Abdoul-Nourou Yigo

Configurations in Google Cloud for a Successful Project

Some particular procedures need to be taken into account for a good cloud set up

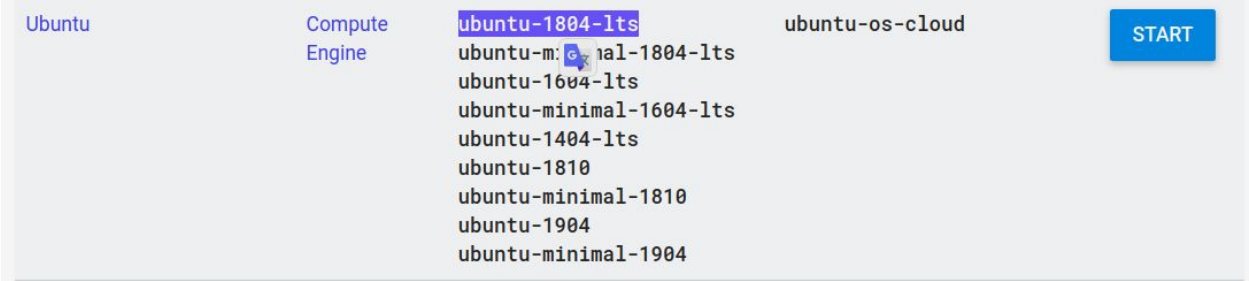
Virtual Machine Creation on Google Cloud

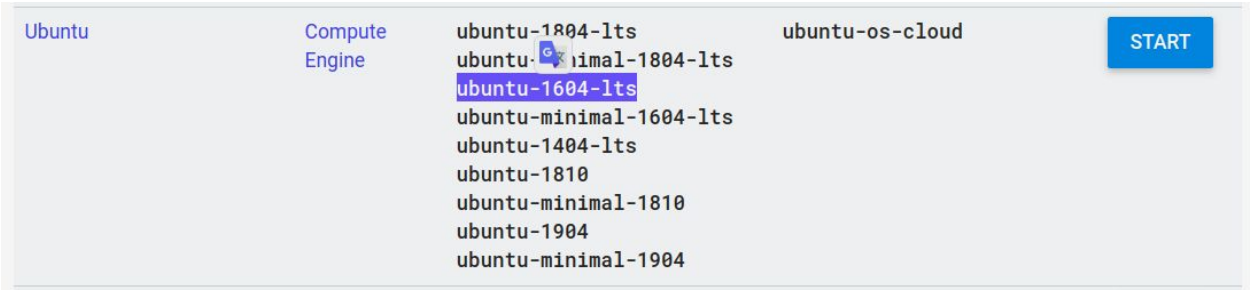
Since the project is based on socket programming, the first thing you would have to do is create VM in Cloud within which you can run your code. The VM would play the role of the server in Google Cloud. You have to carefully choose the VM image (Operating System (OS)) to ensure the right compatibilities when it comes to the installation of some specific packages. Failing to install the right VM could create the following issues:

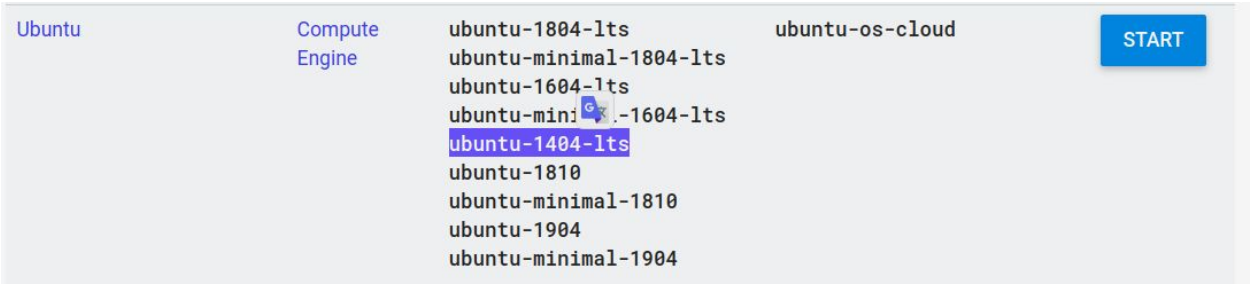
- Difficulties to install the right python packets
    - Without the right python packages, it would be impossible to execute your code on the VM
- In case there is a needs to install some servers such as Apache server and MySQL server. It could be challenging to do the configuration of those servers the wrong OS in Google Cloud.
- If the MySQL server is not installed in the VM, it could be impossible to connect to the SQL instance within Google Cloud.

Characteristic of VM that could be installed

To perform the configuration the server within the VM in Google Cloud. We use Linux Ubuntu with the right version. The following pictures are showing the versions that could be used for the VM set up:







The OS image highlighted could be used to create the VM in Google Cloud. You could access these resources: https://cloud.google.com/compute/docs/images

Command to install the right Python Packages

Since the primary language for the project is Python some important packages needed to be installed. After the VM is freshly installed some crucial need to be installed:

- First of all, you need to login to the VM instance using ssh
- In the terminal execute the following commands to update and upgrade the VM instance:
    - sudo apt update
    - sudo apt dist-upgrade
- Installation of python-pip command:
    - sudo apt install python-pip
- Installation of PyCryptodome by sequentially typing this command:
    - pip apt-get install build-essential python-dev
    - pip install pycryptodomex
    - python -m Cryptodome.SelfTest

In case if you want to use MySQL resources, you would have to install the appropriate packages, but at the same time, you need to appropriately configure MySQL server. For now, this is the command to install the MySQL connector

- pip install mysql-connector-python
- pip install pymysql

With these installations, you are ready to go when it comes to using MySQL resources to manipulate your information within a database.

Installation of MySQL server

Some particular procedures need to be taken for the installation of this server. You need to on the terminal to perform the configuration of the server. The following commands would satisfy the installation requirements:

- sudo apt-get update
- sudo apt-get install mysql-server

Access the Server Remotely

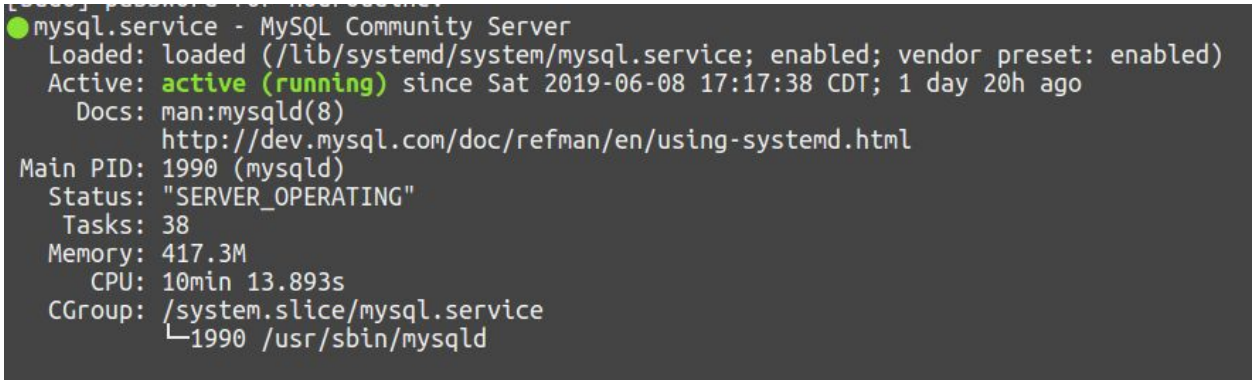
The execution of the following command could ensure remote access to the server:

- sudo ufw allow mysql

Start the Server and Check its Status

To make sure that the server is functioning you could run the following commands:

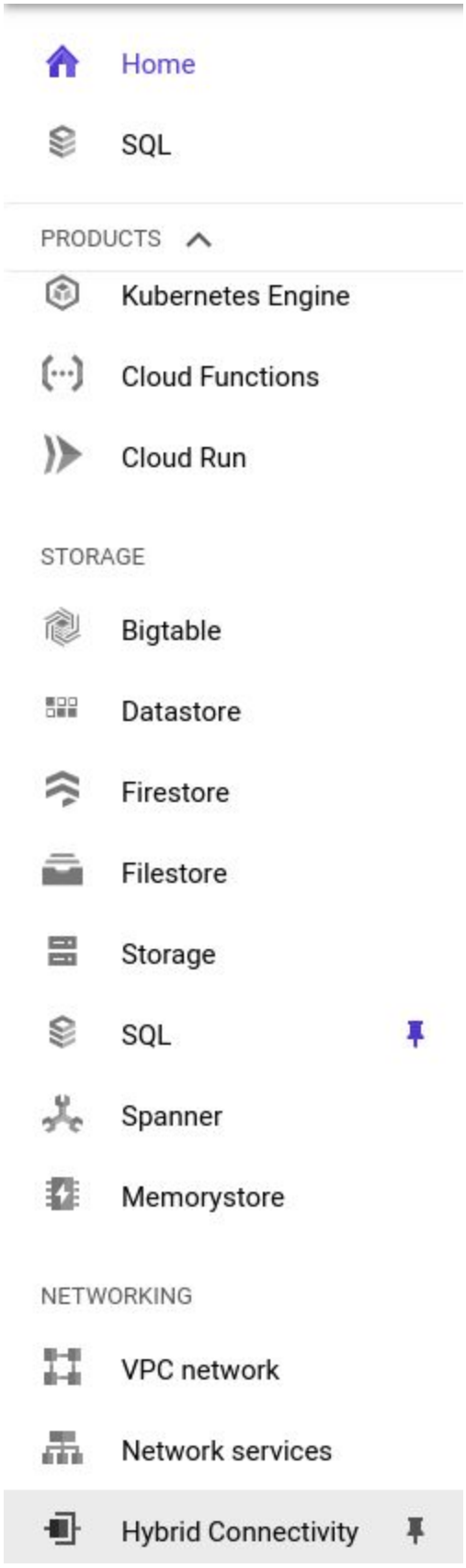
- sudo service mysql start
- Sudo service mysql status
- You should some like the illustration if everything correct works as expected:



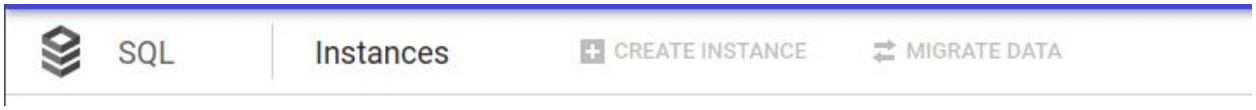
Now the MySQL server is properly installed, you could set up the SQL instance in Google Cloud.

Installation of the SQL instance in Google Cloud

You would need to make a few steps to set up the SQL functionalities. The following illustration would clearly show how to perform the configuration:


Home
SQL
PRODUCTS
Kubernetes Engine
Cloud Functions
Cloud Run
STORAGE
Bigtable
Datastore
Firestore
Filestore
Storage
SQL
Spanner
Memorystore
NETWORKING
VPC network
Network services
Hybrid Connectivity

After clicking the SQL on the menu, you would have to create the SQL server instance



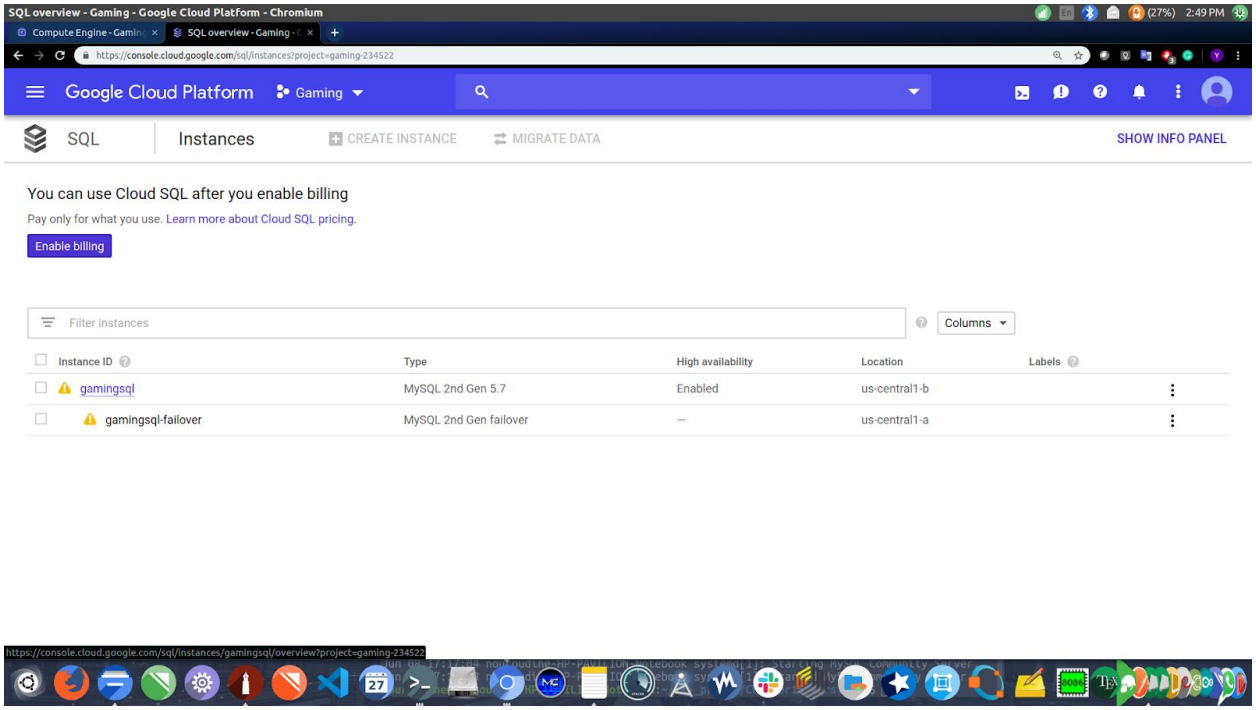
When the instance is well created, you could also set up a failover procedure to backup your data, but this comes at a cost because they charge more. The following illustration is showing how everything is well set up if everything is working well:

Now that the SQL instance is installed you could connect to the VM instance.

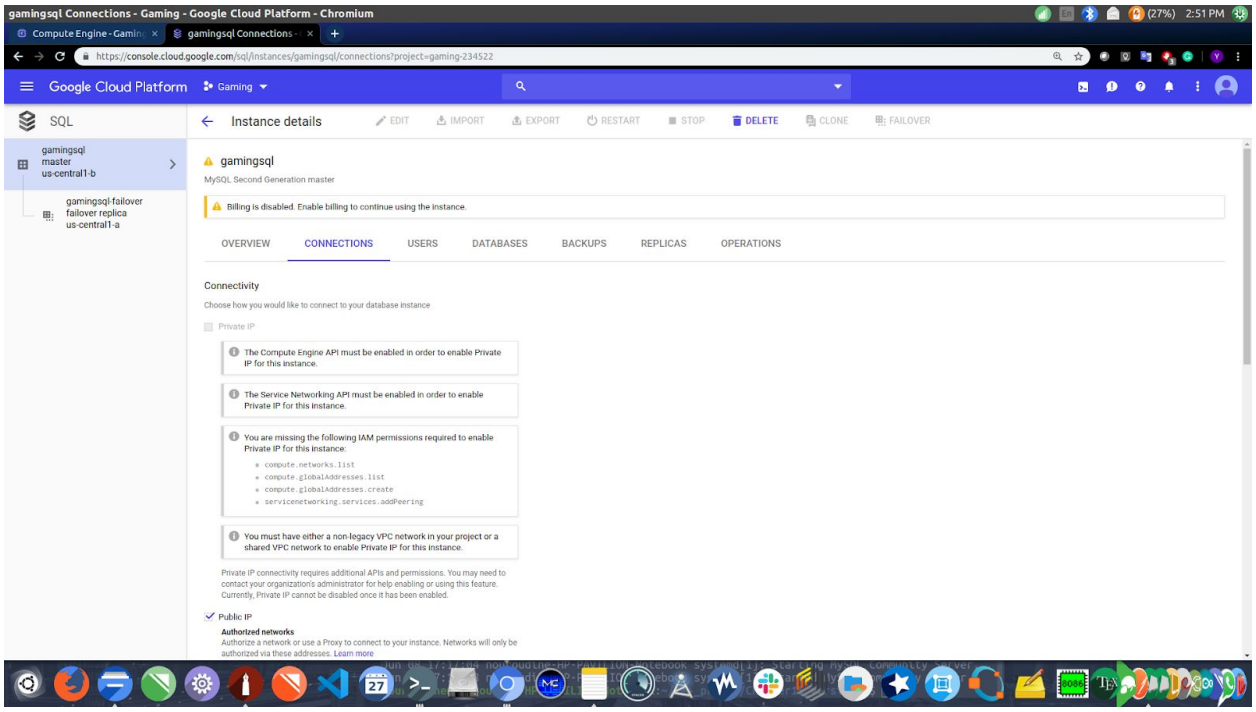
Configuration to Connect the VM instance and the SQL instance

This configuration needs to be done to ensure that the VM instance and the SQL instance can communicate. The following instructions need to be followed:

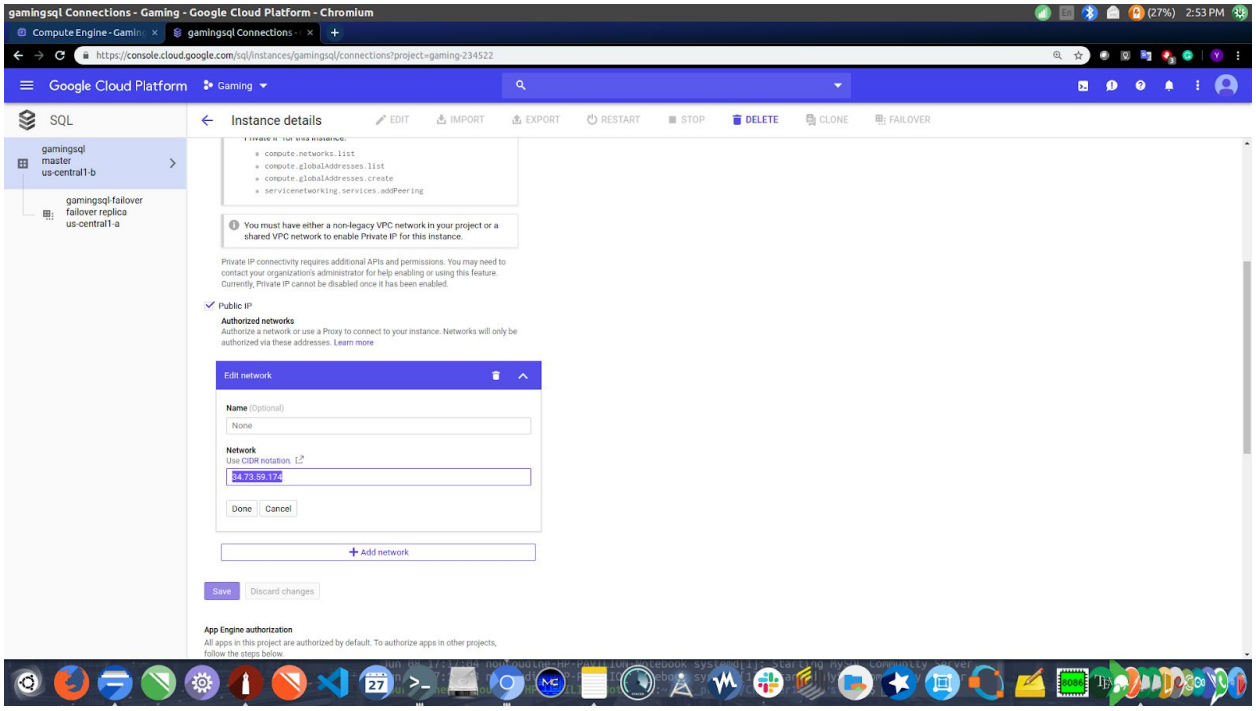
Click on the SQL instance:



Click on CONNECTIONS

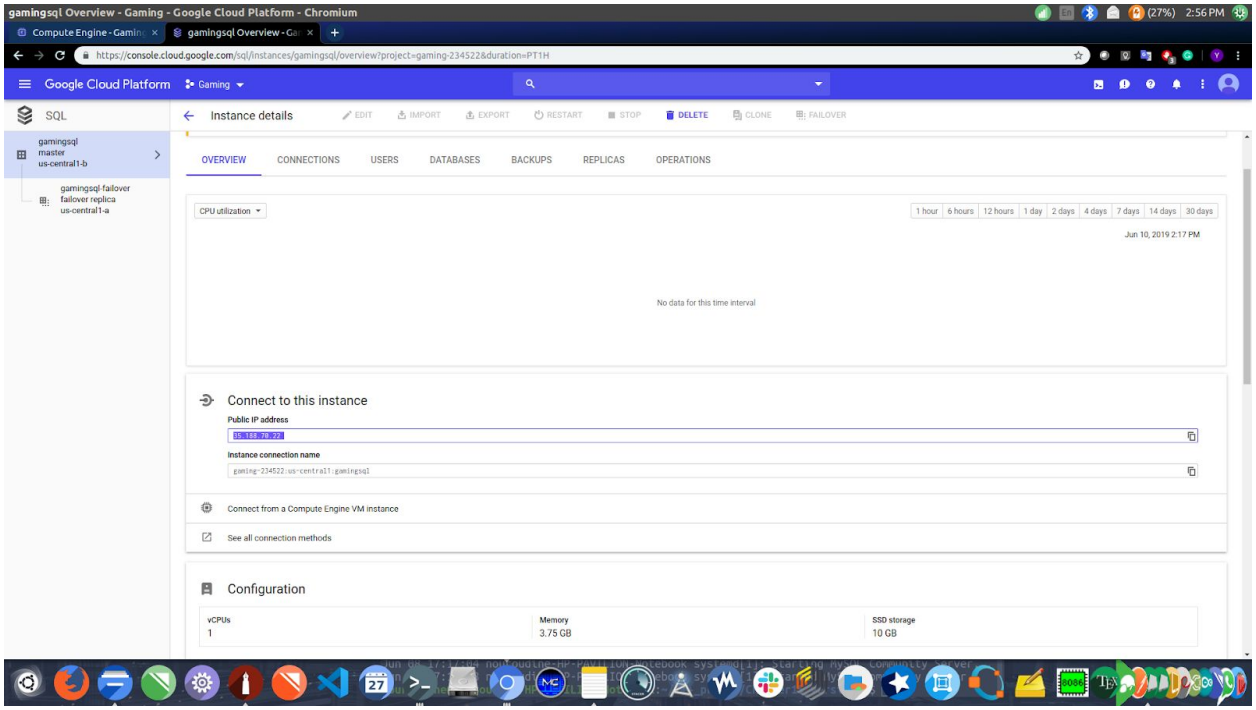


Copy and pass the instance IP address of the VM instance and save:



During the setup, you would also need to create a user account for the root user so that the root user could access the server with valid credentials. You could figure out to do that when you would be doing the configuration of the SQL instance.
After these procedures, everything should work like a charm.

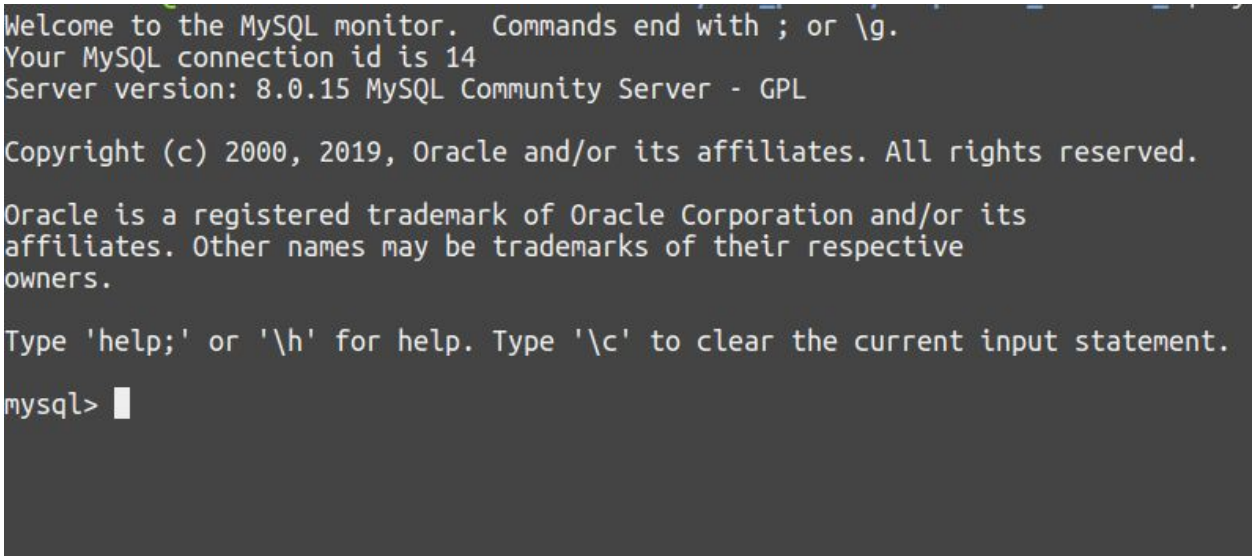
Now it is time to go back to the VM instance to test the connection. For that step, you need to consider the SQL instance IP address. Illustrated in the following illustration:



You need the SQL instance IP address to connect from the VM instance. Before that you need to login to the VM via ssh using the following command in the terminal:

```
mysql --host=SQL_INSTANCE_IP_ADDRESS\
  --user=root --password
```

You must replace the SQL_INSTANCE_IP_ADDRESS with the actual VM IP address as shown in the above picture. If everything works perfectly, you should see the MySQL shell like this:

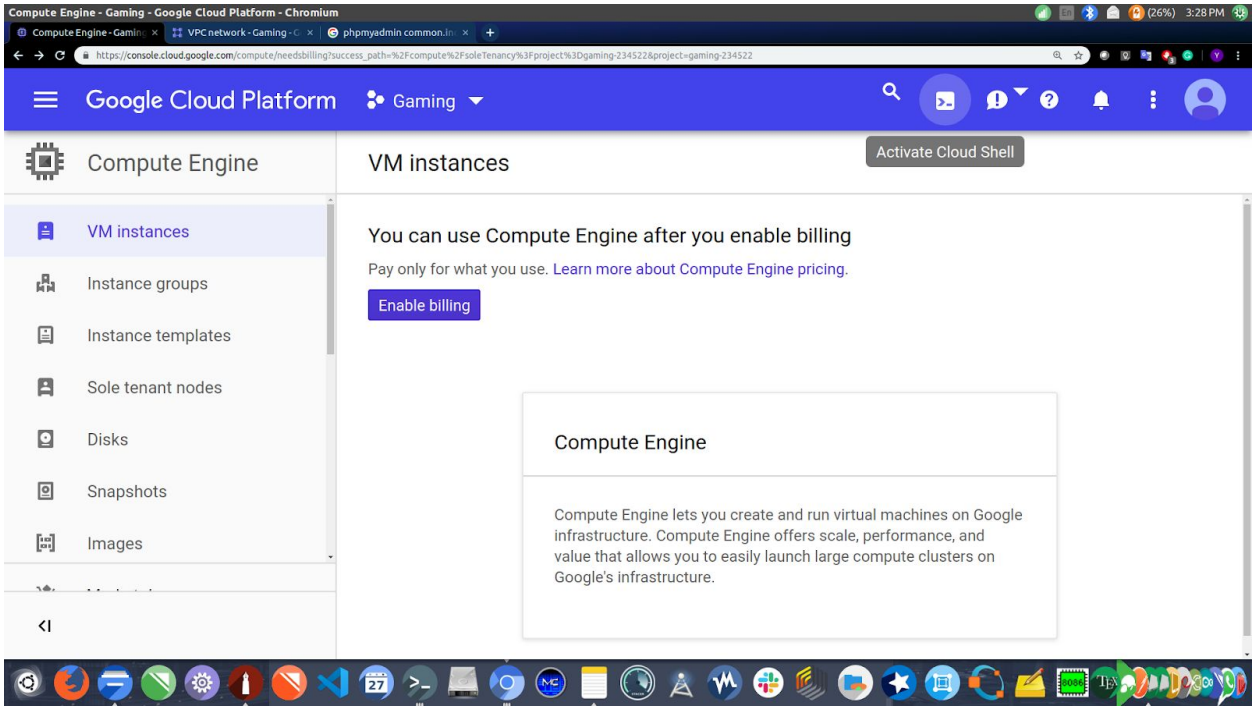


After this result, the VM instance and SQL instance are perfectly connected. You could start creating your tables.

Project Deployment With phpMyAdmin

Without a user interface, the MySQL database manipulation could be challenging. To ease this complexity, you can perform a sophisticated configuration of phpMyAdmin so that you could work effectively and efficiently for more productivity.

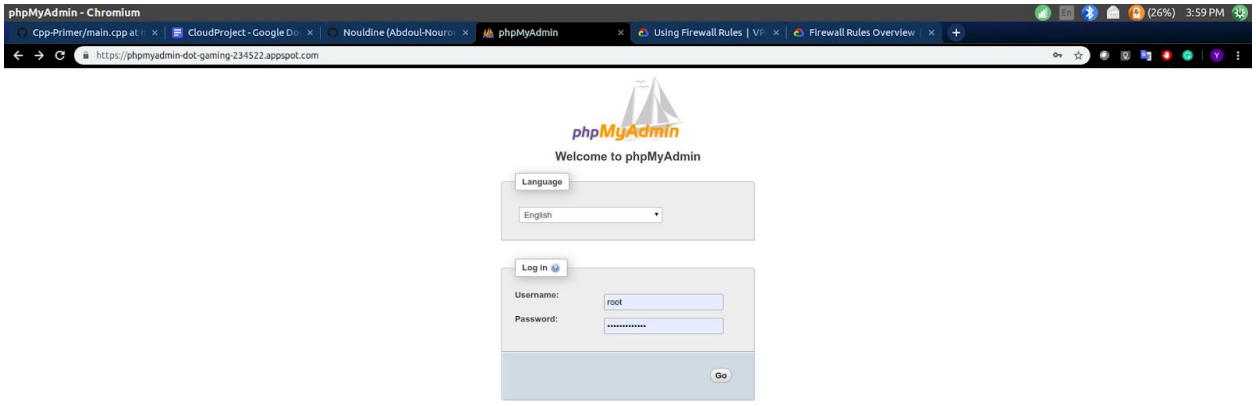
First of all, you need to open the Cloud Shell:



You could follow the instruction from this page to perform the configuration: [phpMyAdmin Configuration]

Note: Please read and think carefully for a successful configuration.

If the configuration is well performed, you should see this:



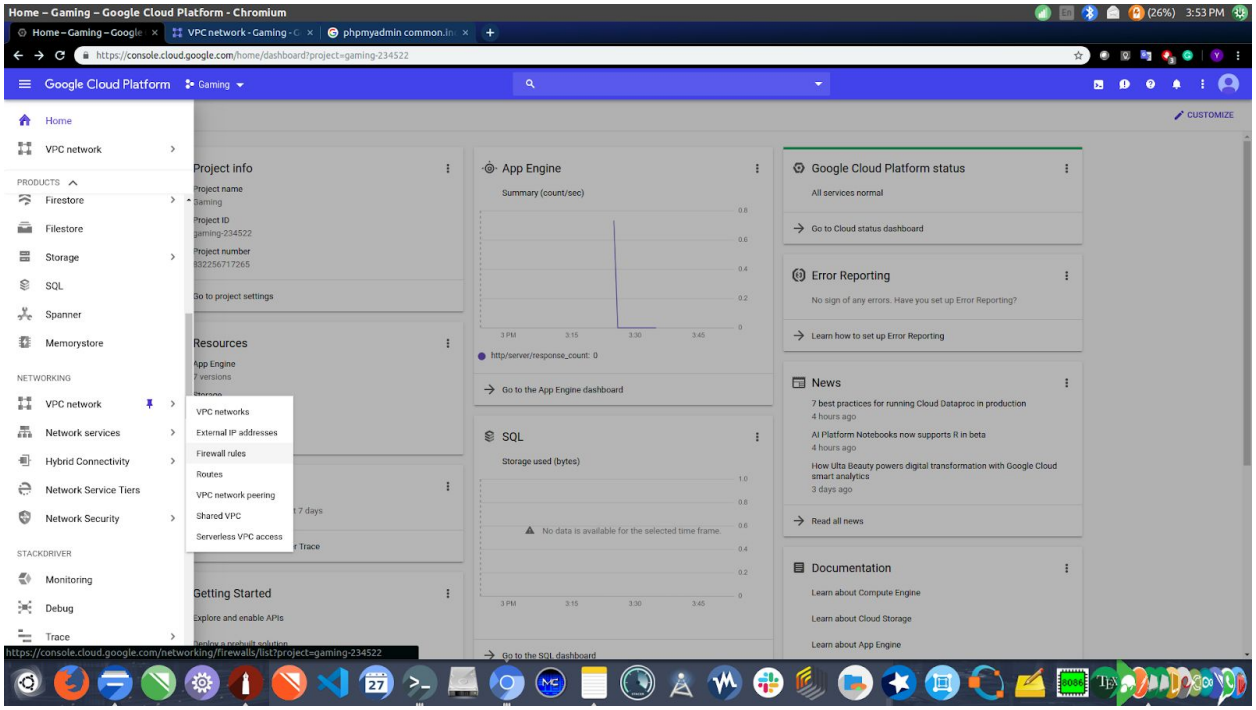
Firewall Rules

Since the project is based on socket programming, it is necessary to understand firewall rules within Google Cloud to execute the right configuration. For instance, you would be using port numbers to be able to communicate from one side to another side under the TCP protocol. In Google Cloud, you need to be able to open a sufficient number of ports so that you could connect the IAM and the CLIENT to the Cloud server.

To be more effective, you could create two different Python scripts to handle requests from the IAM and CLIENT. One of the scripts could handle requests from the IAM server and the other could handle requests from the CLIENT using different port numbers. This would make sure that everything could work smoothly.

To accomplish this setup you could create new firewall rules with a range of TCP ports. For instance, you could open the number of ports you want as long as they are within the port's configuration interval.

Here are the firewall rules in Google Cloud:



Since I do not have enough credit I am not able to display the firewall rules I have configured. However, the following links would show how to set up new firewall rules:

[Firewall Rules]
[Firewall Rules Overview]

Note: I hope that would help. Thanks.