

Exploring Google Cloud Services

Arslan Koshimov

Date of submission: 06.10.2024

Table of Contents

Introduction

The goal of this assignment is to gain hands-on experience with Google Cloud services, specifically focusing on virtual machines, storage solutions, and networking. I will set up and configure different services within Google Cloud and document their findings.

Virtual Machines in Google Cloud

• Create a Virtual Machine (VM) Instance

1. Select options for vm. Actually I changed name to cad-test-vm-linux, and OS to Ubuntu 20.04 LTS. Also ensure to allow SSH, HTTP and HTTPS traffic.

The screenshot shows the Google Cloud 'Create an instance' interface. The 'Operating system and storage' tab is selected, displaying configuration for a VM named 'cad-test-vm-linux'. The OS is Ubuntu 20.04 LTS, the machine type is e2-medium, and the disk size is 10 GB. The 'Monthly estimate' is \$25.46. The 'Equivalent code' tab on the right shows the gcloud command and Terraform code for creating the instance.

2. Copy code from previous step, and run it in console(where you created project and login)

```
koshlov@arian998:~$ gcloud compute instances create cad-test-vm-linux \
--project=triple-baton-397512 \
--zone=us-central1-c \
--machine-type=e2-medium \
--network-interface=network-tier=PREMIUM,stack-type=IPV4_ONLY,subnet=default \
--maintenance-policy=MIGRATE \
--provisioning-model=STANDARD \
--service-account=2439297869-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/service.management.readonly,https://www.googleapis.com/auth/trace.append \
--create-disk=auto-delete=yes,boot=yes,device-name=cad-test-vm-linux,image=projects/ubuntu-os-cloud/global/images/ubuntu-2004-focal-v20240830,mode=rw,size=10,type=pd-balanced \
--no-shielded-vmm \
--shielded-secure-boot \
--shielded-integrity-monitoring \
--reservation-affinity=anyd-gcloud \
Created [https://www.googleapis.com/compute/v1/projects/triple-baton-397512/zones/us-central1-c/instances/cad-test-vm-linux].
NAME: cad-test-vm-linux
ZONE: us-central1-c
MACHINE_TYPE: e2-medium
PREEMPTIBLE:
INTERNAL_IP: 10.128.0.2
EXTERNAL_IP: 34.121.183.217
STATUS: RUNNING
```

3. So, then I closed my billing account, therefor my vm stopped

Failed to start cad-test-vm-linux: This API method requires billing to be enabled. Please enable billing on project #triple-baton-397512 by visiting <https://console.developers.google.com/billing/enable?project=triple-baton-397512> then retry. If you enabled billing for this project recently, wait a few minutes for the action to propagate to our systems and retry.

- **Connect to the VM**

1. Run: **gcloud compute ssh cad-test-vm-linux --zone us-cenrall-c**

- **Install a Web Server (Nginx)**

1. Update package: **apt update**
2. Install nginx: **apt install nginx**
3. Start nginx: **systemctl start nginx**
4. Run: **systemctl enable nginx**, when it will system will start, it will auto start service nginx.
5. Navigate to default directory where it see file for provide content and create simple hello-world html page:
cd /var/www/html
and
echo "<h1>Hello, World!</h1>" | sudo tee index.html
6. Also maybe you want to configure nginx-conf for setup it for global address

Storage Solutions in Google Cloud

- **Create a Cloud Storage Bucket**

1. First of all, login in console (**gcloud auth login**) and go to project.
2. Create bucket:

```
gsutil mb -l us-centrall-c gs://test-bucket-for-cad
```

But I saw:

```
koshimovarslan999@cloudshell:~ (triple-baton-397512)$ gsutil mb -l us-centrall gs://test-bucket-for-cad
Creating gs://test-bucket-for-cad/...
AccessDeniedException: 403 The billing account for the owning project is disabled in state closed
```

- Account billing disabled

3. Let's make it public:
gsutil iam ch allUsers:objectViewer gs://test-bucket-for-cad
4. If you want check permission, run these:
gsutil iam get gs://test-bucket-for-cad
5. Let's upload file test.txt to home directory of vm (/), run these:
gsutil cp test.txt gs://test-bucket-for-cad/

- **Implement Object Lifecycle Management**

1. Create file lifecycle.json by run: touch lifecycle.json and write into:

```
koshimovarslan999@cloudshell:~ (triple-baton-397512)$ cat lifecycle.json
{
  "rule": [
    {
      "action": {"type": "Delete"},
      "condition": {"age": 30}
    }
  ]
}
```

(This configuration will delete objects after 30 days.)

2. Apply the Lifecycle Policy, by run:
gsutil lifecycle set lifecycle.json gs://test-bucket-for-cad/
3. Then, check is rule set:
gsutil lifecycle get gs://test-bucket-for-cad

Networking in Google Cloud

- **Create a VPC Network**

1. Go to VPC networks in the Google Cloud Console:
 - In the Navigation menu, go to VPC network > VPC networks.
 - Click Create VPC network.
2. Configure the VPC:
 - Enter a name for the VPC (e.g., my-vpc).
 - Set Subnet creation mode to Custom.
 - Click Add subnet, and in the pop-up:
 - Name: my-subnet
 - Region: us-central1
 - IP range: 10.0.0.0/24
 - Click Done.
 - Click Create to create the VPC network.

- **Create a Firewall Rule**

1. Go to Firewall rules in the Google Cloud Console:
 - Navigate to VPC network > Firewall.
 - Click Create firewall rule.
2. Configure the rule:
 - Name: allow-icmp
 - Network: Select your VPC (my-vpc).
 - Targets: All instances in the network.
 - Source IP ranges: 0.0.0.0/0 (to allow traffic from any IP).
 - Protocols and ports: Select Specified protocols and ports and check ICMP.
3. Create the firewall rule:
 - Click Create to apply the rule.

- **Connect VM to the VPC**

1. SSH into the VM:
 - In the VM instances list, click SSH next to your VM.
 2. Ping an external server:
 - Run the following command to ping Google's DNS server:
- ```
bash
ping 8.8.8.8
```

---

## Conclusion

This assignment provided hands-on experience with key Google Cloud services:

- **Virtual Machines (VM):** We learned how to create and configure VMs, install software like web servers, and manage server access using SSH. VMs offer flexible, scalable compute resources ideal for hosting applications and running workloads.
- **Cloud Storage Buckets:** We explored creating and managing storage buckets for securely storing and sharing data. Uploading files and setting lifecycle rules for automated data management helped optimize storage costs.
- **Networking:** Configuring firewalls for VMs demonstrated how to control network traffic securely, ensuring services remain accessible yet protected.

---

## References

- VPC Networking - <https://cloud.google.com/vpc/docs/vpc>
- Firewall Rules - <https://cloud.google.com/firewall/docs/firewalls>
- Compute Engine Instances - <https://cloud.google.com/compute/docs/instances>
- Medium - <https://medium.com/petabytz/vpc-creation-on-google-cloud-platform-gcp-67fa397c1106>