DevOps Lab

CLOUD COMPUTE - GCP

NETWORKING

Home tasks

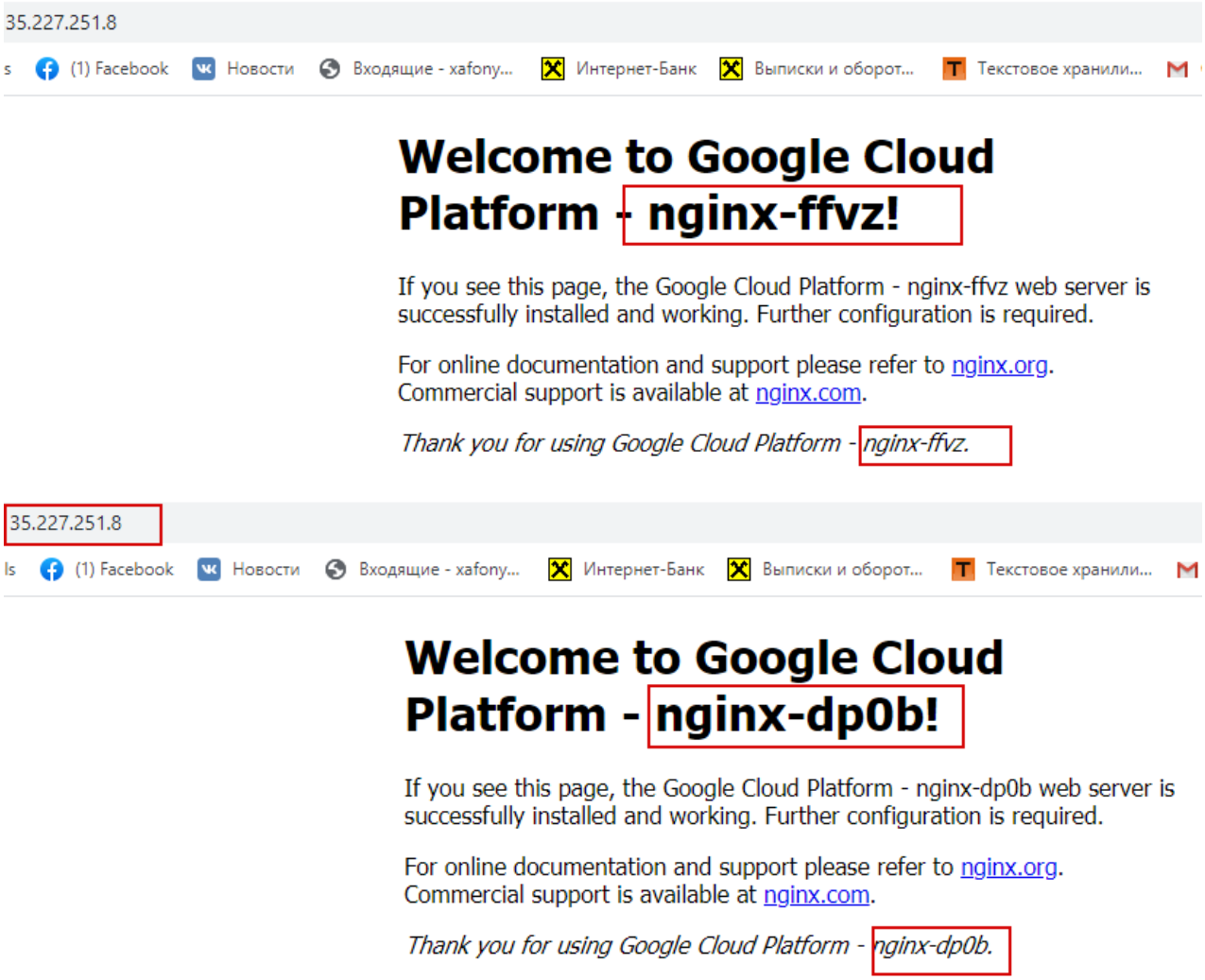
It’s aiming to gain knowledge about Networking in Google Cloud.

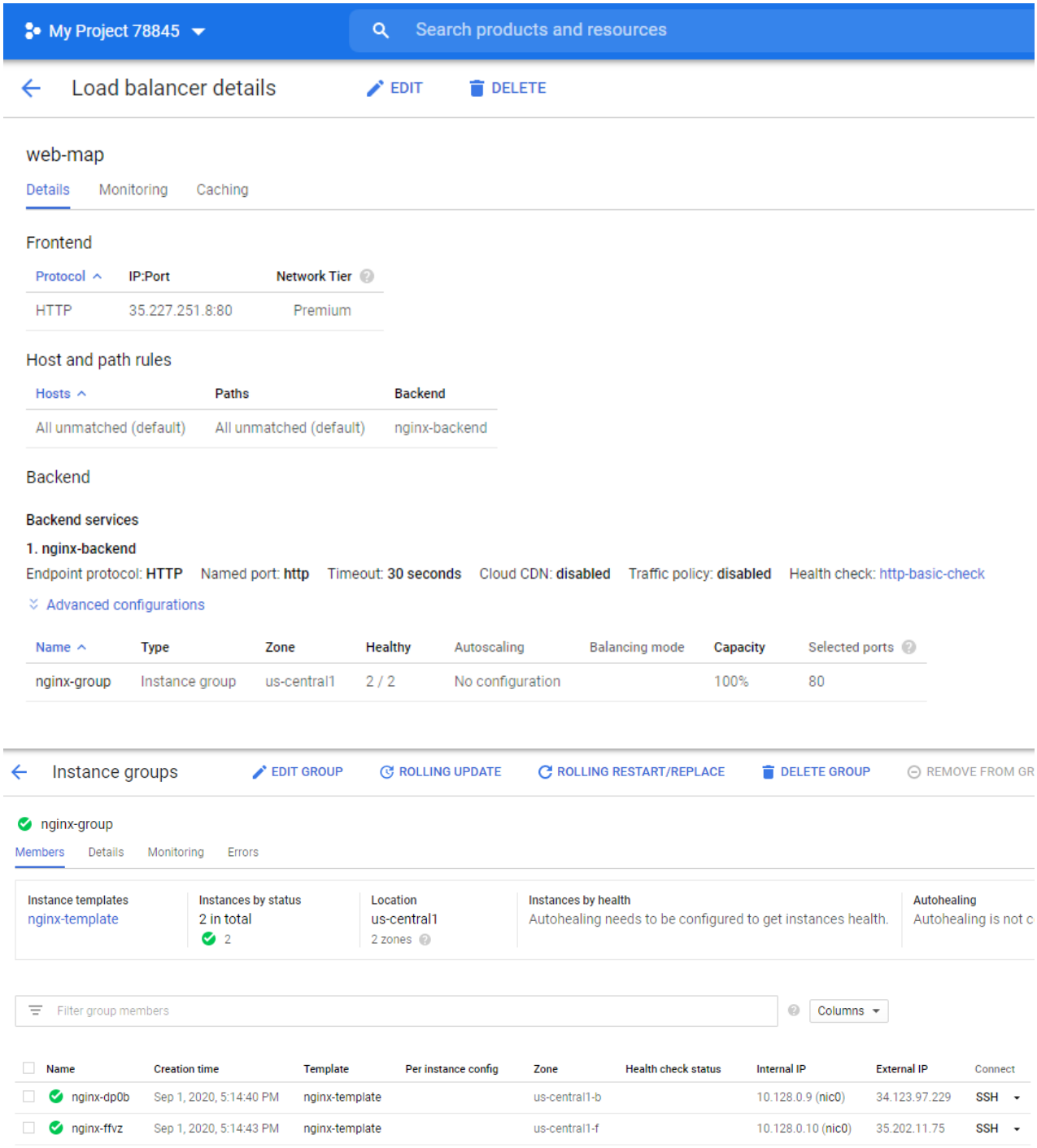
TASK 1

Learn about two types of load balancers in Google Cloud Platform:

- a L3 Network Load Balancer and
- a L7 HTTP(s) Load Balancer.

Lab Link: codelabs: LoadBalancers





TASK 2

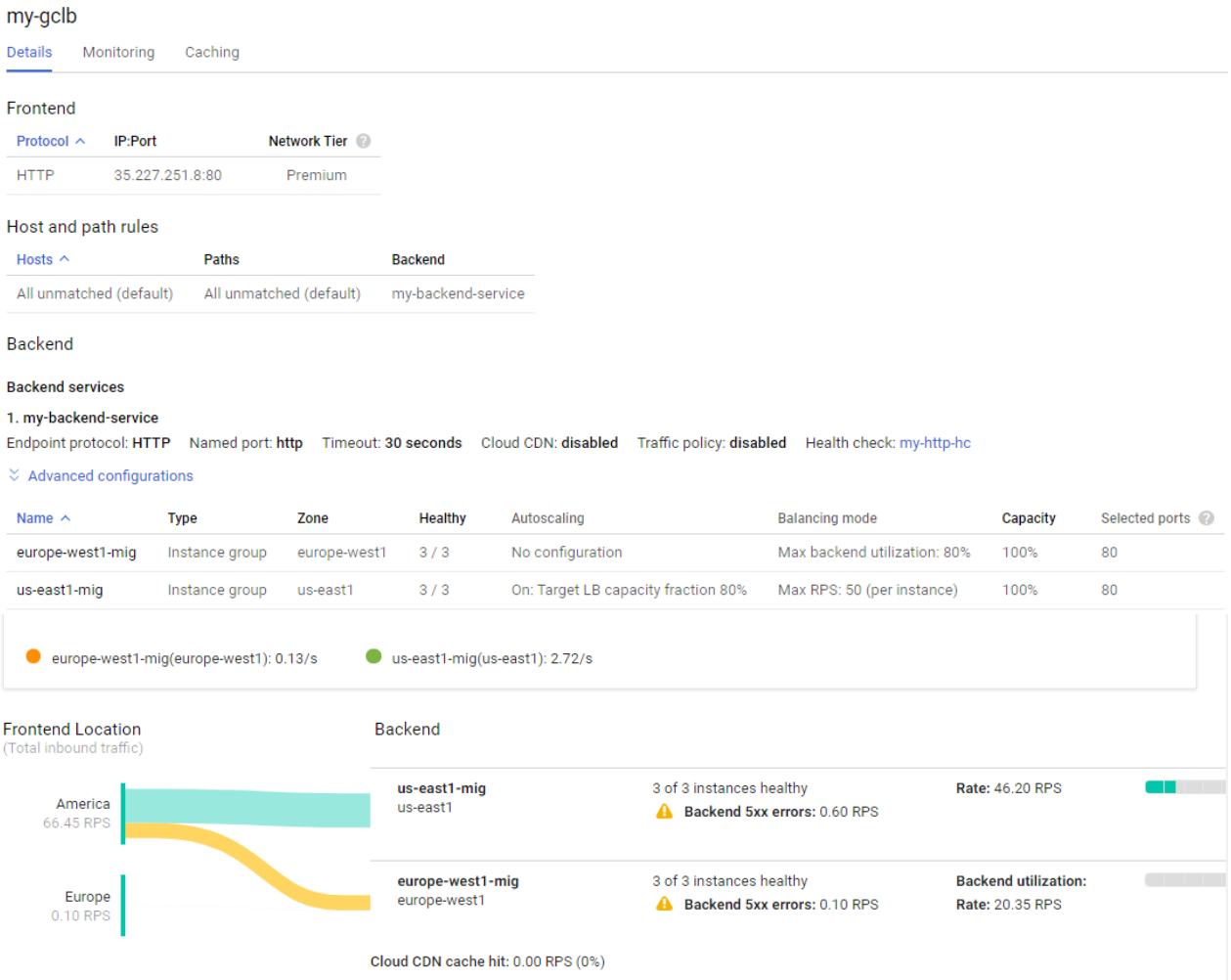
The Objectives are to learn:

- How to measure latency between Google Compute Engine regions and zones
- How to test network connectivity and performance using open source tools
- How to set up up basic firewalling to secure your networks
- How to set up a global HTTP Load Balancer with Managed Instance Groups to automatically scale your resources up and down based on request load
- How to test and monitor your HTTP Load Balancer setup

These exercises are ordered to reflect a common cloud developer experience as follows:

1. Set up your lab environment and learn how to work with your GCP environment.
2. Use of common open source tools to explore your network around the world.
3. Deploy a common use case: use of HTTP Load Balancing and Managed Instance Groups to host a scalable, multi-region web server.
4. Testing and monitoring your network and instances.
5. Cleanup.

Lab Link: codelabs: Neworking 101



TASK 3

The Objectives are to learn:

- Setting up NAT gateways
- How to restrict network traffic that certain tiers of an app cannot talk to each other
- Setting up alternate connectivity options to instances
- Map an external service to look like an internal service
- How to setup an Egress proxy limiting access to specific resources

Lab Link: codelabs: Neworking 102

```
[xafonya_gmail_com@nat-node-gcp-eu ~]$ curl google.com | tail
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  3509  100  3509    0     0   114k      0 --:--:-- --:--:-- --:--:--  118k
<p>Your cache administrator is <a href="mailto:root?subject=CacheErrorInfo%20-%20ERR_ACCESS_DENIED&amp;b
%5Bnone%5D%0D%0ATimeStamp%3A%20Wed,%2002%20Sep%202020%2010%3A32%3A35%20GMT%0D%0A%0D%0AClientIP%3A%20192.
3A%20curl%2F7.29.0%0D%0AAccept%3A%20*%2F*%0D%0AProxy-Connection%3A%20Keep-Alive%0D%0AHost%3A%20google.co
<br>
</div>

<hr>
<div id="footer">
<p>Generated Wed, 02 Sep 2020 10:32:35 GMT by nat-gw-eu (squid/3.5.20)</p>
<!-- ERR_ACCESS_DENIED -->
</div>
</body></html>
[xafonya_gmail_com@nat-node-gcp-eu ~]$
```

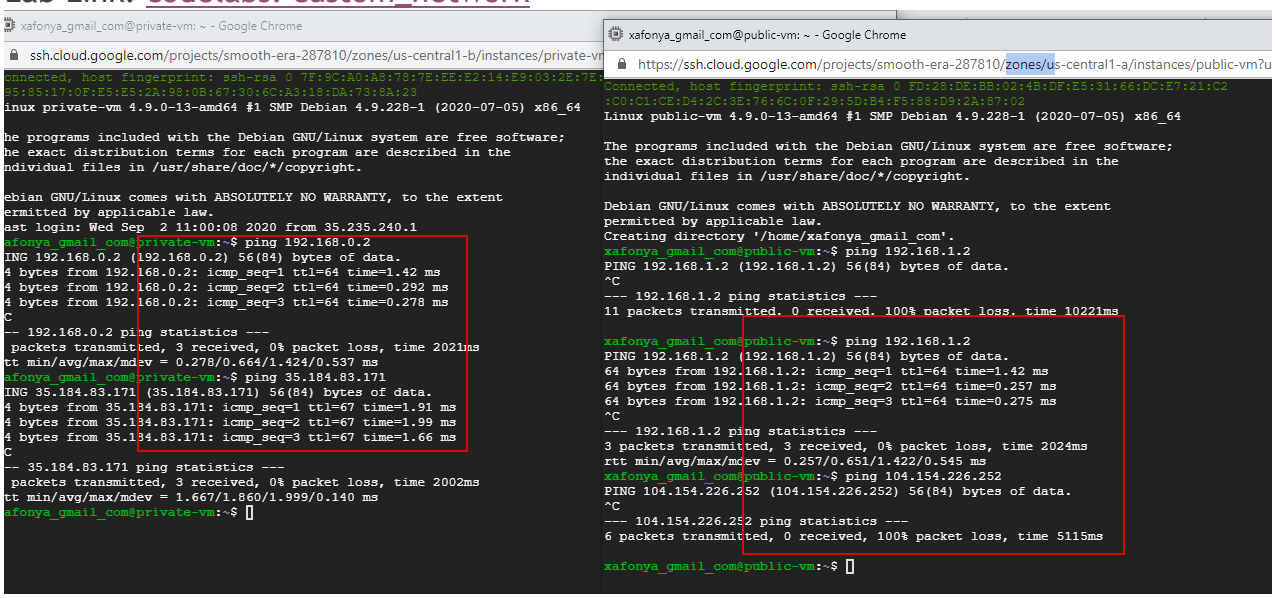
```
  [xafonya_gmail_com@nat-node-gcp-eu ~]$ curl 35.184.83.171 | tail -n 20
    % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                   Dload  Upload   Total   Spent    Left  Speed
  100 10701  100 10701    0     0   743k      0 --:--:-- --:--:-- --:--:--  803k
                  Apache2 package with Debian. However, check <a
                  href="http://bugs.debian.org/cgi-bin/pkgreport.cgi?ordering=normal;archive=0
                  rel="nofollow">existing bug reports</a> before reporting a new bug.
              </p>
              <p>
                  Please report bugs specific to modules (such as PHP and others)
                  to respective packages, not to the web server itself.
              </p>
            </div>




          </div>
        </div>
        <div class="validator">
        </div>
      </body>
    </html>

[xafonya_gmail_com@nat-gw-eu ~]$ ssh nat-node-gcp-eu
Last login: Wed Sep  2 10:07:25 2020 from nat-gw-eu.europe-west1-c.c.smooth-era-287810.internal
[xafonya_gmail_com@nat-node-gcp-eu ~]$ gcloud compute instances list
NAME             ZONE           MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP   EXTERNAL_IP     STATUS
nat-gw-eu        europe-west1-c  n1-standard-1               192.168.20.2  34.77.219.174   RUNNING
nat-node-eu      europe-west1-c  n1-standard-1               192.168.20.3                  RUNNING
nat-node-gcp-eu  europe-west1-c  n1-standard-1               192.168.20.5                  RUNNING
nat-node-w-eu    europe-west1-c  n1-standard-1               192.168.20.4                  RUNNING
faux-on-prem-svc us-central1-f  n1-standard-1               10.128.0.12   35.184.83.171   RUNNING
nat-gw-us        us-central1-f  n1-standard-1               192.168.10.2  34.121.45.5     RUNNING
nat-node-us      us-central1-f  n1-standard-1               192.168.10.3                  RUNNING
nat-node-w-us    us-central1-f  n1-standard-1               192.168.10.4                  RUNNING
[xafonya_gmail_com@nat-node-gcp-eu ~]$
```

TASK 4

The Objectives are to learn:

- Secure app in custom network

Lab Link: codelabs: custom_network

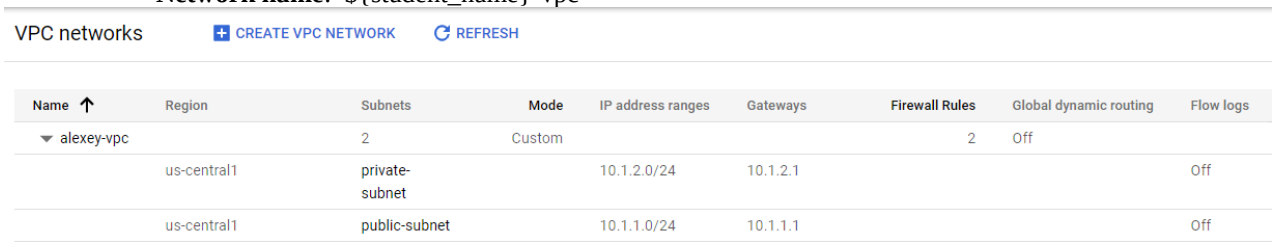


TASK 5

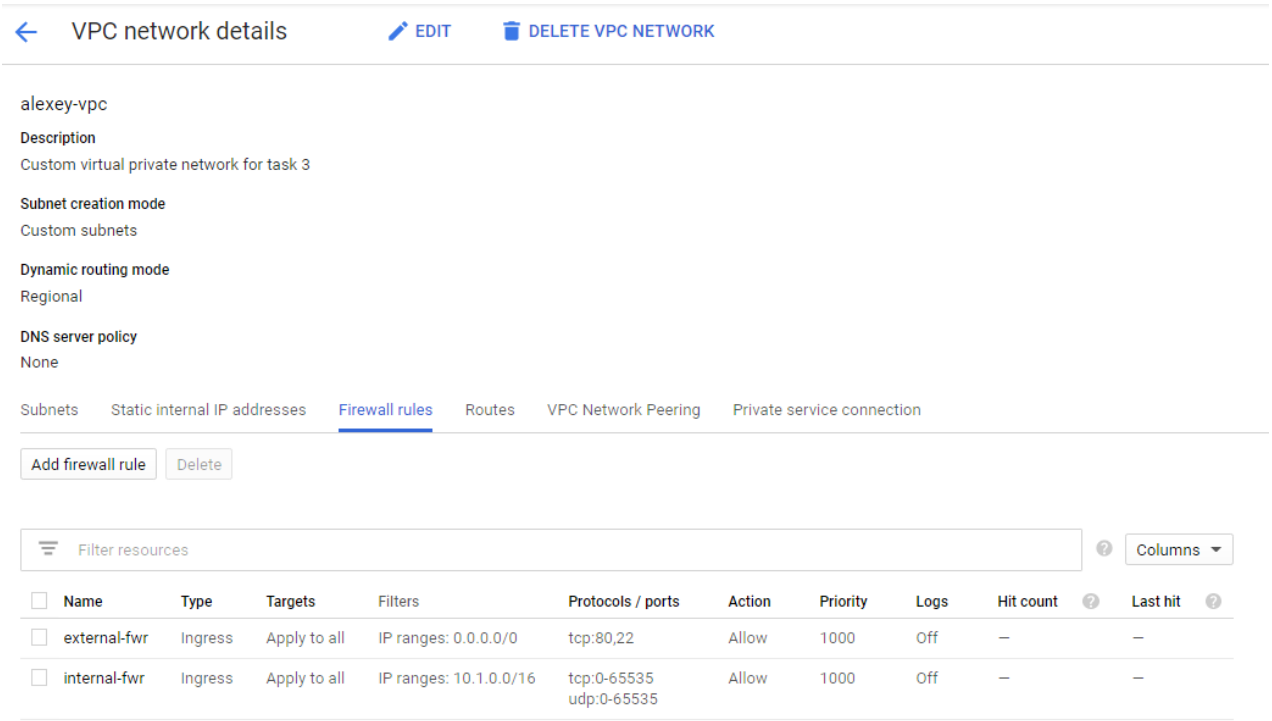
Create network configuration via terraform.
Resources should be used:

1) **google_compute_network** (to create network)
   https://www.terraform.io/docs/providers/google/r/compute_network.html

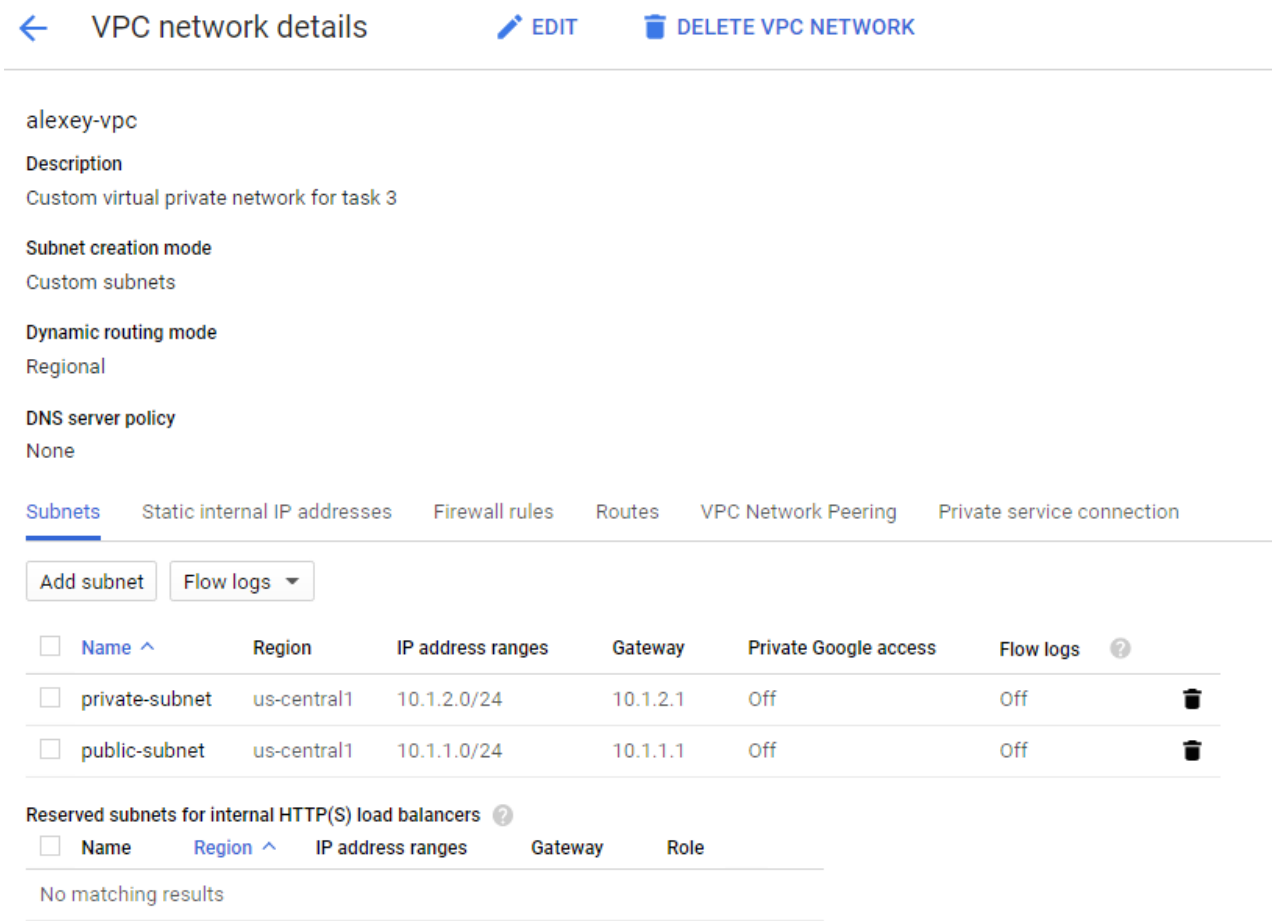
   Network name: ${student_name}-vpc



2) **google_compute_firewall**
   (to create rules for external (allow 80,22) /internal access (allow 0-65535))
   https://www.terraform.io/docs/providers/google/r/compute_firewall.html



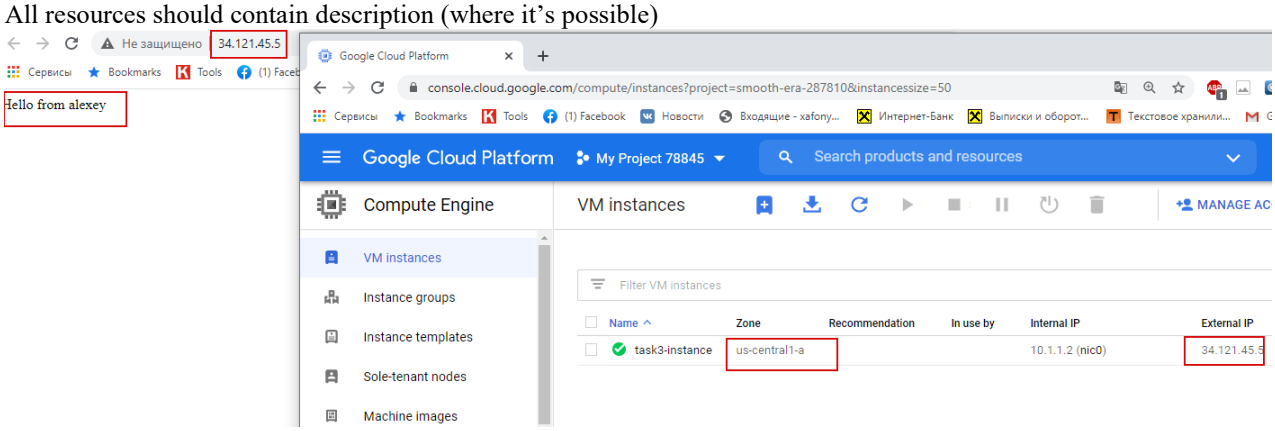
3) **google_compute_subnetwork**
https://www.terraform.io/docs/providers/google/r/compute_subnetwork.html
ranges:
- Public range: 10.”${student_IDnum}”.1.0/24
- Private range: 10.”${student_IDnum}”.2.0/24



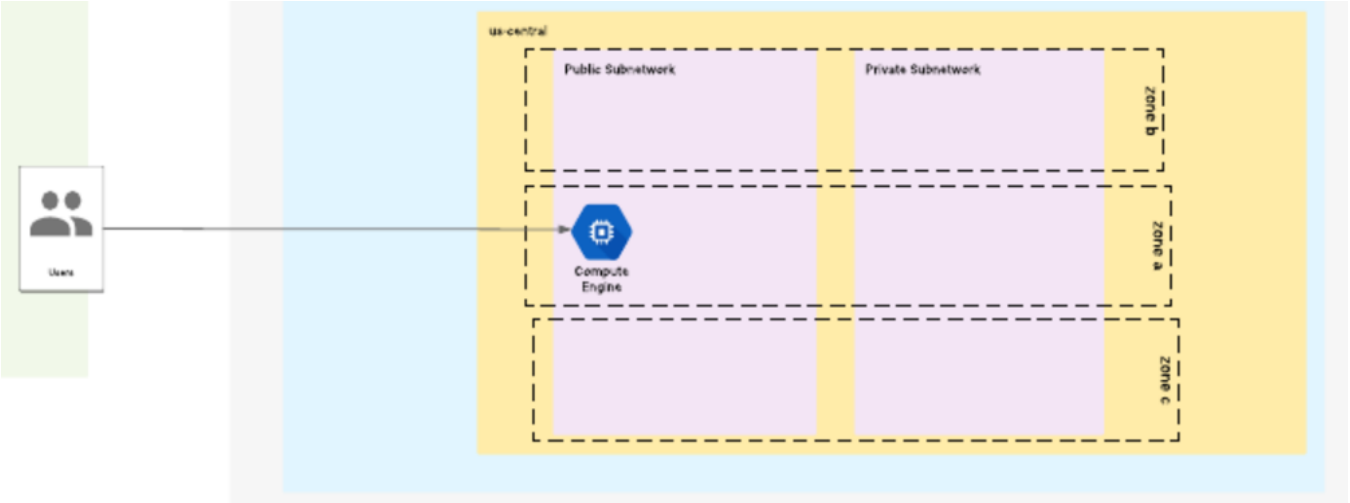
4) **google_compute_instance**
https://www.terraform.io/docs/providers/google/r/compute_instance.html

1. nginx with default page “Hello from ${student_name}”

All resources should contain description (where it’s possible)



Network topology.



All **reports**/code please place into repository:
https://github.com/MNT-Lab/google-cloud-module into appropriate branches: *first char of name + surname*.

For example:
Student: Siarhei Ivanou
Branch Name: **sivanou**

Format depends on case: README.md/scripts/terraform files

Email pattern: [MNT-CD-10.3]-FirstName-LastName

Email should contain the link to personalized branch.