Distributed Systems
SOFE 4790U

Lab 3: Deploying a Circuit Breaking ambassador and a Function-as-a-Service (FaaS)

Group 9
Hemshikha Sultoo - 100670616

Part 1

Requirements like validating the response code, checking the content of the response to detect errors and measuring the response time can be achieved by part 2 and 3. The problem is that it is difficult to monitor services on the cloud on a large scale. Therefore, get the app to perform function checks at endpoints and return the status in order to determine whether everything is running the way it should.

Part 2

Building the Docker Image

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2/DummyServiceContainer (cohesive-cell-362020)$ docker build . -t us.gcr.io/cohesive-cell-362020/dummyservice
Sending build context to Docker daemon  6.656kB
Step 1/7 : FROM node:carbon
carbon: Pulling from library/node
146bd6a88618: Pull complete
9935d0c62ace: Pull complete
db0efb86e806: Pull complete
e705a4c4fd31: Pull complete
c877b722db6f: Pull complete
645c20ec8214: Pull complete
db8fbd9db2fe: Pull complete
1c151cd1b3ea: Pull complete
fbd993995f40: Pull complete
Digest: sha256:a681bf74805b80d03eb21a6c0ef168a976108a287a74167ab593fc953aac34df
Status: Downloaded newer image for node:carbon
 ---> 8eeadf3757f4
Step 2/7 : WORKDIR /usr/src/app
 ---> Running in 22e7b5d18e1f
Removing intermediate container 22e7b5d18e1f
 ---> 6fa4010a14e4
Step 3/7 : COPY package*.json ./
 ---> 076c352d7403
```

```
Successfully built 7e7e31fc6dd2
```

Pushing the image to the GCP container hub

```
362020)$ docker push us.gcr.io/cohesive-cell-362020/dummyservice
Using default tag: latest
The push refers to repository [us.gcr.io/cohesive-cell-362020/dummyservice]
c693f9a0a11f: Pushed
28b512a92839: Pushed
1bd2b9098057: Pushed
c172df178d8d: Pushed
423451ed44f2: Layer already exists
b2aaf85d6633: Layer already exists
88601a85ce11: Layer already exists
42f9c2f9c08e: Layer already exists
99e8bd3efaaf: Layer already exists
bee1e39d7c3a: Layer already exists
1f59a4b2e206: Layer already exists
0ca7f54856c0: Layer already exists
ebb9ae013834: Layer already exists
latest: digest: sha256:19bae3f04be36b4cc4cdfca95d0f578194fbc1b1025b742222e52241ed5a0beb size:
3048
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2/DummyServiceContainer (cohesive-cell-
362020)$
```

Deployed the dummy deployment

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl creat
e -f dummy-deployment.yaml
deployment.apps/dummy-deployment created
```

Deployment exposed via load-balancer

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl expos
e deployment dummy-deployment --port=80 --type=LoadBalancer --name dummy-deployment
service/dummy-deployment exposed
```

Checking for pods, deployments and services

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
dummy-deployment-5598b48c98-rbrcg   1/1     Running   0          3m10s
mongo-0                             1/1     Running   0          40m
mongo-deployment-0                  1/1     Running   0          39m
mongodb-0                           1/1     Running   0          39m
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
dummy-deployment   1/1     1            1           3m29s
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl get services
NAME               TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)        AGE
dummy-deployment   LoadBalancer   10.80.2.107    34.95.55.235    80:30870/TCP   115s
kubernetes         ClusterIP      10.80.0.1      <none>          443/TCP        34d
```

Step 6

```
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ cd ~/SOFE4790U-lab3/part2/
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl create -f nginx-configmap.yaml
configmap/nginx-configuration created
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl create - f circuitbreaker.yaml
Error: must specify one of -f and -k

error: unknown command "- f circuitbreaker.yaml"
See 'kubectl create -h' for help and examples
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl create -f circuitbreaker.yaml
deployment.apps/circuitbreaker created
service/circuitbreaker created
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl get deployments
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
circuitbreaker    0/1     1            0           25s
dummy-deployment  1/1     1            1           91m
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ kubectl get services
NAME              TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)        AGE
circuitbreaker    LoadBalancer   10.80.8.221    34.152.44.77    80:32349/TCP   33s
dummy-deployment  LoadBalancer   10.80.2.107    34.95.55.235    80:30870/TCP   89m
kubernetes        ClusterIP      10.80.0.1      <none>          443/TCP        35d
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$
```

Step 7

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -v http://34.152.44.77
*   Trying 34.152.44.77:80...
* connect to 34.152.44.77 port 80 failed: Connection refused
* Failed to connect to 34.152.44.77 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 34.152.44.77 port 80: Connection refused
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -v http://34.152.44.77
*   Trying 34.152.44.77:80...
* connect to 34.152.44.77 port 80 failed: Connection refused
* Failed to connect to 34.152.44.77 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 34.152.44.77 port 80: Connection refused
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -v http://34.152.44.77
*   Trying 34.152.44.77:80...
* connect to 34.152.44.77 port 80 failed: Connection refused
* Failed to connect to 34.152.44.77 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 34.152.44.77 port 80: Connection refused
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -d "" -s -D - http://34.95.55.235/fakeerrormodeon
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 18
ETag: W/"12-3A0j4woanZCjyZbOLgo7FV9oStI"
Date: Sun, 23 Oct 2022 23:00:11 GMT
Connection: keep-alive

OK FROM 10.76.2.17hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -v http://34.152.44.77
*   Trying 34.152.44.77:80...
* connect to 34.152.44.77 port 80 failed: Connection refused
* Failed to connect to 34.152.44.77 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 34.152.44.77 port 80: Connection refused
```

```
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -d "" -s -D - http://34.95.55.235/fakeerrormodeoff
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 18
ETag: W/"12-3A0j4woanZCjyZbOLgo7FV9oStI"
Date: Sun, 23 Oct 2022 23:01:25 GMT
Connection: keep-alive

OK FROM 10.76.2.17hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ curl -v http://34.152.44.77
*   Trying 34.152.44.77:80...
* connect to 34.152.44.77 port 80 failed: Connection refused
* Failed to connect to 34.152.44.77 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 34.152.44.77 port 80: Connection refused
hem_distributedsystems@cloudshell:~/SOFE4790U-lab3/part2 (cohesive-cell-362020)$ 
```

Video Link of Part 2:
https://drive.google.com/file/d/1tJme84__sRaQ4Z_tXoPtGXgLG3kf9evV/view?usp=sharing

Part 3

```
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ curl -SLsf https://cli.openfaas.com | sudo sh
Finding latest version from GitHub
0.14.11
Downloading package https://github.com/openfaas/faas-cli/releases/download/0.14.11/faas-cli as /tmp/faas-cli
Download complete.

Running with sufficient permissions to attempt to move faas-cli to /usr/local/bin
New version of faas-cli installed to /usr/local/bin

CLI:
 commit:  8820d8e4a15dab900d8a7e8fc271851ccb94012e
 version: 0.14.11
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl -n openfaas get deployments -l "release=openfaas, app=openfaas"
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
alertmanager        1/1     1            1           68m
basic-auth-plugin   1/1     1            1           68m
gateway             1/1     1            1           68m
nats                1/1     1            1           68m
prometheus          1/1     1            1           68m
queue-worker        1/1     1            1           68m
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl rollout status -n openfaas deploy/gateway
deployment "gateway" successfully rolled out
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl get svc -o wide gateway-external -n openfaas
NAME               TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE   SELECTOR
gateway-external   LoadBalancer   10.80.13.199   34.95.12.10   8080:30270/TCP   69m   app=gateway
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ export OPENFAAS_URL="34.95.12.10:8080"
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ PASSWORD=$(kubectl get secret -n openfaas basic-auth -o jsonpath="{.data.basic-authpassword}" | base64 --decode; echo)
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ echo $PASSWORD

hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ PASSWORD=$(kubectl get secret -n openfaas basic-auth -o jsonpath="{.data.basic-auth-password}" | base64 --decode; echo)
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ echo $PASSWORD
866X7GN4dr060fhR4yCy3GOAS
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ echo -n $PASSWORD | faas-cli login --username admin --password-stdin
Calling the OpenFaaS server to validate the credentials...
WARNING! You are not using an encrypted connection to the gateway, consider using HTTPS.
credentials saved for admin http://34.95.12.10:8080
```

```
Total build time: 21.24s
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ docker push us.gcr.io/cohesive-cell-362020/main
Using default tag: latest
The push refers to repository [us.gcr.io/cohesive-cell-362020/main]
bf5093e6e111: Pushed
701ae41eae6b: Pushed
9a3cad1fb65a: Pushed
7a0c936fb321: Pushed
a13ffa773994: Pushed
2478a368d63d: Pushed
9e6acb50dbf4: Pushed
873152d9f32f: Pushed
d86bda25f4c1: Pushed
8074db450cf8: Pushed
93a030f14ec1: Pushed
7f30cde3f699: Layer already exists
fe810f5902cc: Layer already exists
dfd8c046c602: Layer already exists
4fc242d58285: Layer already exists
latest: digest: sha256:a1bd47e764d5fb4fb4bbdd3e586ae69ad37b26352b49fa0ff639145d4e4e122b size: 3659
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ faas-cli deploy -f main.yml
Deploying: main.
WARNING! You are not using an encrypted connection to the gateway, consider using HTTPS.

Deployed. 202 Accepted.
URL: http://34.95.12.10:8080/function/main
```

```
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ curl http://34.95.12.10:80
{}
[1/3]:  "Name": "Square" --> <stdout>
--_curl_-- "Name": "Square"
curl: (3) URL using bad/illegal format or missing URL

[2/3]:  "Color": "Red" --> <stdout>
--_curl_-- "Color": "Red"
curl: (3) URL using bad/illegal format or missing URL

[3/3]:  "Dimensions": 2  --> <stdout>
--_curl_-- "Dimensions": 2
curl: (3) URL using bad/illegal format or missing URL
```

```
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ faas-cli build -f decorator.yml
[0] > Building decorator.
Clearing temporary build folder: ./build/decorator/
Preparing: ./decorator/ build/decorator/function
Building: us.gcr.io/cohesive-cell-362020/decorator:latest with node12 template. Please wait..
Sending build context to Docker daemon  13.31kB
Step 1/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} ghcr.io/openfaas/of-watchdog:0.9.10 as watchdog
 ---> 9f97468a4531
Step 2/31 : FROM --platform=${TARGETPLATFORM:-linux/amd64} node:12-alpine as ship
 ---> bb6d28039b8c
Step 3/31 : ARG TARGETPLATFORM
 ---> Using cache
 ---> 7ac73f6eb716
Step 4/31 : ARG BUILDPLATFORM
 ---> Using cache
 ---> 2a637e31313f
Step 5/31 : COPY --from=watchdog /fwatchdog /usr/bin/fwatchdog
 ---> Using cache
 ---> b61acdb0de26
Step 6/31 : RUN chmod +x /usr/bin/fwatchdog
 ---> Using cache
 ---> 0f7e9ad4da0c
Step 7/31 : RUN apk --no-cache add curl ca-certificates     && addgroup -S app && adduser -S -g app app
 ---> Using cache
 ---> 60be5bc80a35
Step 8/31 : ENV NPM_CONFIG_LOGLEVEL warn
 ---> Using cache
 ---> 49b07c5f9e29
Step 9/31 : RUN chmod 777 /tmp
 ---> Using cache
 ---> 8de74c8c7a36
Step 10/31 : USER app
 ---> Using cache
 ---> d0987aec79bd
```

```
Successfully built c2815ea7c6c8
Successfully tagged us.gcr.io/cohesive-cell-362020/decorator:latest
Image: us.gcr.io/cohesive-cell-362020/decorator:latest built.
[0] < Building decorator done in 11.11s.
[0] Worker done.

Total build time: 11.11s
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$
```

```
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ docker push us.gcr.io/cohesive-cell-362020/ decorator
"docker push" requires exactly 1 argument.
See 'docker push --help'.

Usage:  docker push [OPTIONS] NAME[:TAG]

Push an image or a repository to a registry
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ docker push us.gcr.io/cohesive-cell-362020/decorator
Using default tag: latest
The push refers to repository [us.gcr.io/cohesive-cell-362020/decorator]
588871a7a4ad: Pushed
3255886dfd37: Pushed
71f9db9b1014: Pushed
c58b57f3bb90: Pushed
a13ffa773994: Layer already exists
2478a368d63d: Layer already exists
9e6acb50dbf4: Layer already exists
873152d9f32f: Layer already exists
d86bda25f4c1: Layer already exists
8074db450cf8: Layer already exists
93a030f14ec1: Layer already exists
7f30cde3f699: Layer already exists
fe810f5902cc: Layer already exists
dfd8c046c602: Layer already exists
4fc242d58285: Layer already exists
latest: digest: sha256:faaa967f965cc98a77850ada128d43d7937d4ebda446139f9442f3b5f4995be6 size: 3663
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$
```

```
hem_distributedsystems@cloudshell:~/OpenFaaS (cohesive-cell-362020)$ faas-cli deploy -f decorator.yml
Deploying: decorator.
WARNING! You are not using an encrypted connection to the gateway, consider using HTTPS.

Deployed. 202 Accepted.
URL: http://34.95.12.10:8080/function/decorator
```

Part 3 video Link:
https://drive.google.com/file/d/1QkID_iUKgsNK1BI_SffPup9svU_S7f8j/view?usp=sharing

Design

Persistent volumes (PersistentVolume, PV, subsystem) is about the way that storage is provided to and consumed by users. PVs are resources or storage in the cluster that has been provided by the administrator. PVs allow persistent storage for the containerized apps, that is the application pod will still be able to access previously stored data, even after a restart. This is called providing storage beyond the lifecycle of a pod - which is exactly an example of where PV is used.

Video:
https://drive.google.com/file/d/1dk1QEqLSRXqenw41Z8r7PP3kqaQrF2w3/view?usp=sharing

Creating Persistent Volume YAML to configure pods to use PV for storage

- Status ‘Available’ suggests that the storage space needs to be claimed/requested through a persistent volume claim.

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to cohesive-cell-362020.
Use “gcloud config set project [PROJECT_ID]” to change to a different project.
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ cd pods/storage
-bash: cd: pods/storage: No such file or directory
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ ls
ambassador-deployment.yaml  experiment-deployment.yaml    mongodb-svc.yaml  OpenFaaS    README-cloudshell.txt  web-deployment.yaml
conf.d                      loadbalancer-deployment.yaml  mongo.yaml        output.txt  SOFE4790U-lab3
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl apply -f pv.yaml
persistentvolume/task-pv-volume created
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl get pv task-pv.volume
Error from server (NotFound): persistentvolumes "task-pv.volume" not found
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$ kubectl get pv task-pv-volume
NAME             CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   AGE
task-pv-volume   10Gi       RWO            Retain           Available           manual                  32s
hem_distributedsystems@cloudshell:~ (cohesive-cell-362020)$
```