Andy Yang
DS561
HW8

I first modified my webserver to return the zone it is in, just added this function and returned this instead of the HTML.

```
def get_instance_zone():
    #https://cloud.google.com/compute/docs/metadata/querying-metadata#linux
    metadata_server =
"http://metadata.google.internal/computeMetadata/v1/instance/zone"
    metadata_flavor = {'Metadata-Flavor': 'Google'}

    response = requests.get(metadata_server, headers=metadata_flavor)
    if response.status_code == 200:
        # The response contains the full zone path we just want the last part.
        zone_path = response.text
        zone_name = zone_path.split('/')[-1]
        return zone_name
    else:
        return "Unable to retrieve zone information"
```

Then, after putting two of these in a VM, and making a request to it manually, this is what it looks like:

```
andyyang@crc-dot1x-nat-10-239-69-14 hw8 % python3 http-client.py -d 34.41.171.11
1  -p 8080 -n 1 -i 10000 -v -r 0

/Users/andyyang/Desktop/DS561/hw8/http-client.py:63: DeprecationWarning: ssl.PRO
TOCOL_TLS is deprecated
  ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS)
Requesting 1479 from 34.41.171.111:8080
this is the country: Somalia
200 OK
us-central1-a
andyyang@crc-dot1x-nat-10-239-69-14 hw8 %
```

```
andyyang@crc-dot1x-nat-10-239-69-14 hw8 % python3 http-client.py -d 34.31.220.75
  -p 8080 -n 1 -i 10000 -v -r 0

/Users/andyyang/Desktop/DS561/hw8/http-client.py:63: DeprecationWarning: ssl.PRO
TOCOL_TLS is deprecated
  ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS)
Requesting 8981 from 34.31.220.75:8080
this is the country: Eswatini
200 OK
us-central1-b
andyyang@crc-dot1x-nat-10-239-69-14 hw8 %
```

Now, to put these behind a load balancer:

gcloud compute addresses create network-lb --region us-central1

- Created [https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-east1/addresses/network-lb].

gcloud compute http-health-checks create basic-check --port 8080 --request-path /file/1.html

- Created [https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/global/httpHealthChecks/basic-check].

gcloud compute target-pools create www-pool --region us-central1 --http-health-check basic-check

- Created [https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-central1/targetPools/www-pool].
- NAME REGION SESSION_AFFINITY BACKUP HEALTH_CHECKS
- www-pool us-central1 NONE basic-check

gcloud compute target-pools add-instances www-pool --instances hw8-instance1 --zone us-central1-a

- Updated [https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-central1/targetPools/www-pool].

gcloud compute target-pools add-instances www-pool --instances hw8-instance2 --zone us-central1-b

```
gcloud compute forwarding-rules create www-rule \
   --region  us-central1 \
   --ports 8080 \
   --address network-lb\
   --target-pool www-pool
```

```
gcloud compute forwarding-rules describe www-rule --region us-central1
```

- IPAddress: 35.193.124.241
- IPProtocol: TCP
- creationTimestamp: '2023-11-19T13:04:04.968-08:00'
- description: ''
- fingerprint: y7ltkC1bP-o=
- id: '4817518304857409195'
- kind: compute#forwardingRule
- labelFingerprint: 42WmSpB8rSM=
- loadBalancingScheme: EXTERNAL
- name: www-rule
- networkTier: PREMIUM
- portRange: 8080-8080
- region: https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-central1
- selfLink: https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-central1/forwardingRules/www-rule
- target: https://www.googleapis.com/compute/v1/projects/cloudcomputing-398719/regions/us-central1/targetPools/www-pool

Now, successfully configured my load balancer, this is the IP

```
andyyang@crc-dot1x-nat-10-239-69-14 hw8 % gcloud compute forwarding-rules list

NAME      REGION       IP_ADDRESS      IP_PROTOCOL  TARGET
www-rule  us-central1  35.193.124.241  TCP          us-central1/targetPools/www-
pool
```

Running the http-client against it:

```
andyyang@crc-dot1x-nat-10-239-69-14 hw8 % python3 http-client.py -d 35.193.124.241  -p 8080 -n 5 -i 10000 -v -r 0
/Users/andyyang/Desktop/DS561/hw8/http-client.py:63: DeprecationWarning: ssl.PROTOCOL_TLS is deprecated
  ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS)
Requesting 7323 from 35.193.124.241:8080
this is the country: Latvia
200 OK
us-central1-a
Requesting 2061 from 35.193.124.241:8080
this is the country: Monaco
200 OK
us-central1-b
Requesting 5260 from 35.193.124.241:8080
this is the country: Switzerland
200 OK
us-central1-b
Requesting 4278 from 35.193.124.241:8080
this is the country: Senegal
200 OK
us-central1-b
Requesting 5262 from 35.193.124.241:8080
this is the country: Saudi Arabia
200 OK
us-central1-a
andyyang@crc-dot1x-nat-10-239-69-14 hw8 %
```

- You see a mix of us-central1-a and us-central1-b responses, which we should

REQUIRED QUESTIONS:

Time it takes to notice a dead VM: 40 seconds (from time I clicked stop instance to time I started to notice all us-central1-b's)

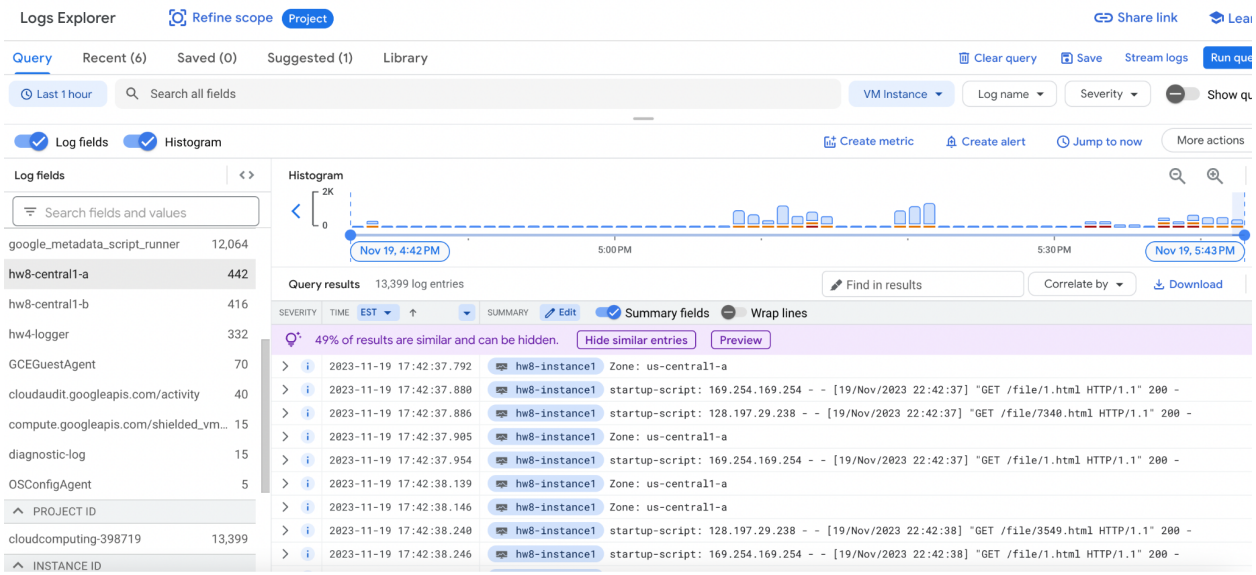
- 10-15 seconds after VM is finished shutting down

Time it takes to notice its back: 42 seconds (again, from when I clicked start, to when I first saw a us-central1-a)

- 10-15 seconds after VM is finished spinning up

Ratio of zone a to zone b

- Each webserver logs to different client



- On the left hand side, you see 442 to zone a and 416 to zone b , roughly = 52:48 ratio