1. Creating Virtual Machines

https://app.pluralsight.com/id/lti/qwiklabs?destinationUrl=https://googlepluralsight.qwiklabs.com/lti_sessions/libraries/gcp-training-content/content/CBL007-CreatingVirtualMachines&originUrl=https://app.pluralsight.com/library/courses/essential-google-cloud-infrastructure-foundation

Objectives

Create several VMs instances of different types with different characteristics.

- small utility VM for administration purposes
- a standard VM
- a custom VM

Task 1: Create a utility virtual machine

Run the following command:

gcloud beta compute --project=qwiklabs-gcp-00-046ec18c8db6 instances create utility-vm --zone=us-central1-c --machine-type=n1-standard-1 --subnet=default --no-address --maintenance-policy=MIGRATE --service-account=100313093755-compute@developer.gserviceaccount.com --scopes=https://www.googleapis.com/auth/devstorage.read_only,https://www.googleapis.com/auth/logging.write,https://www.googleapis.com/auth/monitoring.write,https://www.googleapis.com/auth/servicecontrol,https://www.googleapis.com/auth/service.management.readonly,https://www.googleapis.com/auth/trace.append --image=debian-10-buster-v20200910 --image-project=debian-cloud --boot-disk-size=10GB --boot-disk-type=pd-standard --boot-disk-device-name=utility-vm --no-shielded-secure-boot --shielded-vtpm --shielded-integrity-monitoring --reservation-affinity=any

When prompted for authorization, click to authorize.

The output should contain the following (**do not copy; this is example output**):

```
Created [https://www.googleapis.com/compute/beta/projects/qwiklabs-gcp-00-046ec18c8db6/zones/us-central1-c/instances/utility-vm].
NAME        ZONE           MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP  STATUS
utility-vm  us-central1-c  n1-standard-1               10.128.0.2                RUNNING
```

Task 2: Create a Windows virtual machine

Run the following command:

gcloud beta compute --project=qwiklabs-gcp-00-046ec18c8db6 instances create vm-2 --zone=europe-west2-a --machine-type=n1-standard-2 --subnet=default --network-tier=PREMIUM --maintenance-policy=MIGRATE --service-account=100313093755-compute@developer.gserviceaccount.com --scopes=https://www.googleapis.com/auth/devstorage.read_only,https://www.googleapis.com/auth/logging.write,https://www.googleapis.com/auth/monitoring.write,https://www.googleapis.com/auth/servicecontrol,https://www.googleapis.com/auth/service.management.readonly,https://www.googleapis.com/auth/trace.appen

d --tags=http-server,https-server --image=windows-server-2016-dc-core-v20200908 --image-project=windows-cloud --boot-disk-size=100GB --boot-disk-type=pd-standard --boot-disk-device-name=vm-2 --no-shielded-secure-boot --shielded-vtpm --shielded-integrity-monitoring --reservation-affinity=any

gcloud compute --project=qwiklabs-gcp-00-046ec18c8db6 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

gcloud compute --project=qwiklabs-gcp-00-046ec18c8db6 firewall-rules create default-allow-https --direction=INGRESS --priority=1000 --network=default --action=ALLOW --rules=tcp:443 --source-ranges=0.0.0.0/0 --target-tags=https-server

The output should contain the following (**do not copy; this is example output**):

```
NAME  ZONE            MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP     STATUS
vm-2  europe-west2-a  n1-standard-2               10.154.0.2   34.105.147.150  RUNNING
Creating firewall...⠹Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-046ec18c8db6/global/firewalls/default-allow-http].
Creating firewall...done.
NAME                NETWORK  DIRECTION  PRIORITY  ALLOW   DENY  DISABLED
default-allow-http  default  INGRESS    1000      tcp:80        False
Creating firewall...⠹Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-046ec18c8db6/global/firewalls/default-allow-https].
Creating firewall...done.
NAME                 NETWORK  DIRECTION  PRIORITY  ALLOW    DENY  DISABLED
default-allow-https  default  INGRESS    1000      tcp:443        False
```

Task 3: Create a custom virtual machine

Run the following command:

```
gcloud beta compute --project=qwiklabs-gcp-00-046ec18c8db6 instances create vm-3 --zone=us-west1-b --machine-type=e2-custom-6-32768 --subnet=default --network-tier=PREMIUM --maintenance-policy=MIGRATE --service-account=100313093755-compute@developer.gserviceaccount.com --scopes=https://www.googleapis.com/auth/devstorage.read_only,https://www.googleapis.com/auth/logging.write,https://www.googleapis.com/auth/monitoring.write,https://www.googleapis.com/auth/servicecontrol,https://www.googleapis.com/auth/service.management.readonly,https://www.googleapis.com/auth/trace.append --image=debian-10-buster-v20200910 --image-project=debian-cloud --boot-disk-size=10GB --boot-disk-type=pd-standard --boot-disk-device-name=vm-3 --no-shielded-secure-boot --shielded-vtpm --shielded-integrity-monitoring --reservation-affinity=any
```

The output should contain the following (**do not copy; this is example output**):

```
Created [https://www.googleapis.com/compute/beta/projects/qwiklabs-gcp-00-046ec18c8db6/zones/us-west1-b/instances/vm-3].
NAME  ZONE        MACHINE_TYPE           PREEMPTIBLE INTERNAL_IP  EXTERNAL_IP  STATUS
vm-3  us-west1-b  custom
                  (e2, 6 vCPU, 32.00 GiB)            10.138.0.2   35.247.20.65 RUNNING
```

SSH to your custom VM

```
gcloud compute ssh mv-3
```

Then, run the following commands(independly):

1. `Free` To see information about unused and used memory and swap space on your custom VM
2. **sudo dmidecode -t 17** to see details about the RAM installed on your VM,
3. `nproc` To verify the number of processors
4. `lscpu` To see details about the CPUs installed on your VM
5. `exit`To exit the SSH terminal

2. Working with Virtual Machines

https://app.pluralsight.com/id/lti/qwiklabs?destinationUrl=https://googlepluralsight.qwiklabs.com/lti_sessions/libraries/gcp-training-content/content/CBL016-WorkingVirtualMachines&originUrl=https://app.pluralsight.com/library/courses/essential-google-cloud-infrastructure-foundation

Objectives

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

- Customize an application server
- Install and configure necessary software
- Configure network access
- Schedule regular backups
- Set up maintenance scripts using metadata for graceful startup and shutdown of the server.

Task 1: Create the VM

Run the following command:

gcloud beta compute --project=qwiklabs-gcp-04-98a940638d63 instances create mc-server --zone=us-central1-a --machine-type=e2-medium --subnet=default --address=34.66.142.205 --network-tier=PREMIUM --maintenance-policy=MIGRATE --service-account=249083116004-compute@developer.gserviceaccount.com --scopes=https://www.googleapis.com/auth/servicecontrol,https://www.googleapis.com/auth/service.management.readonly,https://www.googleapis.com/auth/logging.write,https://www.googleapis.com/auth/monitoring.write,https://www.googleapis.com/auth/trace.append,https://www.googleapis.com/auth/devstorage.read_write --tags=minecraft-server --image=debian-9-stretch-v20200910 --image-project=debian-cloud --boot-disk-size=10GB --boot-disk-type=pd-standard --boot-disk-device-name=mc-server --create-disk=mode=rw,size=50,type=projects/qwiklabs-gcp-04-98a940638d63/zones/us-central1-a/diskTypes/pd-standard,name=minecraft-disk,device-name=minecraft-disk --reservation-affinity=any

The output should contain the following (**do not copy; this is example output**):

```
Created [https://www.googleapis.com/compute/beta/projects/qwiklabs-gcp-04-98a940638d63/zones/us-central1-a/instances/mc-server].
NAME       ZONE           MACHINE_TYPE  PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP    STATUS
mc-server  us-central1-a  e2-medium                  10.128.0.2   34.66.142.205  RUNNING
```

copy the external IP somewhere it will be used later

Task 2: Prepare the data disk

Create a directory and format and mount the disk

The disk is attached to the instance, but it is not yet mounted or formatted.

1. To SSH **mc-server**, run the following command:

   ```
   gcloud compute ssh mc-server
   ```

if the following message appears,

```
Did you mean zone [europe-west1-b] for instance: [mc-server] (Y/n)?  n
```

select n and wait to be directed to the mc-server vm path

2. To create a directory that serves as the mount point for the data disk, run the following command:

```
sudo mkdir -p /home/minecraft
```

3. To format the disk, run the following command:

```
sudo mkfs.ext4 -F -E lazy_itable_init=0,\ lazy_journal_init=0,discard \
/dev/disk/by-id/google-minecraft-disk
```

4. To mount the disk, run the following command:

```
sudo mount -o discard,defaults /dev/disk/by-id/google-minecraft-disk
/home/minecraft
```

Task 3: Install and run the application-

Install the Java Runtime Environment (JRE) and the Minecraft server

1. In the SSH terminal for **mc-server**, to update the Debian repositories on the VM, run the following command:

```
sudo apt-get update
```

2. After the repositories are updated, to install the headless JRE, run the following command:

```
sudo apt-get install -y default-jre-headless
```

3. To navigate to the directory where the persistent disk is mounted, run the following command:

```
cd /home/minecraft
```

4. To install **wget**, run the following command:

```
sudo apt-get install wget
```

5. If prompted to continue, type **Y**
6. To download the current Minecraft server JAR file (1.11.2 JAR), run the following command:

```
sudo wget
https://launcher.mojang.com/v1/objects/d0d0fe2b1dc6ab4c65554cb734270872b72dadd6/s
erver.jarrver.jar
```

Initialize the Minecraft server

1. To initialize the Minecraft server, run the following command:

```
sudo java -Xmx1024M -Xms1024M -jar server.jar nogui
```

NB: The Minecraft server won't run unless you accept the terms of the End User Licensing Agreement (EULA).

2. To see the files that were created in the first initialization of the Minecraft server, run the following command:

```
sudo ls -l
```

3. To edit the EULA, run the following command:

```
sudo nano eula.txt
```

4. Change the last line of the file from `eula=false` to `eula=true`
5. Press **Ctrl+O**, **ENTER** to save the file and then press **Ctrl+X** to exit nano.

Create a virtual terminal screen to start the Minecraft server

If you start the Minecraft server again now, it is tied to the life of your SSH session: that is, if you close your SSH terminal, the server is also terminated. To avoid this issue, you can use screen, an application that allows you to create a virtual terminal that can be "detached," becoming a background process, or "reattached," becoming a foreground process. When a virtual terminal is detached to the background, it will run whether you are logged in or not.

1. To install `screen`, run the following command:

   ```
   sudo apt-get install -y screen
   ```
2. To start your Minecraft server in a `screen` virtual terminal, run the following command: (Use the `-S` flag to name your terminal `mcs`)

   ```
   sudo screen -S mcs java -Xmx1024M -Xms1024M -jar server.jar nogui
   ```

Detach from the screen and close your SSH session

1. To detach the screen terminal, press **Ctrl+A**, **Ctrl+D**. The terminal continues to run in the background. To reattach the terminal, run the following command:

   ```
   sudo screen -r mcs
   ```
2. If necessary, exit the screen terminal by pressing **Ctrl+A**, **Ctrl+D**.
3. To exit the SSH terminal, run the following command:

   ```
   Exit
   ```

Congratulations! You set up and customized a VM and installed and configured application software—a Minecraft server

Task 4: Allow client traffic

Up to this point, the server has an external static IP address, but it cannot receive traffic because there is no firewall rule in place. Minecraft server uses TCP port 25565 by default. So you need to configure a firewall rule to allow these connections.

Create a firewall rule

Run the following command in the cloud shell

gcloud compute --project=qwiklabs-gcp-04-98a940638d63 firewall-rules create minecraft-rule --direction=INGRESS --priority=1000 --network=default --action=ALLOW --rules=tcp:25565 --source-ranges=0.0.0.0/0 --target-tags=minecraft-server

The output should contain the following (**do not copy; this is example output**):

```
Creating firewall...⠹Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-04-98a940638d63/global/firewalls/minecraft-rule].
Creating firewall...done.
NAME            NETWORK  DIRECTION  PRIORITY  ALLOW      DENY  DISABLED
minecraft-rule  default  INGRESS    1000      tcp:25565        False
```

Verify server availability

1. Locate and copy the **External IP address** for the **mc-server** VM from the cloud shell
2. Use the following website to test your Minecraft server: https://mcsrvstat.us/

Task 5: Schedule regular backups

Backing up your application data is a common activity. In this case, you configure the system to back up Minecraft world data to Cloud Storage.

1. SSH to **mc-server**, run the following command:

   ```
   gcloud compute ssh mc-server
   ```
2. Create a globally unique bucket name, and store it in the environment variable YOUR_BUCKET_NAME. To make it unique, you can use your Project ID. Run the following command:

   ```
   export YOUR_BUCKET_NAME=qwiklabs-gcp-04-98a940638d63
   ```
3. Verify it with echo:

   ```
   echo $YOUR BUCKET NAME
   ```
4. To create the bucket using the gsutil tool, part of the Cloud SDK, run the following command:

   ```
   gsutil mb gs://$YOUR_BUCKET_NAME-minecraft-backup
   ```

Create a backup script

1. In the mc-server SSH terminal, navigate to your home directory:

   ```
   cd /home/minecraft
   ```
2. To create the script, run the following command:

   ```
   sudo nano /home/minecraft/backup.sh
   ```
3. Copy and paste the following script into the file:

   ```
   #!/bin/bash
   screen -r mcs -X stuff '/save-all\n/save-off\n'
   /usr/bin/gsutil cp -R ${BASH_SOURCE%/*}/world gs://${YOUR_BUCKET_NAME}-minecraft-backup/$(date "+%Y%m%d-%H%M%S")-world
   screen -r mcs -X stuff '/save-on\n'
   ```
5. To make the script executable, run the following command:

   ```
   sudo chmod 755 /home/minecraft/backup.sh
   ```

Test the backup script and schedule a cron job

1. In the mc-server SSH terminal, run the backup script:

   ```
   . /home/minecraft/backup.sh
   ```
2. After the script finishes, return to the Cloud Console.
3. To verify that the backup file was written, on the **Navigation menu**, click **Storage** > **Browser**.

4. Click on the backup bucket name. You should see a folder with a date-time stamp name. Now that you've verified that the backups are working, you can schedule a cron job to automate the task.
5. In the mc-server SSH terminal, open the cron table for editing:

```
sudo crontab -e
```

6. When you are prompted to select an editor, type the number corresponding to **nano**, and press **ENTER**.
7. At the bottom of the cron table, paste the following line:

```
0 */4 * * * /home/minecraft/backup.sh
```

That line instructs cron to run backups every 4 hours.

8. Press **Ctrl+O**, **ENTER** to save the cron table, and press **Ctrl+X** to exit nano.

This creates about 300 backups a month in Cloud Storage, so you will want to regularly delete them to avoid charges. Cloud Storage offers the Object Lifecycle Management feature to set a Time to Live (TTL) for objects, archive older versions of objects, or "downgrade" storage classes of objects to help manage costs.

Task 6: Server maintenance

To perform server maintenance, you need to shut down the server.

Connect via SSH to the server, stop it and shut down the VM

1. In the mc-server SSH terminal, run the following command:

```
sudo screen -r -X stuff '/stop\n'
```

2. Enter cd in the shell to return to the mc-server path

   Output will appear as follows:

```
student-04-7a523dd41f41@mc-server:~$
```

3. Run the following command the stop the mc-server instance:

```
student-04-7a523dd41f41@mc-server:~$ sudo poweroff
```

4. You will be logged out of your SSH session.

To start up your instance again, run the following command:

```
gcloud compute instances start mc-server
```

if the following message appears,

```
Did you mean zone [europe-west1-b] for instance: [mc-server] (Y/n)?  n
```

select n and wait to be directed to the mc-server vm path

Automate server maintenance with startup and shutdown scripts

1. Open a new shell terminal
2. Instead of following the manual process to mount the persistent disk and launch the server application in a screen, you can use metadata scripts to create a startup script and a shutdown script to do this for you.
3. To add the startup script run the following command:

```
gcloud compute instances add-metadata mc-server \
  --metadata startup-script-url=https://storage.googleapis.com/cloud-
training/archinfra/mcserver/startup.sh
```

4. To add the shutdown script run the following command

```
gcloud compute instances add-metadata mc-server \
  --metadata shutdown-script-url=https://storage.googleapis.com/cloud-
training/archinfra/mcserver/shutdown.sh
```

When you restart your instance, the startup script automatically mounts the Minecraft disk to the appropriate directory, starts your Minecraft server in a screen session, and detaches the session. When you stop the instance, the shutdown script shuts down your Minecraft server before the instance shuts down. It's a best practice to store these scripts in Cloud Storage.

4. Restart the mc-server vm by running the following comman:

```
gcloud compute instances start mc-server
```

if the following message appears,

```
Did you mean zone [europe-west1-b] for instance: [mc-server] (Y/n)?  n
```

select n and wait to be directed to the mc-server vm path

The output should contain the following (**do not copy; this is example output**):

```
Updated [https://compute.googleapis.com/compute/v1/projects/qwiklabs-gcp-04-
98a940638d63/zones/us-central1-a/instances/mc-server].
Instance internal IP is 10.128.0.2
Instance external IP is 34.66.142.205
```