

HW8 - David Euijoon Kim - Load Balancer

<https://github.com/dk-davidekim/Google-Cloud-Computing.git>

1. Edit Code (app_one.py)

```
BU > DS561 > ds561-davidekim-U66545284 > hw8 > app_one.py
1  from flask import Flask, request, Response
2  from google.cloud import storage, logging, pubsub_v1
3  import requests
4
5  app = Flask(__name__)
6
7  BANNED_COUNTRIES = ["North Korea", "Iran", "Cuba", "Myanmar",
8                      "Iraq", "Libya", "Sudan", "Zimbabwe", "Syria"]
9
10 @app.route('/', defaults={'path': ''}, methods=['GET', 'POST', 'PUT', 'DELETE', 'HEAD', 'CONNECT', 'OPTIONS', 'TRACE', 'PATCH'])
11 @app.route('/<path:filename>', methods=['GET', 'POST', 'PUT', 'DELETE', 'HEAD', 'CONNECT', 'OPTIONS', 'TRACE', 'PATCH'])
12 def app_one(filename):
13     logging_client = logging.Client(project='ds-561')
14     logger = logging_client.logger('hw8')
15     pub = pubsub_v1.PublisherClient()
16     path = pub.topic_path('ds-561', 'hw3')
17
18     if request.method == 'GET':
19         country = request.headers.get("X-country", "")
20
21         if country in BANNED_COUNTRIES:
22             try:
23                 data = str({'400 Forbidden from country': country})
24                 future = pub.publish(path, data.encode("utf-8"))
25                 message_id = future.result()
26                 logger.log_text(f"Message published with ID: {message_id}")
27             except Exception as e:
28                 logger.log_text(f"PubSub Notification Failed {str(e)}")
29                 logger.log_text(f"Error Code 400: Forbidden: {str(country)}")
30                 return "Permission Denied", 400
31
32         else:
33             try:
34                 filename = filename.replace('bu-ds561-dk98-bucket/', '')
35                 storage_client = storage.Client()
36                 bucket = storage_client.bucket('bu-ds561-dk98-bucket')
37                 blob = bucket.blob(filename)
38                 file_content = blob.download_as_text()
39                 logger.log_text(f"200: {filename}")
40
41                 ## HW8 EDIT
42                 zone = get_zone()
43                 response = Response(file_content)
44                 response.headers['X-VM-Zone'] = zone
45                 ##
46
47                 return response, 200
48
49             except Exception as e:
50                 logger.log_text(f"Error Code 404: {filename}: {str(e)}")
51                 return 'File not found', 404
52
53     else:
54         logger.log_text(f"Error Code 501: {request.method}")
55         return 'Not implemented', 501
56
57 ## HW8 EDIT
58 def get_zone():
59     logging_client = logging.Client(project='ds-561')
60     logger = logging_client.logger('hw8')
61     try:
62         headers = {"Metadata-Flavor": "Google"}
63         r = requests.get("http://metadata.google.internal/computeMetadata/v1/instance/zone", headers=headers)
64         if r.status_code == 200:
65             return r.text.split('/')[1]
66         return "Unknown Zone"
67     except Exception as e:
68         logger.log_text(f"Failed to get zone: {str(e)}")
69         return "Error with Zone"
70
71 ##
72 if __name__ == "__main__":
73     app.run(host='0.0.0.0', port=8080)
```

2. Code (app_two.py) - No Edits

```
BU > DS561 > ds561-davidekim-U66545284 > hw8 > app_two.py
1  from google.cloud import pubsub_v1
2
3  def call_back(message):
4      print(f"Received message: {message.data.decode('utf-8')}")
5      message.ack()
6
7  sub = pubsub_v1.SubscriberClient()
8  path = sub.subscription_path('ds-561', 'app_one_two')
9
10 future = sub.subscribe(path, callback=call_back)
11 print("Listening for messages on: {}".format(path))
12
13 with sub:
14     try:
15         future.result()
16     except KeyboardInterrupt:
17         future.cancel()
```

3. Edit Code (http-client.py)

```
3  import http.client
4  from urllib.parse import urljoin
5  import argparse
6  import random
7
8  import platform
9  import os
10 import ssl
11 from datetime import date
12
13
14 list_of_countries = ['Afghanistan', 'Albania', 'Algeria', 'Andorra',
15                     'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia',
16                     'Australia', 'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain',
17                     'Bangladesh', 'Barbados', 'Belarus', 'Belgium', 'Belize', 'Benin',
18                     'Bhutan', 'Bolivia', 'Bosnia and Herzegovina', 'Botswana',
19                     'Brazil', 'Brunei', 'Bulgaria', 'Burkina Faso', 'Burundi', 'Cabo Verde',
20                     'Cambodia', 'Cameroon', 'Canada', 'Central African Republic',
21                     'Chad', 'Chile', 'China', 'Colombia', 'Comoros',
22                     'Congo, Democratic Republic of the', 'Congo, Republic of the',
23                     'Costa Rica', 'Cote d'Ivoire', 'Croatia', 'Cuba', 'Cyprus',
24                     'Czechia', 'Denmark', 'Djibouti', 'Dominica', 'Dominican Republic',
25                     'Ecuador', 'Egypt', 'El Salvador', 'Equatorial Guinea',
26                     'Eritrea', 'Estonia', 'Eswatini', 'Ethiopia', 'Fiji', 'Finland',
27                     'France', 'Gabon', 'Gambia', 'Georgia', 'Germany', 'Ghana',
28                     'Greece', 'Grenada', 'Guatemala', 'Guinea', 'Guinea-Bissau',
29                     'Guyana', 'Haiti', 'Honduras', 'Hungary', 'Iceland', 'India',
30                     'Indonesia', 'Iran', 'Iraq', 'Ireland', 'Israel', 'Italy',
31                     'Jamaica', 'Japan', 'Jordan', 'Kazakhstan', 'Kenya', 'Kiribati',
32                     'Kosovo', 'Kuwait', 'Kyrgyzstan', 'Laos', 'Latvia', 'Lebanon',
33                     'Lesotho', 'Liberia', 'Libya', 'Liechtenstein', 'Lithuania',
34                     'Luxembourg', 'Madagascar', 'Malawi', 'Malaysia', 'Maldives',
35                     'Mali', 'Malta', 'Marshall Islands', 'Mauritania', 'Mauritius',
36                     'Mexico', 'Micronesia', 'Moldova', 'Monaco', 'Mongolia',
37                     'Montenegro', 'Morocco', 'Mozambique', 'Myanmar', 'Namibia',
38                     'Nauru', 'Nepal', 'Netherlands', 'New Zealand', 'Nicaragua',
39                     'Niger', 'Nigeria', 'North Korea', 'North Macedonia', 'Norway',
40                     'Oman', 'Pakistan', 'Palau', 'Palestine', 'Panama',
41                     'Papua New Guinea', 'Paraguay', 'Peru', 'Philippines', 'Poland', 'Portugal',
42                     'Qatar', 'Romania', 'Russia', 'Rwanda', 'Saint Kitts and Nevis',
43                     'Saint Lucia', 'Saint Vincent and the Grenadines', 'Samoa',
44                     'San Marino', 'Sao Tome and Principe', 'Saudi Arabia', 'Senegal',
45                     'Serbia', 'Seychelles', 'Sierra Leone', 'Singapore', 'Slovakia',
46                     'Slovenia', 'Solomon Islands', 'Somalia', 'South Africa', 'South Korea',
47                     'South Sudan', 'Spain', 'Sri Lanka', 'Sudan', 'Suriname',
48                     'Sweden', 'Switzerland', 'Syria', 'Taiwan', 'Tajikistan',
49                     'Tanzania', 'Thailand', 'Timor-Leste', 'Togo', 'Tonga',
50                     'Trinidad and Tobago', 'Tunisia', 'Turkey', 'Turkmenistan', 'Tuvalu',
51                     'Uganda', 'Ukraine', 'United Arab Emirates (UAE)', 'United Kingdom',
52                     'United States of America (USA)', 'Uruguay', 'Uzbekistan',
53                     'Vanuatu', 'Vatican City (Holy See)', 'Venezuela', 'Vietnam',
54                     'Yemen', 'Zambia', 'Zimbabwe']
55
56 list_of_genders = ['Male', 'Female']
57 list_of_ages = ['0-16', '17-25', '26-35', '36-45', '46-55', '56-65', '66-75', '76+']
58 list_of_incomes = ['0-10k', '10k-20k', '20k-40k', '40k-60k', '60k-100k', '100k-150k', '150k-250k', '250k+']
59
60 cidr_dict = {}
61 used_cidrs = []
62
63 def fix_certs():
64     ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS)
65     ssl_context.verify_mode = ssl.CERT_REQUIRED
66     ssl_context.check_hostname = True
67     ssl_context.load_default_certs()
68     if platform.system().lower() == 'darwin':
69         import certifi
70         ssl_context.load_verify_locations(
71             cafile=os.path.relpath(certifi.where()),
72             capath=None,
73             cadata=None)
74     return ssl_context
75
76 def build_country_cidrs():
77     for country in list_of_countries:
78         # Select a number of CIDRs to associate with that country
79         num_cidrs = random.randrange(1, 10)
80         cnt = 0
81         country_cidrs = []
82         while cnt < num_cidrs:
83             next_cidr = random.randrange(1, 16000000)
84             if not next_cidr in used_cidrs:
85                 used_cidrs.append(next_cidr)
86                 country_cidrs.append(next_cidr)
87                 cnt += 1
88         cidr_dict[country] = country_cidrs
89
90 def select_country():
91     idx = random.randrange(0, len(list_of_countries))
92     return list_of_countries[idx]
93
94 def select_cidr(country):
95     cidr_list = cidr_dict[country]
96     idx = random.randrange(0, len(cidr_list))
97     return cidr_list[idx]
98
99 def make_ip(cidr):
100     octet4 = random.randrange(0, 255)
101     octet3 = cidr % 255
102     cidr = int(cidr / 255)
103     octet2 = cidr % 255
104     octet1 = int(cidr / 255)
105     ip = str(octet1) + '.' + str(octet2) + '.' + str(octet3) + '.' + str(octet4)
106     return ip
107
108 def make_filename(bucketname, dirname, num_files):
109     idx = random.randrange(0, num_files)
110     name = bucketname
111     if dirname != '':
112         name += ('/' + dirname)
113     name += ('/' + str(idx) + '.html')
114     return name
```

```

115 def get_list_item(lst):
116     index = random.randrange(0, len(lst))
117     return lst[index]
118
119 def build_headers(country, ip):
120     headers = {}
121     headers.update({'X-country':country})
122     headers.update({'X-client-IP':ip})
123     headers.update({'X-gender':get_list_item(list_of_genders)})
124     headers.update({'X-age':get_list_item(list_of_ages)})
125     headers.update({'X-income':get_list_item(list_of_incomes)})
126     today = str(date.today())
127     time_of_day = random.randrange(0,24)
128     time_str = today + ' ' + format(time_of_day, '02d') + ':00:00'
129     headers.update({'X-time':time_str})
130     return headers
131
132 def make_request(domain, port, country, ip, filename, use_ssl, ssl_context, follow, verbose):
133     if verbose:
134         print("Requesting ", filename, " from ", domain, port)
135     conn = None
136     if use_ssl:
137         conn = http.client.HTTPSConnection(domain, port, context=ssl_context)
138     else:
139         conn = http.client.HTTPConnection(domain, port)
140
141     headers = build_headers(country, ip)
142     conn.request("GET", filename, headers=headers)
143     res = conn.getresponse()
144     data = res.read()
145     if verbose:
146         print(res.status, res.reason)
147         print(res.msg)
148         print(data)
149
150     ## HWB EDIT
151     zone = res.getheader('X-VM-Zone')
152     if zone:
153         print("X-VM-Zone:", zone)
154     else:
155         print("X-VM-Zone header not found")
156     ##
157
158     if follow:
159         location_header = res.getheader('location')
160         if location_header is not None:
161             filename = urljoin(filename, location_header)
162             make_request(domain, port, country, ip, filename, use_ssl, ssl_context, follow, verbose)
163     conn.close()

```

```

166 def main():
167     ssl_context = fix_certs()
168     parser = argparse.ArgumentParser()
169     parser.add_argument("-d", "--domain", help="Domain to make requests to", type=str, default="www.python.org")
170     parser.add_argument("-b", "--bucket", help="Cloud bucket containing your files. Use none if running local", type=str, default="by-cds533-kthanas1-bucket1")
171     parser.add_argument("-w", "--webdir", help="Directory containing your files. Use none if you did not make one", type=str, default="webdir")
172     parser.add_argument("-n", "--num-requests", help="Number of requests to make", type=int, default=100000)
173     parser.add_argument("-i", "--index", help="Maximum existing file index", type=int, default=100000)
174     parser.add_argument("-p", "--port", help="Server Port", type=int, default=80)
175     parser.add_argument("-f", "--follow", help="Follow Redirects", action="store_true")
176     parser.add_argument("-s", "--ssl", help="Use HTTPS", action="store_true")
177     parser.add_argument("-v", "--verbose", help="Print the responses from the server on stdout", action="store_true")
178     parser.add_argument("-r", "--random", help="Initial random seed", type=int, default=0)
179     args = parser.parse_args()
180     if args.random != 0:
181         random.seed(args.random)
182
183     build_country_cidrs()
184
185     if args.bucket == 'none':
186         args.bucket = ''
187     if args.webdir == 'none':
188         args.webdir = ''
189     # Make the requests
190     for i in range(0,args.num_requests):
191         country = select_country()
192         cidr = select_cidr(country)
193         ip = make_ip(cidr)
194         filename = make_filename(args.bucket, args.webdir, args.index)
195         # If using the default port but have enabled ssl change the default port to be that of SSL
196         if args.ssl and args.port==80:
197             args.port=443
198         make_request(args.domain, args.port, country, ip, filename, args.ssl, ssl_context, args.follow, args.verbose)
199
200 if __name__ == "__main__":
201     main()

```

4. VM Creation

```
gcloud compute instances create www1 \  
  --zone=us-east1-b \  
  --tags=web-server \  
  --machine-type=e2-medium \  
  --image-family=debian-11 \  
  --image-project=debian-cloud
```

```
gcloud compute instances create www2 \  
  --zone=us-east1-c \  
  --tags=web-server \  
  --machine-type=e2-medium \  
  --image-family=debian-11 \  
  --image-project=debian-cloud
```

```
gcloud compute instances create www3 \  
  --zone=us-east1-b \  
  --tags=web-server \  
  --machine-type=e2-medium \  
  --image-family=debian-11 \  
  --image-project=debian-cloud
```

Firewalls

- ☒ Allow HTTP traffic
- ☒ Allow HTTPS traffic
- ☒ Allow Load Balancer Health checks

5. Firewall 8080

```
gcloud compute firewall-rules create allow-web-8080 \  
  --allow=tcp:8080 \  
  --target-tags=web-server \  
  --description="Allow port 8080 access to web servers"
```

```
gcloud compute instances add-tags www1 --zone=us-east1-b --tags=web-server
gcloud compute instances add-tags www2 --zone=us-east1-b --tags=web-server
```

6. Install Dependencies in VMs

```
sudo apt-get install python3-distutils
sudo curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
sudo python3 get-pip.py
```

```
sudo pip install google-cloud-storage
sudo pip install google-cloud-pubsub
sudo pip install google-cloud-logging
sudo pip install Flask
sudo pip install requests
```

7. Send Files

```
gcloud compute scp
/Users/davidekim/Desktop/DataScience/BU/DS561/ds561-davidekim-U66545284/hw8/app_one.py
www1:~/
```

```
gcloud compute scp
/Users/davidekim/Desktop/DataScience/BU/DS561/ds561-davidekim-U66545284/hw8/app_one.py
www2:~/ --zone=us-east1-c
```

```
gcloud compute scp
/Users/davidekim/Desktop/DataScience/BU/DS561/ds561-davidekim-U66545284/hw8/app_two.py
www3:~/
```

8. Load Balancer

```
gcloud compute addresses create network-lb-ip-1 --region us-east1
gcloud compute http-health-checks create basic-check
gcloud compute target-pools create www-pool --region us-east1 --http-health-check basic-check
gcloud compute target-pools add-instances www-pool --instances www1 --instances-zone us-east1-b
gcloud compute target-pools add-instances www-pool --instances www2 --instances-zone us-east1-c
gcloud compute forwarding-rules create www-rule --region us-east1 --ports 8080 --address
network-lb-ip-1 --target-pool www-pool
gcloud compute forwarding-rules describe www-rule --region us-east1
```

9. Test

```
X-VM-Zone: us-east1-c
Requesting bu-ds561-dk98-bucket/hw2_output/6424.html from 35.190.170.219 8080
200 OK
Server: Werkzeug/3.0.1 Python/3.9.2
Date: Sat, 18 Nov 2023 04:02:40 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 33702
X-VM-Zone: us-east1-b
Connection: close
```

```
Press Ctrl+C to quit
128.197.29.228 - - [18/Nov/2023 04:01:48] "GET bu-ds561-dk98-bucket/hw2_output/1803.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:01:55] "GET bu-ds561-dk98-bucket/hw2_output/2893.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:19] "GET bu-ds561-dk98-bucket/hw2_output/2809.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:19] "GET bu-ds561-dk98-bucket/hw2_output/5121.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:20] "GET bu-ds561-dk98-bucket/hw2_output/3174.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:20] "GET bu-ds561-dk98-bucket/hw2_output/3466.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:21] "GET bu-ds561-dk98-bucket/hw2_output/1689.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:22] "GET bu-ds561-dk98-bucket/hw2_output/1023.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:23] "GET bu-ds561-dk98-bucket/hw2_output/2266.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:25] "GET bu-ds561-dk98-bucket/hw2_output/4440.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:25] "GET bu-ds561-dk98-bucket/hw2_output/3544.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:26] "GET bu-ds561-dk98-bucket/hw2_output/7722.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:26] "GET bu-ds561-dk98-bucket/hw2_output/1052.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:27] "GET bu-ds561-dk98-bucket/hw2_output/2536.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:27] "GET bu-ds561-dk98-bucket/hw2_output/4273.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:28] "GET bu-ds561-dk98-bucket/hw2_output/481.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:29] "GET bu-ds561-dk98-bucket/hw2_output/4291.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:29] "GET bu-ds561-dk98-bucket/hw2_output/5404.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:30] "GET bu-ds561-dk98-bucket/hw2_output/4806.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:30] "GET bu-ds561-dk98-bucket/hw2_output/570.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:32] "GET bu-ds561-dk98-bucket/hw2_output/14992.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:34] "GET bu-ds561-dk98-bucket/hw2_output/1069.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:35] "GET bu-ds561-dk98-bucket/hw2_output/848.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:36] "GET bu-ds561-dk98-bucket/hw2_output/9990.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:36] "GET bu-ds561-dk98-bucket/hw2_output/8028.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:37] "GET bu-ds561-dk98-bucket/hw2_output/8235.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:38] "GET bu-ds561-dk98-bucket/hw2_output/2384.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:38] "GET bu-ds561-dk98-bucket/hw2_output/10136.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:39] "GET bu-ds561-dk98-bucket/hw2_output/4354.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:41] "GET bu-ds561-dk98-bucket/hw2_output/2123.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:41] "GET bu-ds561-dk98-bucket/hw2_output/7663.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:42] "GET bu-ds561-dk98-bucket/hw2_output/2927.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:21] "GET bu-ds561-dk98-bucket/hw2_output/6473.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:22] "GET bu-ds561-dk98-bucket/hw2_output/9202.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:23] "GET bu-ds561-dk98-bucket/hw2_output/10508.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:23] "GET bu-ds561-dk98-bucket/hw2_output/14888.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:24] "GET bu-ds561-dk98-bucket/hw2_output/2971.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:24] "GET bu-ds561-dk98-bucket/hw2_output/4215.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:24] "GET bu-ds561-dk98-bucket/hw2_output/8852.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:25] "GET bu-ds561-dk98-bucket/hw2_output/317.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:28] "GET bu-ds561-dk98-bucket/hw2_output/3235.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:30] "GET bu-ds561-dk98-bucket/hw2_output/3712.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:31] "GET bu-ds561-dk98-bucket/hw2_output/2248.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:31] "GET bu-ds561-dk98-bucket/hw2_output/10057.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:31] "GET bu-ds561-dk98-bucket/hw2_output/7661.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:32] "GET bu-ds561-dk98-bucket/hw2_output/2679.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:32] "GET bu-ds561-dk98-bucket/hw2_output/5087.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:33] "GET bu-ds561-dk98-bucket/hw2_output/3687.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:33] "GET bu-ds561-dk98-bucket/hw2_output/10549.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:34] "GET bu-ds561-dk98-bucket/hw2_output/7827.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:34] "GET bu-ds561-dk98-bucket/hw2_output/8389.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:35] "GET bu-ds561-dk98-bucket/hw2_output/8060.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:36] "GET bu-ds561-dk98-bucket/hw2_output/8668.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:37] "GET bu-ds561-dk98-bucket/hw2_output/91.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:38] "GET bu-ds561-dk98-bucket/hw2_output/1686.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:39] "GET bu-ds561-dk98-bucket/hw2_output/9243.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:40] "GET bu-ds561-dk98-bucket/hw2_output/6424.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:40] "GET bu-ds561-dk98-bucket/hw2_output/4402.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:40] "GET bu-ds561-dk98-bucket/hw2_output/2489.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:41] "GET bu-ds561-dk98-bucket/hw2_output/6337.html HTTP/1.1" 404 -
128.197.29.228 - - [18/Nov/2023 04:02:42] "GET bu-ds561-dk98-bucket/hw2_output/2570.html HTTP/1.1" 200 -
128.197.29.228 - - [18/Nov/2023 04:02:43] "GET bu-ds561-dk98-bucket/hw2_output/5129.html HTTP/1.1" 200 -
```

<input type="checkbox"/>	<input checked="" type="checkbox"/>	www1	us-east1-b	www-pool	10.142.0.35 (nic0)	34.148.13.88 🔗 (nic0)	SSH	▼	⋮
<input type="checkbox"/>	<input checked="" type="checkbox"/>	www2	us-east1-c	www-pool	10.142.0.36 (nic0)	34.148.149.114 🔗 (nic0)	SSH	▼	⋮
<input type="checkbox"/>	<input checked="" type="checkbox"/>	www3	us-east1-b		10.142.0.38 (nic0)	34.75.133.64 (nic0)	SSH	▼	⋮

Google Cloud

DS 561

Search (/) for resources, docs, products, and more

VM instances

Network services

Load balancing

Cloud DNS

Cloud CDN

Cloud NAT

Traffic Director

Service Directory

Cloud Domains

Private Service Connect

SSL policies

Target pool details

DELETE

In order to edit target pools use load balancer mutations, gcloud or REST API.

www-pool

In use by [www-8080-rule](#)

Region [us-east1](#)

Session affinity [None](#)

Health check [basic-check](#)

VM instances [VIEW VM INSTANCES](#)

EQUIVALENT REST

Instance ↑

Zone

35,190,170,219

[www1](#)

us-east1-b

✓

[www2](#)

us-east1-c

✓

```
python3 http-client.py --ss 35.190.170.219 -b hu-ds651-dk8r-bucket -w h2_output -n 100000 -l 11000 -p 8000 -v -f
riatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.<ncp>ncp=HREF=1448.html"> This is a link </><ncp>ncp=lorum ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.<ncp>ncp=lorum ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.<ncp>ncp=lorum ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.<ncp>ncp=lorum ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.<ncp>ncp=HREF=5528.html"> This is a link </><ncp>ncp=body</html></ncp>
</w2-zone>
Requesting hu-ds651-dk8r-bucket/h2_output/4532.html from 35.190.170.219 8800
Traceback (most recent call last):
File "/Users/davidkim/Desktop/DataScience/HW/DS651/da6561-davidkim-U66545284/hw/http-client.py", line 201, in module
main()
File "/Users/davidkim/Desktop/DataScience/HW/DS651/da6561-davidkim-U66545284/hw/http-client.py", line 198, in main
args=args, url=args.url, port=args.port, country=args.country, filename=args.ssi, ssl_context=args.forgo_sslcontext, verbose=True)
File "/Users/davidkim/Desktop/DataScience/HW/DS651/da6561-davidkim-U66545284/hw/http-client.py", line 142, in make_request
conn.request("GET", filename, headers=headers)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 1265, in request
self._send_request(method, url, body, headers, encode_chunked)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 1331, in _send_request
self._encode_headers(headers, encode_chunked)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 1280, in _encode_headers
self._send_output(message_body, encode_chunked=encode_chunked)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 1040, in _send_output
self.send(msg)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 980, in send
self._buffer.append(data)
File "/Users/davidkim/tensorflow/1/b/python3.9/http/client.py", line 946, in connect
self.sock = self._create_connection()
File "/Users/davidkim/tensorflow/1/b/python3.9/socket.py", line 844, in create_connection
raise err
File "/Users/davidkim/tensorflow/1/b/python3.9/socket.py", line 832, in create_connection
ConnectionRefusedError: [Errno 61] Connection refused
```

END