To check the created pods in which node it is created

Steps to create

Step 1:

1. Create the server with ubuntu
    - Select the ubuntu
    - Select the t2 micro
    - Increase the storage to 25
2. Change to the root user
3. Update the server apt update -y

Step 2:

1. Install the docker packages by using commands

sudo apt install curl wget apt-transport-https -y
sudo curl -fsSL https://get.docker.com -o get-docker.sh
chmod 777 get-docker.sh
sh get-docker.sh

2. Start the docker by command systemctl start docker
3. Check the status of docker by command systemctl status docker

Step 3:

1. Install the kubectl packages

sudo curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
chmod +x kubectl

2. It can provide the executable permissions to the kubectl file

Step 4:

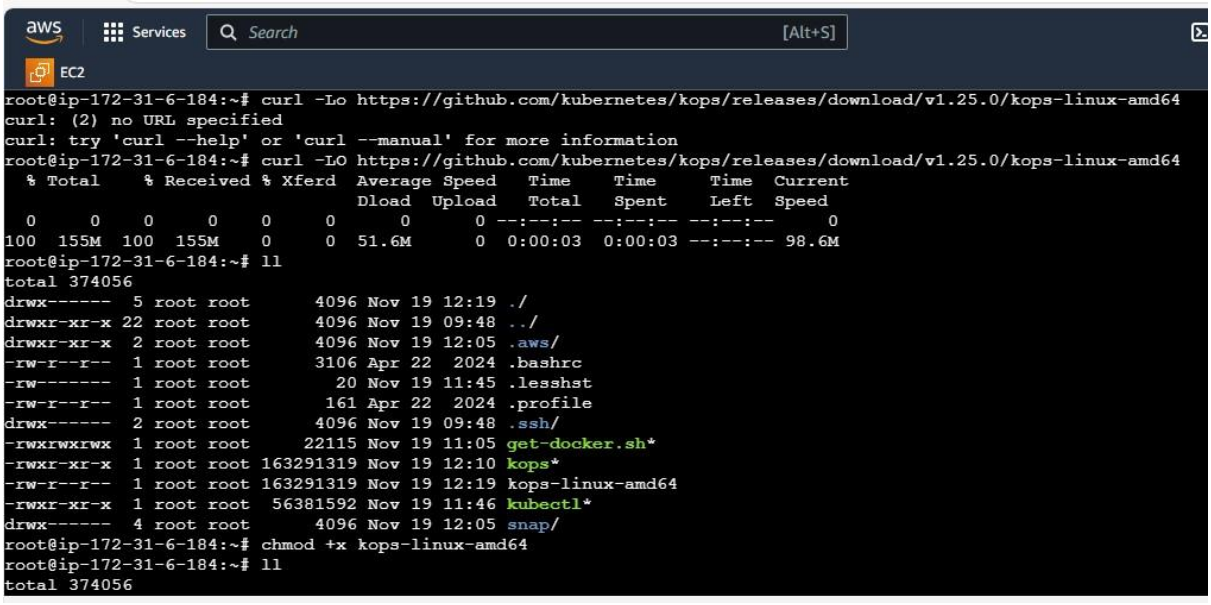
1. Give **aws configure**
2. Give snap info **aws-cli**
3. Give snap install **aws-cli --channel=v1/stable --classic** to install the aws cli
4. Create user in iam and give access permissions to the user
5. Provide access key and secret key in cli

Step 5:

1. To install the kops by command is

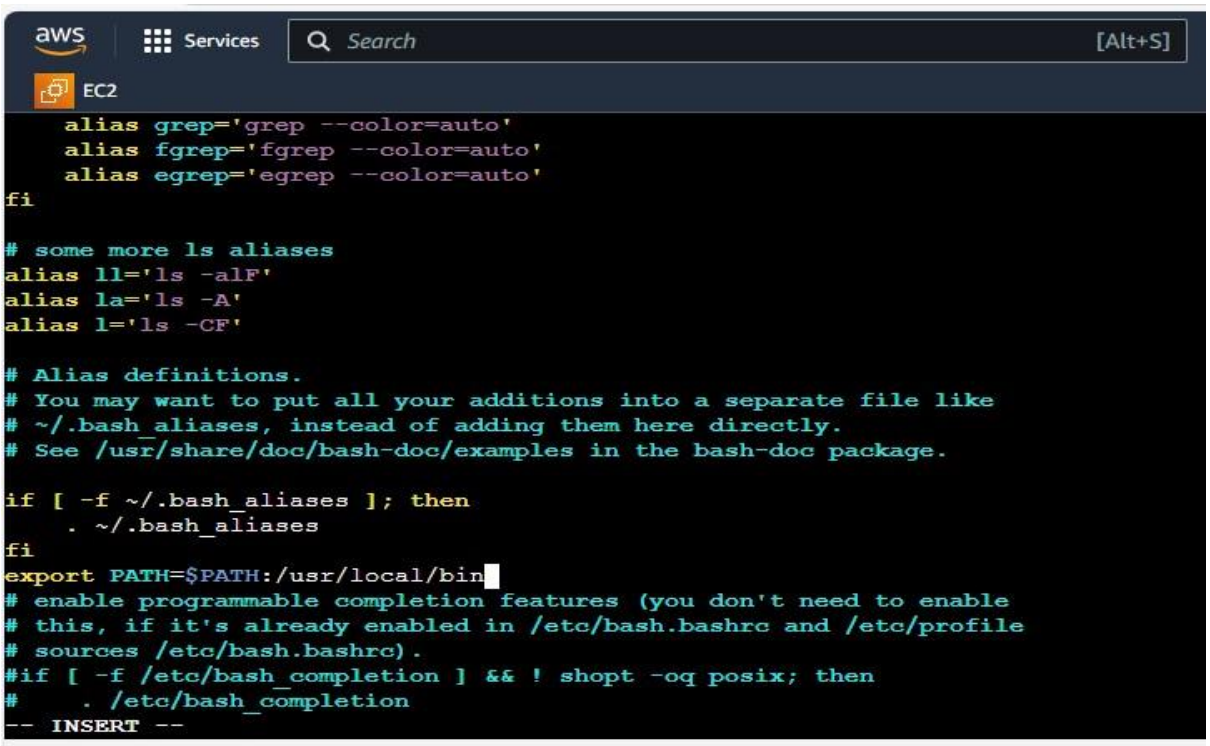
curl -LO https://github.com/kubernetes/kops/releases/download/v1.25.0/kops-linux-amd64
chmod +x kops-linux-amd64
mv kops-linux-amd64 /usr/local/bin/kops
mv kubectl /usr/local/bin/kubectl

2. After performing all the commands in terminal as shown in below figure



3. Give ll -a to list the files

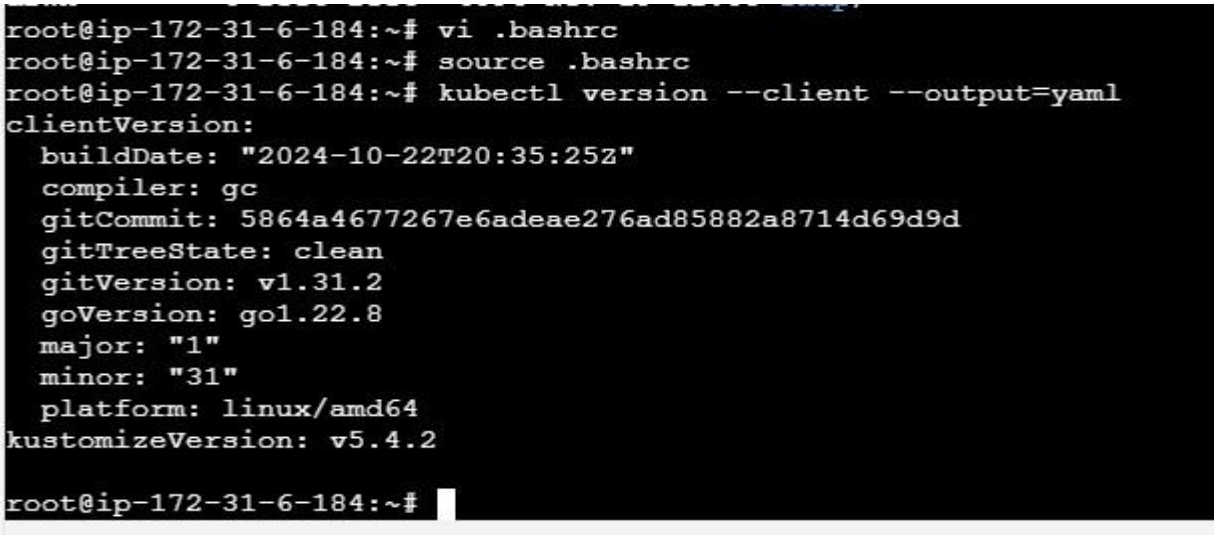
4. Go inside of the file <u>vi .bashrc</u> to add the content in file



5. Save and quit from the editor

6. Type **source .bashrc** command

7. Type **kubectl version --client --output=yaml** it will show the version in yaml file format as shown in below figure



Step 6:

1. Create the s3 bucket from cli and store the entire cluster information in the created bucket
2. Create the bucket from cli by the command

```
aws s3api create-bucket --bucket mani420 --region us-east-1
```

3. To enable the versioning from cli by through command

```
aws s3api put-bucket-versioning --bucket mani420 --region us-east-1 --versioning-configuration Status=Enabled
```

4. Exporting the bucket in kops by the command

```
export kops_state_store=s3://mani420
```

Step 7:

1. We need to create the cluster
2. We can create the cluster by the command

kops create cluster --name raj.k8s.local --state=s3://mani420 --zones us-east-1a --master-size t2.medium --node-size t2.micro

3. It can create the s3 bucket, autoscaling, load balancer automatically in N.virginia as shown in below figure
4. To edit the node information

kops edit ig --name=raj.k8s.local nodes-us-east-1a --state=s3://mani420

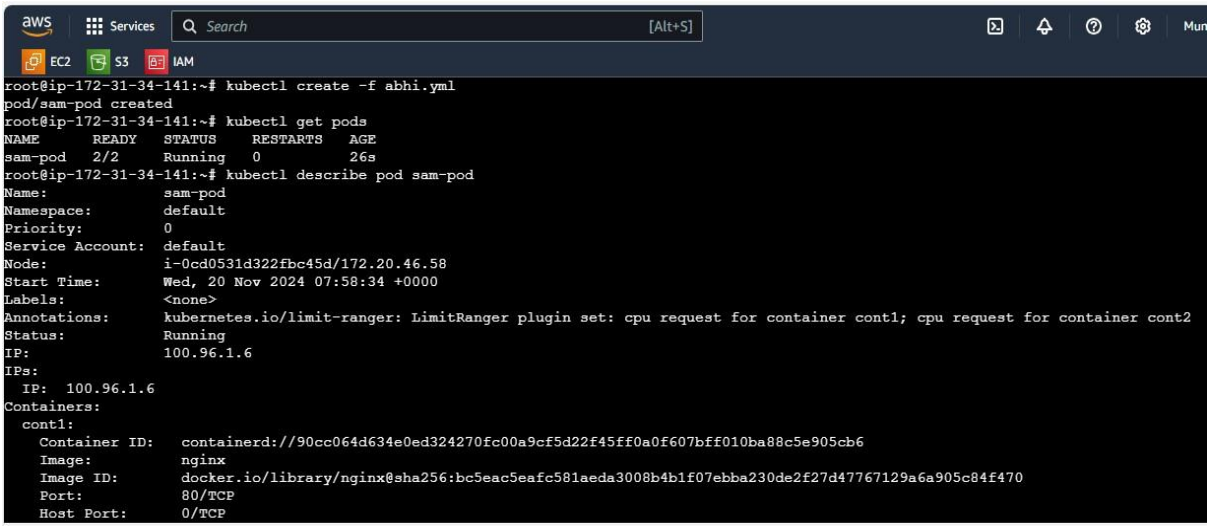
```
Suggestions:
 * list clusters with: kops get cluster
 * edit this cluster with: kops edit cluster raj.k8s.local
 * edit your node instance group: kops edit ig --name=raj.k8s.local nodes-us-east-1a
 * edit your master instance group: kops edit ig --name=raj.k8s.local master-us-east-1a

Finally configure your cluster with: kops update cluster --name raj.k8s.local --yes --admin
```

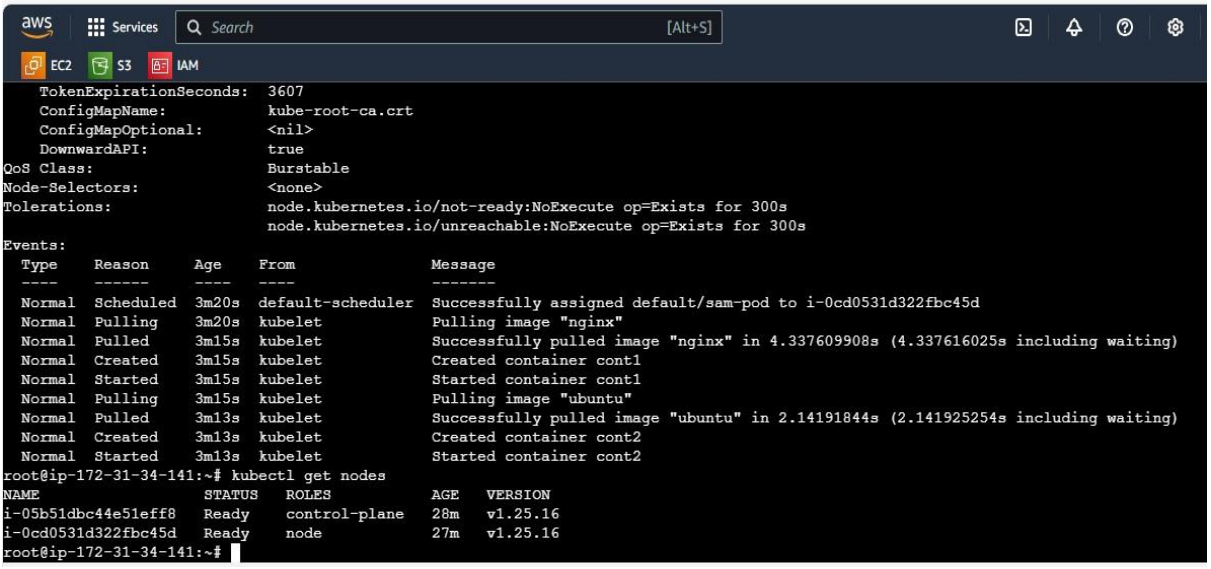
5. We need to change max and min then save and exit
6. If the instances are not created then update the cluster, then the instances, load balancers, autoscaling will be created by the command

kops update cluster --name raj.k8s.local --yes --admin --state=s3://mani420

7. Create yaml file with vi abhi.yml

8. Type **kubectl create -f abhi.yml to create the pods**
9. Type **kubectl describe pods sam-pod** to check on which node pod is created





10. Pod is created in the worker node