In master node only kuberentes commads will work. I.e., no pod in worker node so no Kubernetes commands only docker commands.

In Kubernetes we didn’t use kubeadm becoz we are using cloud , i.e., elastic ips load balancers.

Kubectl get nodes 🡪 shows nodes and status

Kubectl describe node masternode

Systemctl restart kubelet

free -m

vi /var/log/ messages -- log file

escape+shift g

**task:**

1. how to access remote Kubernetes cluster from my laptop?
2. How to create pod without deployment?

kubectl run podname –image=podname.

kubectl run httpdpod –image=httpd

1. How to provide access to Kubernetes for a particular user/ normal user?
2. How to create a container in Kubernetes?

Unlike docker containers cannot be created in k8 .

In Kubernetes containers are created in pod ( pods are created by process called deployment(manages pods when down)

Kubectl create deployment deploymentname –image=application name

Kubectl create deployment vamsi –image=httpd

kubectl get deployment

kubectl get pods

kubectl delete pod podname

kubectl delete deployment deploymentname

kubectl get pods -o wide

**How to login to workernode from master node?**

For that we need to open one port 10250 in worker nodes

Then type command in master node

kubectl exec -it podname (from get pods) -- /bin/bash

Now we logged into pod that means we login to container.

Exit

kubectl cluster-info

Lec 67:

**Tasks**:

How to host my application without port number? With ip address?

How to reduce image size? Methods?

Can I maintain multiple masters?

How to add new worker node, delete worker node?

If a master node down ?what will happen? Will my application work?

**Commands** **list**

Kubectl create deployment dep1 –image=httpd

Kubectl expose deployment dep1 –type=NodePort –port=80 –target-port=80

There are 3 types of services

Cluster ip

Nodeport

Load balancer

Service : means method to expose the application to the world. Creates a virtual logical machine

Kubectl get services

Deleting a service🡺 kubectl delete servicename(httpd1)

**Node port:**

In node port type we should know worker node ip address ,if there are number of worker nodes how should we get. That’s the disadvantage.

Generally we don’t use node port type. We use load balancer method. But in kubeadm loadbalancer method is not supported.

**How to access a service / how to expose a application?**

kubectl expose deployment httpd1 --type=NodePort --port=80 --target-port=80

unlike docker (docker has 2 ports 1. Container port and 2 host port) Kubernetes has 3 ports involved

1)target port = pod port

if we open pod port it will open to container also.

2) port= service port

3) node port = by default it assigns between 30000-32700 ( any one port) no need to give while exposing service

After that kubectl get services edit security groups and add workernode port in workernode in ec2

Browser🡪 workernodeip:workernode port

**Task :**

How to open application with only ip not port number included?

Default port for browser = 80

I.e., how to redirect any port to port number 80? In docker / Kubernetes

How to create my website without port number

**Lecture 68:**

**Task:**

I have one master and multiple worker if I have problem with master can I make a worker node to master? How?

**Load balancer method**

Kubectl expose deploymet httpd2 –type=LoadBalancer --port=80 –target-port=80

Kubectl get services

Kubectl describe service servicename(httpd2)

Actually load balancer should create an external ip but it always shows pending becoz it requires additional configuration. There are multiple ways for that ( ex installing cloud manager ) as it was long process we avoid using for clouds. That is why when we go with cloud we avoid using kubeadm instead we use kops method.

Kubectl describe service servicename(httpd2)