**Cluster creation using kops --> (1 master , 2 worker nodes)**

for single or one node cluster creation minikube is used...

for multi node cluster creation kubeadm and kops is used...

I have used kops for cluster creation.

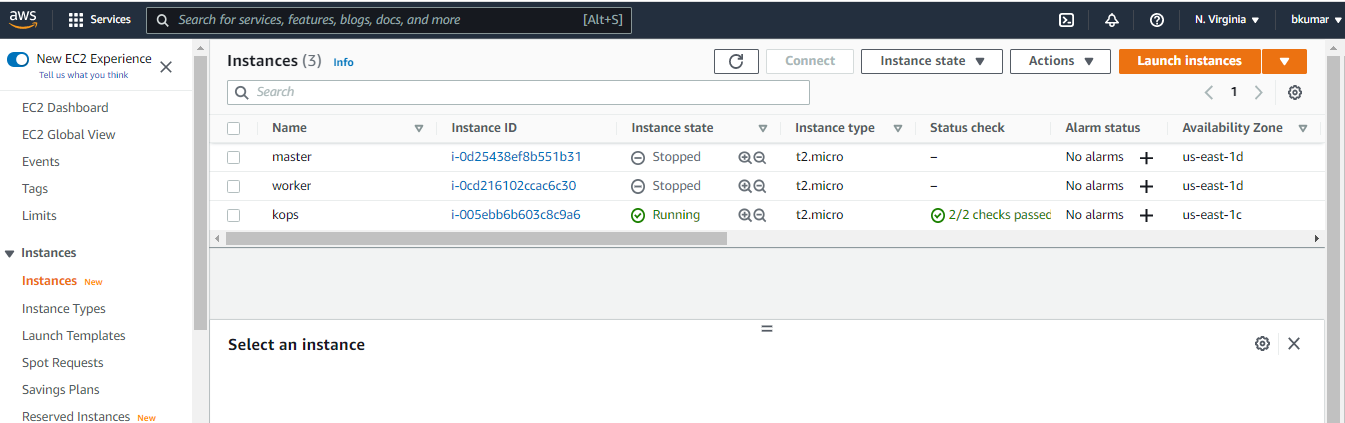
**Pre-requisites -->**

----> Domain for k8s dns records --> e.g-> kops.mustangtech.xyz.

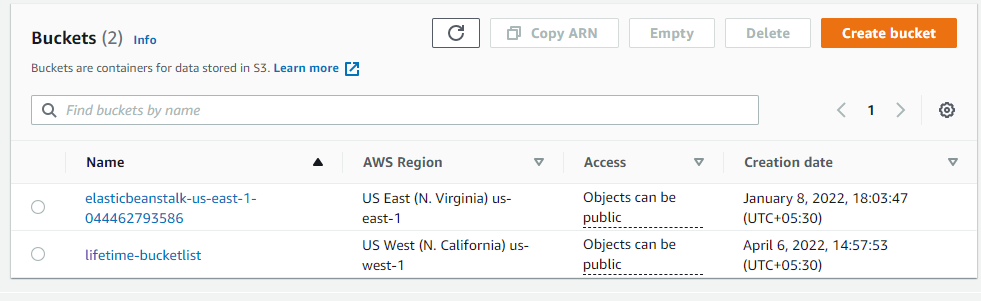
----> create a linux and setup --> kops,kubectl,sshkeys,awscli.

----> login onto aws console --> s3 bucket , IAM user for aws cli , route53 for dns hosted zones.

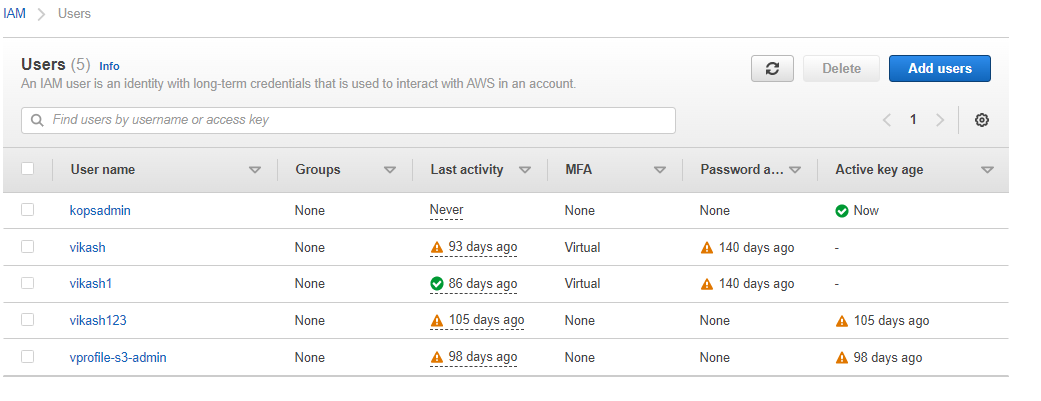
Create a ec2 instance --> ubuntu 20.04 ami



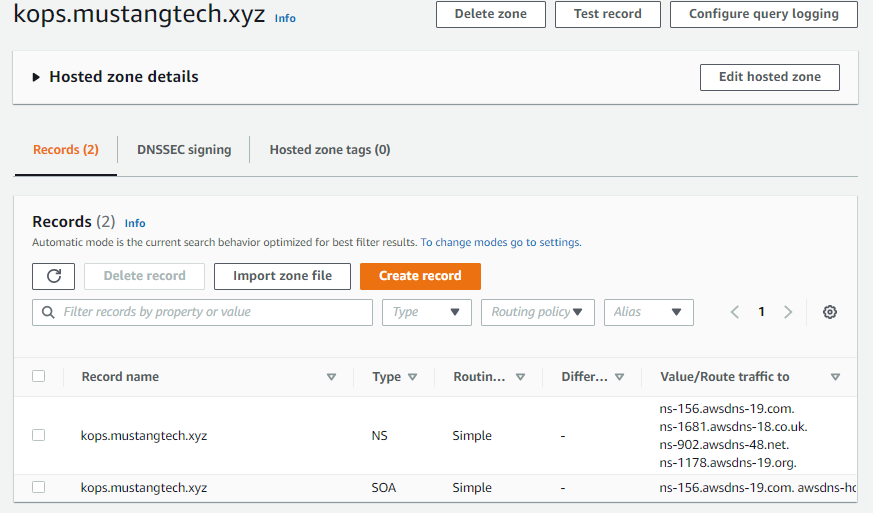
Create a s3 bucket to keep the state file --> bucketname --> lifetime-bucketlist



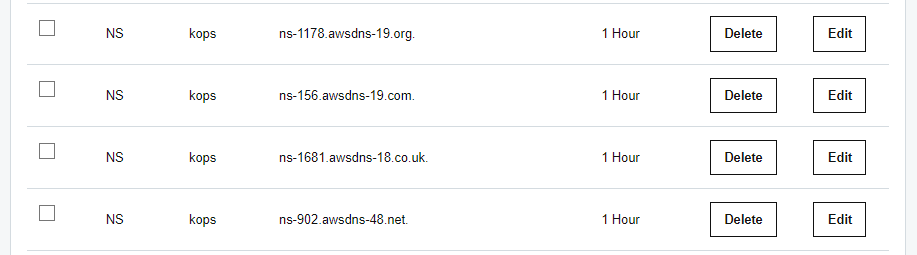
Create a IAM user - kopsadmin -->



Create a dns hosted zones --> kops.mustangtech.xyz



add or route this traffic to godaddy account --> dns records -->



Now login to ec2 instance -->

we have to generate the ssh key by **ssh-keygen** command so, that it will automatically push the public key to the worker nodes instances.

install awscli -->

--> sudo apt update && sudo apt update awscli -y.

Now, aws configure --> provide the required credentials.

Now install kubectl and kops from k8s docx -->

kubectl -->

https://kubernetes.io/docs/tasks/tools/

executable permission - sudo chmod +x ./kubectl

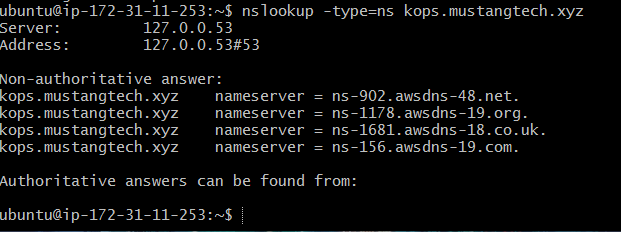
sudo mv kubectl /usr/local/bin

kops -->

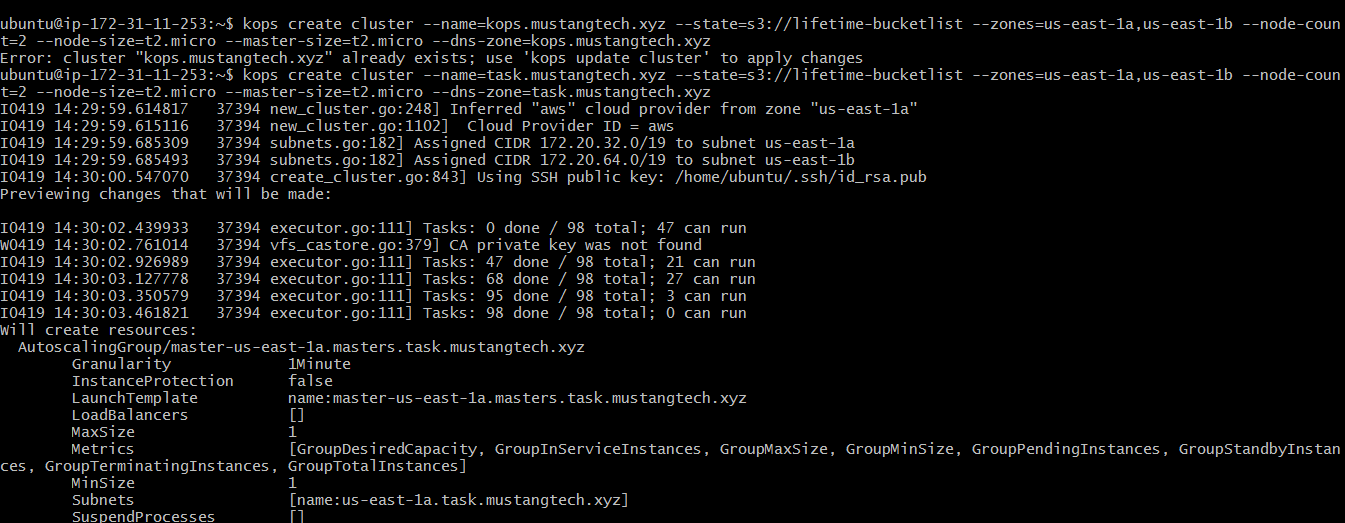
https://kubernetes.io/docs/setup/production-environment/tools/kops/

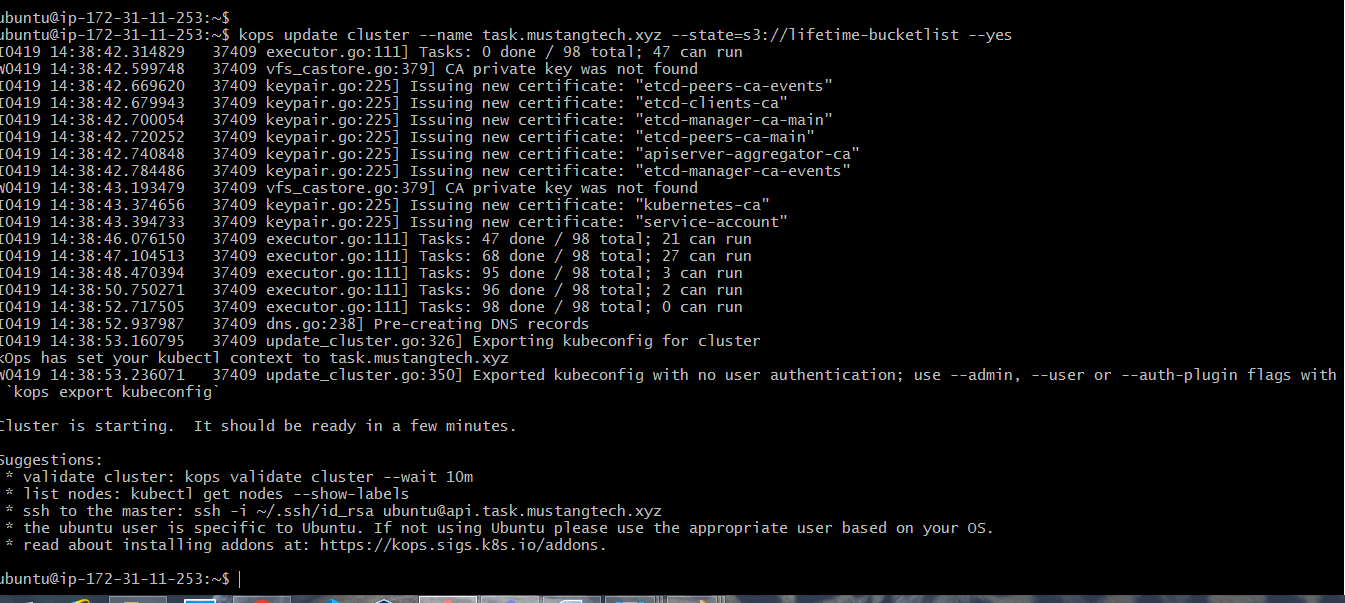
executable permission - sudo chmod +x kops

sudo mv kops /usr/local/bin

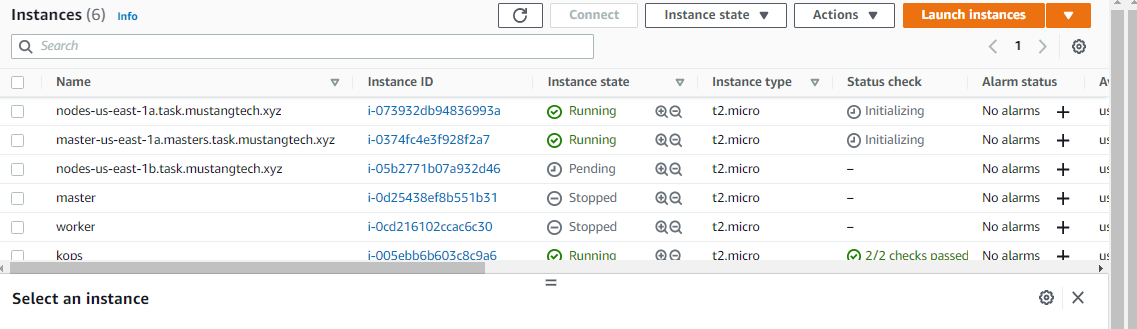


Now create cluster using kops command -->

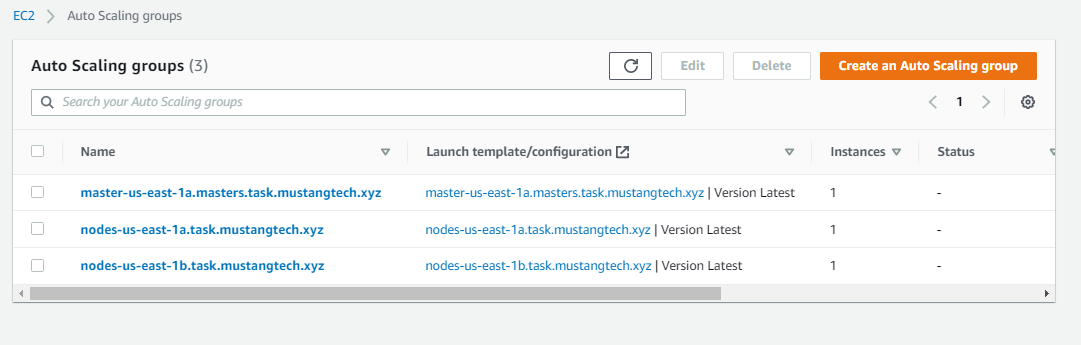




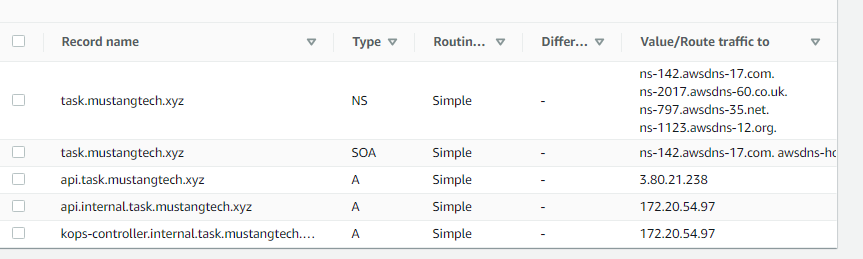
master -1 , node -2 has been created -->



followed by autoscalling groups -->

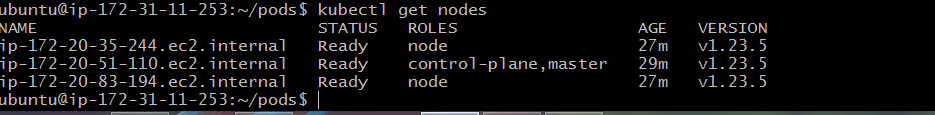


route53 dns hosted zones with new entries -->



commands -->

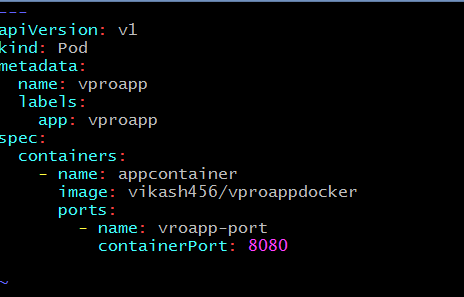
kubectl get nodes



Create pod by putting contents inside the file -->

creating pod from image which is present in my docker hub registry.

vim vproapp.yaml



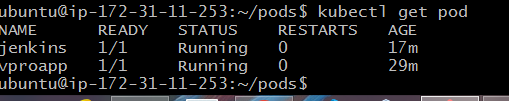
know ,

kubectl create -f vproapp.yaml

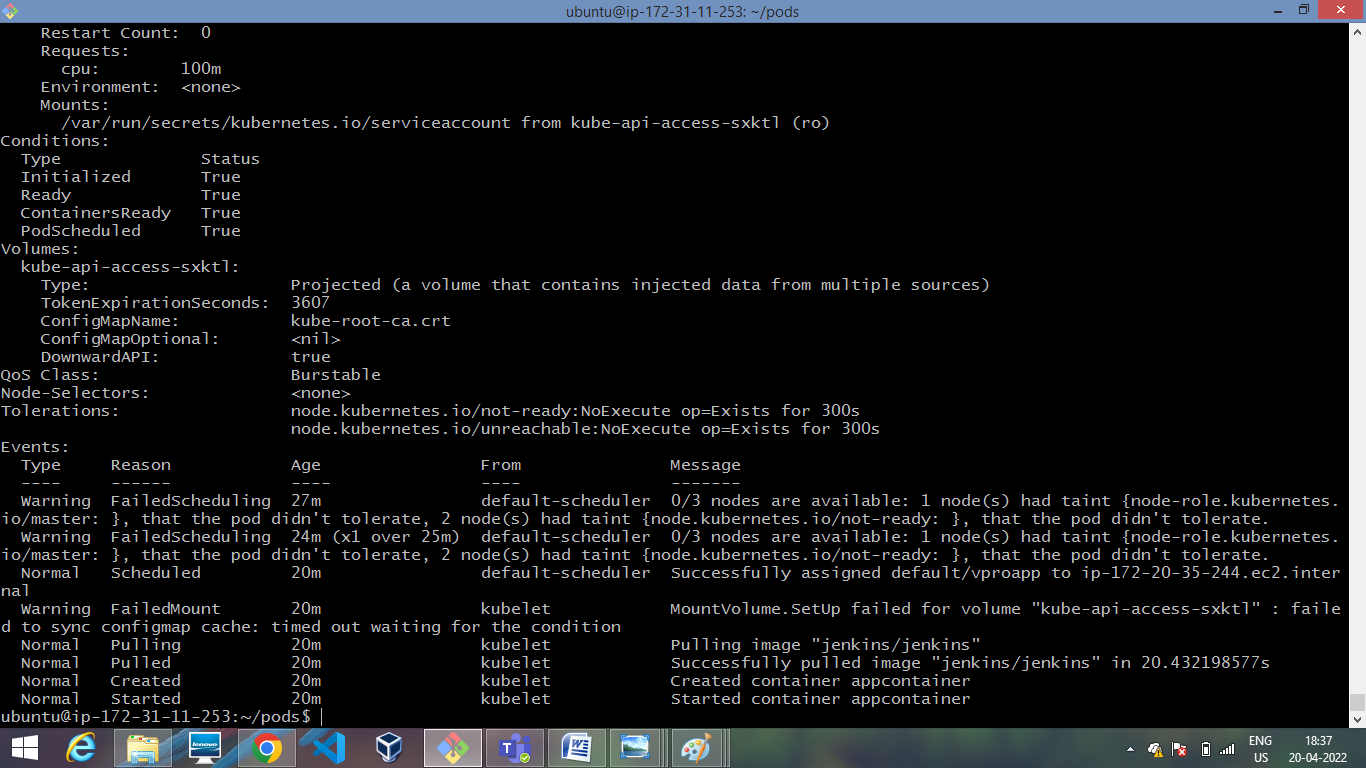


for extra information of the pod we need to give the following command-->

kubectl get pod

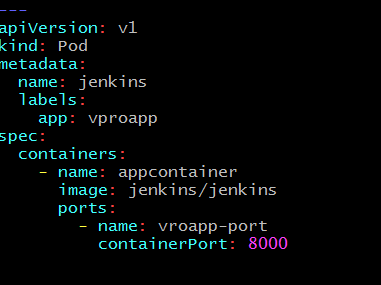


kubectl describe pod vproapp (addational info of the pod)



Creating pod from jenkins official image from docker hub registry-->

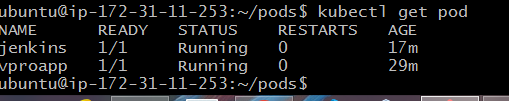
vim defination.yaml



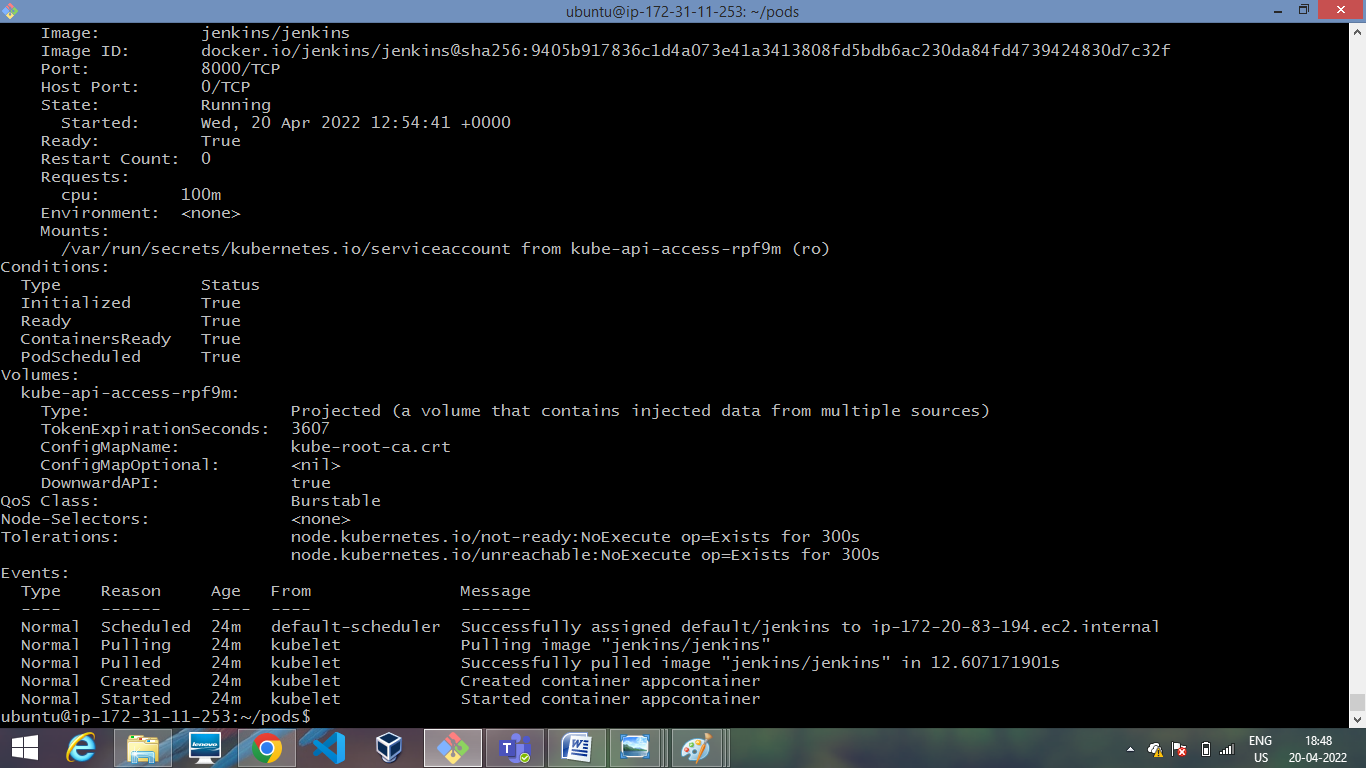
kubectl create -f defination.yaml



kubectl get pod

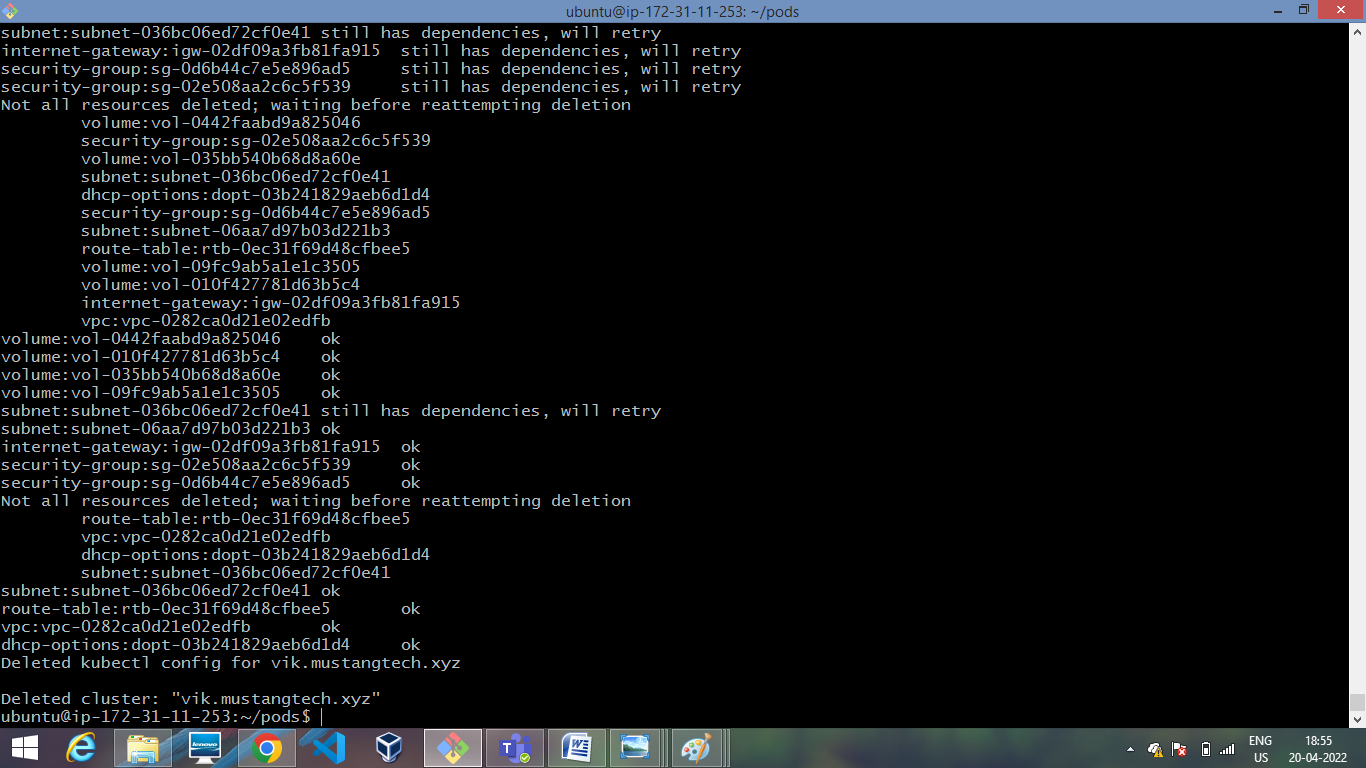


kubectl describe pod jenkins (addational info of the pod)



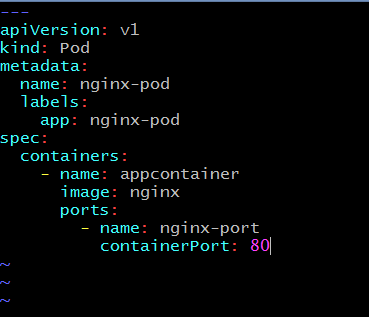
command for deleting the cluster -->

kops delete cluster --name=vik.mustangtech.xyz --state=s3://lifetime-bucketlist --yes

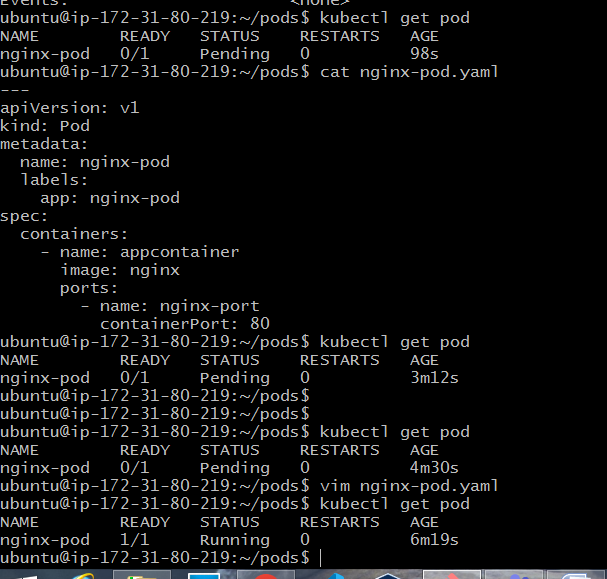


Created a nginx pod -->

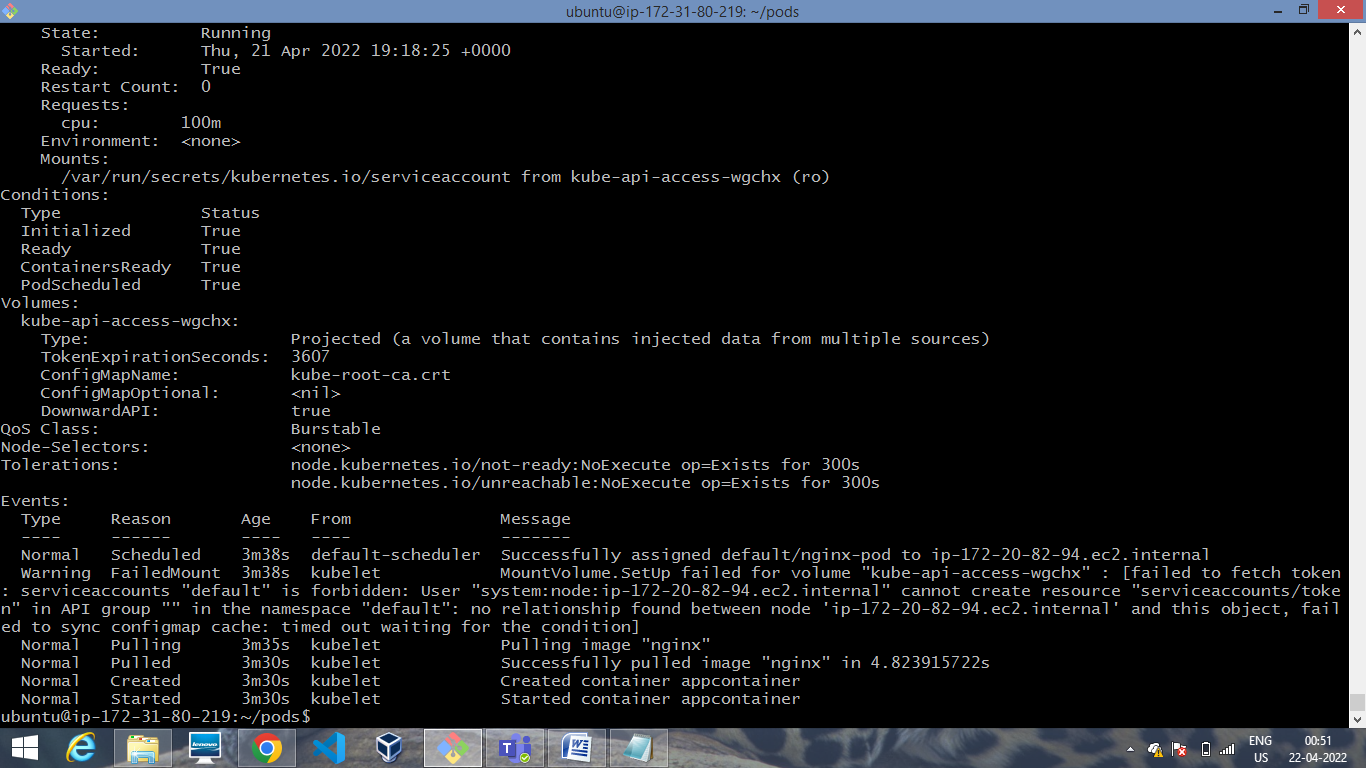
vim nginx-pod.yaml



kubectl get pod

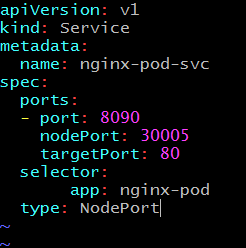


kubectl describe pod nginx-pod

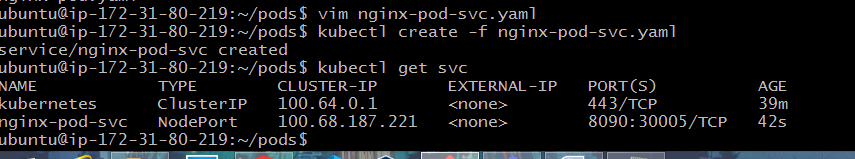


Now , create service file for nginx pod so it will be accessed by users -->

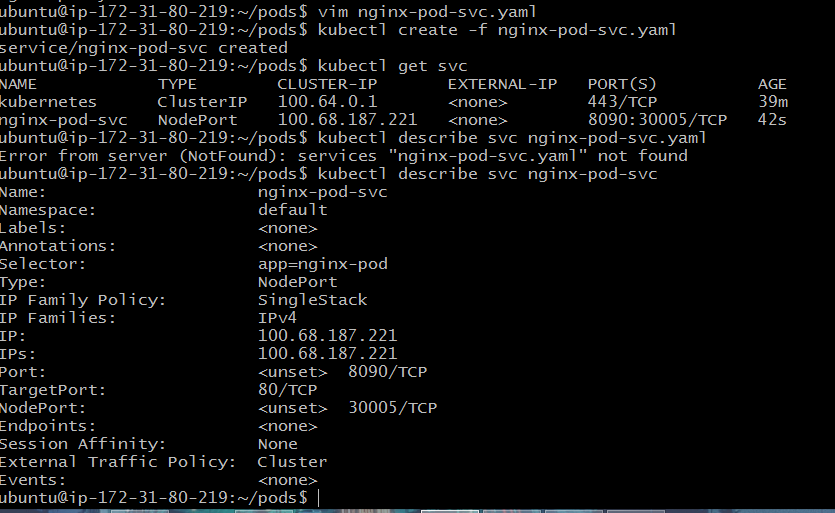
vim nginx-pod-svc.yaml



kubectl create -f nginx-pod-svc and kubectl get svc

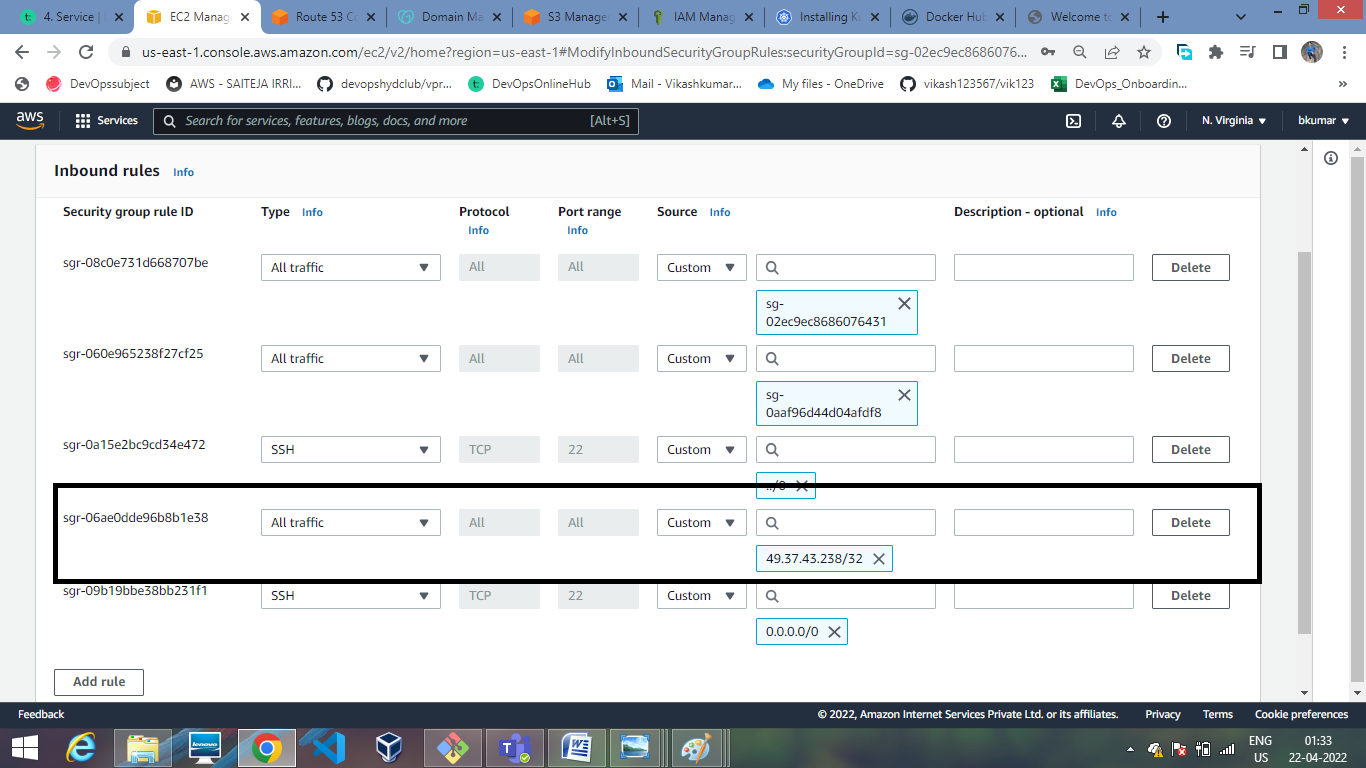


kubectl describe svc nginx-pod-svc

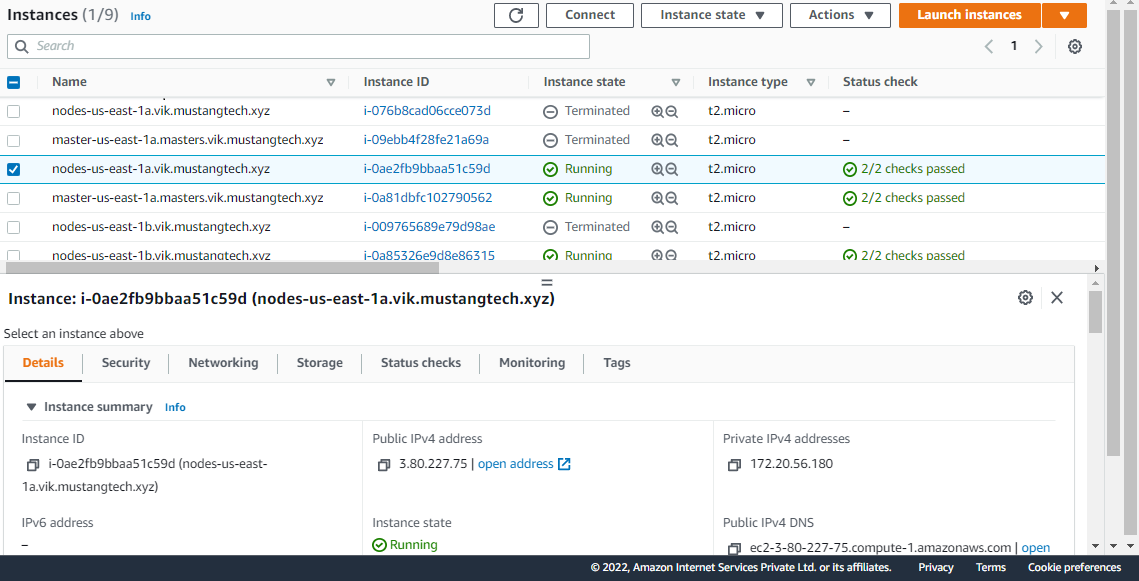


Now by taking any nodes security group --> because the pod which we have created will be running on all the nodes i.e, worker and master nodes.

allow , all traffic -- from myip



Now take public ip of worker node with the node port which we have taken (30005) in nginx-pod-svc file -->



3.80.227.75:30005 in terminal

