Kubernetes Course 2025

INSTALL

```
git clone https://github.com/niklaushirt/training.git
cd training
```

https://hub.docker.com

Container

BUILD

```
cd /Users/nhirt/TEMP/training
cd demo-app/k8sdemo/
bat ./Dockerfile
```

```
podman build -t myimage:1.0.0 .
podman images
```

PODMAN

```
podman context ls

podman --connection podman-machine-default-root run \
    --rm \
    --privileged \
    --detach \
    --name portainer \
    -v portainer_data:/data \
    -v /var/run/docker.sock:/var/run/docker.sock \
    -p 9443:9443 \
    -p 8000:8000 \
    portainer/portainer-ce
```

https://localhost:9443

```
podman --connection podman-machine-default-root kill portaine
podman --connection podman-machine-default-root volume rm por
```

RUN

```
podman run --rm -ti myimage:1.0.0 /bin/bash
uname -a
```

```
podman run --rm -ti -e MYVAR=foo -p 8080:3000 myimage:1.0.0
```

http://localhost:8080

```
podman ps [-a]

podman stop/kill <CONTAINER_ID>
```

REGISTRY

```
podman login docker.io

podman tag myimage:1.0.0 docker.io/niklaushirt/myimage:1.0.0

podman images
podman push docker.io/niklaushirt/myimage:1.0.0

podman pull docker.io/niklaushirt/myimage:latest
```

https://hub.docker.com/u/niklaushirt

```
podman build --platform linux/arm64/v8 --platform linux/amd64
podman manifest inspect myimage-mf|yq -P
```

K8s

```
cd /Users/nhirt/TEMP/training

kubectl get nodes
```

RUNNING

```
kubectl run nginx-demo --image=docker.io/nginx
kubectl get pods
```

```
bat ./deployment/pod.yaml
kubectl create -f ./deployment/pod.yaml
kubectl apply -f ./deployment/pod.yaml

kubectl get pods
```

```
kubectl get pods nginx -o yaml|yq
```

```
kubectl get pods
echo "-----"
kubectl get pods -l release=april2023
```

SERVICES

```
kubectl create -f ./deployment/service.yaml
```

```
kubectl port-forward --namespace default pod/nginx 8000:80
```

http://localhost:8000

DEPLOYMENT

```
bat ./deployment/deployment.yaml

kubectl apply -f ./deployment/deployment.yaml

kubectl get pods
kubectl get deployments
kubectl port-forward --namespace default deployment/nginx-dep
```

http://localhost:8000

```
kubectl port-forward --namespace default service/nginx-servic
```

CONFIGURATION

```
bat ./deployment/deployment-cm.yaml

kubectl apply -f ./deployment/deployment-cm.yaml
```

APPLIED

https://niklaushirt.github.io/k8s*training*web/04-kubernetes-basics/04-kubernetes-basics-03

```
bat ./deployment/demoapp-backend-mac.yaml

bat ./deployment/demoapp-mac.yaml
```

```
kubectl create -f ./deployment/demoapp-backend-mac.yaml
kubectl create -f ./deployment/demoapp-backend-service.yaml

kubectl create -f ./deployment/demoapp-mac.yaml
kubectl create -f ./deployment/demoapp-service.yaml
kubectl port-forward --namespace default service/k8sdemo-serv
```

http://localhost:3000

STORAGE

```
kubectl get StorageClasses
```

```
bat ./volumes/1-simple-mysql-pv.yaml
kubectl apply -f ./volumes/1-simple-mysql-pv.yaml
kubectl get pv
kubectl describe pv mysql-pv-volume
```

```
bat ./volumes/2-simple-mysql-pvc.yaml
kubectl apply -f ./volumes/2-simple-mysql-pvc.yaml
kubectl get pvc
```

```
bat ./volumes/3-simple-mysql-deployment.yaml
kubectl apply -f ./volumes/3-simple-mysql-deployment.yaml
kubectl get deployment
```

```
k9s

mysql -ppassword

CREATE DATABASE testing;
show databases;
exit;
```

ADVANCED

```
kubectl apply -f ./deployment/deployment-READY.yaml
kubectl apply -f ./deployment/deployment-LIVENESS.yaml
kubectl apply -f ./deployment/deployment-OOM.yaml
kubectl apply -f ./deployment/deployment-crashloop.yaml
kubectl apply -f ./deployment/deployment-imagepull.yaml
kubectl apply -f ./deployment/deployment-imagepull-sa.yaml
```

LIMITS

```
bat ./deployment/deployment-READY.yaml
bat ./deployment/deployment-LIVENESS.yaml
```

PROBES

```
bat ./deployment/deployment-OOM.yaml
```

RBAC

https://niklaushirt.github.io/k8s*training*web/06-security/06-security-04

```
rakkess resource pods --namespace default
```

https://github.com/alcideio/rbac-tool/releases

```
rbac-tool analysis -o table
rbac-tool viz --include-namespaces default --include-pods-onl

rbac-tool lookup -e '^kubeadm:.*'

rbac-tool lookup -e '.*demo.*'
```

```
bat ./deployment/critical.yaml
kubectl apply -f ./deployment/critical.yaml
```

SECURITY

```
conftest test -p ./conftest/src/examples/kubernetes/policy ./
```

```
kubescape list controls

kubescape scan

kubescape scan control C-0067,C-0066,C-0035,C-0012,C-0057,C-0
```

```
kubescape list frameworks

kubescape scan framework DevOpsBest
kubescape scan framework DevOpsBest -v
kubescape scan framework nsa -v
kubescape scan framework mitre -v
kubescape scan framework cis-v1.23-t1.0.1 -v -f html> result.
open result.html
```

```
# Minimize wildcard use in Roles
kubescape scan control C-0187 -v

# Privileged container
kubescape scan control C-0057 -v

# Ensure that the cluster-admin role is only used where requi
kubescape scan control C-0185 -v

# Workload with administrative roles
kubescape scan control C-0272 -v

# Applications credentials in configuration file
kubescape scan control C-0012 -v
```

DEBUG

```
bat ./deployment/deployment-crashloop.yaml
bat ./deployment/deployment-imagepull.yaml
bat ./deployment/deployment-imagepull-sa.yaml
```

CLEAN-UP

```
kubectl delete -f ./deployment/deployment-crashloop.yaml
kubectl delete -f ./deployment/deployment-imagepull.yaml
kubectl delete -f ./deployment/deployment-imagepull-sa.yaml
kubectl delete -f ./deployment/deployment-OOM.yaml
kubectl delete -f ./deployment/deployment-READY.yaml
kubectl delete -f ./deployment/deployment-LIVENESS.yaml


kubectl delete -f ./volumes/3-simple-mysql-deployment.yaml
kubectl delete -f ./volumes/2-simple-mysql-pvc.yaml
kubectl delete -f ./volumes/1-simple-mysql-pv.yaml
kubectl delete -f ./deployment/demoapp-backend-mac.yaml
kubectl delete -f ./deployment/demoapp-backend-service.yaml
kubectl delete -f ./deployment/demoapp-mac.yaml
kubectl delete -f ./deployment/demoapp-service.yaml
kubectl delete -f ./deployment/deployment.yaml
kubectl delete -f ./deployment/deployment-cm.yaml
kubectl delete -f ./deployment/pod.yaml
kubectl delete -f ./deployment/service.yaml
kubectl delete -f ./deployment/critical.yaml

kubectl delete pods nginx-demo
kubectl delete pods nginx
podman --connection podman-machine-default-root kill portaine
```

```
kubectl debug $(kubectl get po -n default -l app=crashloop --no-headers|
\awk '{print $1;}') -it --copy-to=my-debugger --container=mycontainer -- sh
```

```
kubectl debug $(kubectl get po -n default -l app=crashloop --no-headers|
\awk '{print $1;}') -it --image=busybox kubectl debug $(kubectl get po -n
default -l app=crashloop --no-headers| \awk '{print $1;}') -it --image=busybox
--copy-to=my-debugger
```

```
kubectl debug $(kubectl get po -n default -l app=crashloop --no-headers|
\awk '{print $1;}') -it --image=busybox
```