Cómo trabajar con Cloud Build

1 hora5 créditos

**Descripción general**

En este lab, compilarás una imagen de contenedor de Docker a partir del código proporcionado y un Dockerfile con Cloud Build. Luego, subirás el contenedor a Container Registry.

**Objetivos**

En este lab, aprenderás a realizar las siguientes tareas:

* Usar Cloud Build para compilar y enviar contenedores
* Usar Container Registry para almacenar e implementar contenedores

**Configuración del lab**

Accede al lab

En cada lab, recibirá un proyecto de Google Cloud y un conjunto de recursos nuevos por tiempo limitado y sin costo adicional.

1. Accede a Qwiklabs desde una **ventana de incógnito**.
2. Ten en cuenta el tiempo de acceso del lab (por ejemplo, 1:15:00) y asegúrate de finalizarlo en el plazo asignado.  
   No existe una función de pausa. Si lo necesita, puede reiniciar el lab, pero deberá hacerlo desde el comienzo.
3. Cuando esté listo, haga clic en **Comenzar lab**.
4. Anote las credenciales del lab (el **nombre de usuario** y la **contraseña**). Las usarás para acceder a la consola de Google Cloud.
5. Haga clic en **Abrir Google Console**.
6. Haga clic en **Usar otra cuenta**, copie las credenciales para **este** lab y péguelas en el mensaje emergente que aparece.  
   Si usa otras credenciales, se generarán errores o **incurrirá en cargos**.
7. Acepta las condiciones y omite la página de recursos de recuperación.

**Nota:** No hagas clic en **Finalizar lab**, a menos que lo hayas terminado o desees reiniciarlo. Esta acción borrará tu trabajo y quitará el proyecto.

Después de completar los pasos iniciales de acceso, se abrirá el panel del proyecto.

**Tarea 1: Confirma que estén habilitadas las APIs necesarias**

1. Anota el nombre de tu proyecto de Google Cloud. Este valor se muestra en la barra superior de la consola de Google Cloud. Comenzará con qwiklabs-gcp-, seguido de números hexadecimales.
2. En la consola de Google Cloud, en **Menú de navegación** (Ícono del menú de navegación), haz clic en **APIs y servicios**.
3. Haz clic en **Habilitar APIs y servicios**.
4. En el recuadro **Buscar API y servicios**, escribe Cloud Build.
5. En la tarjeta resultante de la API de Cloud Build, si no ves un mensaje que confirme que está habilitada la API, haz clic en el botón HABILITAR.
6. Usa el botón Atrás para volver a la pantalla anterior con un cuadro de búsqueda. En ese cuadro, escribe Container Registry.
7. En la tarjeta resultante de la API de Google Container Registry, si no ves un mensaje que confirme que está habilitada la API, haz clic en el botón HABILITAR.

**Tarea 2: Compila contenedores con Dockerfile y Cloud Build**

Puedes escribir archivos de configuración de compilación para brindar instrucciones a Cloud Build sobre qué tareas realizar cuando se compila un contenedor. Estos archivos de compilación se pueden usar para recuperar dependencias, ejecutar pruebas de unidades, realizar análisis y mucho más. En esta tarea, crearás un Dockerfile y lo usarás como una secuencia de comandos de configuración de compilación con Cloud Build. También crearás una secuencia de comandos de shell simple (quickstart.sh), que representará una aplicación dentro del contenedor.

1. En la barra de título de la consola de Google Cloud, haz clic en **Activar Cloud Shell**.
2. Cuando se te solicite, haz clic en **Continuar**.

Cloud Shell se abrirá en la parte inferior de la ventana de la consola de Google Cloud.

1. Crea un archivo quickstart.sh vacío con el editor de texto nano:

nano quickstart.sh

Se copió correctamente

content\_copy

1. Agrega las siguientes líneas al archivo quickstart.sh:

#!/bin/sh

echo "Hello, world! The time is $(date)."

Se copió correctamente

content\_copy

1. Guarda el archivo y cierra nano presionando las teclas **CTRL + X**. Luego, presiona **Y** seguido de **Intro**.
2. Crea un archivo Dockerfile vacío con el editor de texto nano:

nano Dockerfile

Se copió correctamente

content\_copy

1. Agrega el siguiente comando de Dockerfile:

FROM alpine

Se copió correctamente

content\_copy

Esto le indica a la compilación que use la imagen base de Alpine Linux.

1. Agrega el siguiente comando al final del Dockerfile:

COPY quickstart.sh /

Se copió correctamente

content\_copy

Esto agrega la secuencia de comandos quickstart.sh al directorio / de la imagen.

1. Agrega el siguiente comando al final del Dockerfile:

CMD ["/quickstart.sh"]

Se copió correctamente

content\_copy

Esto configura la imagen para ejecutar la secuencia de comandos /quickstart.sh cuando se crea y se ejecuta el contenedor asociado.

Ahora, el Dockerfile debería verse de la siguiente manera:

FROM alpine

COPY quickstart.sh /

CMD ["/quickstart.sh"]

Se copió correctamente

content\_copy

1. Guarda el archivo y cierra nano presionando las teclas **CTRL + X**. Luego, presiona **Y** seguido de **Intro**.
2. En Cloud Shell, ejecuta el siguiente comando para hacer que la secuencia de comandos quickstart.sh sea ejecutable:

chmod +x quickstart.sh

Se copió correctamente

content\_copy

1. En Cloud Shell, ejecuta el siguiente comando para compilar la imagen de contenedor de Docker en Cloud Build:

gcloud builds submit --tag gcr.io/${GOOGLE\_CLOUD\_PROJECT}/quickstart-image .

Se copió correctamente

content\_copy

**Nota:**No olvides incluir el punto (“.”) al final del comando. Este signo especifica que el código fuente se encuentra en el directorio de trabajo actual durante el tiempo de compilación.

Cuando se completa la compilación, tu imagen de Docker compilada se envía a Container Registry.

1. En la consola de Google Cloud, en **Menú de navegación** (Ícono del menú de navegación), haz clic en **Container Registry** > **Imágenes**.

La imagen de Docker quickstart-image aparecerá en la lista.

**Tarea 3: Compila contenedores con un archivo de configuración de compilación y Cloud Build**

Cloud Build también admite archivos de configuración de compilación personalizados. En esta tarea, incorporarás un contenedor de Docker existente usando un archivo de compilación personalizado con formato YAML mediante Cloud Build.

1. En Cloud Shell, ingresa el siguiente comando para clonar el repositorio en el entorno de Cloud Shell del lab:

git clone https://github.com/GoogleCloudPlatform/training-data-analyst

Se copió correctamente

content\_copy

1. Crea un vínculo simbólico como un acceso directo al directorio de trabajo:

ln -s ~/training-data-analyst/courses/ak8s/v1.1 ~/ak8s

Se copió correctamente

content\_copy

1. Cambia al directorio que contenga los archivos de muestra de este lab:

cd ~/ak8s/Cloud\_Build/a

Se copió correctamente

content\_copy

En este directorio, se te proporcionaron un archivo de configuración de Cloud Build de muestra personalizado, llamado cloudbuild.yaml, copias del Dockerfile y la secuencia de comandos quickstart.sh que creaste en la primera tarea.

1. En Cloud Shell, ejecuta el siguiente comando para ver el contenido de cloudbuild.yaml.

cat cloudbuild.yaml

Se copió correctamente

content\_copy

Verás lo siguiente:

steps:

- name: 'gcr.io/cloud-builders/docker'

args: [ 'build', '-t', 'gcr.io/$PROJECT\_ID/quickstart-image', '.' ]

images:

- 'gcr.io/$PROJECT\_ID/quickstart-image'

Este archivo le indica a Cloud Build que use Docker para compilar una imagen con la especificación del Dockerfile en el directorio local actual, que la etiquete con gcr.io/$PROJECT\_ID/quickstart-image ($PROJECT\_ID es una variable de sustitución que Cloud Build propaga automáticamente con el ID del proyecto asociado) y que, luego, envíe esa imagen a Container Registry.

1. En Cloud Shell, ejecuta el siguiente comando para iniciar Cloud Build usando cloudbuild.yaml como el archivo de configuración de compilación:

gcloud builds submit --config cloudbuild.yaml .

Se copió correctamente

content\_copy

El resultado de la compilación en Cloud Shell debería ser el mismo que antes. Cuando se completa la compilación, se envía una versión nueva de la misma imagen a Container Registry.

1. En la consola de Google Cloud, en **Menú de navegación** (Ícono del menú de navegación), haz clic en **Container Registry** > **Imágenes** y, luego, en quickstart-image.

Ahora habrá dos versiones de quickstart-image en la lista.

Haz clic en *Revisar mi progreso* para verificar el objetivo.

Compilar dos imágenes de contenedor en Cloud Build

Revisar mi progreso

1. En la consola de Google Cloud, en **Menú de navegación** (Ícono del menú de navegación), haz clic en **Cloud Build** > **Historial**.

Aparecerán dos compilaciones en la lista.

1. Haz clic en el ID de la compilación que se encuentra al principio de la lista.

Se muestran los detalles de la compilación, incluido su registro.

**Tarea 4. Compila y prueba contenedores con un archivo de configuración de compilación y Cloud Build**

El verdadero poder de los archivos de configuración de compilación personalizados es su capacidad de realizar otras acciones, en paralelo o en secuencia, además de solo compilar contenedores: ejecutar pruebas en tus contenedores recientemente compilados, enviarlos a varios destinos y, también, implementarlos en Kubernetes Engine.

En este lab, veremos un ejemplo simple: un archivo de configuración de compilación que prueba el contenedor que compiló y, luego, informa el resultado a su entorno de llamada.

1. En Cloud Shell, cambia al directorio que contenga los archivos de muestra de este lab:

cd ~/ak8s/Cloud\_Build/b

Se copió correctamente

content\_copy

Tal como ocurrió antes, en este directorio se te proporcionaron la secuencia de comandos quickstart.sh y el archivo de configuración de Cloud Build de muestra personalizado llamado cloudbuild.yaml. Estos se modificaron ligeramente a fin de demostrar la capacidad de Cloud Build para probar los contenedores que compiló.

También hay un Dockerfile, que es idéntico al que se usó en la tarea anterior.

1. En Cloud Shell, ejecuta el siguiente comando para ver el contenido de cloudbuild.yaml.

cat cloudbuild.yaml

Se copió correctamente

content\_copy

Verás lo siguiente:

steps:

- name: 'gcr.io/cloud-builders/docker'

args: [ 'build', '-t', 'gcr.io/$PROJECT\_ID/quickstart-image', '.' ]

- name: 'gcr.io/$PROJECT\_ID/quickstart-image'

args: ['fail']

images:

- 'gcr.io/$PROJECT\_ID/quickstart-image

Además de sus acciones anteriores, este archivo de configuración de compilación ejecuta la imagen quickstart-image que creó. En esta tarea, se modificó la secuencia de comandos quickstart.sh para que simule una prueba fallida cuando se le pasa un argumento ['fail'].

1. En Cloud Shell, ejecuta el siguiente comando para iniciar Cloud Build usando cloudbuild.yaml como el archivo de configuración de compilación:

gcloud builds submit --config cloudbuild.yaml .

Se copió correctamente

content\_copy

Verás el resultado del comando, que finaliza con un texto como el siguiente:

**Resultado**

Finished Step #1

ERROR

ERROR: build step 1 "gcr.io/ivil-charmer-227922klabs-gcp-49ab2930eea05/quickstart-image" failed: exit status 127

---------------------------------------------------------------------------------------------------------

ERROR: (gcloud.builds.submit) build f3e94c28-fba4-4012-a419-48e90fca7491 completed with status "FAILURE"

1. Confirma que tu shell de comandos detecte que no se pudo realizar la compilación:

echo $?

Se copió correctamente

content\_copy

El comando responderá con un valor distinto de cero. Si hubieras incorporado esta compilación a una secuencia de comandos, tu secuencia podría responder de manera inesperada ante el error en la compilación.

Haz clic en *Revisar mi progreso* para verificar el objetivo.

Compilar y probar contenedores con un archivo de configuración de compilación y Cloud Build

Revisar mi progreso

**Finalice su lab**

Cuando haya completado el lab, haga clic en **Finalizar lab**. Google Cloud Skills Boost quitará los recursos que usó y limpiará la cuenta.

Tendrá la oportunidad de calificar su experiencia en el lab. Seleccione la cantidad de estrellas que corresponda, ingrese un comentario y haga clic en **Enviar**.

La cantidad de estrellas indica lo siguiente:

* 1 estrella = Muy insatisfecho
* 2 estrellas = Insatisfecho
* 3 estrellas = Neutral
* 4 estrellas = Satisfecho
* 5 estrellas = Muy satisfecho

Puede cerrar el cuadro de diálogo si no desea proporcionar comentarios.

Para enviar comentarios, sugerencias o correcciones, use la pestaña **Asistencia**.

Copyright 2020 Google LLC. All rights reserved. Google y el logotipo de Google son marcas de Google LLC. Los demás nombres de productos y empresas pueden ser marcas de las respectivas empresas a las que estén asociados.