Virtual Cloud Computing Assignment

A PG Diploma Project Report

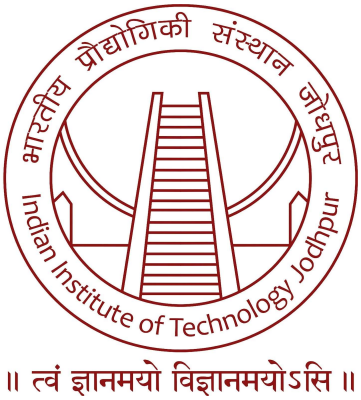
Submitted by

Naresh krishna Vemuri(G23AI2025)

Under the Supervision

Of

Sumit Karla



Department of Artificial Intelligence

Indian Institute of Technology Jodhpur

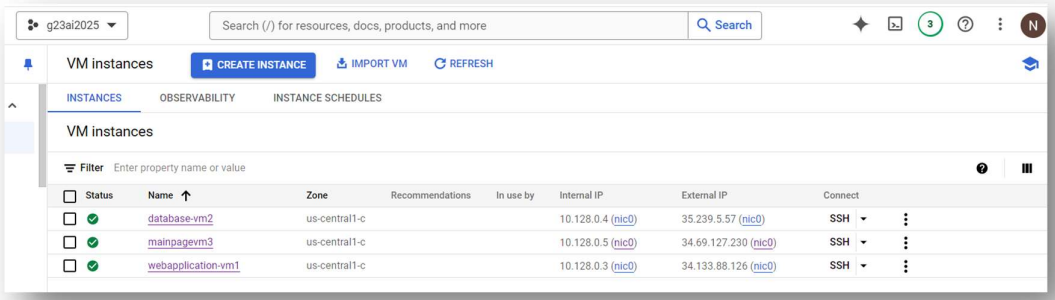
Objective:

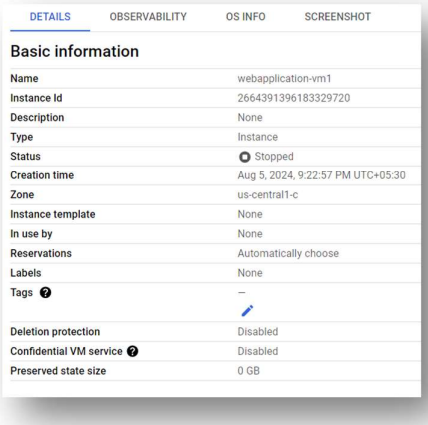
The goal is to create a web application accessible via a VM on GCP. The application will interact with Mongo database hosted on another VM. A third VM will act as a reverse proxy to forward traffic to the web application server.

Created three VM's:

- webapplication-vm1: Hosts the web application (Apache, PHP)
- database-vm2: Hosts the Mongodb database
- mainpagevm3: Acts as a reverse proxy to forward traffic to VM1

Screenshot:





Steps and Procedures for creating:

1. **VM1 Setup:**

   a) Create VM1:

   - zone: us-central1
   - Machine type: e2-micro (1 vCPUs, 10 GB memory)
   - Image: ubuntu
   - Firewall: Allow Http and Allow Https

   b) Install Python and Flask:

```
sudo apt update

sudo apt install apache2 -y

sudo apt install python3-pip -y

pip3 install flask pymongo
```

c) Copy the code from Github repo and upload it to VM1. execute app.py

https://github.com/g23ai2025/VCC_GCP_Assignment2/tree/main

```
python3 app.py
```

2. VM2 Setup:
   a) Create VM2:
      - zone: us-central1
      - Machine type: e2-micro (1 vCPUs, 10 GB memory)
      - Image: ubuntu
      - Firewall: Allow Http and Allow Https

   b) Install MongoDB:

```
sudo apt update

sudo apt install python3-pip -y

sudo apt install -y mongodb

sudo systemctl start mongodb

sudo systemctl enable mongodb

sudo vi /etc/mongodb.conf
```

      - Open mongodb.conf file and change bind IP to 0.0.0.0 and enable port 27017.
      - After update restart mongodb.

3. VM3 Setup:
   a) Create VM3:
      - zone: us-central1
      - Machine type: e2-micro (1 vCPUs, 10 GB memory)
      - Image: ubuntu
      - Firewall: Allow Http and Allow Https
   b) Install Apache setup:

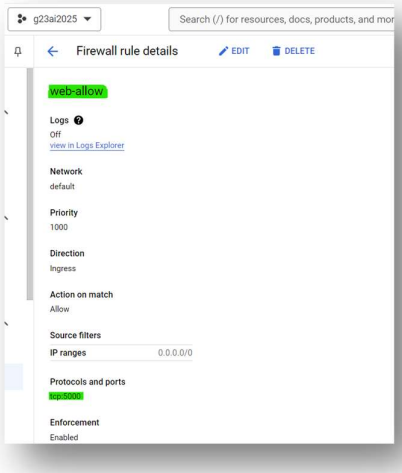
```
step1: exe following commands
        sudo apt-get update
        sudo apt-get install apache2
        sudo a2enmod proxy
        sudo a2enmod proxy_http
        sudo systemctl restart apache2

step2: Open 000-defalut.conf file and Add proxy configurations.

      $ sudo vi /etc/apache2/sites-available/000-default.conf

          #Proxy Configuration
          ProxyRequests on
          ProxyPass / http://34.133.88.126:5000/
          ProxyPassReverse / http://34.133.88.126:5000/

step3: Restart Apache2
      $ sudo systemctl restart apache2
```

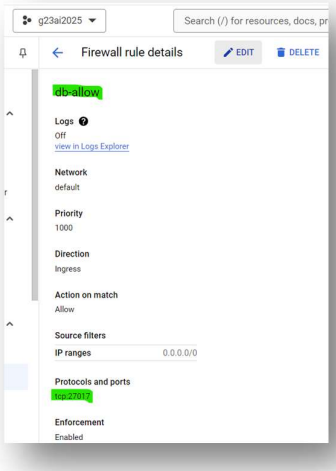
Firewall Policy Setup:

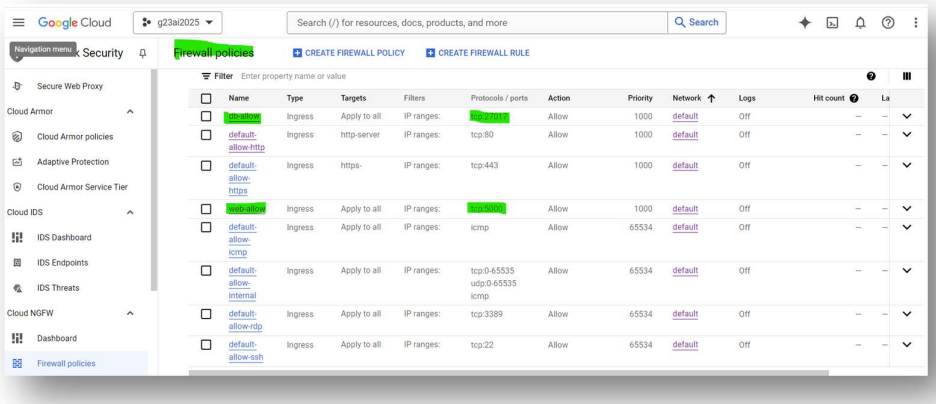
My **web** application is running on **5000** port and **MongoDB** is running on **27017** Port so we need to add both port in firewall to access the web application by creating firewall policy.

a) Creating Firewall Policy:

- In Google Clound Console click on Navigation Menu
- Click on **VPC Network**
- Click on **Firewall**
- Click on **Create firewall Rule**
  i. webapplication-vm1 to allow port **5000**
  ii. database-vm2 to allow port **27017**







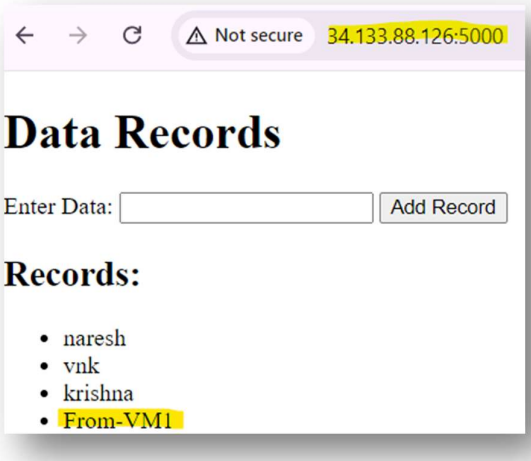
Accessing the Application:

Step1:

1. Connect to VM1 and execute app.py file

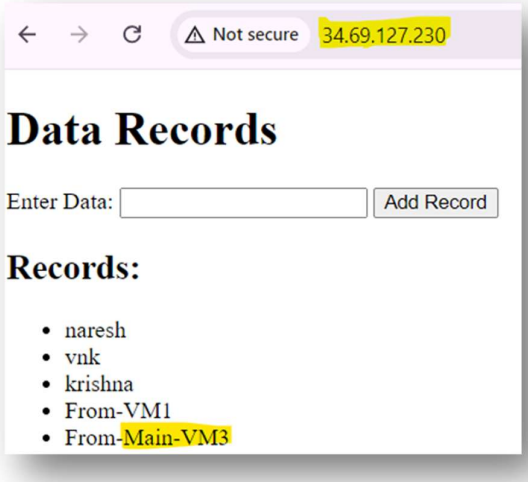
```
g23ai2025@webapplication-vm1:~/app$ python3 app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deplo
tead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://10.128.0.3:5000
Press CTRL+C to quit
49.37.134.164 - - [05/Aug/2024 17:32:37] "GET / HTTP/1.1" 200 -
34.69.127.230 - - [05/Aug/2024 17:45:56] "GET / HTTP/1.1" 200 -
```

2. **Accessing web application using VM1 external IP (34.133.88.126:5000)**



Step2:

Open Browser and access the web application using external IP of **VM3** (**34.69.127.230**)



Conclusion

By following the steps outlined above, we have successfully set up a web application on VM1, a Mongodb database on VM2, and configured VM3 to act as a reverse proxy to forward traffic to VM1. The web application is now accessible via VM3's external IP, and data is correctly inserted and retrieved from the Mongodb database on VM2.