
    - exigir a los distribuidores y usuarios finales externos que acepten términos que lo protejan en la misma medida que este acuerdo;
    - mostrar su aviso de propiedad intelectual válido en sus programas; y
    - indemnizar, defender y eximir de responsabilidad a Microsoft de cualquier reclamación, incluidos honorarios de abogados, relacionados con la distribución o el uso de sus programas.
  - c. Restricciones de Distribución. No podrá
    - alterar ningún aviso de propiedad intelectual, marca o patente en el Código distribuible;
    - utilizar las marcas de Microsoft en los nombres de sus programas de una forma que sugiera que estos provienen de Microsoft o que esta los respalda;

- distribuir el Código distribuible para que se ejecute en una plataforma que no sea Windows;
  - incluir Código distribuible en programas malintencionados, engañosos o ilícitos; o
  - modificar o distribuir el código fuente de cualquier Código distribuible para que cualquier parte de él esté sujeta a una Licencia excluida. Una Licencia excluida es una que exige, como condición de uso, modificación o distribución que
    - el código se revele o distribuya en código fuente; u
    - otros tengan el derecho a modificarlo.
2. SERVICIOS DE SOPORTE TÉCNICO PARA EL COMPLEMENTO. Microsoft proporciona los servicios de soporte técnico para el presente software que se describen en [www.support.microsoft.com/common/international.aspx](http://www.support.microsoft.com/common/international.aspx).

## 8) Imagen de Docker base de Windows - visualfsharpools, v 4.0

(términos de licencia disponibles en:<https://github.com/dotnet/fsharp/blob/main/License.txt>)

Copyright (c) Microsoft Corporation. Todos los derechos reservados.

Con licencia bajo la licencia de Apache, versión 2.0 (la "Licencia"); no puede usar estos archivos excepto de conformidad con la Licencia. Puede obtener una copia de la Licencia en

<http://www.apache.org/licenses/LICENSE-2.0>

A menos que lo exija la legislación vigente o que se acuerde por escrito, el software distribuido bajo la licencia se distribuye "TAL CUAL", SIN GARANTÍAS NI CONDICIONES DE NINGÚN TIPO, ya sean explícitas o implícitas. Consulte la Licencia para conocer los permisos y limitaciones específicos en virtud de la Licencia.

## 9) Imagen de Docker base de Windows - netfx-pcl-reference-assemblies-4.6

TÉRMINOS DE LICENCIA DEL SOFTWARE DE MICROSOFT

ENSAMBLADOS DE REFERENCIA DE BIBLIOTECA DE CLASES PORTABLES DE MICROSOFT .NET - 4.6

-----

Los presentes términos de licencia constituyen un acuerdo entre Microsoft Corporation (o, en función de donde resida, una de sus filiales) y usted. Por favor, léalos. Se aplican al software citado anteriormente. Los términos también se aplican a

- actualizaciones,
- complementos,
- servicios basados en Internet, y
- servicios de soporte técnico

de Microsoft para este software, a menos que haya otros términos que acompañen a dichos elementos. En tal caso, se aplican dichos términos.

AL HACER USO DEL SOFTWARE, ACEPTA ESTOS TÉRMINOS. SI NO LOS ACEPTA, NO UTILICE EL SOFTWARE.

-----

SI CUMPLE ESTOS TÉRMINOS DE LICENCIA, TIENE LOS DERECHOS PERPETUOS QUE SE INDICAN A CONTINUACIÓN.

1. DERECHOS DE USO Y DE INSTALACIÓN. Puede instalar y utilizar cualquier número de copias del software para diseñar, desarrollar y probar los programas.
2. REQUISITOS Y/O DERECHOS DE USO DE LICENCIAS ADICIONALES
  - a. Código distribuible. Puede distribuir el software en programas de herramienta de desarrollador que usted desarrolle, para permitir a los clientes de sus programas desarrollar bibliotecas portables para utilizarlas en cualquier dispositivo o sistema operativo, siempre que cumpla con los términos siguientes.
    - i. Derecho a Utilizar y Distribuir. El software es "Código distribuible".
      - Código distribuible. Puede copiar y distribuir el código objeto del software.
      - Distribución de Terceros. Puede permitir a los distribuidores de sus programas que copien y distribuyan el Código distribuible como parte de esos programas.
    - ii. Requisitos de distribución. Para cualquier Código distribuible que distribuya, debe
      - agregarle una funcionalidad primaria significativa en sus programas;
      - exigir a los distribuidores y a sus clientes que acepten términos que lo protejan en la misma medida que este acuerdo;
      - mostrar su aviso de propiedad intelectual válido en sus programas; y
      - indemnizar, defender y eximir de responsabilidad a Microsoft de cualquier reclamación, incluidos honorarios de abogados, relacionados con la distribución o el uso de sus programas.
    - iii. Restricciones de Distribución. No podrá
      - alterar ningún aviso de propiedad intelectual, marca o patente en el Código distribuible;
      - utilizar las marcas de Microsoft en los nombres de sus programas de una forma que sugiera que estos provienen de Microsoft o que esta los respalda;
      - incluir Código distribuible en programas malintencionados, engañosos o ilícitos; o
      - modificar o distribuir el Código distribuible para que cualquier parte de él esté sujeta a una Licencia excluida. Una Licencia excluida es una que exige, como condición de uso, modificación o distribución que
        - el código se revele o distribuya en código fuente; u
        - otros tengan el derecho a modificarlo.
3. ÁMBITO DE LA LICENCIA. El software se cede bajo licencia y no es objeto de venta. Este acuerdo solo le otorga algunos derechos de uso del software. Microsoft se reserva todos los demás derechos. A menos que la legislación aplicable le otorgue más derechos a pesar de esta limitación, solo podrá utilizar el software tal como se permite expresamente en este acuerdo. Al hacerlo, deberá ajustarse a las limitaciones técnicas del software que solo permiten utilizarlo de determinadas formas. No podrá
  - eludir las limitaciones técnicas del software;
  - utilizar técnicas de ingeniería inversa, descompilar o desensamblar el software, excepto y en la medida que lo permita expresamente la legislación aplicable, a pesar de esta limitación;
  - publicar el software para que otros lo copien; o
  - alquilar o prestar el software.
4. COMENTARIOS. Puede proporcionar comentarios sobre el software. Si envía comentarios acerca de la aplicación a Microsoft, otorga a Microsoft, sin cargo alguno, el derecho de usar, compartir y comercializar sus comentarios de cualquier modo y para cualquier fin. También otorga a terceros, sin cargo alguno, los derechos de patente necesarios para que sus productos, tecnologías y servicios usen o interactúen con cualquier pieza específica de un software de Microsoft o un servicio que incluya los comentarios. Usted no enviará comentarios sujetos a licencia que obliguen a Microsoft a conceder

su software o documentación bajo licencia a terceros por el hecho de que se incluyan en ellos sus comentarios. Dichos derechos se mantendrán en vigor tras la finalización del presente acuerdo.

5. TRANSFERENCIA A TERCEROS. El primer usuario del software podrá transferirlo y este acuerdo, directamente a un tercero. Antes de la transferencia, dicha parte debe aceptar que este acuerdo se aplica a la transferencia y al uso del software. El primer usuario debe desinstalar el software antes de transferirlo por separado desde el dispositivo. El primer usuario no puede conservar ninguna copia.
6. RESTRICCIONES EN MATERIA DE EXPORTACIÓN. El software está sujeto a las leyes de exportación y a los reglamentos de los Estados Unidos. Debe cumplir todas las leyes y los reglamentos de exportación nacionales e internacionales que se aplican al software. Las leyes incluyen restricciones sobre destinos, usuarios finales y uso final. Para obtener información adicional, consulte [www.microsoft.com/exporting](http://www.microsoft.com/exporting).
7. SERVICIOS DE SOPORTE TÉCNICO. Dado que este software se suministra "tal cual", es posible que no se proporcionen servicios de soporte técnico para el mismo.
8. ACUERDO COMPLETO. Este acuerdo y los términos aplicables a complementos, actualizaciones, servicios basados en Internet y servicios de soporte técnico que utilice constituyen el acuerdo completo para el software y todos los servicios de soporte técnico que prestamos.
9. LEGISLACIÓN APLICABLE.
  - a. Estados Unidos. Si adquirió el software en los Estados Unidos, la legislación del estado de Washington regirá la interpretación de este acuerdo, así como las reclamaciones por incumplimiento, independientemente de los principios de conflicto de leyes. La legislación del estado donde reside rige todas las demás reclamaciones, incluidas las reclamaciones bajo las leyes de protección al consumidor del estado, leyes de competencia desleal o perjuicio.
  - b. Fuera de los Estados Unidos. Si adquiere el software en otro país, se aplicará la legislación de dicho país.
10. EFECTO LEGAL. En este acuerdo se describen determinados derechos legales. Es posible que tenga otros derechos en virtud de las leyes de su país. También podría tener derechos con respecto a la parte a la que adquirió el software. Este acuerdo no modifica sus derechos de los que dispone en virtud de la legislación de su país si dicha legislación no lo permite.
11. DESCARGO DE RESPONSABILIDAD DE LA GARANTÍA. EL SOFTWARE SE CONCEDE BAJO LICENCIA "TAL CUAL". USTED ASUME EL RIESGO DE UTILIZARLO. MICROSOFT NO PROPORCIONA NINGUNA GARANTÍA EXPRESA, ASÍ COMO TAMPOCO AVALES O CONDICIONES. PUEDE TENER DERECHOS DE CONSUMIDOR O GARANTÍAS LEGALES ADICIONALES SEGÚN SU LEGISLACIÓN LOCAL QUE ESTE ACUERDO NO PUEDE CAMBIAR. EN LA MEDIDA QUE LO PERMITE LA LEGISLACIÓN LOCAL, MICROSOFT EXCLUYE LAS GARANTÍAS IMPLÍCITAS DE COMERCIABILIDAD, IDONEIDAD PARA UN DETERMINADO FIN Y NO INFRACCIÓN.

PARA AUSTRALIA: TIENE GARANTÍA LEGALES PREVISTAS SEGÚN LA LEY DE CONSUMO AUSTRALIANA Y NINGUNA PARTE DE ESTOS TÉRMINOS PRETENDE AFECTAR A DICHOS DERECHOS.

12. LIMITACIÓN Y EXCLUSIÓN DE LOS RECURSOS Y LOS DAÑOS Y PERJUICIOS. PUEDE RECUPERAR DE MICROSOFT Y SUS PROVEEDORES ONLY DAÑOS DIRECTOS EE. UU. HASTA 5,00 USD. USTED NO PUEDE RECUPERAR NINGÚN OTRO DAÑO, INCLUYENDO CONSECUENTES, LUCRO CESANTE, DAÑOS ESPECIALES, INDIRECTOS O INCIDENTALES.

Esta limitación se aplica a

- cualquier aspecto relacionado con el software, los servicios, el contenido (incluido el código) en sitios de Internet de terceros o programas de terceros; y
- reclamaciones por infracción de contrato, infracción de garantía o condición, responsabilidad estricta, negligencia u otros perjuicios en la medida permitida por la legislación aplicable.

También se aplicará incluso si Microsoft conocía o debería haber conocido la posibilidad de que se produjesen dichos daños. La limitación o exclusión precedente podría no aplicarse a su caso porque el país podría no admitir la exclusión o limitación de daños accidentales, resultantes o de otra índole.

## 10) Imagen de Docker base de Windows - visualcppbuildtools, v 14.0.25420.1

(términos de licencia disponibles en: <https://www.visualstudio.com/license-terms/mt644918/>)

HERRAMIENTAS DE COMPILACIÓN DE MICROSOFT VISUAL C++

TÉRMINOS DE LICENCIA DEL SOFTWARE DE MICROSOFT

HERRAMIENTAS DE COMPILACIÓN DE MICROSOFT VISUAL C++

-----

Los presentes términos de licencia constituyen un acuerdo entre Microsoft Corporation (o, en función de donde resida, una de sus filiales) y usted. Se aplican al software citado anteriormente. Los términos también se aplican a cualquier servicio o actualizaciones de Microsoft para el software, salvo en la medida en que dispongan de distintos términos.

-----

SI CUMPLE ESTOS TÉRMINOS DE LICENCIA, TIENE LOS DERECHOS QUE SE INDICAN A CONTINUACIÓN.

### 1. DERECHOS DE USO Y DE INSTALACIÓN.

a. Un usuario puede utilizar copias del software para desarrollar y probar sus aplicaciones.

2. DATOS. El software podría recopilar información acerca de usted y de su uso del software y enviarlo a Microsoft. Microsoft podría utilizar esta información para proporcionar servicios y mejorar nuestros productos y servicios. Es posible darse de baja en muchas de estas situaciones, pero no en todas, tal y como se describe en la documentación del producto. También hay algunas características en el software que le permitirían recopilar datos de los usuarios de sus aplicaciones. Si utiliza estas características para permitir la recopilación de datos en sus aplicaciones, debe cumplir con la legislación aplicable, que incluye la entrega de notificaciones pertinentes a los usuarios de sus aplicaciones. Puede obtener más información sobre la recopilación y el uso de datos en la documentación de ayuda y en la declaración de privacidad en <http://go.microsoft.com/fwlink/?LinkID=528096>. El uso que hace del software funciona como su consentimiento para estas prácticas.

### 3. TÉRMINOS PARA COMPONENTES ESPECÍFICOS.

a. Implementaciones de servidor. El software podría contener algunos componentes del servidor de compilación en archivos BuildServer.TXT u otros archivos enumerados en la lista de BuildServer que sigue a estos Términos de licencia de Microsoft Software. Puede copiar e instalar dichos elementos, si se incluyen en el software, en sus máquinas de compilación. Usted y otros miembros de su organización podrían utilizar estos elementos en las máquinas de compilación únicamente con el fin de compilar, crear, verificar y archivar sus aplicaciones o para llevar a cabo pruebas de control de calidad o rendimiento como parte del proceso de compilación.

b. Plataformas de Microsoft. El software podría incluir componentes de Microsoft Windows; Microsoft Windows Server; Microsoft SQL Server; Microsoft Exchange; Microsoft Office y Microsoft SharePoint. Estos componentes se rigen por acuerdos independientes y por sus propias políticas de soporte de productos, tal y como se describe en los términos de licencia que se encuentran en el directorio de instalación de dicho componente o en la carpeta "Licencias" que acompaña al software.

c. Componentes de terceros. El software podría incluir componentes de terceros con avisos legales independientes o que se rigen por otros acuerdos, tal y como se describe en el archivo ThirdPartyNotices que acompaña al software. Incluso si tales componentes se rigen por otros acuerdos, también se aplican los descargos de responsabilidad y las limitaciones y exclusiones de daños siguientes.

d. Administradores de paquetes. El software podría incluir administradores de paquetes, como NuGet, que le ofrece la opción de descargar otros paquetes de software de Microsoft y de terceros para



utilizar con la aplicación. Estos paquetes incluyen sus propias licencias y no son objeto del presente acuerdo. Microsoft no distribuye, otorga licencia ni proporciona ninguna garantía para ningún paquete de terceros.

4. **ÁMBITO DE LA LICENCIA.** El software se cede bajo licencia y no es objeto de venta. Este acuerdo solo le otorga algunos derechos de uso del software. Microsoft se reserva todos los demás derechos. A menos que la legislación aplicable le otorgue más derechos a pesar de esta limitación, solo podrá utilizar el software tal como se permite expresamente en este acuerdo. Al hacerlo, deberá ajustarse a las limitaciones técnicas del software que solo permiten utilizarlo de determinadas formas. Para obtener más información, consulte <https://docs.microsoft.com/en-us/legal/information-protection/software-license-terms#1-installation-and-use-rights>. No podrá
  - eludir las limitaciones técnicas del software;
  - utilizar técnicas de ingeniería inversa, descompilar o desensamblar el software, así como tampoco intentar hacerlo, excepto y en la medida que lo exijan los términos de licencia de terceros que rigen el uso de ciertos componentes de código abierto que se podrían incluir con el software;
  - eliminar, minimizar, bloquear o modificar ninguna notificación de Microsoft o sus proveedores;
  - utilizar el software de ninguna manera que esté en contra de la ley; o
  - compartir, publicar, alquilar o dar el software en préstamo, ni entregarlo como solución hospedada independiente para que otros lo utilicen.
5. **RESTRICCIONES EN MATERIA DE EXPORTACIÓN.** Debe cumplir con todas las leyes y reglamentos, nacionales e internacionales, en materia de exportación que sean de aplicación al software, lo que incluye restricciones en cuanto a destino, usuarios finales y uso final. Para obtener más información sobre las restricciones en materia de exportación, visite ([aka.ms/exporting](https://aka.ms/exporting)).
6. **SERVICIOS DE SOPORTE TÉCNICO.** Dado que este software se suministra "tal cual", es posible que no se proporcionen servicios de soporte técnico para el mismo.
7. **ACUERDO COMPLETO.** Este acuerdo y los términos aplicables a complementos, actualizaciones, servicios basados en Internet y servicios de soporte técnico que utilice constituyen el acuerdo completo para el software y los servicios de soporte técnico.
8. **LEGISLACIÓN APLICABLE.** Si adquirió el software en los Estados Unidos, la legislación del estado de Washington regirá la interpretación de este acuerdo, así como las reclamaciones por incumplimiento, y la legislación del estado donde usted resida regirá todas las demás reclamaciones. Si adquiere el software en otro país, se aplicará la legislación de dicho país.
9. **DERECHOS DEL CONSUMIDOR, VARIACIONES REGIONALES.** En este acuerdo se describen determinados derechos legales. Puede que usted tenga otros derechos, incluidos derechos del consumidor, según la legislación de su estado o país. Por separado e independientemente de su relación con Microsoft, también puede tener derechos con respecto a la parte de la que obtuvo el software. Este acuerdo no modifica los otros derechos de los que dispone en virtud de la legislación de su estado, país o región si dicha legislación no lo permite. Por ejemplo, si adquirió el software en una de las regiones que figuran a continuación, o si se aplican leyes obligatorias del país o la región, las siguientes disposiciones se aplican a su caso:
  - **Australia.** Tiene garantías legales previstas según la Ley de Consumo australiana y ninguna parte de este acuerdo pretende afectar a dichos derechos.
  - **Canadá.** Si adquirió este software en Canadá, para dejar de recibir actualizaciones, puede desactivar la función de actualizaciones automáticas, desconectar su dispositivo de Internet (sin embargo, cuando vuelva a conectarse a Internet el software volverá a comprobar e instalar actualizaciones) o desinstalar el software. La documentación del producto, de haberla, también puede especificar cómo desactivar las actualizaciones para su dispositivo o software específico.
  - **Alemania y Austria.**
    - **Garantía.** El software que disponga de la debida licencia funcionará esencialmente como se describe en la documentación de Microsoft que lo acompaña. Sin embargo, Microsoft no otorga ninguna garantía contractual en relación con el software licenciado.
    - **Limitación de responsabilidad.** En caso de comportamiento intencional, negligencia grave, reclamaciones basadas en la Ley de Responsabilidad por Productos, así como en caso de muerte, daños corporales o lesiones físicas, Microsoft es responsable conforme a la legislación.



Sujeto a la cláusula anterior (ii), Microsoft solo será responsable de negligencia leve si Microsoft incumple las obligaciones contractuales materiales, cuyo cumplimiento permite la debida ejecución de este acuerdo, y cuya infracción haría peligrar el propósito de este acuerdo y la confianza de una parte en su cumplimiento (denominadas "obligaciones cardinales"). En otros casos de negligencia leve, Microsoft no será responsable de dicha negligencia.

10EFECTO LEGAL. En este acuerdo se describen determinados derechos legales. Es posible que tenga otros derechos en virtud de las leyes de su estado o país. Este acuerdo no modifica sus derechos de los que dispone en virtud de la legislación de su estado o país si dicha legislación no lo permite. Sin limitación de lo anterior, para Australia, TIENE GARANTÍA LEGALES PREVISTAS SEGÚN LA LEY DE CONSUMO AUSTRALIANA Y NINGUNA PARTE DE ESTOS TÉRMINOS PRETENDE AFECTAR A DICHS DERECHOS

11DESCARGO DE RESPONSABILIDAD DE LA GARANTÍA. EL SOFTWARE SE CONCEDE BAJO LICENCIA "TAL CUAL". USTED ASUME EL RIESGO DE UTILIZARLO. MICROSOFT NO PROPORCIONA NINGUNA GARANTÍA EXPRESA, ASÍ COMO TAMPOCO AVALES O CONDICIONES. EN LA MEDIDA QUE LO PERMITE LA LEGISLACIÓN LOCAL, MICROSOFT EXCLUYE LAS GARANTÍAS IMPLÍCITAS DE COMERCIABILIDAD, IDONEIDAD PARA UN DETERMINADO FIN Y NO INFRACCIÓN.

12LIMITACIÓN Y EXCLUSIÓN DE LOS DAÑOS Y PERJUICIOS. PUEDE RECUPERAR DE MICROSOFT Y SUS PROVEEDORES ONLY DAÑOS DIRECTOS EE. UU. HASTA 5,00 USD. USTED NO PUEDE RECUPERAR NINGÚN OTRO DAÑO, INCLUYENDO CONSECUENTES, LUCRO CESANTE, DAÑOS ESPECIALES, INDIRECTOS O INCIDENTALES.

Esta limitación se aplica a (a) nada relacionado con el software, servicios, contenido, (incluido código) en sitios de Internet o aplicaciones de terceros; y (b) reclamaciones por incumplimiento del contrato, incumplimiento de garantía o condición, responsabilidad estricta, negligencia u otros perjuicios en la medida permitida por la ley vigente.

También se aplicará incluso si Microsoft conocía o debería haber conocido la posibilidad de que se produjesen dichos daños. La limitación o exclusión precedente podría no aplicarse a su caso porque el país podría no admitir la exclusión o limitación de daños accidentales, resultantes o de otra índole.

## 11) Imagen de Docker base de Windows - microsoft-windows-netfx3-ondemand-package.cab

### TÉRMINOS DE LICENCIA COMPLEMENTARIOS DEL SOFTWARE DE MICROSOFT

MICROSOFT .NET FRAMEWORK 3.5 SP1 PARA SISTEMA OPERATIVO MICROSOFT WINDOWS

-----

Microsoft Corporation (o, en función de donde resida, una de sus filiales) le otorga licencia para este complemento. Si posee licencias para utilizar software de sistema operativo Windows (al que se aplica este complemento) (el "software"), puede utilizar este complemento. No puede utilizarlo si no tiene una licencia para el software. Puede utilizar una copia de este complemento con cada copia de licencia válida del software.

Los siguientes términos de licencia describen términos de uso adicionales para este complemento. Los presentes términos y los términos de licencia para el software se aplican al uso que usted haga del complemento. Si hay un conflicto, se aplican estos términos de licencia complementarios.

AL HACER USO DE ESTE COMPLEMENTO, ACEPTA ESTOS TÉRMINOS. SI NO LOS ACEPTA, NO UTILICE ESTE COMPLEMENTO.

-----

Si cumple estos términos de licencia, tiene los derechos que se indican a continuación.

1. SERVICIOS DE SOPORTE TÉCNICO PARA EL COMPLEMENTO. Microsoft proporciona los servicios de soporte técnico para el presente software que se describen en [www.support.microsoft.com/common/international.aspx](http://www.support.microsoft.com/common/international.aspx).
2. PRUEBAS DE BANCO DE PRUEBAS DE MICROSOFT .NET. Este software incluye los componentes .NET Framework, Windows Communication Foundation, Windows Presentation Foundation y Windows Workflow Foundation de los sistemas operativos Windows (Componentes .NET). Puede realizar pruebas internas de banco de pruebas de los componentes de .NET. Puede divulgar los resultados de cualquier prueba de banco de pruebas de los componentes de .NET, siempre que cumpla con las condiciones establecidas en <http://go.microsoft.com/fwlink/?LinkID=66406>.

Sin perjuicio de cualquier otro acuerdo que puede tener con Microsoft, si divulga los resultados de dichas pruebas de banco de pruebas, Microsoft tendrá el derecho a divulgar los resultados de las pruebas de referencia que realice de sus productos que compitan con el componente de .NET correspondiente, siempre que cumpla con las mismas condiciones establecidas en <http://go.microsoft.com/fwlink/?LinkID=66406>.

## 12) Imagen de Docker base de Windows - dotnet-sdk

(disponible en <https://github.com/dotnet/core/blob/main/LICENSE.TXT>)

La Licencia MIT (MIT)

Copyright (c) Microsoft Corporation

Se concede permiso, libre de cargos, a cualquier persona que obtenga una copia de este software y de los archivos de documentación asociados (el "Software"), para utilizar el Software sin restricción, incluyendo sin limitación los derechos a usar, copiar, modificar, fusionar, publicar, distribuir, sublicenciar, y/o vender copias del Software, y a permitir a las personas a las que se les proporcione el Software a hacer lo mismo, sujeto a las siguientes condiciones:

El aviso de copyright anterior y este aviso de permiso se incluirán en todas las copias o partes sustanciales del Software.

EL SOFTWARE SE PROPORCIONA "TAL CUAL", SIN GARANTÍAS DE NINGÚN TIPO, EXPLÍCITAS NI IMPLÍCITAS, LO QUE INCLUYE, SIN QUE SIRVA DE LIMITACIÓN, LAS GARANTÍAS DE COMERCIALIZACIÓN, IDONEIDAD PARA UN FIN DETERMINADO Y AUSENCIA DE INFRACCIÓN. LOS AUTORES O TITULARES DE LOS DERECHOS DE AUTOR (COPYRIGHT) NO SERÁN RESPONSABLES EN NINGÚN CASO ANTE NINGUNA RECLAMACIÓN, DAÑO, PERJUICIO NI RESPONSABILIDAD DE NINGÚN OTRO TIPO, YA SEA CONTRACTUAL, EXTRA CONTRACTUAL O DE OTRA ÍNDOLE, QUE SE DERIVEN DEL SOFTWARE O ESTÉN RELACIONADOS CON ÉL, CON SU UTILIZACIÓN O CON OTROS USOS DE ESTE.

# Historial de revisiones de la guía del usuario de AWS CodeBuild

En la siguiente tabla se describen los cambios importantes que se han realizado en la documentación desde la última versión de AWS CodeBuild. Para obtener notificaciones sobre las actualizaciones de esta documentación, puede suscribirse a una fuente RSS.

- Última versión de la API: 06/10/2016
- Última actualización de la documentación: 4 de octubre de 2021

update-history-change	update-history-description	update-history-date
<a href="#">Modo de informe por lotes (p. 279)</a>	Ahora CodeBuild le permite seleccionar cómo se envían los estados de compilación por lotes al proveedor de origen de un proyecto. Para obtener más información, consulte <a href="#">Modo de informe por lotes</a> .	4 de octubre de 2021
<a href="#">Nuevo tipo de computación (p. 169)</a>	CodeBuild ahora admite un pequeño tipo de cómputo ARM. Para obtener más información, consulte <a href="#">Tipos de computación del entorno de compilación</a> .	13 de septiembre de 2021
<a href="#">Proyectos de compilación pública (p. 279)</a>	Ahora CodeBuild le permite hacer que los resultados de compilación de sus proyectos de compilación estén disponibles para el público sin necesidad de acceder a un AWS account. Para obtener más información, consulte <a href="#">Proyectos de compilación pública</a> .	11 de agosto de 2021
<a href="#">Depuración de sesiones para compilaciones por lotes (p. 460)</a>	CodeBuild ahora admite la depuración de sesiones para compilaciones por lotes. Para obtener más información, consulte <a href="#">build-graphylista de compilación</a> .	3 de marzo de 2021
<a href="#">Límite de compilación concurrente a nivel (p. 460)</a>	Ahora CodeBuild le permite limitar el número de compilaciones simultáneas de un proyecto de compilación. Para obtener más información, consulte <a href="#">Configuración del proyecto</a> y <a href="#">ConcurrentBuildLimit</a> .	16 de febrero de 2021

<a href="#">Nueva propiedad buildspec: s3-prefijo (p. 460)</a>	CodeBuild proporciona ahora el <code>s3-prefix</code> propiedad buildspec para artefactos que permite especificar un prefijo de ruta para los artefactos que se cargan en Amazon S3. Para obtener más información, consulte <a href="#">prefijo s3</a> .	9 de febrero de 2021
<a href="#">Nueva propiedad buildspec: en caso de fallo (p. 460)</a>	CodeBuild proporciona ahora el <code>on-failure</code> propiedad buildspec para fases de compilación que le permite determinar qué ocurre cuando falla una fase de compilación. Para obtener más información, consulte <a href="#">en caso de error</a> .	9 de febrero de 2021
<a href="#">Nueva propiedad buildspec: exclude-paths (p. 460)</a>	CodeBuild proporciona ahora el <code>exclude-paths</code> propiedad buildspec para artefactos que le permite excluir rutas de los artefactos de compilación. Para obtener más información, consulte <a href="#">rutas de exclusión</a> .	9 de febrero de 2021
<a href="#">Nueva propiedad buildspec: habilitar enlaces simbólicos (p. 460)</a>	CodeBuild proporciona ahora el <code>enable-symlinks</code> propiedad buildspec para artefactos que permite conservar vínculos simbólicos en un artefacto ZIP. Para obtener más información, consulte <a href="#">habilitar enlaces simbólicos</a> .	9 de febrero de 2021
<a href="#">Mejora del nombre del artefacto Buildspec (p. 460)</a>	CodeBuild ahora permite el <code>artifacts/name</code> propiedad para contener información de ruta. Para obtener más información, consulte <a href="#">Nombre de</a> .	9 de febrero de 2021
<a href="#">Informes de cobertura de códigos (p. 460)</a>	CodeBuild ahora proporciona informes de cobertura de código. Para obtener más información, consulte <a href="#">Informes de cobertura de códigos</a> .	30 de julio de 2020
<a href="#">Compilaciones Batch lotes (p. 460)</a>	CodeBuild ahora admite la ejecución de compilaciones simultáneas y coordinadas de un proyecto. Para obtener más información, consulte <a href="#">Generaciones Batch en CodeBuild</a> .	30 de julio de 2020

<a href="#">Imagen de Windows Server 2019 (p. 460)</a>	CodeBuild ahora proporciona una imagen de compilación de Windows Server Core 2019. Para obtener más información, consulte <a href="#">Imágenes de Docker proporcionadas por CodeBuild</a> .	20 de julio de 2020
<a href="#">Administrador de sesiones (p. 460)</a>	Ahora CodeBuild te permite pausar una compilación en ejecución y luego usar AWS Systems Manager Session Manager para conectarse al contenedor de compilación y ver el estado del contenedor. Para obtener más información, consulte <a href="#">Administrador de sesiones</a> .	20 de julio de 2020
<a href="#">Tema actualizado (p. 460)</a>	CodeBuild ahora admite la especificación de un shell para usar en sus entornos de compilación en el archivo buildspec. Para obtener más información, consulte <a href="#">Referencia de especificaciones de compilación</a> .	25 de junio de 2020
<a href="#">Generación de informes de pruebas con marcos de pruebas (p. 460)</a>	Se han agregado varios temas que describen cómo generar informes de pruebas de CodeBuild con varios marcos de pruebas. Para obtener más información, consulte <a href="#">Generación de informes de pruebas con marcos de pruebas</a> .	29 mayo de 2020
<a href="#">Temas actualizados (p. 460)</a>	Ahora, CodeBuild admite agregar etiquetas a grupos de informes. Para obtener más información, consulte <a href="#">ReportGroup</a> .	21 de mayo de 2020
<a href="#">Compatibilidad de informes de prueba (p. 460)</a>	La compatibilidad con CodeBuild para informes de prueba ya está disponible con carácter general.	21 de mayo de 2020

<a href="#">Temas actualizados (p. 460)</a>	CodeBuild ahora admite la creación de filtros webhook para Github y Bitbucket que desencadenan compilaciones solo cuando el mensaje de confirmación del encabezado coincide con la expresión especificada. Para obtener más información, consulte la <a href="#">solicitud de extracción de GitHub</a> y el <a href="#">ejemplo de filtro de webhook</a> y la <a href="#">solicitud de extracción de Bitbucket</a> y el <a href="#">ejemplo de filtro de webhook</a> .	6 de mayo de 2020
<a href="#">Temas nuevos (p. 460)</a>	ahora admite el uso compartido de recursos de proyectos de compilación y de grupos de informes. Para obtener más información, consulte <a href="#">Uso de proyectos compartidos</a> y <a href="#">Uso de grupos de informes compartidos</a> .	13 de diciembre de 2019
<a href="#">Temas nuevos y actualizados (p. 460)</a>	CodeBuild ahora admite los informes de pruebas durante la ejecución de un proyecto de compilación. Para obtener más información, consulte <a href="#">Uso de informes de pruebas</a> , <a href="#">Crear un informe de pruebas</a> y <a href="#">Crear un informe de pruebas mediante la muestra de AWS CLI</a> .	25 de noviembre de 2019
<a href="#">Tema actualizado (p. 460)</a>	Ahora, CodeBuild admite los tipos de entorno GPU y Arm de Linux, y <code>2xlarge</code> Tipo de computación. Para obtener más información, consulte <a href="#">Tipos de computación del entorno de compilación</a> .	19 de noviembre de 2019
<a href="#">Temas actualizados (p. 460)</a>	Ahora, CodeBuild admite números de compilación en todas las compilaciones, exportación de variables de entorno y AWS Secrets Manager Integración de. Para obtener más información, consulte la información sobre <a href="#">variables exportadas</a> y <a href="#">Secrets Manager</a> en <a href="#">Sintaxis de la especificación de compilación</a> .	6 de noviembre de 2019

<a href="#">Nuevo tema (p. 460)</a>	Ahora, CodeBuild admite reglas de notificación. Puede utilizar las reglas de notificación para notificar a los usuarios los cambios importantes en los proyectos de compilación. Para obtener más información, consulte <a href="#">Create a notification rule</a> .	5 de noviembre de 2019
<a href="#">Temas actualizados (p. 460)</a>	ahora es compatible con los runtimes de Android versión 29 y Go versión 1.13. Para obtener más información, consulte <a href="#">Imágenes de Docker proporcionadas por CodeBuild</a> y <a href="#">Sintaxis de buildspec</a> .	10 de septiembre de 2019
<a href="#">Temas actualizados (p. 460)</a>	Al crear un proyecto, ahora puede elegir la imagen administrada de Amazon Linux 2 (AL2). Para obtener más información, consulte <a href="#">Imágenes de Docker proporcionadas por CodeBuild</a> y <a href="#">Versiones del entorno de ejecución en el ejemplo del archivo buildspec de CodeBuild</a> .	16 de agosto de 2019
<a href="#">Tema actualizado (p. 460)</a>	Cuando cree un proyecto, ahora podrá deshabilitar el cifrado de los registros de S3 y, si utiliza un repositorio de origen basado en Git, incluir submódulos de Git. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación en CodeBuild</a> .	8 de marzo de 2019
<a href="#">Nuevo tema (p. 460)</a>	Ahora, CodeBuild admite el almacenamiento en caché local. Cuando cree una compilación, podrá especificar uno de los cuatro modos de almacenamiento en caché local. Para obtener más información, consulte <a href="#">Almacenamiento en caché de compilaciones en Code</a> .	21 de febrero de 2019
<a href="#">Temas nuevos (p. 460)</a>	CodeBuild ahora admite grupos de filtros de webhooks para especificar eventos que desencadenan una compilación. Para obtener más información, consulte <a href="#">Filtrar eventos de webhooks de GitHub</a> y <a href="#">Filtrar eventos de webhooks de Bitbucket</a> .	8 de febrero de 2019

<a href="#">Nuevo tema (p. 460)</a>	En la Guía del usuario de CodeBuild, se explica ahora cómo utilizar CodeBuild con un servidor proxy. Para obtener más información, consulte <a href="#">Uso de CodeBuild con un servidor proxy</a> .	4 de febrero de 2019
<a href="#">Temas actualizados (p. 460)</a>	Ahora, CodeBuild admite el uso de una imagen Amazon ECR que está en otra AWS account. Se han actualizado varios temas para reflejar este cambio, como <a href="#">Ejemplo de Amazon ECR para CodeBuild</a> , <a href="#">Creación de un proyecto de compilación</a> , y <a href="#">Crear un rol de servicio de CodeBuilding</a> .	24 de enero de 2019
<a href="#">Compatibilidad con registros de Docker privados (p. 460)</a>	CodeBuild ahora admite el uso de una imagen de Docker almacenada en un registro privado como entorno de ejecución de. Para obtener más información, consulte <a href="#">Registro privado con AWS Secrets Manager</a> <a href="#">Ejemplo de</a> .	24 de enero de 2019
<a href="#">Tema actualizado (p. 460)</a>	CodeBuild ahora admite el uso de un token de acceso para conectar a repositorios de GitHub (con un token de acceso personal) y Bitbucket (con una contraseña de aplicación). Para obtener más información, consulte la sección <a href="#">Crear un proyecto de compilación (consola)</a> y <a href="#">Uso de tokens de acceso con su proveedor de código fuente</a> .	6 de diciembre de 2018
<a href="#">Tema actualizado (p. 460)</a>	Ahora, CodeBuild admite nuevas métricas de compilación que miden la duración de cada fase en una compilación. Para obtener más información, consulte <a href="#">Métricas de CloudWatch</a> .	15 de noviembre de 2018
<a href="#">Tema de la política de puntos de enlace de la VPC (p. 460)</a>	Los puntos de enlace de Amazon VPC para CodeBuild ahora admiten políticas. Para obtener más información, consulte <a href="#">Creación de una política de puntos de enlace de la VPC para CodeBuild</a> .	9 de noviembre de 2018
<a href="#">Contenido actualizado (p. 460)</a>	Los temas se han actualizado para reflejar la experiencia con la nueva consola.	30 de octubre de 2018



<a href="#">Ejemplo de Amazon EFS (p. 460)</a>	CodeBuild puede montar un sistema de archivos de Amazon EFS durante una compilación utilizando los comandos de un archivo buildspec del proyecto. Para obtener más información, consulte <a href="#">Ejemplo de Amazon EFS para CodeBuild</a> .	26 de octubre de 2018
<a href="#">Webhooks de BitBucket (p. 460)</a>	CodeBuild ahora admite webhooks cuando utiliza BitBucket para su repositorio. Para obtener más información, consulte <a href="#">Ejemplo de solicitud de extracción de Bitbucket para CodeBuild</a> .	2 de octubre de 2018
<a href="#">Registros de S3 (p. 460)</a>	Ahora, CodeBuild admite los registros de compilación en un bucket de S3. Anteriormente, únicamente se podían compilar registros con CloudWatch Logs. Para obtener más información, consulte <a href="#">Creación de un proyecto</a> .	17 de septiembre de 2018
<a href="#">Varios orígenes de entrada y varios artefactos de salida (p. 460)</a>	Ahora, CodeBuild admite proyectos que utilizan más de un origen de entrada y que publican más de un conjunto de artefactos. Para obtener más información, consulte <a href="#">Ejemplo de varios orígenes de entrada y varios artefactos de entrada</a> y <a href="#">Ejemplo de integración de CodePipeline con CodeBuild y varias fuentes de entrada y varios artefactos de salida</a> .	30 de agosto de 2018
<a href="#">Ejemplo de control de versiones semánticas (p. 460)</a>	La Guía del usuario de CodeBuild incluye ahora un ejemplo basado en casos de uso que muestra cómo utilizar el control de versiones semánticas para nombres de artefacto en el momento de la compilación. Para obtener más información, consulte <a href="#">Ejemplo para utilizar el control de versiones semánticas para asignar nombre a artefactos de compilación</a> .	14 de agosto de 2018

<a href="#">Ejemplo de nuevo sitio web estático (p. 460)</a>	La Guía del usuario de CodeBuild incluye ahora un ejemplo basado en casos de uso que muestra cómo alojar la salida de la compilación en un bucket de S3. El ejemplo utiliza casos de soporte recientes sobre los artefactos de compilación sin cifrar. Para obtener más información, consulte <a href="#">Crear un sitio web estático con la salida de compilación alojada en un bucket de Amazon S3</a> .	14 de agosto de 2018
<a href="#">Soporte para la anulación de un nombre de artefacto con el control de versiones semánticas (p. 460)</a>	Ahora puede usar el control de versiones semánticas para especificar un formato que CodeBuild utiliza para asignar nombre a artefactos de compilación. Esto resulta útil porque un artefacto de compilación con un nombre de codificación fija sobrescribe artefactos de compilación anteriores que utilicen el mismo nombre de codificación fija. Por ejemplo, si se activa una compilación varias veces al día, ahora puede añadir una marca de tiempo a su nombre de artefactos. Cada nombre de artefacto de compilación es único y no sobrescribe los artefactos de compilaciones anteriores.	7 de agosto de 2018
<a href="#">Soporte de artefactos de compilación sin cifrar (p. 460)</a>	Ahora, CodeBuild admite las compilaciones con artefactos de compilación sin cifrar. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación (consola)</a> .	26 de julio de 2018
<a href="#">Support alarmas y las métricas de Amazon CloudWatch (p. 460)</a>	CodeBuild ahora proporciona la integración con métricas y alarmas CloudWatch. Puede utilizar la consola de CodeBuild o CloudWatch para monitorear compilaciones en el nivel del proyecto y la cuenta. Para obtener más información, consulte este tema sobre el <a href="#">monitoreo de compilaciones</a> .	19 de julio de 2018

<a href="#">Compatibilidad con la notificación del estado de una compilación (p. 460)</a>	CodeBuild ahora puede informar sobre el estado del inicio y finalización de una compilación al proveedor de código fuente. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación en CodeBuild</a> .	10 de julio de 2018
<a href="#">Se han añadido variables de entorno a la documentación CodeBuild (p. 460)</a>	La página <a href="#">Variables de entorno en entornos de compilación</a> se ha actualizado con las variables de entorno CODEBUILD_BUILD_ID, CODEBUILD_LOG_PATH y CODEBUILD_START_TIME.	9 de julio de 2018
<a href="#">Support confinally bloque en el archivo buildspec (p. 460)</a>	La documentación de CodeBuild se ha actualizado con información detallada sobre el <code>finally</code> en un archivo buildspec. Los comandos del bloque <code>finally</code> siempre se ejecutan después de los comandos de su bloque de comandos correspondiente. Para obtener más información, consulte <a href="#">Sintaxis de la especificación de compilación</a> .	20 de junio de 2018
<a href="#">Notificaciones de actualización del agente de Co (p. 460)</a>	La documentación de CodeBuild se ha actualizado con información detallada sobre cómo se puede utilizar Amazon SNS para recibir una notificación cuando se publiquen nuevas versiones del agente de CodeBuild. Para obtener más información, consulte <a href="#">Recepción de notificaciones sobre nuevas AWS CodeBuild versiones del agente</a> .	15 de junio de 2018

## Actualizaciones anteriores

En la siguiente tabla, se describen los cambios importantes de cada versión de la Guía del usuario de AWS CodeBuild anteriores a junio de 2018.

Cambio	Descripción	Fecha
Compatibilidad con compilaciones de Windows	CodeBuild ahora admite compilaciones para la plataforma Microsoft Windows Server, incluido un entorno de compilación preconfigurado para .NET Core 2.0 en Windows. Para obtener más información,	25 de mayo de 2018

Cambio	Descripción	Fecha
	consulte <a href="#">Ejemplos de Microsoft Windows para CodeBuild (p. 31)</a> .	
Compatibilidad con la idempotencia de la compilación	Cuando se ejecuta el comando <code>start-build</code> con la AWS Command Line Interface (AWS CLI), se puede especificar que la compilación sea idempotente. Para obtener más información, consulte <a href="#">Ejecutar una compilación (AWS CLI) (p. 282)</a> .	15 de mayo de 2018
Compatibilidad con la anulación de más opciones de un proyecto de compilación	Ahora se pueden anular más opciones de un proyecto de compilación al crear una compilación. Las anulaciones solo se aplican a esa compilación. Para obtener más información, consulte <a href="#">Ejecutar una compilación en AWS CodeBuild (p. 281)</a> .	15 de mayo de 2018
Compatibilidad con puntos de enlace de la VPC	Ahora es posible utilizar puntos de enlace de la VPC para mejorar la seguridad de las compilaciones. Para obtener más información, consulte <a href="#">Uso de puntos de enlace de la VPC (p. 182)</a> .	18 de marzo de 2018
Compatibilidad con los disparadores	Ahora puede crear desencadenadores para programar las compilaciones a intervalos regulares. Para obtener más información, consulte <a href="#">Crear desencadenadores de AWS CodeBuild (p. 233)</a> .	28 de marzo de 2018
Documentación de los puntos de enlace de FIPS	Ahora, puede obtener información acerca de cómo utilizar elAWS Command Line Interface(AWS CLI) o unAWSSDK de indicar a CodeBuild que use uno de los cuatro puntos de enlace que cumplen el estándar federal de procesamiento de información (FIPS). Para obtener más información, consulte <a href="#">Especificación del punto de enlace de AWS CodeBuild (p. 404)</a> .	28 de marzo de 2018

Cambio	Descripción	Fecha
AWS CodeBuild disponible en Asia Pacífico (Mumbai), Europa (París) y América del Sur (São Paulo)	AWS CodeBuild ya está disponible en las regiones Asia Pacífico (Mumbai), Europa (París) y América del Sur (São Paulo). Para obtener más información, consulte <a href="#">AWS CodeBuild en la Referencia general de Amazon Web Services</a> .	28 de marzo de 2018
Compatibilidad con GitHub Enterprise Server	ahora CodeBuild puede compilar código fuente almacenado en un repositorio de GitHub Enterprise Server. Para obtener más información, consulte <a href="#">Ejemplo de GitHub Enterprise Server (p. 107)</a> .	25 de enero de 2018
Compatibilidad con la profundidad del clon de Git	CodeBuild ahora admite la creación de un clon superficial con un historial truncado en el número especificado de confirmaciones. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación (p. 197)</a> .	25 de enero de 2018
Compatibilidad con VPC	Las compilaciones habilitadas para VPC ahora pueden tener acceso a los recursos de su VPC. Para obtener más información, consulte <a href="#">Compatibilidad con VPC (p. 180)</a> .	27 de noviembre de 2017
Compatibilidad con el almacenamiento en caché de dependencias	Ahora, CodeBuild admite el almacenamiento en caché de las dependencias. Esto permite a CodeBuild guardar algunas partes reutilizables del entorno de compilación en la caché y usarlas en las distintas compilaciones.	27 de noviembre de 2017
Compatibilidad con las insignias de compilación	CodeBuild ahora admite el uso de insignias de compilación, que proporcionan una imagen incrustable generada dinámicamente (insignia) que muestra el estado de la última compilación de un proyecto. Para obtener más información, consulte <a href="#">Ejemplo de insignias de compilación (p. 78)</a> .	27 de noviembre de 2017

Cambio	Descripción	Fecha
AWS ConfigIntegración de	AWS Configahora admite CodeBuild comoAWS, lo que significa que el servicio puede realizar un seguimiento de sus proyectos de CodeBuild. Para obtener más información acerca deAWS Config, consulte <a href="#">Ejemplo de AWS Config (p. 65)</a> .	20 de octubre de 2017
Recompilación automática del código fuente actualizado en repositorios de GitHub	Si su código fuente está almacenado en un repositorio de GitHub, puede permitir que AWS CodeBuild lo recompile siempre que se envíe un cambio de código al repositorio. Para obtener más información, consulte <a href="#">Ejemplo de filtros de webhooks y solicitudes de extracción de GitHub (p. 113)</a> .	21 de septiembre de 2017
Nuevas formas de almacenar y recuperar variables de entorno confidenciales o grandes en el almacén de parámetros de Amazon EC2 Systems Manager	A partir de ahora, puede utilizar laAWS CodeBuildconsola oAWS CLIpara recuperar variables de entorno confidenciales o de gran tamaño almacenadas en el almacén de parámetros de Amazon EC2 Systems Manager. También puede utilizar laAWS CodeBuildpara almacenar estos tipos de variables de entornos en el almacén de parámetros de Amazon EC2 Systems Manager. Anteriormente, solo podía recuperar estos tipos de variables de entorno incluyéndolas en una especificación de compilación o ejecutando comandos de compilación para automatizar la AWS CLI. Solo podía almacenar estos tipos de variables de entornos mediante la consola del almacén de parámetros de Amazon EC2 Systems Manager. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación (p. 197)</a> , <a href="#">Cambiar la configuración de un proyecto de compilación (p. 254)</a> , y <a href="#">Ejecutar una compilación (p. 281)</a> .	14 de septiembre de 2017

Cambio	Descripción	Fecha
Compatibilidad con la eliminación de compilaciones	Ahora puede eliminar compilaciones en AWS CodeBuild. Para obtener más información, consulte <a href="#">Eliminación de compilaciones (p. 304)</a> .	31 de agosto de 2017
Una forma actualizada de recuperar variables de entorno confidenciales o grandes almacenadas en el almacén de parámetros de Amazon EC2 Systems Manager mediante una especificación de compilación	AWS CodeBuild ahora le permite usar más fácilmente una especificación de compilación para recuperar variables de entorno confidenciales o de gran tamaño almacenadas en el almacén de parámetros de Amazon EC2 Systems Manager. Anteriormente, solo podía recuperar estos tipos de variables de entorno ejecutando comandos de compilación para automatizar la AWS CLI. Para obtener más información, consulte <a href="#">la parameter-store mapeado en Sintaxis de buildspec (p. 137)</a> .	10 de agosto de 2017
AWS CodeBuild admite Bitbucket	Ahora, CodeBuild puede compilar código fuente almacenado en un repositorio de Bitbucket. Para obtener más información, consulte <a href="#">Creación de un proyecto de compilación (p. 197)</a> y <a href="#">Ejecutar una compilación (p. 281)</a> .	10 de agosto de 2017
AWS CodeBuild disponible en EE. UU. Oeste (Norte de California), Europa (Londres) y Canadá (Central)	AWS CodeBuild ya está disponible en las regiones EE. UU. Oeste (Norte de California), Europa (Londres) y Canadá (Central). Para obtener más información, consulte <a href="#">AWS CodeBuild</a> en la Referencia general de Amazon Web Services.	29 de junio de 2017
Nombres y ubicaciones alternativos de archivos buildspec admitidos	Ahora puede especificar un nombre o ubicación alternativos de un archivo buildspec para usarlos en un proyecto de compilación, en lugar de un archivo buildspec predeterminado llamado <code>buildspec.yml</code> en la raíz del código fuente. Para obtener más información, consulte <a href="#">Nombre de archivo y ubicación de almacenamiento de buildspec (p. 137)</a> .	27 de junio de 2017

Cambio	Descripción	Fecha
Ejemplo de notificaciones de compilación actualizado	CodeBuild ahora proporciona compatibilidad integrada con las notificaciones de compilación a través de Amazon CloudWatch Events y Amazon Simple Notification Service (Amazon SNS). El <a href="#">Ejemplo de notificaciones de compilación (p. 81)</a> anterior se ha actualizado para mostrar este nuevo comportamiento.	22 de junio de 2017
Se ha añadido un ejemplo de Docker en imagen personalizada	Se ha añadido un ejemplo que muestra cómo utilizar CodeBuild y una imagen de compilación de Docker personalizada para ejecutar una imagen de Docker. Para obtener más información, consulte la <a href="#">Ejemplo de Docker en una imagen personalizada (p. 99)</a> .	7 de junio de 2017
Obtener código fuente de solicitudes de extracción de GitHub	Cuando ejecuta una compilación con CodeBuild basada en código fuente almacenado en un repositorio de GitHub, ahora puede especificar un ID de solicitud de extracción de GitHub para la compilación. También puede especificar un ID de confirmación, un nombre de ramificación o un nombre de etiqueta en su lugar. Para obtener más información, consulte la <a href="#">Versión de origenValor enEjecutar una compilación (consola) (p. 282)</a> o <a href="#">elsourceVersionValor enEjecutar una compilación (AWS CLI) (p. 282)</a> .	6 de junio de 2017



Cambio	Descripción	Fecha
Versión de especificación de compilación actualizada	Se ha publicado una nueva versión del formato buildspec. La versión 0.2 soluciona el problema de que CodeBuild ejecutara cada comando de compilación en una instancia distinta del shell predeterminado. También, en la versión 0.2, <code>environment_variables</code> ahora se llama <code>env</code> y <code>plaintext</code> ahora se llama <code>variables</code> . Para obtener más información, consulte <a href="#">Referencia de la especificación de compilación para CodeBuild (p. 136)</a> .	9 de mayo de 2017
Dockerfiles para imágenes de compilación disponibles en GitHub	Las definiciones para muchas de las imágenes de compilación proporcionadas por AWS CodeBuild están disponibles como Dockerfiles en GitHub. Para obtener más información, consulte la columna Definition de la tabla en <a href="#">Imágenes de Docker proporcionadas por CodeBuild (p. 161)</a> .	2 de mayo de 2017
AWS CodeBuild disponible en Europa (Fráncfort), Asia Pacífico (Singapur), Asia Pacífico (Sídney) y Asia Pacífico (Tokio)	AWS CodeBuild ya está disponible en las regiones Europa (Fráncfort), Asia Pacífico (Singapur), Asia Pacífico (Sídney) y Asia Pacífico (Tokio). Para obtener más información, consulte <a href="#">AWS CodeBuild</a> en la Referencia general de Amazon Web Services.	21 de marzo de 2017
Compatible con acciones de prueba de CodePipeline para CodeBuild	Ahora puede añadir a una canalización de CodePipeline una acción de prueba que use CodeBuild. Para obtener más información, consulte <a href="#">Agregar una acción de prueba de CodeBuild a una canalización (consola de CodePipeline) (p. 418)</a> .	8 de marzo de 2017

Cambio	Descripción	Fecha
Los archivos buildspec permiten obtener la salida de la compilación de los directorios de nivel superior seleccionados	Los archivos buildspec le permiten ahora especificar directorios de nivel superior individuales para los que puede indicar a CodeBuild en artefactos de salida de la compilación. Para ello utilice la asignación <code>base-directory</code> . Para obtener más información, consulte <a href="#">Sintaxis de buildspec (p. 137)</a> .	8 de febrero de 2017
Variables de entorno integradas	AWS CodeBuild proporciona variables de entorno integradas adicionales para que las use en sus compilaciones. Estas variables de entorno describen la entidad que inició la compilación, la dirección URL del repositorio de código fuente, el ID de versión de código fuente, etc. Para obtener más información, consulte <a href="#">Variables de entorno en los entornos de compilación (p. 172)</a> .	30 de enero de 2017
AWS CodeBuild disponibles en EE.UU. Este (Ohio)	AWS CodeBuild ya está disponible en la región EE.UU. Este (Ohio). Para obtener más información, consulte <a href="#">AWS CodeBuild en la Referencia general de Amazon Web Services</a> .	19 de enero de 2017
Información sobre el funcionamiento del shell y los comandos	CodeBuild ejecuta cada comando que especifique en una instancia independiente del shell predeterminado de un entorno de compilación. Este comportamiento predeterminado puede producir algunos efectos secundarios imprevistos para los comandos. Le recomendamos algunos enfoques para solucionar este comportamiento predeterminado si fuera necesario. Para obtener más información, consulte <a href="#">Shells y comandos de los entornos de compilación (p. 171)</a> .	9 de diciembre de 2016

Cambio	Descripción	Fecha
Información sobre las variables de entorno	CodeBuild proporciona varias variables de entorno que puede utilizar en los comandos de compilación. También puede definir sus propias variables de entorno. Para obtener más información, consulte <a href="#">Variables de entorno en los entornos de compilación (p. 172)</a> .	7 de diciembre de 2016
Tema sobre solución de problemas	Ahora dispone de información sobre solución de problemas. Para obtener más información, consulte <a href="#">Solución de problemas de AWS CodeBuild (p. 427)</a> .	5 de diciembre de 2016
Versión inicial del complemento de Jenkins	Esta es la versión inicial del complemento CodeBuild Jenkins. Para obtener más información, consulte <a href="#">Uso de AWS CodeBuild con Jenkins (p. 420)</a> .	5 de diciembre de 2016
Versión inicial de la Guía de usuario	Esta es la versión inicial de la Guía del usuario de CodeBuild.	1 de diciembre de 2016

# Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el [Glosario de AWS](#) en la Referencia general de AWS.

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.