**Project 3**

**These notes use the Google Cloud console terminal from within your project, to do this from a Windows terminal, you’ll find that majority of the steps can be completed by substituting the backslash \ with caret ^. Creating bash scripts that will automatically be added to the instances upon creation through the Windows terminal can be a little tricky, and you may find it easier to just do it manually from the inside of the instance after creation. The code below will include those scripts and will work for most Linux terminals.**

1. **Set default region and zones, in this case we are doing this in region us-central1 and zone us-central1-a**

gcloud config set compute/region us-central1

**Then**

gcloud config set compute/zone us-central1-a

1. **Create multiple web server instances www1, www2, www3 within your default zone**

gcloud compute instances create www1 \

--zone= us-central1-a \

--tags=network-lb-tag \

--machine-type=e2-small \

--image-family=debian-11 \

--image-project=debian-cloud \

--metadata=startup-script='#!/bin/bash

apt-get update

apt-get install apache2 -y

service apache2 restart

echo "

<h3>Web Server: www1</h3>" | tee /var/www/html/index.html'

**Then**

gcloud compute instances create www1 \

--zone= us-central1-a \

--tags=network-lb-tag \

--machine-type=e2-small \

--image-family=debian-11 \

--image-project=debian-cloud \

--metadata=startup-script='#!/bin/bash

apt-get update

apt-get install apache2 -y

service apache2 restart

echo "

<h3>Web Server: www2</h3>" | tee /var/www/html/index.html'

**Then**

gcloud compute instances create www3 \

--zone= us-central1-a \

--tags=network-lb-tag \

--machine-type=e2-small \

--image-family=debian-11 \

--image-project=debian-cloud \

--metadata=startup-script='#!/bin/bash

apt-get update

apt-get install apache2 -y

service apache2 restart

echo "

<h3>Web Server: www3</h3>" | tee /var/www/html/index.html'

**Create a firewall rule to allow external traffic**

gcloud compute firewall-rules create www-firewall-network-lb \

--target-tags network-lb-tag --allow tcp:80

1. **Configure the load balancing server**

gcloud compute addresses create network-lb-ip-1 \

--region us-central1

**Add HTTP help check, add a target pool within the default region, then add instances to pool**

gcloud compute http-health-checks create basic-check

**Then**

gcloud compute target-pools create www-pool \

--region Region --http-health-check basic-check

**Then**

gcloud compute target-pools add-instances www-pool \

--instances www1,www2,www3

**Add a forwarding rule to the pool**

gcloud compute forwarding-rules create www-rule \

--region us-central1 \

--ports 80 \

--address network-lb-ip-1 \

--target-pool www-pool

1. **Create HTTP load balancer then a template based managed instance group**

gcloud compute instance-templates create lb-backend-template \

--region=us-central1 \

--network=default \

--subnet=default \

--tags=allow-health-check \

--machine-type=e2-medium \

--image-family=debian-11 \

--image-project=debian-cloud \

--metadata=startup-script='#!/bin/bash

apt-get update

apt-get install apache2 -y

a2ensite default-ssl

a2enmod ssl

vm\_hostname="$(curl -H "Metadata-Flavor:Google" \

http://169.254.169.254/computeMetadata/v1/instance/name)"

echo "Page served from: $vm\_hostname" | \

tee /var/www/html/index.html

systemctl restart apache2'

**Then**

gcloud compute instance-groups managed create lb-backend-group \

--template=lb-backend-template --size=2 --zone=us-central1-a

**Then**

gcloud compute firewall-rules create fw-allow-health-check \

--network=default \

--action=allow \

--direction=ingress \

--source-ranges=130.211.0.0/22,35.191.0.0/16 \

--target-tags=allow-health-check \

--rules=tcp:80

**Then , static ip for load balancer**

gcloud compute addresses create lb-ipv4-1 \

--ip-version=IPV4 \

--global

**Then, create health check**

gcloud compute health-checks create http http-basic-check \

--port 80

1. **Create backend service and add instance group to service**

gcloud compute backend-services create web-backend-service \

--protocol=HTTP \

--port-name=http \

--health-checks=http-basic-check \

--global

**Then**

gcloud compute backend-services add-backend web-backend-service \

--instance-group=lb-backend-group \

--instance-group-zone=us-central1-a \

--global

**Then, create a URL map to route incoming traffic to a proxy**

gcloud compute url-maps create web-map-http \

--default-service web-backend-service

**Then**

gcloud compute target-http-proxies create http-lb-proxy \

--url-map web-map-http

**Then**

gcloud compute forwarding-rules create http-content-rule \

--address=lb-ipv4-1\

--global \

--target-http-proxy=http-lb-proxy \

**Test load balancer(web-map-http) by visiting it’s ip. Make sure to turn off https and use http.**