**INSTALLATION**

***Checking installation manual***

<https://minikube.sigs.k8s.io/docs/start/>

***Download installation binary***

Ozlems-MBP:Desktop ademmecit$ mkdir adjust

Ozlems-MBP:Desktop ademmecit$ cd adjust/

Ozlems-MBP:adjust ademmecit$ curl -LO <https://storage.googleapis.com/minikube/releases/latest/minikube-darwin-amd64>

***Install minikube***

Ozlems-MBP:adjust ademmecit$ ls -l

total 164024

-rw-r--r-- 1 ademmecit staff 68393536 29 Oct 15:08 minikube-darwin-amd64

Ozlems-MBP:adjust ademmecit$ sudo install minikube-darwin-amd64 /usr/local/bin/minikube

Password:

***Starting minikube***

Ozlems-MBP:adjust ademmecit$ minikube start

😄 minikube v1.23.2 on Darwin 10.15.7

✨ Automatically selected the hyperkit driver

💾 Downloading driver docker-machine-driver-hyperkit:

> docker-machine-driver-hyper...: 65 B / 65 B [----------] 100.00% ? p/s 0s

> docker-machine-driver-hyper...: 508.21 KiB / 10.53 MiB [>\_] 4.71% ? p/s ?

> docker-machine-driver-hyper...: 10.53 MiB / 10.53 MiB 100.00% 1.28 MiB p

🔑 The 'hyperkit' driver requires elevated permissions. The following commands will be executed:

$ sudo chown root:wheel /Users/ozlemadem/.minikube/bin/docker-machine-driver-hyperkit

$ sudo chmod u+s /Users/ozlemadem/.minikube/bin/docker-machine-driver-hyperkit

💿 Downloading VM boot image ...

> minikube-v1.23.1.iso.sha256: 65 B / 65 B [-------------] 100.00% ? p/s 0s

> minikube-v1.23.1.iso: 225.22 MiB / 225.22 MiB 100.00% 1.14 MiB p/s 3m18s

👍 Starting control plane node minikube in cluster minikube

💾 Downloading Kubernetes v1.22.2 preload ...

> preloaded-images-k8s-v13-v1...: 511.69 MiB / 511.69 MiB 100.00% 426.81 K

🔥 Creating hyperkit VM (CPUs=2, Memory=2200MB, Disk=20000MB) ...

🐳 Preparing Kubernetes v1.22.2 on Docker 20.10.8 ...

▪ Generating certificates and keys ...

▪ Booting up control plane ...

▪ Configuring RBAC rules ...

🔎 Verifying Kubernetes components...

▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5

🌟 Enabled addons: storage-provisioner, default-storageclass

❗ /usr/local/bin/kubectl is version 1.17.4, which may have incompatibilites with Kubernetes 1.22.2.

▪ Want kubectl v1.22.2? Try 'minikube kubectl -- get pods -A'

🏄 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default

***Post-installation steps to control installation & making alias on shell***

Ozlems-MBP:adjust ademmecit$ minikube kubectl -- get po -A

> kubectl.sha256: 64 B / 64 B [--------------------------] 100.00% ? p/s 0s

> kubectl: 50.63 MiB / 50.63 MiB [-----------] 100.00% 693.77 KiB p/s 1m15s

NAMESPACE NAME READY STATUS RESTARTS AGE

kube-system coredns-78fcd69978-b6lqp 1/1 Running 0 129m

kube-system etcd-minikube 1/1 Running 0 129m

kube-system kube-apiserver-minikube 1/1 Running 0 129m

kube-system kube-controller-manager-minikube 1/1 Running 0 129m

kube-system kube-proxy-lrc4d 1/1 Running 0 129m

kube-system kube-scheduler-minikube 1/1 Running 0 129m

kube-system storage-provisioner 1/1 Running 2 (10m ago) 129m

Ozlems-MBP:adjust ademmecit$ kubectl get po -A

NAMESPACE NAME READY STATUS RESTARTS AGE

kube-system coredns-78fcd69978-b6lqp 1/1 Running 0 130m

kube-system etcd-minikube 1/1 Running 0 130m

kube-system kube-apiserver-minikube 1/1 Running 0 130m

kube-system kube-controller-manager-minikube 1/1 Running 0 130m

kube-system kube-proxy-lrc4d 1/1 Running 0 130m

kube-system kube-scheduler-minikube 1/1 Running 0 130m

kube-system storage-provisioner 1/1 Running 2 (11m ago) 130m

Ozlems-MBP:adjust ademmecit$ alias kubectl="minikube kubectl --"

***Checking minkube status on dashboard by using a browser***

Ozlems-MBP:adjust ademmecit$ minikube dashboard

🔌 Enabling dashboard ...

▪ Using image kubernetesui/dashboard:v2.3.1

▪ Using image kubernetesui/metrics-scraper:v1.0.7

🤔 Verifying dashboard health ...

🚀 Launching proxy ...

🤔 Verifying proxy health ...

🎉 Opening http://127.0.0.1:53883/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default browser...

Graphical user interface, text, application, email

Description automatically generated

Ozlems-MBP:adjust ademmecit$ mkdir homework

Ozlems-MBP:adjust ademmecit$ cd homework/

***GIT CONFIGURATION***

*Forking app repository*

Fork app repository on github (<https://github.com/sawasy/http_server>) to personal repository

*Cloning personal repository to local*

Ozlems-MBP:homework ademmecit$ git clone https://github.com/ademmecit/http\_server

Cloning into 'http\_server'...

remote: Enumerating objects: 16, done.

remote: Counting objects: 100% (16/16), done.

remote: Compressing objects: 100% (15/15), done.

remote: Total 16 (delta 5), reused 5 (delta 0), pack-reused 0

Unpacking objects: 100% (16/16), 4.41 KiB | 282.00 KiB/s, done.

Ozlems-MBP:homework ademmecit$ git config --global user.name "Adem Mecit"

Ozlems-MBP:homework ademmecit$ git config --global user.email "adem.mecit@gmail.com"

Ozlems-MBP:homework ademmecit$ ssh-keygen -t rsa -b 4096

Generating public/private rsa key pair.

Enter file in which to save the key (/Users/ozlemadem/.ssh/id\_rsa):

/Users/ozlemadem/.ssh/id\_rsa already exists.

Overwrite (y/n)? y

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /Users/ozlemadem/.ssh/id\_rsa.

Your public key has been saved in /Users/ozlemadem/.ssh/id\_rsa.pub.

The key fingerprint is: xxxx

***Add SSH key to github***

***Testing the connectivity to the github***

Ozlems-MBP:homework ademmecit$ ssh -T git@github.com

Warning: Permanently added the RSA host key for IP address '140.82.121.4' to the list of known hosts.

Enter passphrase for key '/Users/ozlemadem/.ssh/id\_rsa':

Hi ademmecit! You've successfully authenticated, but GitHub does not provide shell access.

Ozlems-MBP:homework ademmecit$ git init

Reinitialized existing Git repository in /Users/ozlemadem/Desktop/adjust/homework/.git/

***Run Docker Daemon on Client***

Ozlems-MBP:homework ademmecit$ open --background -a Docker

***Creating docker image***

*Creating Dockerfile*

Ozlems-MBP:homework ademmecit$ cat Dockerfile

FROM ruby:latest

EXPOSE 80

COPY http\_server.rb .

CMD ruby http\_server.rb

*Create docker image :*

Ozlems-MBP:homework ademmecit$ docker build -t homework .

Ozlems-MBP:homework ademmecit$ docker build -t homework .

[+] Building 2.7s (8/8) FINISHED

=> [internal] load build definition from Dockerfile 0.0s

=> => transferring dockerfile: 36B 0.0s

=> [internal] load .dockerignore 0.0s

=> => transferring context: 2B 0.0s

=> [internal] load metadata for docker.io/library/ruby:latest 2.2s

=> [auth] library/ruby:pull token for registry-1.docker.io 0.0s

=> [internal] load build context 0.0s

=> => transferring context: 36B 0.0s

=> [1/2] FROM docker.io/library/ruby:latest@sha256:179ae4287b6bb0b9f03d09cd1f10c2aba7e34c60183c838b2ea00bcdd85c43d7 0.0s

=> CACHED [2/2] COPY http\_server.rb . 0.0s

=> exporting to image 0.0s

=> => exporting layers 0.0s

=> => writing image sha256:778f0e2ba7fb9a826ce44eb876503be9e21120d334566fe13709cfe4d0e68f4e 0.0s

=> => naming to docker.io/library/homework 0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

*Check docker image :*

Ozlems-MBP:homework ademmecit$ docker image ls

REPOSITORY TAG IMAGE ID CREATED SIZE

homework latest 778f0e2ba7fb 14 minutes ago 881MB

ademmecit/docker101tutorial latest 5be237b9a37a 5 hours ago 28.3MB

docker101tutorial latest 5be237b9a37a 5 hours ago 28.3MB

alpine/git latest 0deb7380d708 10 days ago 27.4MB

101309896996.dkr.ecr.eu-west-1.amazonaws.com/wemesh latest 559d85e8bce5 13 months ago 1GB

v8/docker\_wemesh latest 559d85e8bce5 13 months ago 1GB

v7/docker\_wemesh latest d0dc8dcf12f2 14 months ago 1GB

101309896996.dkr.ecr.eu-west-1.amazonaws.com/wemesh <none> d0dc8dcf12f2 14 months ago 1GB

v6/docker\_wemesh latest 53bd2ac85061 15 months ago 1GB

101309896996.dkr.ecr.eu-west-1.amazonaws.com/wemesh <none> 53bd2ac85061 15 months ago 1GB

v4/docker\_wemesh latest 91c32ec67294 16 months ago 1GB

v5/docker\_wemesh latest 91c32ec67294 16 months ago 1GB

101309896996.dkr.ecr.eu-west-1.amazonaws.com/wemesh <none> 91c32ec67294 16 months ago 1GB

v3/docker\_wemesh latest 44bac70279de 16 months ago 1GB

101309896996.dkr.ecr.eu-west-1.amazonaws.com/wemesh <none> 44bac70279de 16 months ago 1GB

node latest dcda6cd5e439 16 months ago 942MB

*Run container image on port 3000 (mapping container port 80 for external access)*

Ozlems-MBP:homework ademmecit$ docker run -dp 3000:80 homework

c43beef693c54d97d3406d11c0bdf5dd0ca80112844026d65d811d132eba2f8d

*Check running container :*

Ozlems-MBP:homework ademmecit$ docker container ls

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

c43beef693c5 homework "/bin/sh -c 'ruby ht…" 8 seconds ago Up 7 seconds 0.0.0.0:3000->80/tcp, :::3000->80/tcp festive\_wescoff

*Reach container terminal to check ruby version :*

Ozlems-MBP:homework ademmecit$ docker exec -it c43beef693c5 /bin/sh

# ruby -v

ruby 3.0.2p107 (2021-07-07 revision 0db68f0233) [x86\_64-linux]

*Use Postman for GET request*

No response to user and getting container is stopping

*Checking container console for the error messages*

Ozlems-MBP:homework ademmecit$ docker container ls

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

c43beef693c5 homework "/bin/sh -c 'ruby ht…" 11 minutes ago Up 5 minutes 0.0.0.0:3000->80/tcp, :::3000->80/tcp magical\_wright

Ozlems-MBP:homework ademmecit$ docker container logs c43beef693c5

http\_server.rb:7:in `readpartial': end of file reached (EOFError)

from http\_server.rb:7:in `block in <main>'

from http\_server.rb:5:in `loop'

from http\_server.rb:5:in `<main>'

GET / HTTP/1.1

GET / HTTP/1.1

*Using locally installed rubby to test the application*

Use ruby installed directly on to the computer in order to eleminate potential docker or container issues and found some issues with http server code and added some lines for troubleshhoing and finally found the error

Ozlems-MBP:homework ademmecit$ cat http\_server.rb

require 'socket'

server = TCPServer.new('0.0.0.0', 80)

loop {

client = server.accept

request = client.readpartial(2048)

method, path, version = request.lines[0].split

puts "#{method} #{path} #{version}"

client.write("HTTP/1.1 200\r\n") # 1

client.write("Content-Type: text/html\r\n") # 2

client.write("\r\n") # 3

if path == "/healthcheck"

puts "healthcheck"

client.write("OK")

else

puts "other"

client.write("Well, hello there!")

end

client.close

}

Added headers to response page

Added for troubleshooting (no rubby experience)

Server side app can be able to parse URL and gives approprite response

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

***Git Update***

*Create Personal Access Token*

Creating personal Access token to make updates

*Initialize repository for git*

Ozlems-MBP:http\_server ademmecit$ git init

Reinitialized existing Git repository in /Users/ozlemadem/Desktop/adjust/homework/http\_server/.git/

*Checking repository status adding new files, commiting changes and updating public repository*

Ozlems-MBP:http\_server ademmecit$ git status

On branch main

Your branch is up to date with 'origin/main'.

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: Dockerfile

new file: HomeworkDocumentation.docx

modified: http\_server.rb

Ozlems-MBP:http\_server ademmecit$ git add -A

Ozlems-MBP:http\_server ademmecit$ git commit -m 'bug fix and docker implementation'

[main 5546010] bug fix and docker implementation

3 files changed, 13 insertions(+), 1 deletion(-)

create mode 100644 Dockerfile

create mode 100644 HomeworkDocumentation.docx

Ozlems-MBP:http\_server ademmecit$ git push

Enumerating objects: 7, done.

Counting objects: 100% (7/7), done.

Delta compression using up to 4 threads

Compressing objects: 100% (5/5), done.

Writing objects: 100% (5/5), 168.32 KiB | 16.83 MiB/s, done.

Total 5 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/ademmecit/http\_server

75b7acc..5546010 main -> main

Ozlems-MBP:http\_server ademmecit$ eval $(minikube docker-env)

Ozlems-MBP:http\_server ademmecit$ docker build -t homework:0.0.1 -f homework-dockerfile .

Sending build context to Docker daemon 657.9kB

Step 1/4 : FROM ruby:latest

---> 090e42145ec2

Step 2/4 : EXPOSE 80

---> Using cache

---> efaba386bf71

Step 3/4 : COPY http\_server.rb .

---> 9be598f86241

Step 4/4 : CMD ruby http\_server.rb

---> Running in 44d99dc06597

Removing intermediate container 44d99dc06597

---> 05c9a0647c38

Successfully built 05c9a0647c38

Successfully tagged homework:0.0.1

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

Ozlems-MBP:http\_server ademmecit$ cat homework-deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: homework

labels:

app: homework

spec:

replicas: 2

selector:

matchLabels:

app: homework

template:

metadata:

labels:

app: homework

spec:

containers:

- name: homework

image: homework:0.0.1

ports:

- containerPort: 80

Ozlems-MBP:http\_server ademmecit$ kubectl create -f homework\_deployment.yaml

deployment.apps/homework created

Ozlems-MBP:http\_server ademmecit$ kubectl get deployments

NAME READY UP-TO-DATE AVAILABLE AGE

homework 2/2 2 2 11s

Ozlems-MBP:http\_server ademmecit$ kubectl get pods

NAME READY STATUS RESTARTS AGE

homework-5589c9f9fd-9jkjj 1/1 Running 0 8m8s

homework-5589c9f9fd-g5vc7 1/1 Running 0 8m8s

Ozlems-MBP:http\_server ademmecit$ cat homework-service.yaml

kind: Service

apiVersion: v1

metadata:

name: homework

spec:

selector:

app: homework

ports:

- nodePort: 32500

port: 8081

protocol: TCP

targetPort: 80

type: LoadBalancer

Ozlems-MBP:http\_server ademmecit$ kubectl create -f homework-service.yaml

service/homework created

Ozlems-MBP:http\_server ademmecit$ kubectl get services

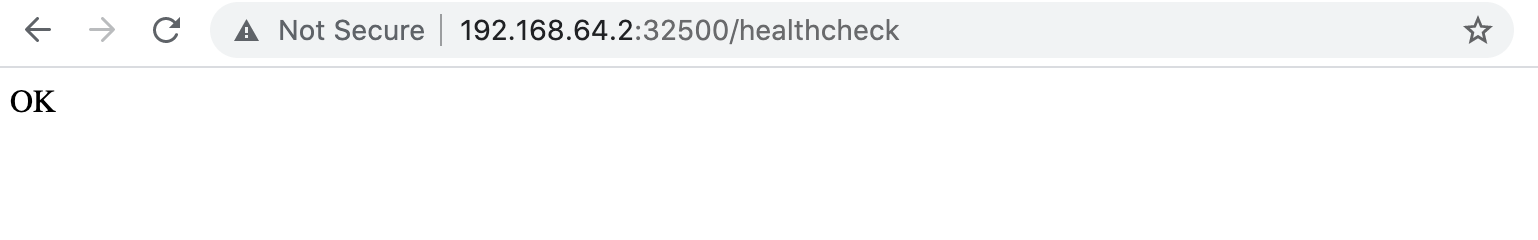
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

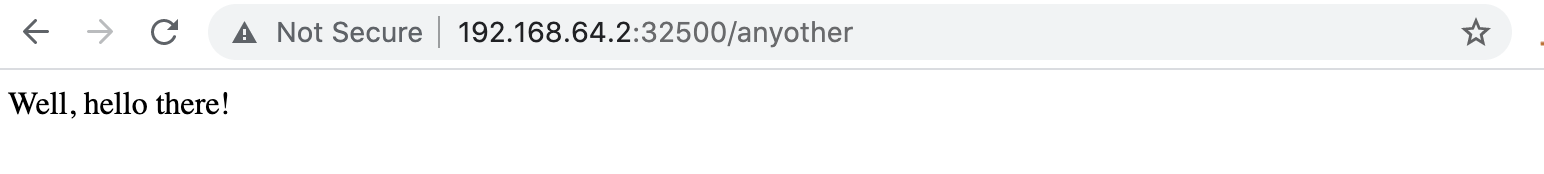
homework LoadBalancer 10.102.155.39 <pending> 8081:32500/TCP 15s

kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 30h

Ozlems-MBP:http\_server ademmecit$ minikube service homework --url

http://192.168.64.2:32500





*Hardening Solution With livenessProbe and readinessProbe*

Ozlems-MBP:http\_server ademmecit$ cat homework-deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: homework

labels:

app: homework

spec:

replicas: 2

selector:

matchLabels:

app: homework

template:

metadata:

labels:

app: homework

spec:

containers:

- name: homework

image: homework:0.0.1

ports:

- containerPort: 80

livenessProbe:

httpGet:

path: /healthcheck

port: 80

periodSeconds: 3

readinessProbe:

httpGet:

path: /healthcheck

port: 80

periodSeconds: 1

Check containers readiness just after spinning it to make service available quickly as much as possible and repeat each 1 seconds until it is available to recieve traffic.

Check containers liveness every 3 seconds and terminate it if there is a failure.