Day 23 Revisit (Docker)

* Docker is a DevOps Tool mainly used for containerizing the application.
* Container = Source Code + Related Libraries + Environment to Run
* Virtual Machine vs Docker
* Dual Booting, Hyperviser
* WSL – Windows Sub System for Linux
* Installing Docker [Enable WSL, Hyperviser, Virtualization Platform] – Turn windows feature on or off.
* Downloaded Docker Desktop from official website.
* Docker shares the same kernel across the containers.
* Docker containers are light weight alternative to VM (Virtual Machine)
* Docker Architecture (Docker Client, Docker Engine, Docker Registry)
* Docker Client (CLI [Command Prompt or terminal] & GUI [docker Desktop])
* Docker Engine (Docker Deamon[dockerd], Docker Images, Containers)
* Docker Hub (It’s a cloud version to store and mange all the docker images) – Similar to mvnrepository, npmrepository and github
* Pulled docker image from official site (hello-world)
* Images – Class, Container – Running version of image.
* Docker build, run, compose, images, ps, rmi, rm
* Creating docker image from scratch – demo (docker-java)
* Dockerfile – steps to create docker image. FROM, WORKDIR, COPY, RUN, PORT
* Host & Guest OS
* Used MySQL docker image (demo)

Day 24 Agenda

* Introduction to Kubernetes
* Introduction to container orchestration
* Installing Minikube
* Container Orchestration
* Cluster, Control Pane, Nodes & Pods
* Replication & Scaling
* deploying a spring boot application with k8s

Kubernetes – K8S (I18N)

K8S is an open-source system for automating deployment, scaling & managing containerized applications.

K8S – Container Orchestration Tool (One of the DevOps tool)

* Never Outgrow (Scaling)
* Runs Anywhere

**POD** - This object indicates the processes which are running in the cluster. (Small & simplest unit)

**Node**

A **node** is nothing but a single host, which is used to run the virtual or physical machines. A node in the Kubernetes cluster is also known as a minion.

**Service**

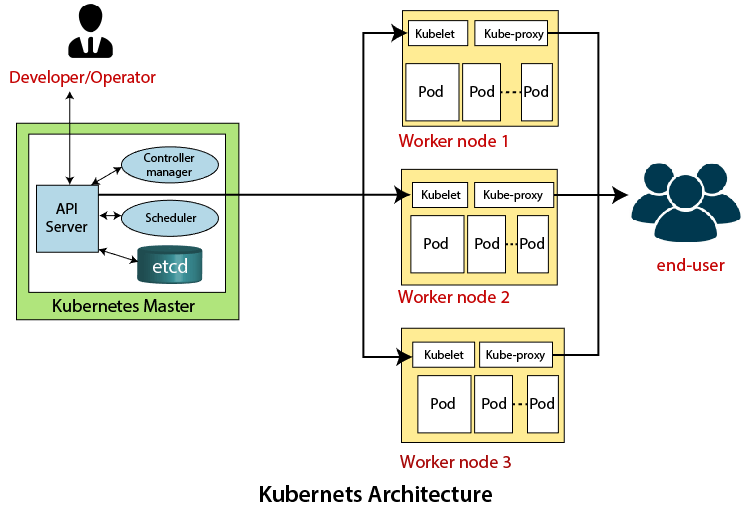
A **service** in a Kubernetes is a logical set of pods, which works together. With the help of services, users can easily manage load balancing configurations.

**ReplicaSet**

A **ReplicaSet** in the Kubernetes is used to identify the particular number of pod replicas are running at a given time. It replaces the replication controller because it is more powerful and allows a user to use the "set-based" label selector

K8S Arch

* Master Node [Control Plane] (API Server, etcd, scheduler, ctrler mgr)
* Worker Node (slave) – many in number [kubelet, kube proxy (k-proxy), Pods]



[**https://kubernetes.io/docs/home/**](https://kubernetes.io/docs/home/)

<https://labs.play-with-k8s.com/>

<https://kubernetes.io/docs/tutorials/kubernetes-basics/>

Installing K8S

* Minikube
* Kubeadm (k8s Administrator)
* Kubectl (k8s controller)

Install Kubectl using curl command (kubectl is a cluster mgmt. command line tool)

Kubectl allows you to run commands against Kubernetes clusters. You can use kubectl to deploy applications, inspect and manage cluster resources, and view logs.

curl.exe -LO “<https://dl.k8s.io/release/v1.28.4/bin/windows/amd64/kubectl.exe>”

Installing minikube

<https://codelabs.developers.google.com/codelabs/cloud-springboot-kubernetes#0>

<https://levioconsulting.com/insights/dockerize-a-spring-boot-application-with-tomcat/>