Lab 5 – Containers and Kubernetes

Aleksandra Stachniak

Task 1

```
C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker>docker run --rm -v C:/Users/olas0/.aws:/root/.aws aws-cli --version
aws-cli/2.10.1 Python/3.8.20 Linux/5.15.153.1-microsoft-standard-WSL2 source-sandbox/x86_64.alpine.3 prompt/off
```

```
C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker>docker run --rm -v C:/Users/olas0/.aws:/root/.aws aws-cli s3 ls
2024-11-12 18:40:23 astachniak-bucket-lsc
```

```
C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker>docker images
REPOSITORY                          TAG       IMAGE ID       CREATED             SIZE
aws-cli                             latest    826c5d81ffa6   About an hour ago   600MB
```

Task 2

Commands used to set up the entire deployment:

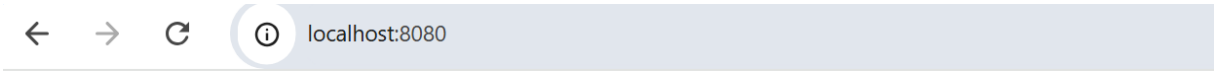
- .\kind.exe create cluster --name astachniak-cluster --config=lscconfig.yaml
- .\helm.exe install nfs-server-provisioner nfs-ganesha-server-and-external-provisioner/nfs-server-provisioner --set storageClass.name=lscdocker
- kubectl apply --context kind-astachniak-cluster --filename=pvc.yaml
- kubectl apply --context kind-astachniak-cluster --filename=deployment.yaml
- kubectl apply --context kind-astachniak-cluster --filename=service.yaml
- kubectl apply --context kind-astachniak-cluster --filename=job.yaml

```
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> .\kind.exe --version
kind version 0.25.0
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> .\kind.exe create cluster --name astachniak-cluster --config=lscconfig.yaml
Creating cluster "astachniak-cluster" ...
 ✓ Ensuring node image (kindest/node:v1.31.2) 🖼
 ✓ Preparing nodes 📦
 ✓ Writing configuration 📜
 ✓ Starting control-plane 🕹️
 ✓ Installing CNI 🔌
 ✓ Installing StorageClass 💾
Set kubectl context to "kind-astachniak-cluster"
You can now use your cluster with:

kubectl cluster-info --context kind-astachniak-cluster

Thanks for using kind! 😊
```

```
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> .\helm.exe repo add nfs-ganesha-server-and-external-provisioner https://kubernetes-
sigs.github.io/nfs-ganesha-server-and-external-provisioner
"nfs-ganesha-server-and-external-provisioner" has been added to your repositories
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> .\helm.exe repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "nfs-ganesha-server-and-external-provisioner" chart repository
Update Complete. ⎈Happy Helming!⎈
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> .\helm.exe install nfs-server-provisioner nfs-ganesha-server-and-external-provision
er/nfs-server-provisioner --set storageClass.name=lscdocker
NAME: nfs-server-provisioner
LAST DEPLOYED: Tue Nov 12 21:37:16 2024
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The NFS Provisioner service has now been installed.

A storage class named 'lscdocker' has now been created
and is available to provision dynamic volumes.
```

```
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl apply --context kind-astachniak-cluster --filename=pvc.yaml
persistentvolumeclaim/lsc-lab5-pvc created
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl apply --context kind-astachniak-cluster --filename=deployment.yaml
deployment.apps/nginx-deployment created
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl apply --context kind-astachniak-cluster --filename=service.yaml
service/nginx-service created
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl apply --context kind-astachniak-cluster --filename=job.yaml
job.batch/copy-content-job created
```

```
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl get services
NAME                      TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)
                                      AGE
kubernetes                ClusterIP   10.96.0.1       <none>        443/TCP
                                      29m
nfs-server-provisioner    ClusterIP   10.96.173.23    <none>        2049/TCP,2049/UDP,32803/TCP,32803/UDP,20048/TCP,20048/UDP,875/TCP,
875/UDP,111/TCP,111/UDP,662/TCP,662/UDP   15m
nginx-service             NodePort    10.96.159.191   <none>        80:30000/TCP
                                      9m12s
PS C:\Users\olas0\OneDrive\Pulpit\aws-cli-docker> kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
Handling connection for 8080
```



Hello from Kubernetes!