Kubernetes

- A container orchestration tool hosted by cloud native computing foundation (CNCF), developed by Google

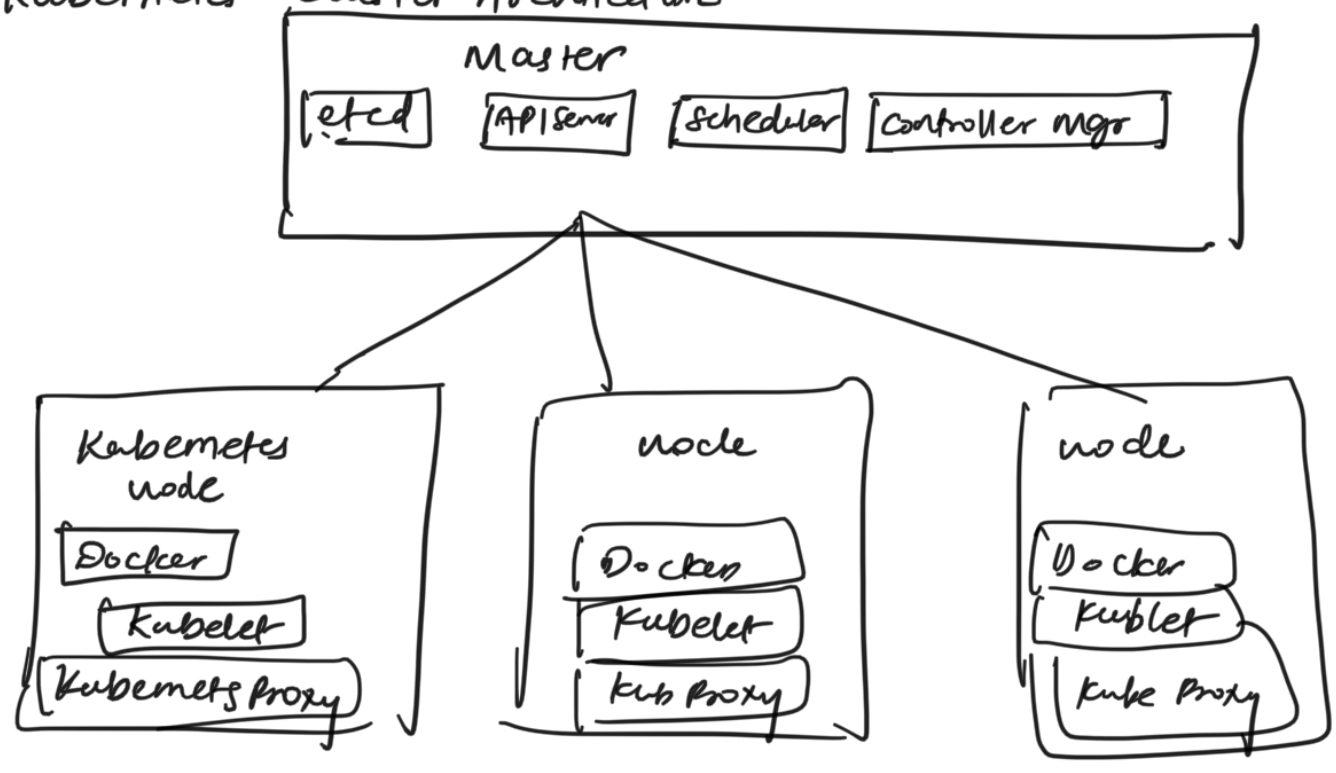
Features
- CICD
- containerized infrastructure
- Application centric management
- Auto scalable infrastructure
- Env consistency across dev, test, prod
- Loosely coupled infrastructure.
- High density of resource utilization

- It can run application on clusters of physical and virtual machine infrastructure
- Can also run apps on cloud.

What do orchestration tools offer?
1. High Availability
2. Scalability
3. Disaster Recovery

Kubelet - kubernetes process that makes it possible for the cluster to talk/communicate with each other and execute some tasks on those nodes

Kubernetes - Cluster Architecture



etcd: kubs backed key value store
- accessible to everyone
- may have sensitive info.

API Server: Entrypoint to K8S Cluster
- provides all operations on the cluster.

Controller Manager : Keeps track of what is happening in the cluster
- collect & send info to API server. continuously.
- make changes to bring current status to desired state.

Scheduler - ensures pod placement
- distribute workload.
- track utilization of working load on cluster nodes.

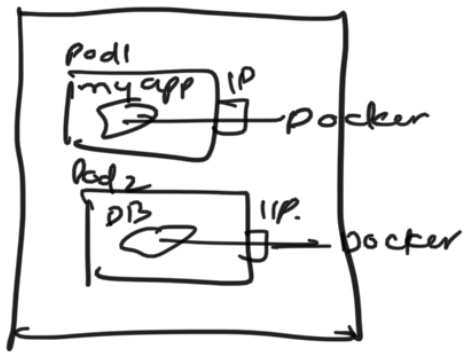
Node Components
Docker

Kubelet - for communication
- interacts with etcd to read config details
etcd interact with master to recieved commands & work.
- manage network rules, port forwarding.

K8S Proxy - make service available to external host
- load balancing.
- make sure networking env is predictable
- and accessible as well as isolated.
- manages pods on node, volume, secrets, health check, create new containers etc.

We can have multiple replicas of Master in case one fails.

Pod:
- smallest unit of K8S
- Abstraction over container
- Usually 1 application per pod
- Each pod has its own IP address.
- New IP address on recreation



Service
- a permanent IP address connected to a pod.
- helps in load balaneing & scaling.
- REST object in K8S. whose definition can be posted to K8S api server on master to create new instance.
- a logical set of pods.

eg.
```
apiVersion: v1
kind: Service
metadata:
    name: Tutorial
```

```
spec:
  ports:
    -port : 8080
     targetPort : 31999
```

Types of service

- ClusterIP
  - restrict service within cluster

NodePort
expose to static port on deployed node
clusterIP is automatically created.

Load Balancer
It uses cloud providers load balancer.
NodePort & ClusterIP are created automatically to which external lb will route

Configmap: map of configuration
- external configuration of your application
  eg DB_URL : mongo-db-service
- You don't have to build new images everytime the db cert is changed
- Dont put credentials here.

Secrets.
- used to store secret data
- base 64 encoded

Volumes
- Attaches your physical storage to the pod

Deployment
- blueprint for my-app pods.
- You create Deployments
- can scale up/down
- if one pod is not available we use other replica.
- DB cant be replicated using deployment

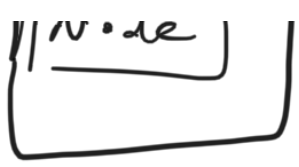
Stateful set
- meant for databases.
- to maintain consistency in db

DB's are often hosted outside K8S cluster

Ingress - route traffic to cluster.

Minikube
- creates virtualbox on laptop
- Node runs in that Virtualbox

- 1 Node K8S cluster
- for testing purposes.



Kubectl.

- a way to interact with cluster
- a command line tool
- API Server → UI, API, CLI (Kubectl)

Installation

brew update
brew install hyperkit
brew install minikube (will install kubectl automatically)

minikube start --vm-driver=hyperkit
(which hypervisor to use.)

kubectl get nodes → get status of nodes.

minikube status
kubectl version → Client & Server version.
kubectl get pods
kubectl get services

Deployment - abstraction over pods

kubectl create deployment NAME --image=image
[--dry-run] [options]

kubectl create deployment nginx-depl --image=nginx
kubectl get deployment
kubectl get pod
kubectl logs [podname]
kubectl describe pod [podname]
kubectl exec -it podname --bin/bash
kubectl apply -f configfile.yaml → add all options, image, name in configfile & use it

kubectl apply -f secrets.yaml.

nodePort 30000 - 32767.

clusterIP - Internal
LoadBalance - External.