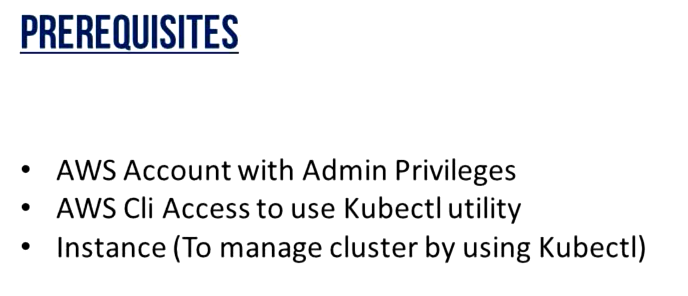
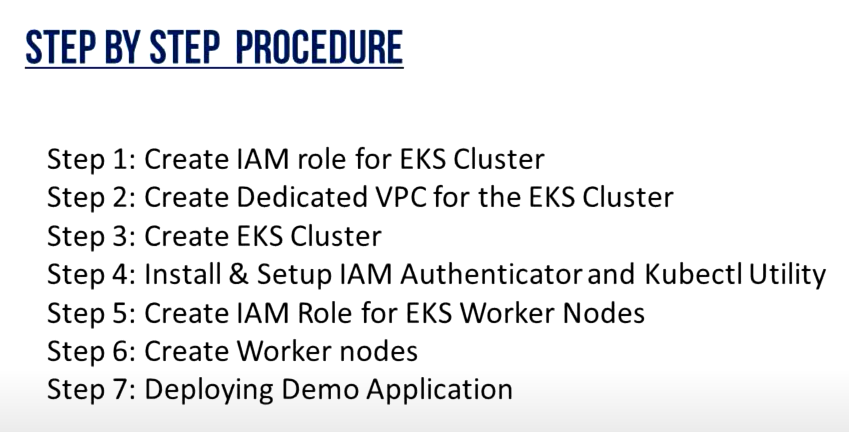
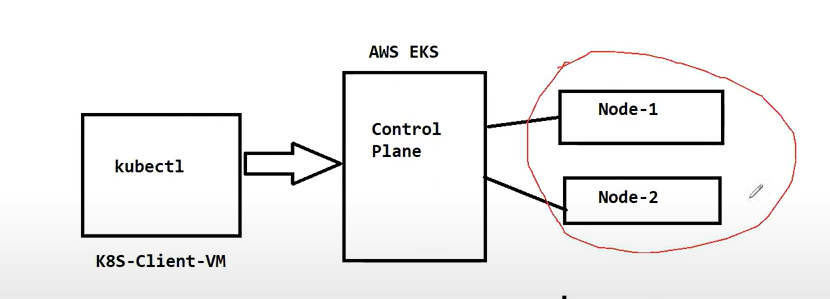
**EKS CLUSTER**

****

**1-4 FOR Master node**

**5-6 for worker nodes**

****

****

**This control plane is provided by AWS , we need to create worker nodes and attach to control plane (EKS).**

**Now I am going to take one more machine, this machine I will be calling as kubernetes client vM, in this Iam going to install kubectl now this client machine I will use to exectute the commands with the control palne. Nodes and client machines we are going to create.**

**Worker Nodes and Client machine we have to install**

**Worker Nodes is attached to EKS**

**Client machine I will install kubectl to execute Kubernetes commands**

**Steps to Create EKS Cluster in AWS**

**Step 1. Create VPC   
using cloud formation template**

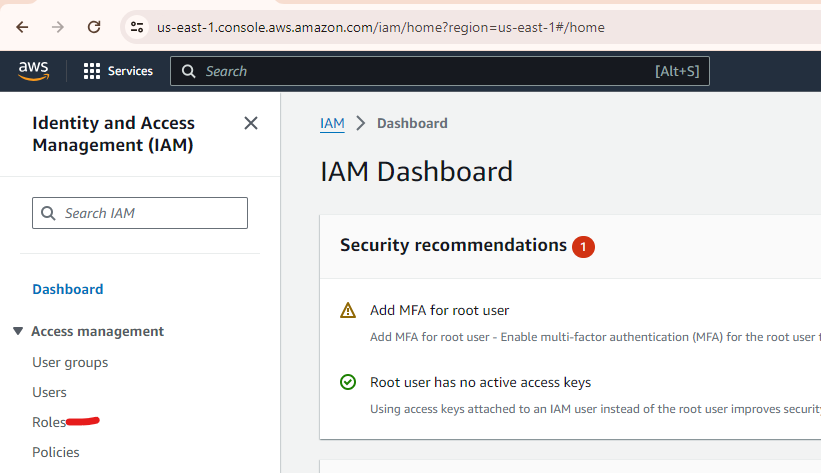
**Search for cloud formation template**

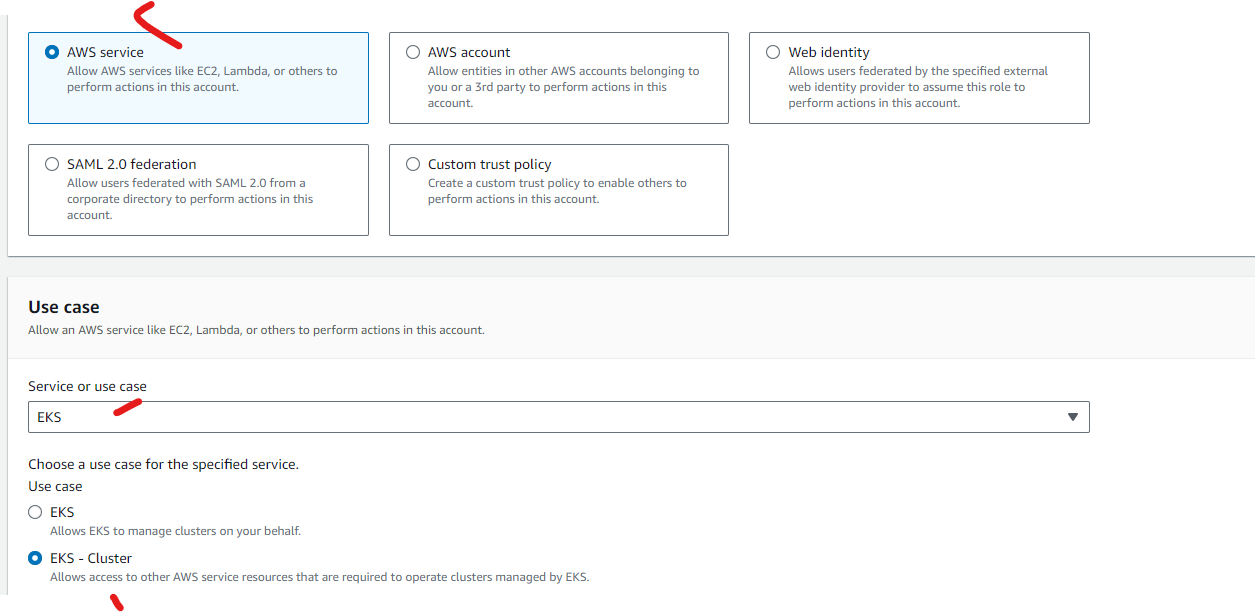
* **Stack and enter below URL**[**https://s3.us-west-2.amazonaws.com/amazon-eks/cloudformation/2020-10-29/amazon-eks-vpc-private-subnets.yaml**](https://s3.us-west-2.amazonaws.com/amazon-eks/cloudformation/2020-10-29/amazon-eks-vpc-private-subnets.yaml)

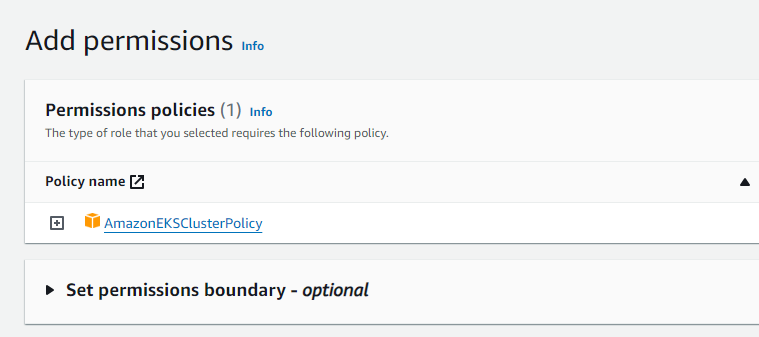
**Step2 : Create IAM role on AWS**

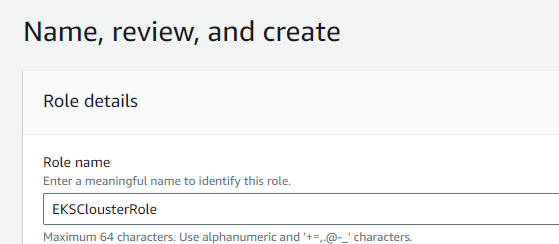
* **Entity type: AWS Service**
* **Select use case as “EKS” 🡺 EKS cluster**
* **Role Name : EKSClusterRole ( you can give any name for the Role)**

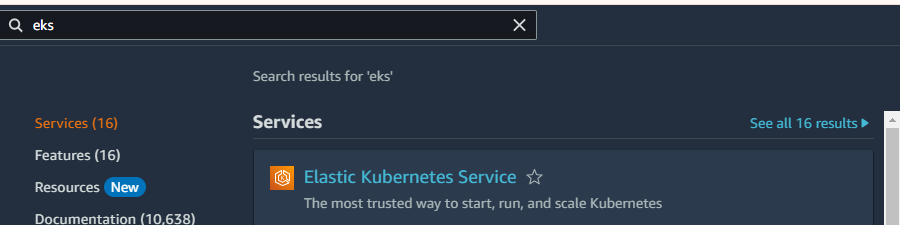
**Navigate to IAM and click on Roles**

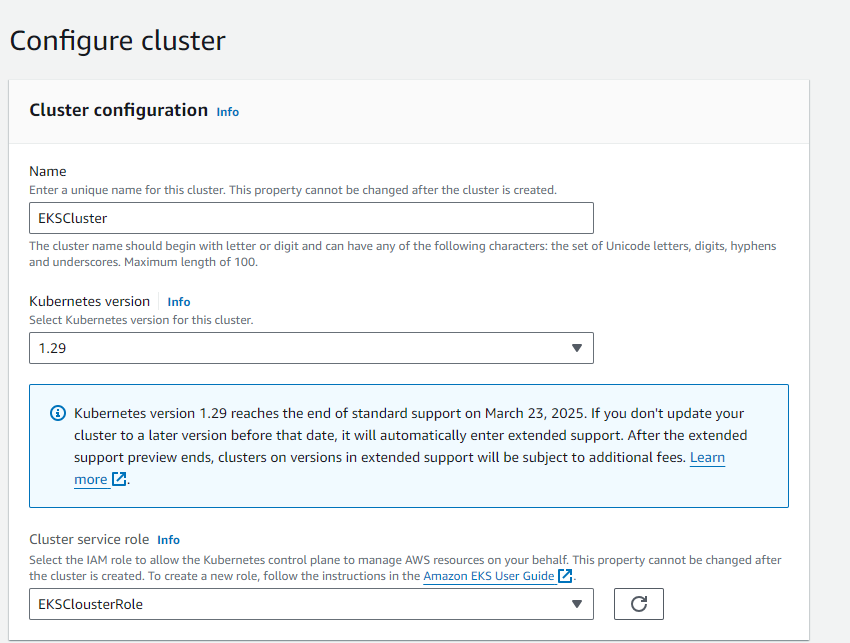
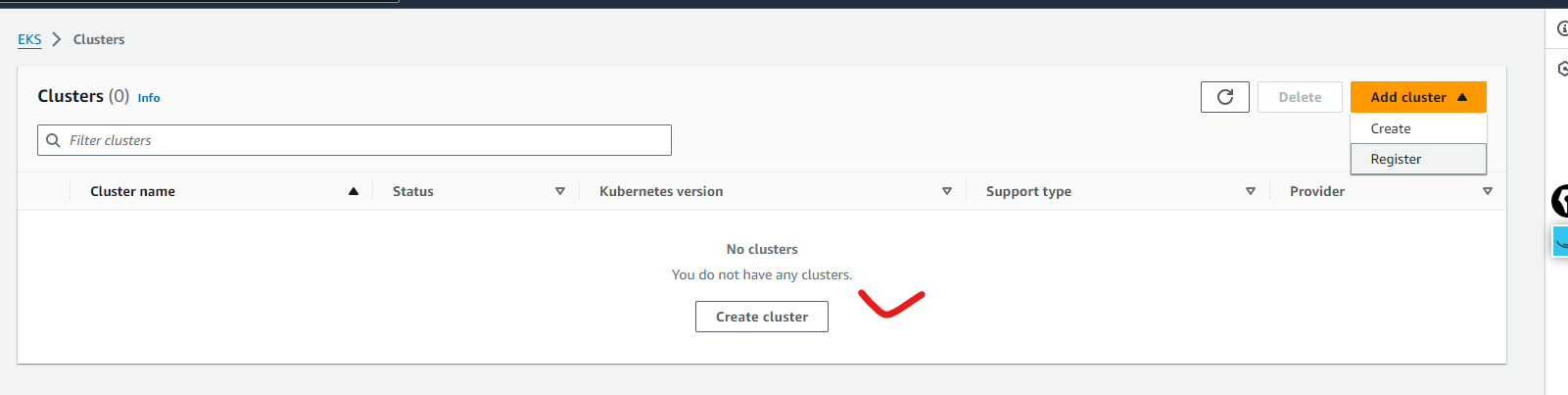
****

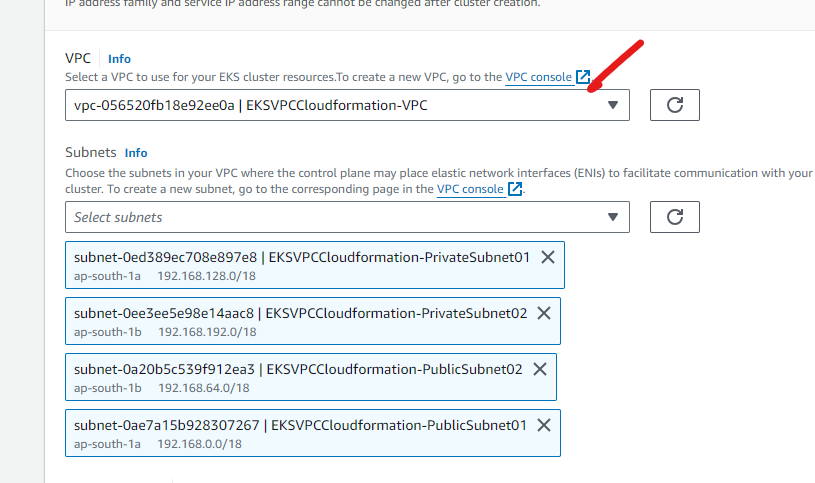
****

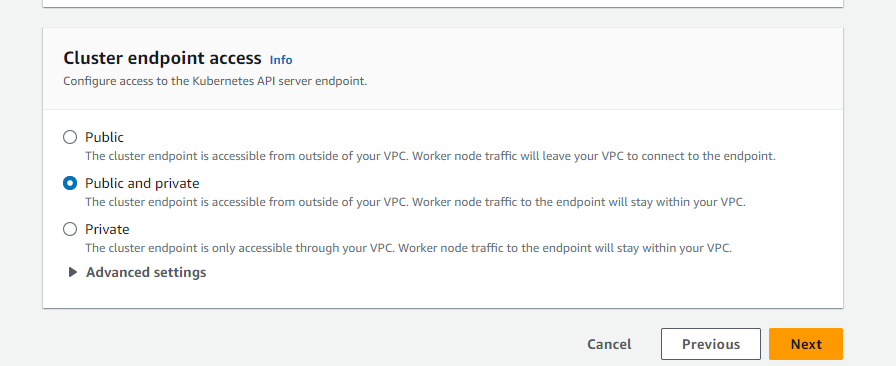
****

****

****

****

****

****

**By setting public and private. Cluster can be access outside vpc and worker nodes can access using private endpoints**

**Next🡺 Next 🡺 Next🡺 Create**

**Now we need AKSclientVM to install kubectl**

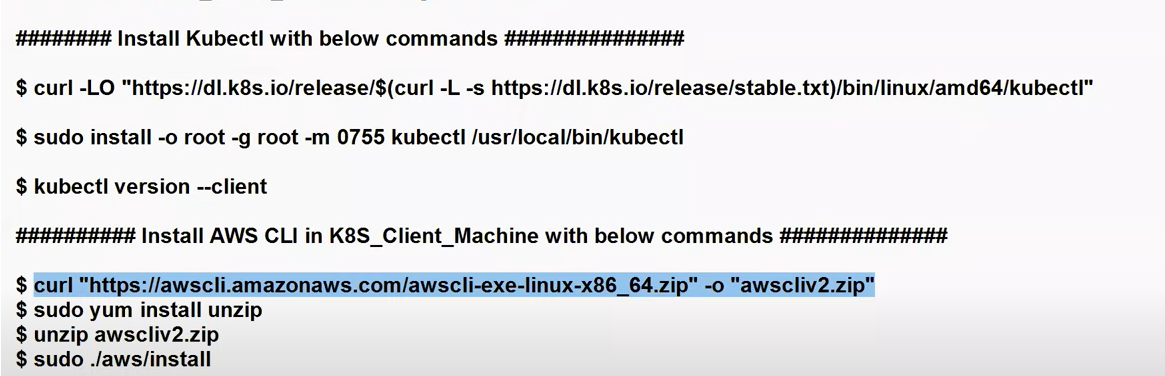
**Login into VM**

[**https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/#install-kubectl-binary-with-curl-on-linux**](https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/#install-kubectl-binary-with-curl-on-linux)

curl -LO [https://dl.k8s.io/release/**$(**curl -L -s https://dl.k8s.io/release/stable.txt**)**/bin/linux/arm64/kubectl](https://dl.k8s.io/release/$(curl%20-L%20-s%20https://dl.k8s.io/release/stable.txt)/bin/linux/arm64/kubectl)

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

kubectl version –client

****

**========== Configure AWS Cli with Credentials ================**

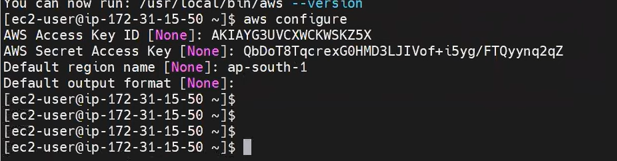
**This is root account security credentials**

**Accesskey: AKIA3MZIKP7VURWCZRNG**

**Secretkey: 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**

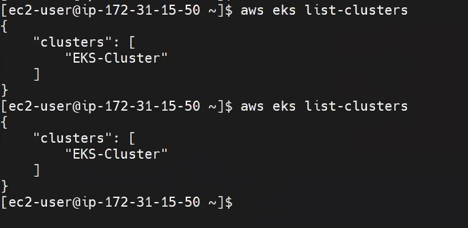
**# aws configure**

**Note: we can use root user accesskey and credentials**

****

**Once Cluster is created you can see what are the clusters are available**

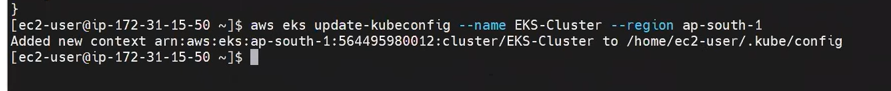
**# aws eks list-clusters**

****

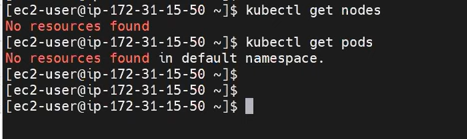
**======== update kubeconfig file in remote machine from cluster using below command ========**

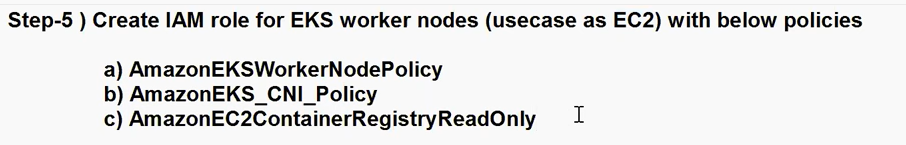
**# aws eks update-kubeconfig –name <cluster-name> --region <region-code>**

**Ex: aws eks update-kubeconfig –name ekscluster –region us-west-1**

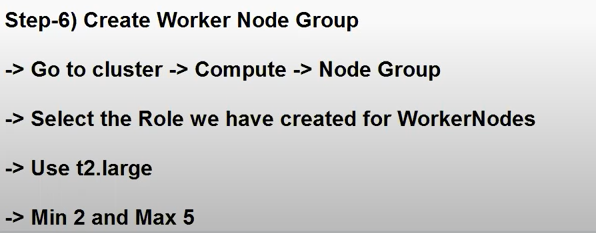
**  
  
# kubectl get nodes**

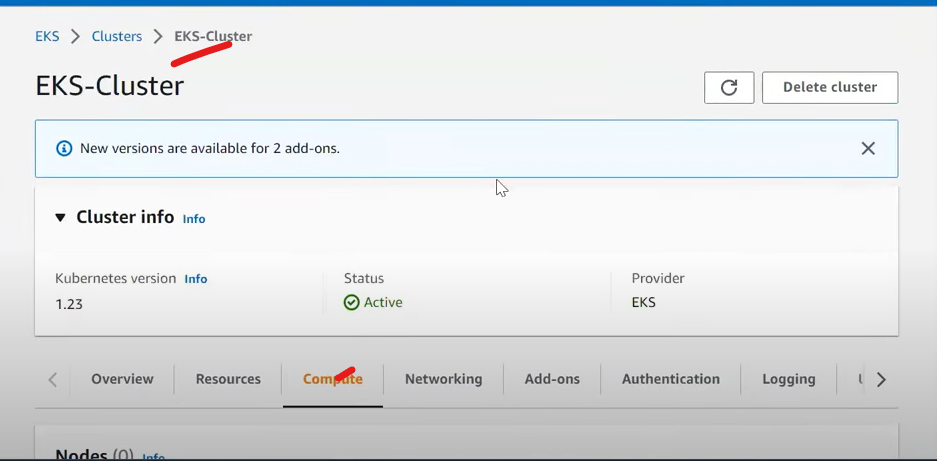
**# kubectl get pods**

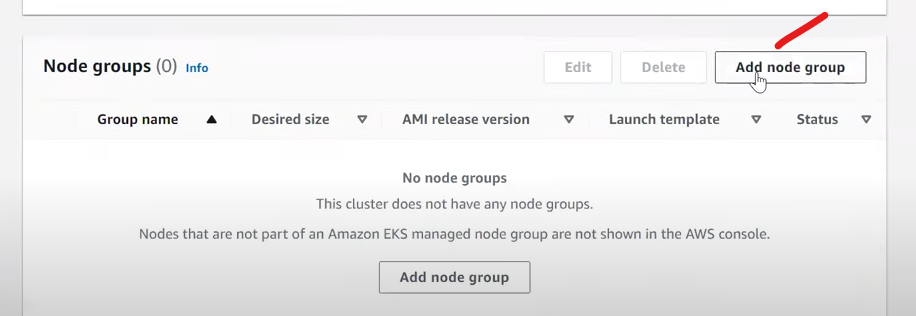
****

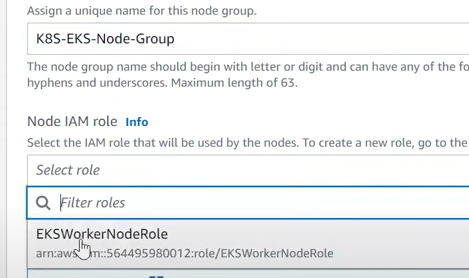
****

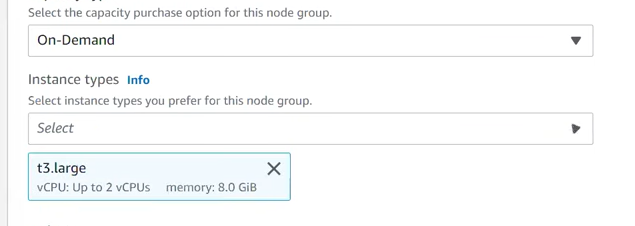
We need to create worker Node group in the EKS cluster 🡺 Navigate to Cluster

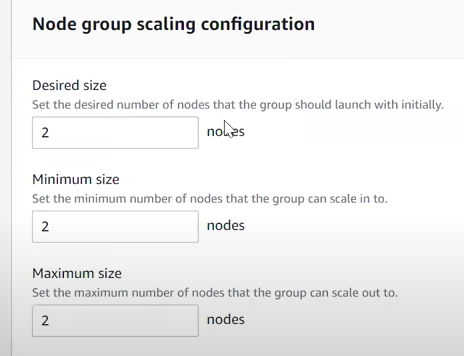
****

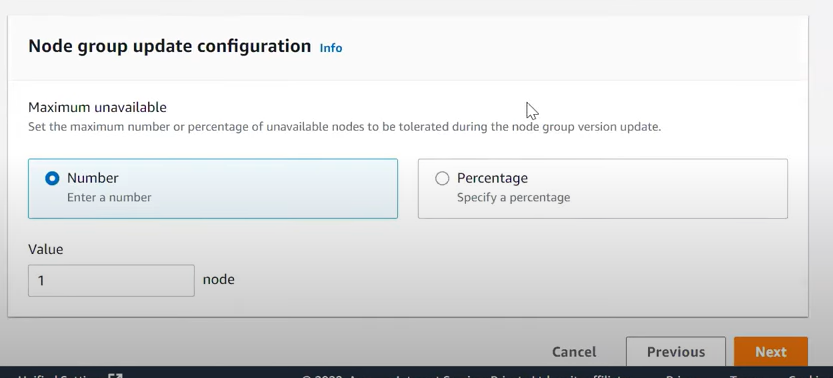
****

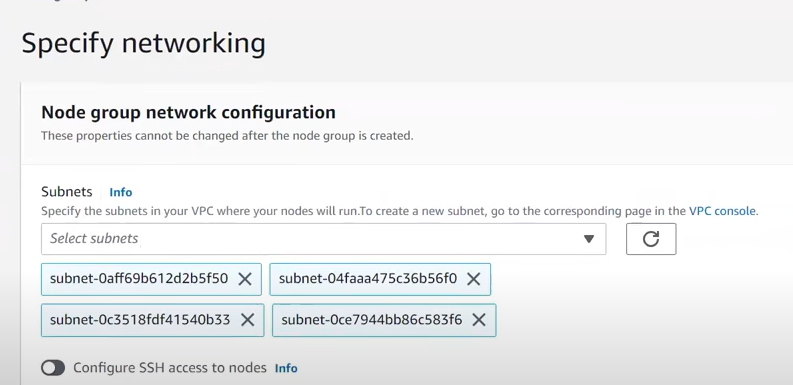
****

****

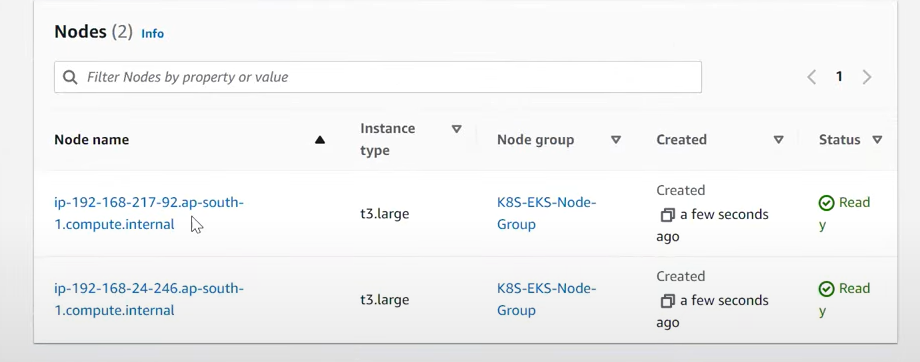
****

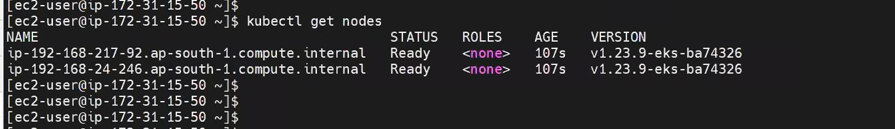
****

****

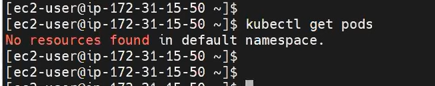
****

**Create  
2 worker nodes are ready**

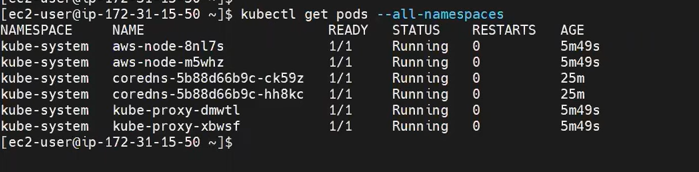
****

****

**# kubectl get pods**

****

**# kubectl get pods -all-namespaces**

****

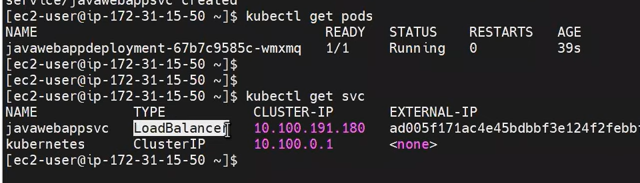
**Copy the yaml file**

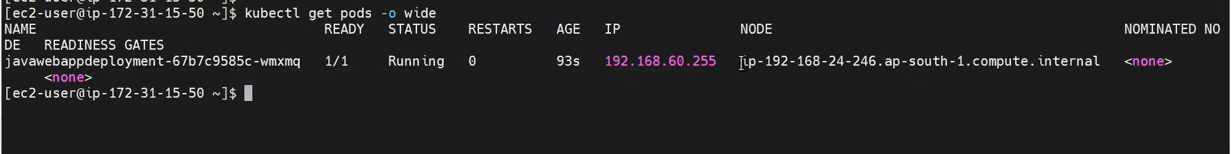
**Vi nginxwebapp.yaml**

**Paste the code**

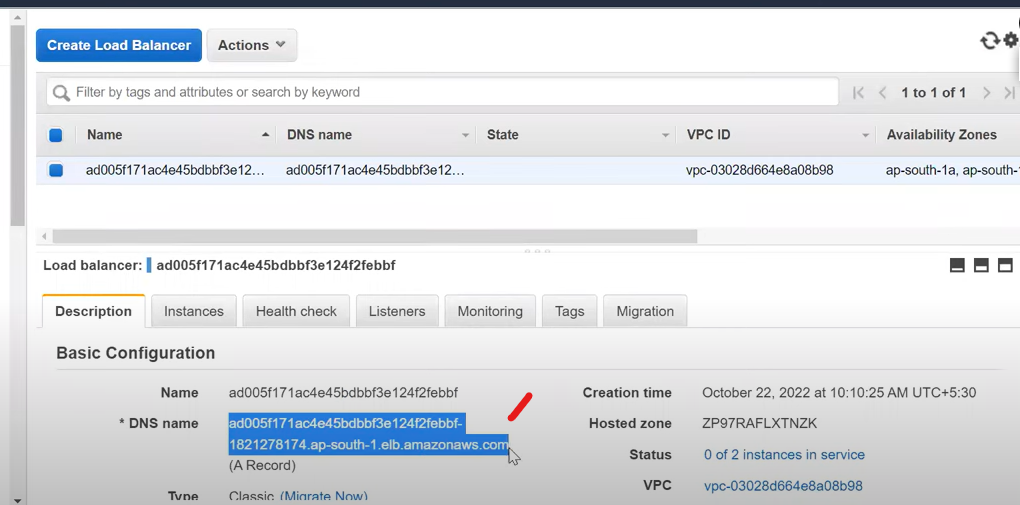
**:wq**

**# kubectl apply -f nginxwebapp.yaml**

****

****

**Lets access loadbalancer DNS**

****