Google Cloud Platform Funda... | GCP Fundamentals: Getting St... | GCP Fundamentals: Getting St...

https://googlecloud.qwiklabs.com/classrooms/3803/labs/29713

← GCP Fundamentals: Getting Started with Deployment Manager and Stackdriver

Start Lab 00:30:00

Score —/10

GCP Fundamentals: Getting Started with Deployment Manager and Stackdriver

30 minutes Free ★★★★☆ Rate Lab

Overview

In this lab, you create a deployment using Deployment Manager and use it to maintain a consistent state of your deployment. You will also view resource usage in a VM instance using Stackdriver.

Objectives

In this lab, you will learn how to perform the following tasks:

- Create a Deployment Manager deployment.
- Update a Deployment Manager deployment.
- View the load on a VM instance using Stackdriver.

Task 1: Sign in to the Google Cloud Platform (GCP) Console

What you'll need

To complete this lab, you'll need:

- Access to a standard internet browser (Chrome browser recommended).
- Time. Note the lab's **Completion** time in Qwiklabs. This is an estimate of the time it should take to complete all steps. Plan your schedule so you have time to complete the lab. Once you start the lab, you will not be able to pause and return later (you begin at step 1 every time you start a lab).
- The lab's **Access** time is how long your lab resources will be available. If you finish your lab with access time still available, you will be able to explore the Google Cloud Platform or work on any section of the lab that was marked "if you have time". Once the Access time runs out, your lab will end and all resources will terminate.
- You **DO NOT** need a Google Cloud Platform account or project. An account, project and associated resources are provided to you as part of this lab.
- If you already have your own GCP account, make sure you do not use it for this lab.
- If your lab prompts you to log into the console, **use only the student account provided to you by the lab**. This prevents you from incurring charges for lab activities in your personal GCP account.

Start your lab

When you are ready, click **Start Lab**. You can track your lab's progress with the status bar at the top of your screen.

> **Important** What is happening during this time? Your lab is spinning up GCP resources for you behind the scenes, including an account, a project, resources within the project, and permission for you to control the resources needed to run the lab. This means that instead of spending time manually setting up a project and building resources from scratch as part of your lab, you can begin learning more quickly.

Find Your Lab's GCP Username and Password

To access the resources and console for this lab, locate the Connection Details panel in Qwiklabs. Here you will find the account ID and password for the account you will use to log in to the Google Cloud Platform:

CONNECTION DETAILS
OPEN GOOGLE CONSOLE
USERNAME
google822-student@qwiklabs.net
PASSWORD
TZjR4X786

If your lab provides other resource identifiers or connection-related information, it will appear on this panel as well.

Task 2: Confirm that needed APIs are enabled

1. Make a note of the name of your GCP project. This value is shown in the top bar of the Google Cloud Platform Console. It will be of the form `qwiklabs-gcp-` followed by hexadecimal numbers.
2. In the GCP Console, on the **Navigation menu** (☰), click **APIs & services**.
3. Scroll down in the list of enabled APIs, and confirm that these APIs are enabled:
   - Cloud Deployment Manager v2 API
   - Cloud Runtime Configuration API
   - Stackdriver Monitoring API
4. If one or more APIs is missing, click the **Enable APIs and Services** button at top. Search for the above APIs by name and enable each for your current project. (You noted the name of your GCP project above.)

Task 3: Create a Deployment Manager deployment

1. On the **Google Cloud Platform** menu, click **Activate Cloud Shell** (>_). If a dialog box appears, click **Start Cloud Shell**.
2. For your convenience, place the zone that Qwiklabs assigned you to into an environment variable called MY_ZONE. At the Cloud Shell prompt, type this partial command:

```
export MY_ZONE=
```

followed by the zone that Qwiklabs assigned you to. Your complete command will look like this:

```
export MY_ZONE=us-central1-f
```

3. At the Cloud Shell prompt, download an editable Deployment Manager template:

```
gsutil cp gs://cloud-training/gcpfcoreinfra/mydeploy.yaml mydeploy.yaml
```

4. Insert your Google Cloud Platform project ID into the file in place of the string `PROJECT_ID` using this command:

```
sed -i -e 's/PROJECT_ID/'$DEVSHELL_PROJECT_ID/ mydeploy.yaml
```

5. Insert your assigned Google Cloud Platform zone into the file in place of the string `ZONE` using this command:

```
sed -i -e 's/ZONE/'$MY_ZONE/ mydeploy.yaml
```

6. View the `mydeploy.yaml` file, with your modifications, with this command:

```
cat mydeploy.yaml
```

The file will look something like this:

```
resources:
- name: my-vm
  type: compute.v1.instance
  properties:
    zone: us-central1-a
    machineType: zones/us-central1-a/machineTypes/n1-standard-1
    metadata:
      items:
      - key: startup-script
        value: "apt-get update"
    disks:
    - deviceName: boot
      type: PERSISTENT
      boot: true
      autoDelete: true
      initializeParams:
        sourceImage: https://www.googleapis.com/compute/v1/projects
/debian-cloud/global/images/debian-9-stretch-v20180806
    networkInterfaces:
    - network: https://www.googleapis.com/compute/v1/projects/qwiklabs-
gcp-dcdf854d278b50cd/global/networks/default
      accessConfigs:
      - name: External NAT
        type: ONE_TO_ONE_NAT
```

> Do not use the above text literally in your own **mydeploy.yaml** file. Be sure that the zone that is named on the **zone:** and **machineType:** lines in your file matches the zone to which Qwiklabs assigned you. Be sure that the GCP project ID on the **network:** line in your file matches the project ID to which Qwiklabs assigned you, not the one in this example.

7. Build a deployment from the template:

```
gcloud deployment-manager deployments create my-first-depl --config
mydeploy.yaml
```

When the deployment operation is complete, the **gcloud** command displays a list of the resources named in the template and their current state.

8. Confirm that the deployment was successful. In the GCP Console, on the **Navigation menu** (☰), click **Compute Engine > VM instances**. You will see that a VM instance called **my-vm** has been created, as specified by the template.
9. Click on the VM instance's name to open its VM instance details screen.
10. Scroll down to the **Custom metadata** section. Confirm that the startup script you specified in your Deployment Manager template has been installed.

Click *Check my progress* to verify the objective.

Create a Deployment Manager deployment
Check my progress

Task 4: Update a Deployment Manager deployment

1. Return to your Cloud Shell prompt. Launch the `nano` text editor to edit the **mydeploy.yaml** file:

```
nano mydeploy.yaml
```

2. Find the line that sets the value of the startup script, `value: "apt-get update"`, and edit it so that it looks like this:

```
      value: "apt-get update; apt-get install nginx-light -y"
```

Do not disturb the spaces at the beginning of the line. The YAML templating language relies on indented lines as part of its syntax. As you edit the file, be sure that the `v` in the word `value` in this new line is immediately below the `k` in the word `key` on the line above it.

3. Press **Ctrl+O** and then press **Enter** to save your edited file.
4. Press **Ctrl+X** to exit the **nano** text editor.
5. Return to your Cloud Shell prompt. Enter this command to cause Deployment Manager to update your deployment to install the new startup script:

```
gcloud deployment-manager deployments update my-first-depl --config
mydeploy.yaml
```

Wait for the **gcloud** command to display a message confirming that the update operation was completed successfully.

6. In the GCP console, on the **Navigation menu** (☰), click **Compute Engine > VM instances**.
7. Click on the **my-vm** VM instance's name to open its **VM instance details** pane.
8. Scroll down to the **Custom metadata** section. Confirm that the startup script has been updated to the value you declared in your Deployment Manager template.

Click *Check my progress* to verify the objective.

Update the Deployment Manager deployment
Check my progress

Task 5: View the Load on a VM using Stackdriver

1. In the GCP Console, on the **Navigation menu** (☰), click **Compute Engine > VM instances**.
2. To open a command prompt on the **my-vm** instance, click **SSH** in its row in the **VM instances** list.
3. In the ssh session on **my-vm**, execute this command to create a CPU load:

```
dd if=/dev/urandom | gzip -9 >> /dev/null &
```

This Linux pipeline forces the CPU to work on compressing a continuous stream of random data.

> Leave the window containing your ssh open while you proceed with the lab.

4. In the GCP Console, on the **Navigation menu** (☰), under **Stackdriver**, click **Monitoring**.
5. On the Stackdriver welcome screen (**Create your free Stackdriver account**), confirm that the Google Cloud Platform project shown is the one to which Qwiklabs assigned you. Click **Create Workspace**.
6. In the **Add Google Cloud Platform projects to monitor** dialog, confirm that the GCP project that Qwiklabs created for you is shown as selected, and then click **Continue**.
7. In the **Monitor AWS accounts** dialog, click **Skip AWS Setup**.
8. In the **Install the Stackdriver Agents** dialog, click **Continue**.

Installing the Stackdriver agents into your VM instances is generally recommended, because it allows Stackdriver to monitor and log more information. But this action is not required for this lab.

9. On the **Get Reports by Email** screen, click **No reports**, and then click **Continue**. After a short delay, Stackdriver displays the message **Finished initial collection**.
10. Click **Launch monitoring**.
11. On the **Welcome to Stackdriver Monitoring** screen, click **Resources > Metrics Explorer**.
12. In the **Metrics Explorer**'s **Metric** pane, select the resource type **GCE VM instance** and the metric **CPU usage**.

In the resulting graph, notice that CPU usage increased sharply a few minutes ago.

13. Terminate your workload generator. Return to your ssh session on **my-vm** and enter this command:

```
kill %1
```

Congratulations!

In this lab, you used Deployment Manager to create a deployment using a template, and you demonstrated Deployment Manager's ability to bring a deployment into compliance with a template. You also used Stackdriver to view resource consumption on a VM instance.

End your lab

When you have completed your lab, click **End Lab**. Qwiklabs removes the resources you've used and cleans the account for you.

You will be given an opportunity to rate the lab experience. Select the applicable number of stars, type a comment, and then click **Submit**.

The number of stars indicates the following:

- 1 star = Very dissatisfied
- 2 stars = Dissatisfied
- 3 stars = Neutral
- 4 stars = Satisfied
- 5 stars = Very satisfied

You can close the dialog box if you don't want to provide feedback.

For feedback, suggestions, or corrections, please use the **Support** tab.

Manual Last Updated: April 01, 2019
Lab Last Tested: April 01, 2019

More resources

Read the documentation on Google Cloud Deployment Manager.

Read the documentation on Stackdriver.