PROFESSIONAL DATA ENGINEER *- Google Cloud Platform*

TITLE: "Introduction to Google Cloud - Compute Engine"

Author:

- *Name: "Vignesh Sekar"*
- *Designation: "Multi Cloud Architect"*
- *Tags: [Google Cloud, DataEngineer, Python, PySpark, SQL, BigData]*

Compute Engine:

1. An instance is a **virtual machine (VM)** hosted on Google's infrastructure. You can create an instance or create a group of managed instances by using the Google Cloud console, the Google Cloud CLI, or the Compute Engine API.

APIs - what and why

1. API (Application Programming Interface) is a software program that gives developers access to computing resources and data.
2. APIs adhere to specific rules and methods to clearly communicate requests and responses.
3. The ability to access data and computing resources greatly increases a developer's efficiency.
4. It is much easier to use an API than to build every single program, method, or dataset from scratch.
5. APIs are built on the principle of abstraction—you don't need to understand the inner workings or complexities of an API to use it in your own environment.
6. APIs are built with the developer in mind and often don't offer a graphical user interface (GUI).
7. APIs are often used in web development, machine learning, data science, and system administration workflows.

Task 1. Create a virtual machine using the Cloud console

1. In the Navigation menu (Navigation menu), click Compute Engine > VM instances.
2. Click CREATE INSTANCE.
3. On the Create an Instance page, for Name, type my-vm-1.
4. For Region and Zone, select the region and zone .

5. For Machine type, accept the default.
6. For Boot disk, if the Image shown is not Debian GNU/Linux 10 (Buster), click Change and select Debian GNU/Linux 10 (Buster).
7. Leave the defaults for Identity and API access unmodified.
8. For Firewall, click Allow HTTP traffic.Leave all other defaults unmodified.
9. To create and launch the VM, click Create.

Task 2. Create a virtual machine with 'gcloud' on Cloud Shell

0. export PROJECT_ID=$(gcloud config get-value project) export ZONE=$(gcloud config get-value compute/zone) export REGION=$(gcloud config get-value compute/region)

Set the region to us-central1 a. gcloud config set compute/region us-central1 2. To view the project region setting, run the following command b. gcloud config get-value compute/region 3. Set the zone to us-central1-a c. gcloud config set compute/zone us-central1-a 4. To view the project zone setting, run the following command d. gcloud config get-value compute/zone

1. In Cloud Shell, run the following command
    a. gcloud compute instances create gcelab21 --machine-type e2-medium --zone $ZONE (or) a1. gcloud compute instances create gcelab2
    --machine-type "n1-standard-1"
    --image-project "debian-cloud"
    --image-family "debian-10"
    --subnet "default"
2. Type Y and press Enter at the location prompt
3. List the compute instance available in the project
    a. gcloud compute instances list
4. List the gcelab2 virtual machine
    b. gcloud compute instances list --filter="name=('gcelab2')"
5. List the Firewall rules in the project
    c. gcloud compute firewall-rules list

- Add a tag to the virtual machine:
    - gcloud compute instances add-tags gcelab2 --tags http-server,https-server
- Update the firewall rule to allow:
    - gcloud compute firewall-rules create default-allow-http --direction=INGRESS --priority=1000 --network=default --action=ALLOW --rules=tcp:80 --source-ranges=0.0.0.0/0 --target-tags=http-server
- List the firewall rules for the project:
    - gcloud compute firewall-rules list --filter=ALLOW:'80'

1. List the Firewall rules for the default network
    d. gcloud compute firewall-rules list --filter="network='default'"
2. List the Firewall rules for the default network where the allow rule matches an ICMP rule
    e. gcloud compute firewall-rules list --filter="NETWORK:'default' AND ALLOW:'icmp'"
3. To connect to your VM with SSH, run the following command
    b. gcloud compute ssh gcelab2 --zone $ZONE

4. To continue, type Y
5. To leave the passphrase empty, press ENTER twice.
6. Install nginx web server on to virtual machine
    c. sudo apt install -y nginx
    - Confirm that NGINX is running:
        -
            c. ps auwx | grep nginx
    - To see the web page, return to the Cloud Console and click the External IP link in the row for your machine, or add the
        - External IP value to http://EXTERNAL_IP/ in a new browser window or tab.
7. You don't need to do anything here, so to disconnect from SSH and exit the remote shell, run the following command

d. exit

8. View the available logs on the system

e. gcloud logging logs list

9. View the logs that relate to compute resources

f. gcloud logging logs list --filter="compute"

10. Read the logs related to the resource type of gce_instance

g. gcloud logging read "resource.type=gce_instance" --limit 5

11. Read the logs for a specific virtual machine

h. gcloud logging read "resource.type=gce_instance AND labels.instance_name='gcelab2'" --limit 5

Here is a breakdown of the command so you understand what is happening:

1. gcloud compute allows you to manage your Compute Engine resources in a format that's simpler than the Compute Engine API.
2. instances create creates a new instance.
3. gcelab2 is the name of the VM.
4. The --machine-type flag specifies the machine type as e2-medium.
5. The --zone flag specifies where the VM is created.
6. If you omit the --zone flag, the gcloud tool can infer your desired zone based on your default properties. Other required instance settings, such as machine type and image, are set to default values if not specified in the create command.

Other Lab - learners hands-on --- After SSH into the instance

- Free -m

- Free —giga
- Lscpu

  #!/bin/bash

  - sudo su
  - apt update
  - apt install apache2
  - ls /var/www/html
  - echo "Hello World!"
  - echo "Hello World!" > /var/www/html/index.html
  - echo $(hostname)
  - echo $(hostname -i)
  - echo "Hello World from $(hostname)"
  - echo "Hello World from $(hostname) $(hostname -i)"
  - echo "Hello world from $(hostname) $(hostname -i)" > /var/www/html/index.html
  - sudo service apache2 start
  - sudo service apache2 stop

Get information on commands

The gcloud command line interface comes with built-in documentation that you can access at any time. Useful information like optional flags and expected arguments can be accessed whenever you need it.

1. To view the gcloud compute instance create manual, run:
   a. gcloud compute instances create --help (or) gcloud -h (or) gcloud config --help
2. Press ENTER or the space bar to scroll through the content.
3. Press Q to exit.
4. The --help flag provides a verbose output for commands. If you want a short summary, use the -h flag.
5. Run the following command in Cloud Shell
   a. gcloud compute instances create -h
6. Run the following command in Cloud Shell
   a. gcloud compute instances describe gcelab2

Resources:

- https://cloud.google.com/compute/docs/instances (https://cloud.google.com/compute/docs/instances)