Course 1 - 10 Cloud / Kubernetes

# K8 Architecture - Step by Step

Good references:

<https://www.redhat.com/en/topics/containers/kubernetes-architecture>

<https://medium.com/google-cloud/kubernetes-101-pods-nodes-containers-and-clusters-c1509e409e16>

<https://phoenixnap.com/kb/understanding-kubernetes-architecture-diagrams>

<https://kubernetes.io/docs/concepts/workloads/pods/>

\<https://www.cncf.io/the-childrens-illustrated-guide-to-kubernetes/>

Experimenting with K8

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

# Step 0 - Design a service that will run on multiple machines (cluster)

Service should stay up, even when a few machines are down

## Step 1 - master and worker



## Step 2 - Control Proxies

How can master communicate with workers nodes?



## Step 3 - How To Applications

We run them as docker containers



## Step 4 - Grouping / Managing Applications

Containers are grouped into PODs



## Step 5 - Networking

Containers communicate with each other using kube-proxy



## Step 6 - More Control Services

Scheduler - obvious

Etcd - distributed database that stores the state of cluster (source of truth)



## FInal Service Deployment



# Lab 9.10 - Deploying Docker via K8 (updated instructions)

$ git clone https://github.com/Anuj1990/Docker.git

$ cd Docker

$ docker build . -t dockerapp

$ kubectl get pods

$ sudo cp /etc/kubernetes/admin.conf $HOME/

$ sudo chown $(id -u):$(id -g) $HOME/admin.conf

$ export KUBECONFIG=$HOME/admin.conf

$ kubectl run dockerapp --image=dockerapp --port=80

$ kubectl get pods

$ kubectl get deployment

$ kubectl create deployment mydeploy --image dockerapp

$ kubectl get deployment

$ kubectl get services

$ kubectl expose deployments/mydeploy --port=80 --target-port=80 --type=NodePort

$ kubectl get services

# diagnosing error

$ kubectl describe pod dockerapp

# How to Reset K8

# reset

$ sudo kubeadm reset

# delete config

$ sudo rm -rf ~/.kube

# reboot to be absolutely sure

$ sudo reboot