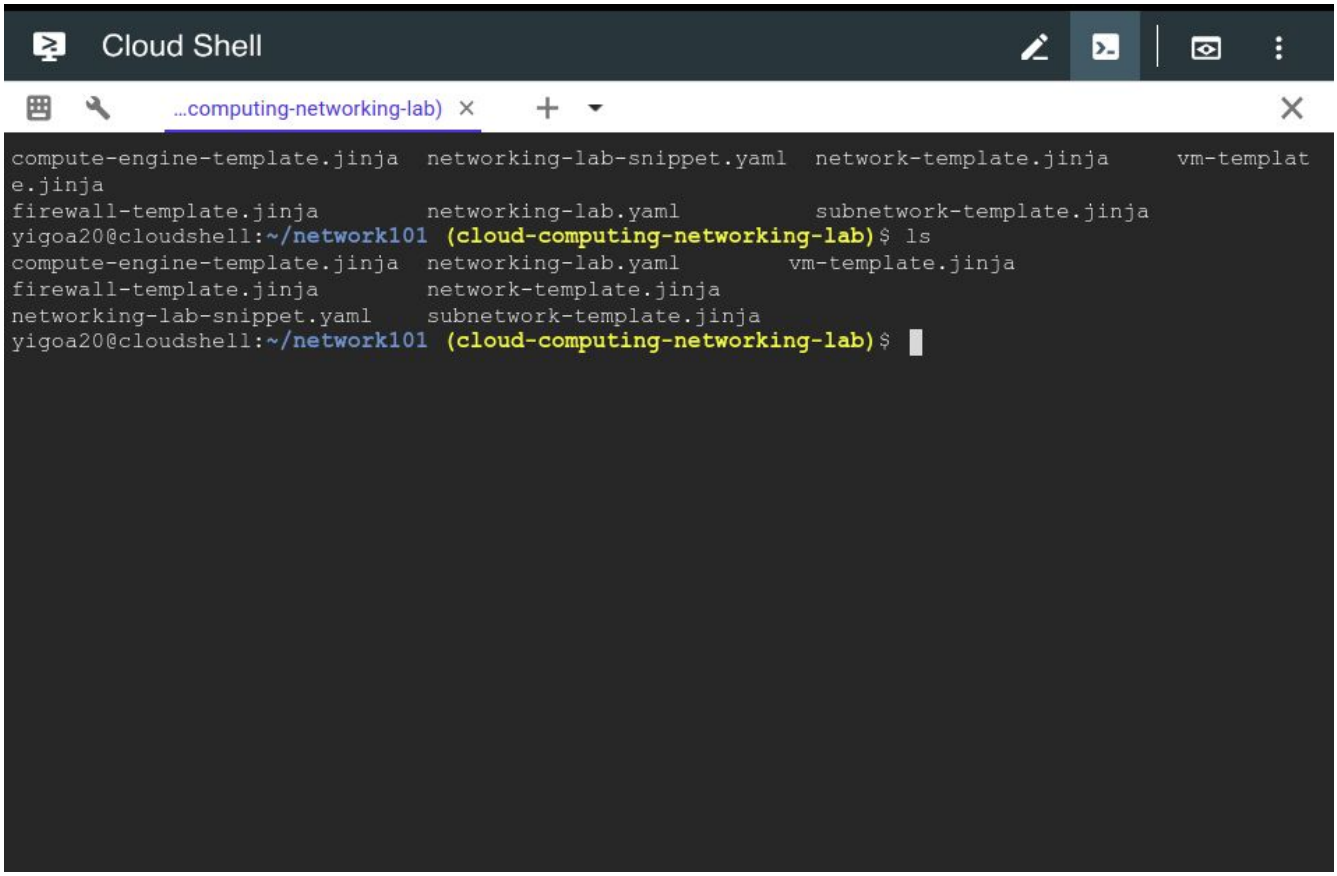
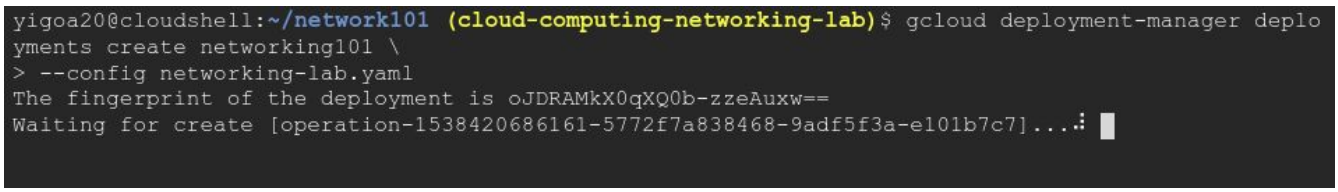
Abdoul-Nourou Yigo
Dr. Jiazhen Zhou
Cloud Computing
October 1, 2018

Assignment 2: Google cloud Report

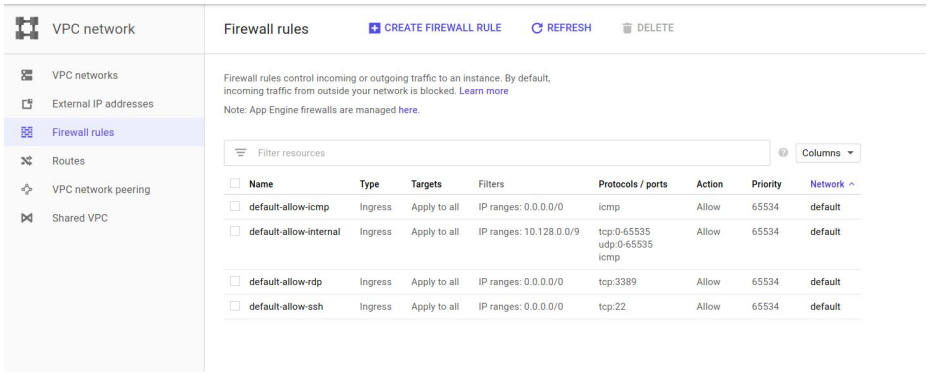
These are the procedures followed in the Google Cloud platform:

The first process is to set up the instances on the platform, so the following picture illustrates this procedure:



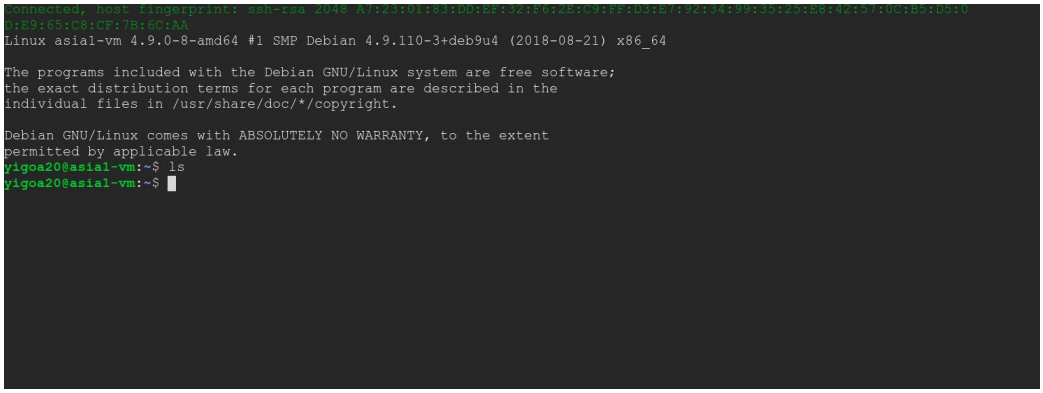


The following is the firewall rules from the console and the terminal:

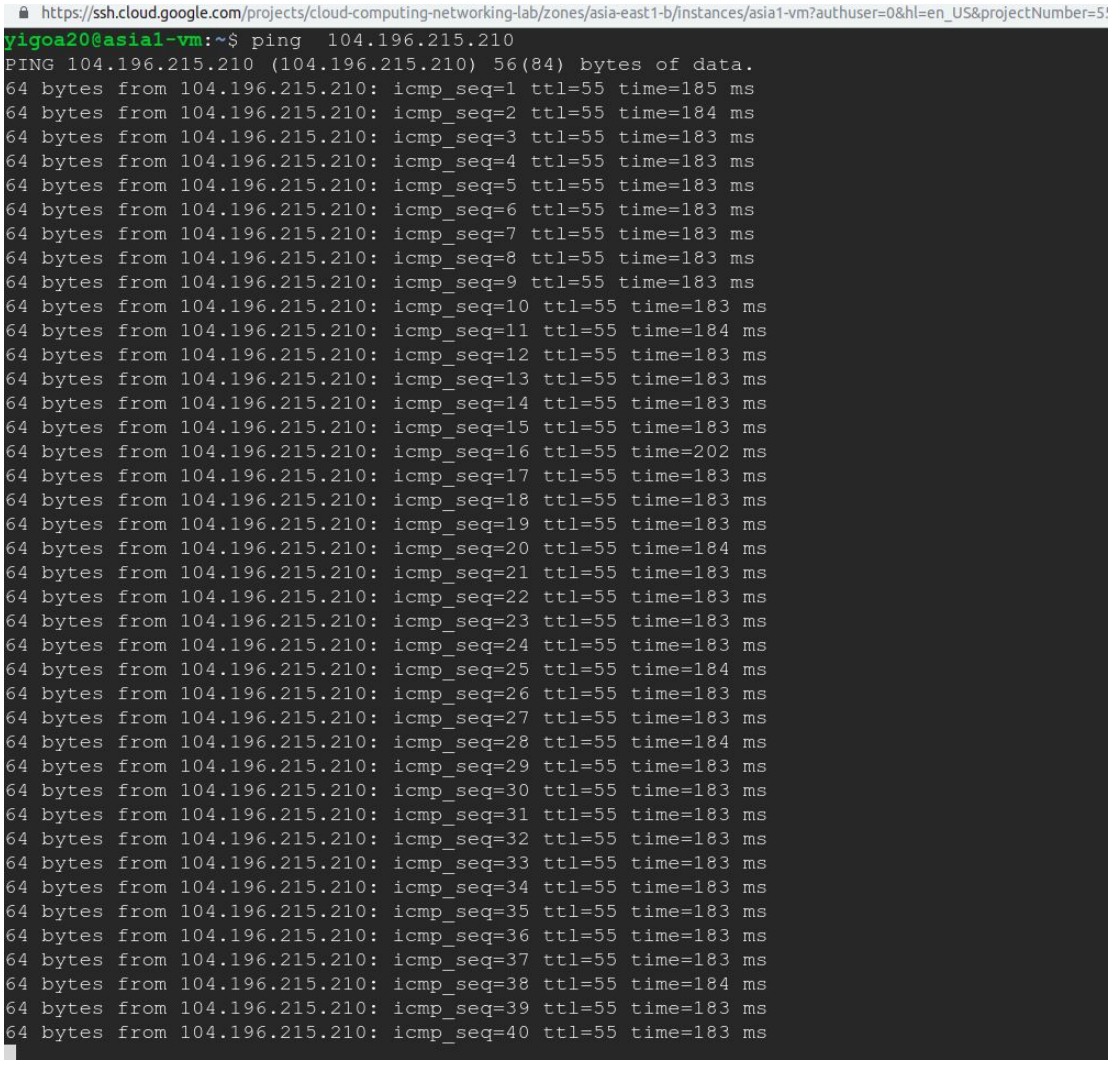



```
{#
Copyright 2016 Google Inc. All rights reserved.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
#}

resources:
- name: {{ env["name"] }}-allow-internal
  type: compute.v1.firewall
  properties:
    network: $(ref.{{ properties["network"] }}.selfLink)
    sourceRanges: ["10.0.0.0/8"]
    allowed:
    - IPProtocol: TCP
      ports: ["0-65535"]
    - IPProtocol: UDP
      ports: ["0-65535"]
    - IPProtocol: ICMP
- name: {{ env["name"] }}-allow-ssh
  type: compute.v1.firewall
  properties:
    network: $(ref.{{ properties["network"] }}.selfLink)
    sourceRanges: ["0.0.0.0/0"]
    allowed:
    - IPProtocol: TCP
      ports: ["22"]
- name: {{ env["name"] }}-allow-icmp
  type: compute.v1.firewall
  properties:
    network: $(ref.{{ properties["network"] }}.selfLink)
```

After all the configuration setup I was able to login in the instance using the SSH command illustrates by the following picture:



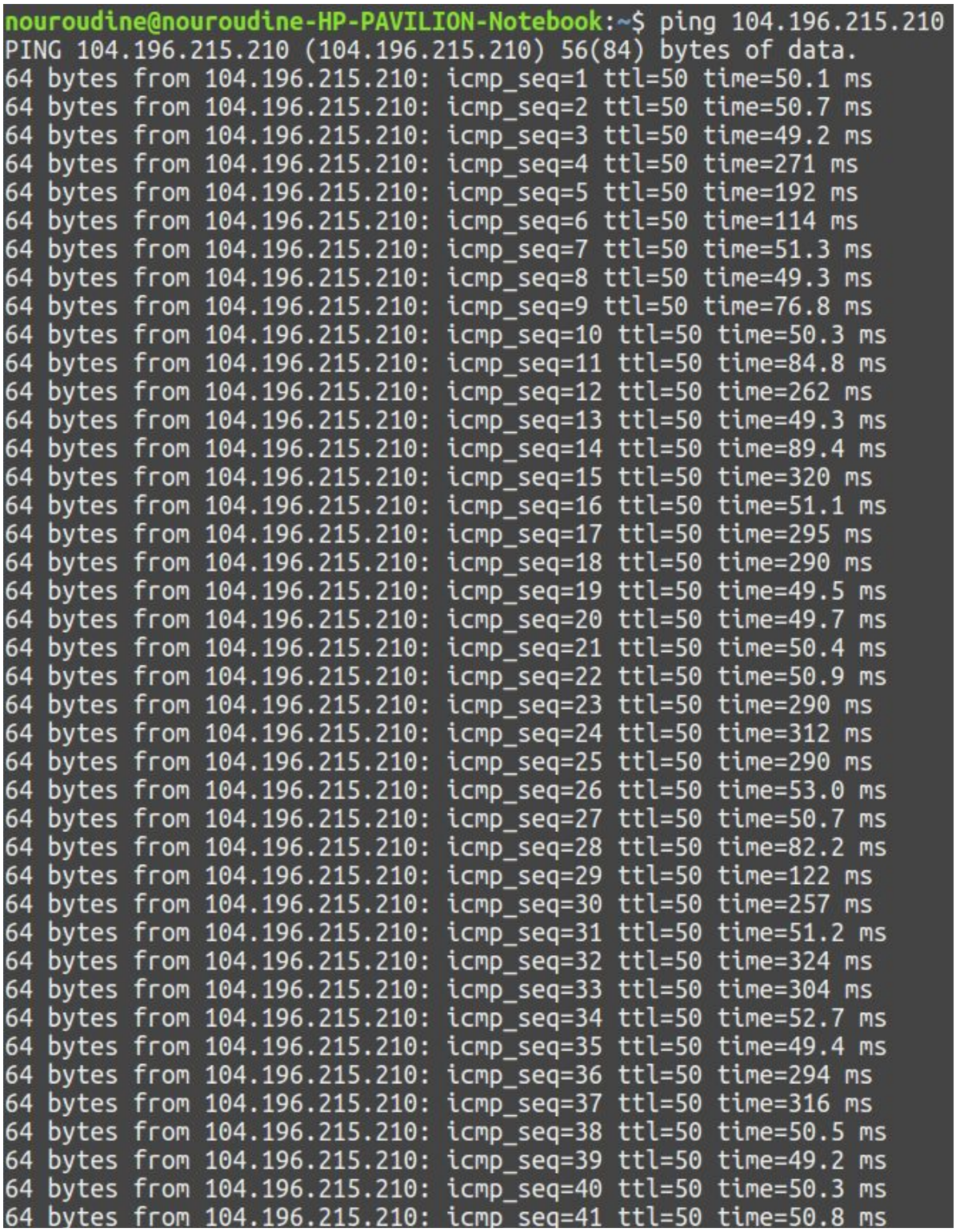
The next step was to use a ping command between two different instances to observe the latency:
This following illustration is showing a time between an instance from the US and Asia.



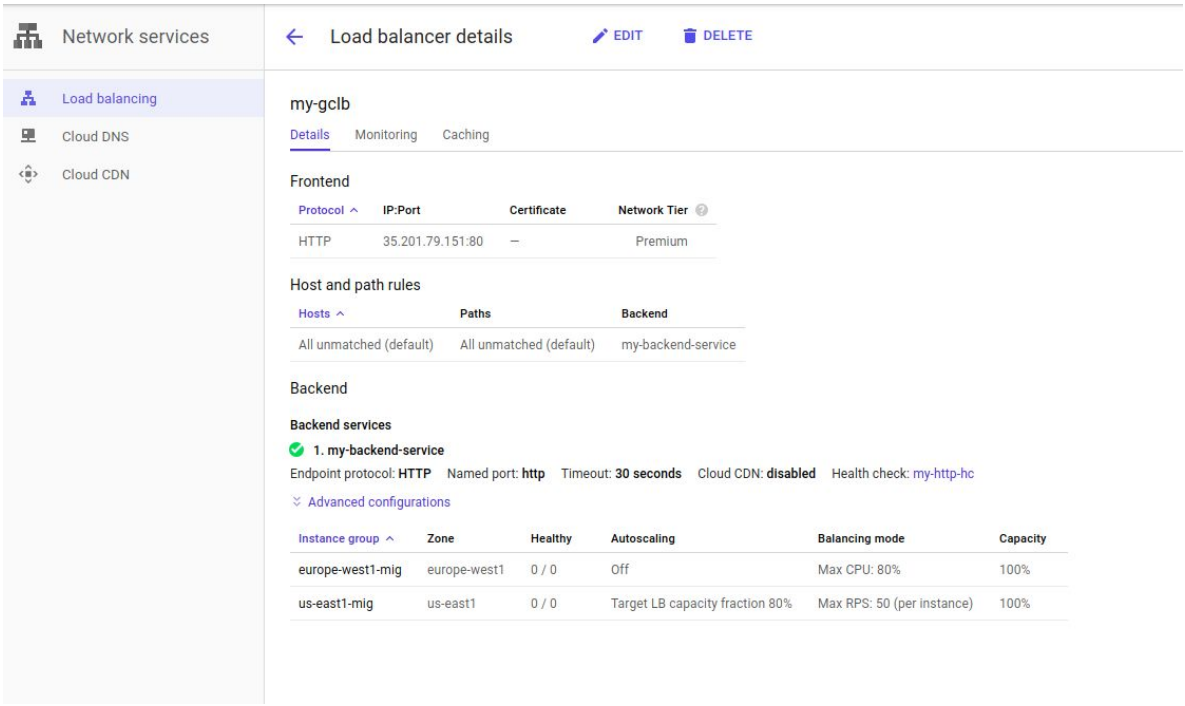
I have also performed a test within the US instances, so the following picture is a test on those instances (us-east1-b to us-west1-b):

```
yigoa20@e1-vm:~$ ping 35.233.180.195
PING 35.233.180.195 (35.233.180.195) 56(84) bytes of data.
64 bytes from 35.233.180.195: icmp_seq=1 ttl=57 time=68.4 ms
64 bytes from 35.233.180.195: icmp_seq=2 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=3 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=4 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=5 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=6 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=7 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=8 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=9 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=10 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=11 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=12 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=13 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=14 ttl=57 time=67.1 ms
64 bytes from 35.233.180.195: icmp_seq=15 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=16 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=17 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=18 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=19 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=20 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=21 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=22 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=23 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=24 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=25 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=26 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=27 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=28 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=29 ttl=57 time=66.8 ms
64 bytes from 35.233.180.195: icmp_seq=30 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=31 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=32 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=33 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=34 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=35 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=36 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=37 ttl=57 time=66.6 ms
64 bytes from 35.233.180.195: icmp_seq=38 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=39 ttl=57 time=66.7 ms
64 bytes from 35.233.180.195: icmp_seq=40 ttl=57 time=66.7 ms
```

The last test was from my machine to another instance:



The next step is to create a load balancer to manage traffic from in the backend and frontend using the HTTP protocol. Therefore, the following illustration is showing then configuration interface.

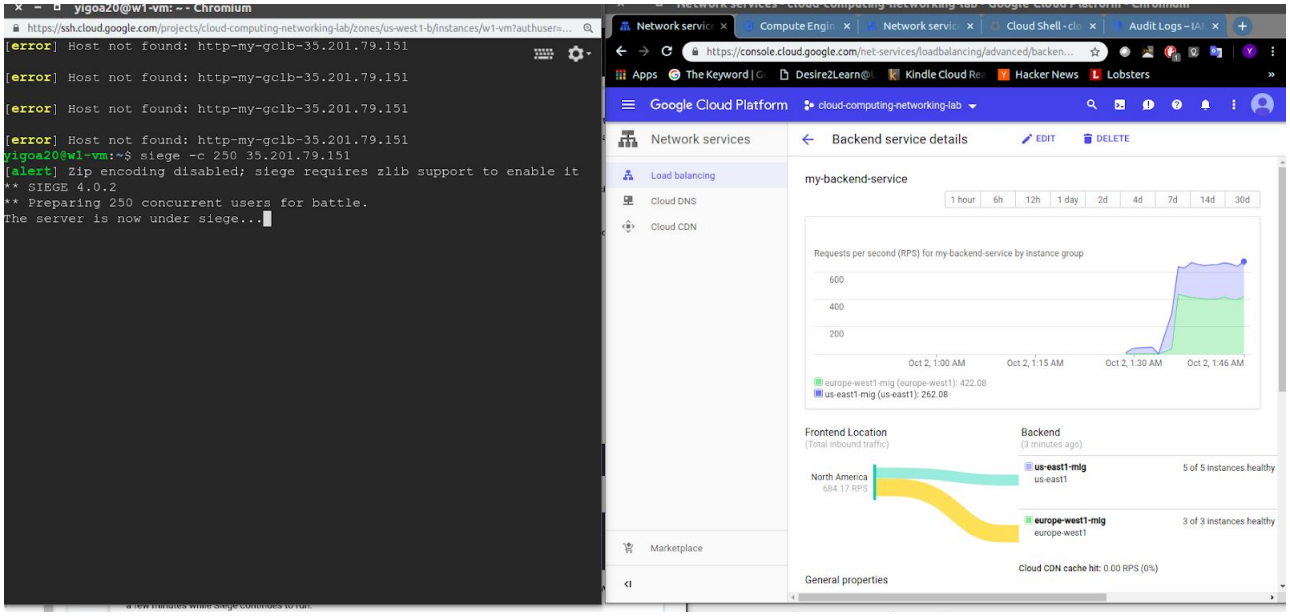


I have tried to ping the load balancer and this is the result:

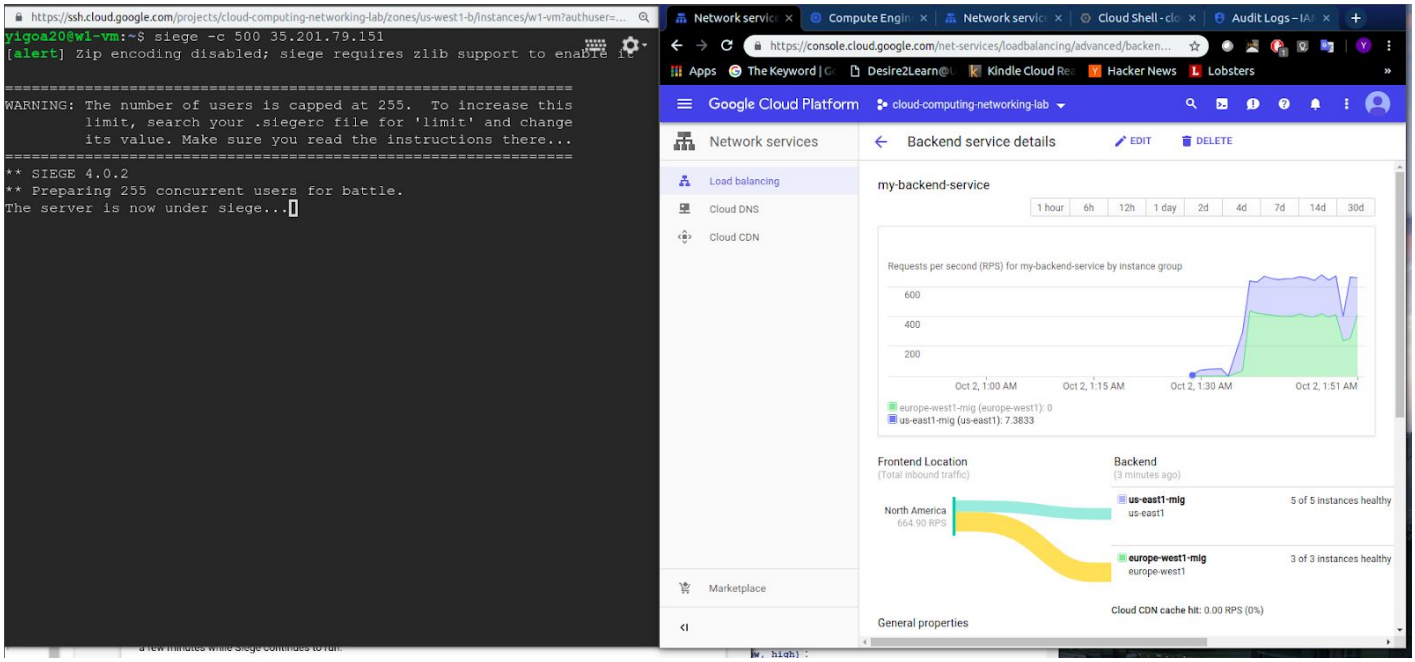
```
nouroudine@nouroudine-HP-PAVILION-Notebook:~$ ping 35.201.79.1
PING 35.201.79.151 (35.201.79.151) 56(84) bytes of data.
64 bytes from 35.201.79.151: icmp_seq=1 ttl=50 time=116 ms
64 bytes from 35.201.79.151: icmp_seq=2 ttl=50 time=16.0 ms
64 bytes from 35.201.79.151: icmp_seq=3 ttl=50 time=46.3 ms
64 bytes from 35.201.79.151: icmp_seq=4 ttl=50 time=15.6 ms
64 bytes from 35.201.79.151: icmp_seq=5 ttl=50 time=16.0 ms
64 bytes from 35.201.79.151: icmp_seq=6 ttl=50 time=15.7 ms
64 bytes from 35.201.79.151: icmp_seq=7 ttl=50 time=15.9 ms
64 bytes from 35.201.79.151: icmp_seq=8 ttl=50 time=50.7 ms
64 bytes from 35.201.79.151: icmp_seq=9 ttl=50 time=21.0 ms
64 bytes from 35.201.79.151: icmp_seq=10 ttl=50 time=15.9 ms
64 bytes from 35.201.79.151: icmp_seq=11 ttl=50 time=16.6 ms
64 bytes from 35.201.79.151: icmp_seq=12 ttl=50 time=15.7 ms
64 bytes from 35.201.79.151: icmp_seq=13 ttl=50 time=16.6 ms
64 bytes from 35.201.79.151: icmp_seq=14 ttl=50 time=15.4 ms
64 bytes from 35.201.79.151: icmp_seq=15 ttl=50 time=17.6 ms
64 bytes from 35.201.79.151: icmp_seq=16 ttl=50 time=15.3 ms
64 bytes from 35.201.79.151: icmp_seq=17 ttl=50 time=18.8 ms
64 bytes from 35.201.79.151: icmp_seq=18 ttl=50 time=21.8 ms
64 bytes from 35.201.79.151: icmp_seq=19 ttl=50 time=15.6 ms
64 bytes from 35.201.79.151: icmp_seq=20 ttl=50 time=16.4 ms
64 bytes from 35.201.79.151: icmp_seq=21 ttl=50 time=16.1 ms
64 bytes from 35.201.79.151: icmp_seq=22 ttl=50 time=15.4 ms
64 bytes from 35.201.79.151: icmp_seq=23 ttl=50 time=17.0 ms
64 bytes from 35.201.79.151: icmp_seq=24 ttl=50 time=18.7 ms
64 bytes from 35.201.79.151: icmp_seq=25 ttl=50 time=20.7 ms
64 bytes from 35.201.79.151: icmp_seq=26 ttl=50 time=16.5 ms
64 bytes from 35.201.79.151: icmp_seq=27 ttl=50 time=16.0 ms
64 bytes from 35.201.79.151: icmp_seq=28 ttl=50 time=16.0 ms
64 bytes from 35.201.79.151: icmp_seq=29 ttl=50 time=15.9 ms
64 bytes from 35.201.79.151: icmp_seq=30 ttl=50 time=15.8 ms
64 bytes from 35.201.79.151: icmp_seq=31 ttl=50 time=15.3 ms
64 bytes from 35.201.79.151: icmp_seq=32 ttl=50 time=17.9 ms
64 bytes from 35.201.79.151: icmp_seq=33 ttl=50 time=15.2 ms
```

The following step, I am using Siege to test the load balancer performance:

Stimulation with 250 concurrent users:



I have tried to push the stimulation to 500, but the maximum number of users is 255 according to the warning message:


https://ssh.cloud.google.com/projects/cloud-computing-networking-lab/zones/us-west1-b/instances/w1-vm?authuser=...
yigoa20@w1-vm:~$ siege -c 500 35.201.79.151
[alert] Zip encoding disabled; siege requires zlib support to enable it
WARNING: The number of users is capped at 255. To increase this
limit, search your .siegerc file for 'limit' and change
its value. Make sure you read the instructions there...
** SIEGE 4.0.2
** Preparing 255 concurrent users for battle.
The server is now under siege...
Network servic
Compute Engin
Network servic
Cloud Shell - clo
Audit Logs – IAM
https://console.cloud.google.com/net-services/loadbalancing/advanced/backen...
Apps
The Keyword | G
Desire2Learn@U
Kindle Cloud Rea
Hacker News
Lobsters
Google Cloud Platform
cloud-computing-networking-lab
Network services
Load balancing
Cloud DNS
Cloud CDN
Marketplace
Backend service details
EDIT
DELETE
my-backend-service
1 hour
6h
12h
1 day
2d
4d
7d
14d
30d
Requests per second (RPS) for my-backend-service by instance group
600
400
200
Oct 2, 1:00 AM
Oct 2, 1:15 AM
Oct 2, 1:30 AM
Oct 2, 1:51 AM
europe-west1-mig (europe-west1): 0
us-east1-mig (us-east1): 7.3833
Frontend Location
(Total inbound traffic)
North America
664.90 RPS
Backend
(3 minutes ago)
us-east1-mig
us-east1
5 of 5 instances healthy
europe-west1-mig
europe-west1
3 of 3 instances healthy
Cloud CDN cache hit: 0.00 RPS (0%)
General properties