Google Cloud Fundamentals: Getting Started with Compute Engine

Overview

In this lab, you will create virtual machines (VMs) and connect to them. You will also create connections between the instances.

Objectives

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

Create a Compute Engine virtual machine using the Google Cloud Platform (GCP) Console.

Create a Compute Engine virtual machine using the gcloud command-line interface.

Connect between the two instances.

Task 2: Create a virtual machine using the GCP Console

In the Navigation menu (Navigation menu), click Compute Engine > VM instances.

Click Create.

On the Create an Instance page, for Name, type my-vm-1.

For Region and Zone, select the region and zone assigned by Qwiklabs.

For Machine type, accept the default.

For Boot disk, if the Image shown is not Debian GNU/Linux 10 (Buster), click Change and select Debian GNU/Linux 10 (Buster).

Leave the defaults for Identity and API access unmodified.

For Firewall, click Allow HTTP traffic.

Leave all other defaults unmodified.

To create and launch the VM, click Create.

Note: The VM can take about two minutes to launch and be fully available for use.

Click Check my progress to verify the objective.

Create a virtual machine using the GCP Console

Task 3: Create a virtual machine using the gcloud command line

In GCP console, on the top right toolbar, click the Open Cloud Shell button.

Cloud Shell icon

Click Continue. cloudshell_continue.png

To display a list of all the zones in the region to which Qwiklabs assigned you, enter this partial command gcloud compute zones list | grep followed by the region that Qwiklabs or your instructor assigned you to.

Your completed command will look like this:

```
gcloud compute zones list | grep us-central1
```

Choose a zone from that list other than the zone to which Qwiklabs assigned you. For example, if Qwiklabs assigned you to region us-central1 and zone us-central1-a you might choose zone us-central1-b.

To set your default zone to the one you just chose, enter this partial command gcloud config set compute/zone followed by the zone you chose.

Your completed command will look like this:

```
gcloud config set compute/zone us-central1-b
```

To create a VM instance called my-vm-2 in that zone, execute this command:

```
gcloud compute instances create "my-vm-2" \

--machine-type "n1-standard-1" \

--image-project "debian-cloud" \

--image-family "debian-10" \

--subnet "default"
```

Note: The VM can take about two minutes to launch and be fully available for use.

To close the Cloud Shell, execute the following command:

```
exit
```

Click Check my progress to verify the objective.

Create a virtual machine using the gcloud command line

Task 4: Connect between VM instances

In the Navigation menu (Navigation menu), click Compute Engine > VM instances.

You will see the two VM instances you created, each in a different zone.

Notice that the Internal IP addresses of these two instances share the first three bytes in common. They reside on the same subnet in their Google Cloud VPC even though they are in different zones.

To open a command prompt on the my-vm-2 instance, click SSH in its row in the VM instances list.

Use the ping command to confirm that my-vm-2 can reach my-vm-1 over the network:

```
ping my-vm-1.us-central1-a
```

Notice that the output of the ping command reveals that the complete hostname of my-vm-1 is my-vm-1.us-central1-a.c.PROJECT_ID.internal, where PROJECT_ID is the name of your Google Cloud Platform project. GCP automatically supplies Domain Name Service (DNS) resolution for the internal IP addresses of VM instances.

Press Ctrl+C to abort the ping command.

Use the ssh command to open a command prompt on my-vm-1:

```
ssh my-vm-1.us-central1-a
```

If you are prompted about whether you want to continue connecting to a host with unknown authenticity, enter yes to confirm that you do.

At the command prompt on my-vm-1, install the Nginx web server:

```
sudo apt-get install nginx-light -y
```

Use the nano text editor to add a custom message to the home page of the web server:

```
sudo nano /var/www/html/index.nginx-debian.html
```

Use the arrow keys to move the cursor to the line just below the h1 header. Add text like this, and replace YOUR_NAME with your name:

```
Hi from YOUR_NAME
```

Press Ctrl+O and then press Enter to save your edited file, and then press Ctrl+X to exit the nano text editor.

Confirm that the web server is serving your new page. At the command prompt on my-vm-1, execute this command:

```
curl http://localhost/
```

The response will be the HTML source of the web server's home page, including your line of custom text.

To exit the command prompt on my-vm-1, execute this command:

```
exit
```

You will return to the command prompt on my-vm-2

To confirm that my-vm-2 can reach the web server on my-vm-1, at the command prompt on my-vm-2, execute this command:

```
curl http://my-vm-1.us-central1-a/
```

The response will again be the HTML source of the web server's home page, including your line of custom text.

In the Navigation menu (Navigation menu), click Compute Engine > VM instances.

Copy the External IP address for my-vm-1 and paste it into the address bar of a new browser tab. You will see your web server's home page, including your custom text.

If you forgot to click Allow HTTP traffic when you created the my-vm-1 VM instance, your attempt to reach your web server's home page will fail. You can add a firewall rule to allow inbound traffic to your instances, although this topic is out of scope for this course.

Congratulations!

In this lab, you created virtual machine (VM) instances in two different zones and connected to them using ping, ssh, and HTTP.