**Kubernetes Task -2**

**Task Description:**

Create the K8s EKS,further you have to do the deployment of the Nginx application and access the application outside the cluster.

=====================================================

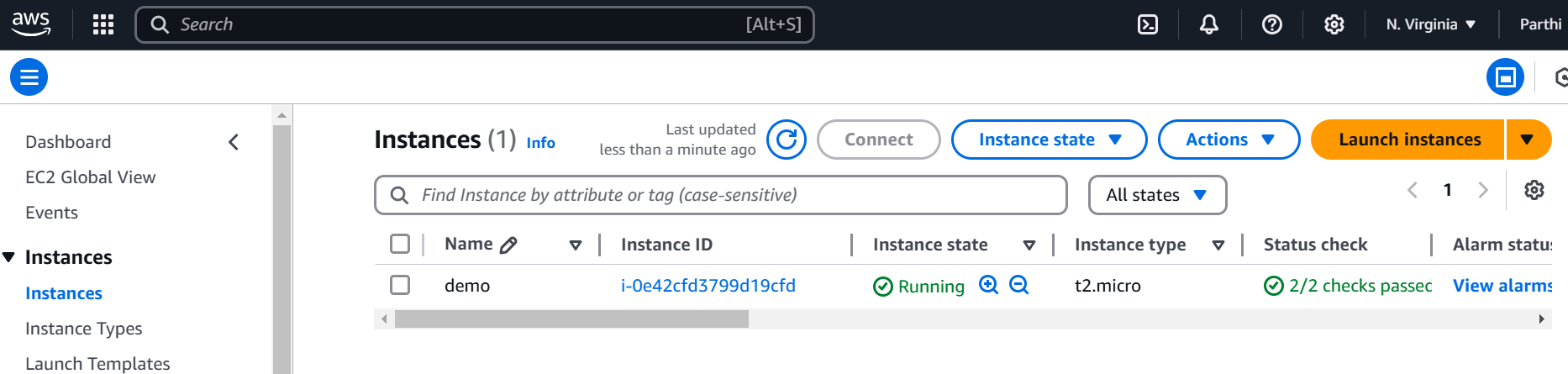
Steps:

1. Setup the Ec2 Instance to create the EKS cluster.
2. Setup aws, kubectl & eksctl
3. Create the IAM role and attach to Ec2Instance.
4. Creating the EKS cluster
5. Creating the deployment and service
6. Expose the application in the website
7. Delete the cluster

=========================================================

**1.Setup the Ec2 Instance to create the EKS cluster.**

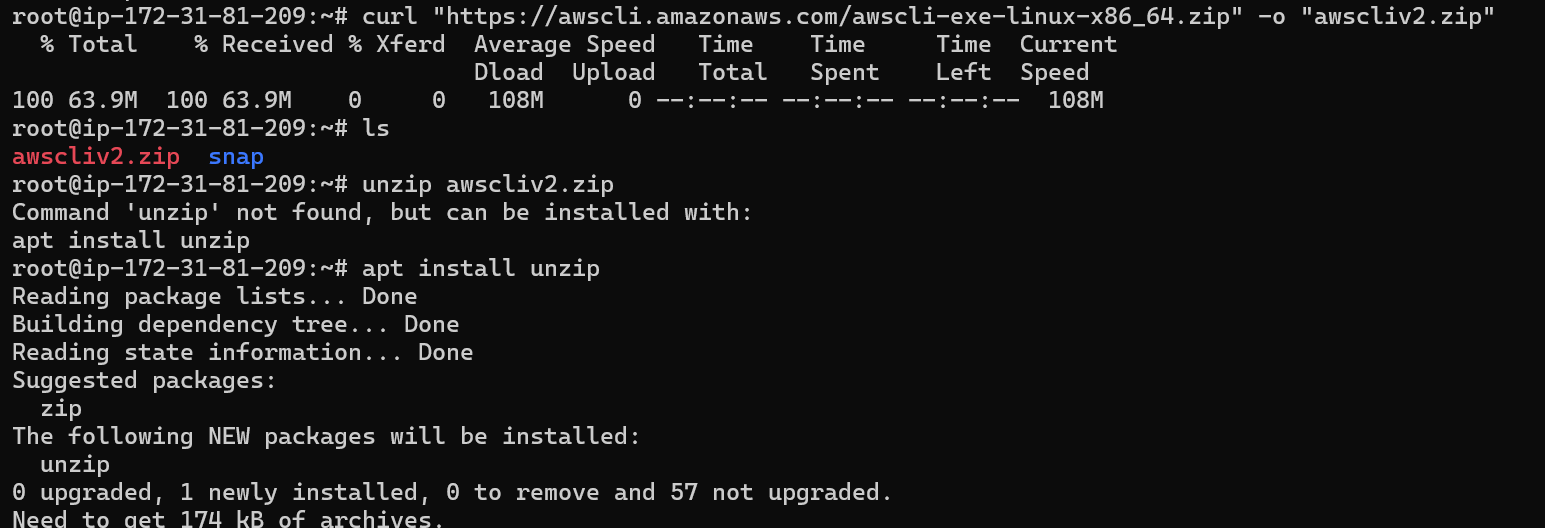
* Created the Ec2 Instance.

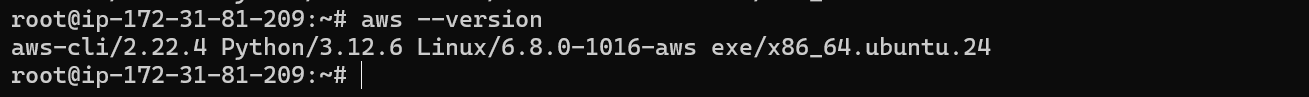


**2.Setup aws, kubectl & eksctl**

* To use the below url to download the aws cli.

| curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"  unzip awscliv2.zip  sudo ./aws/install |
| --- |



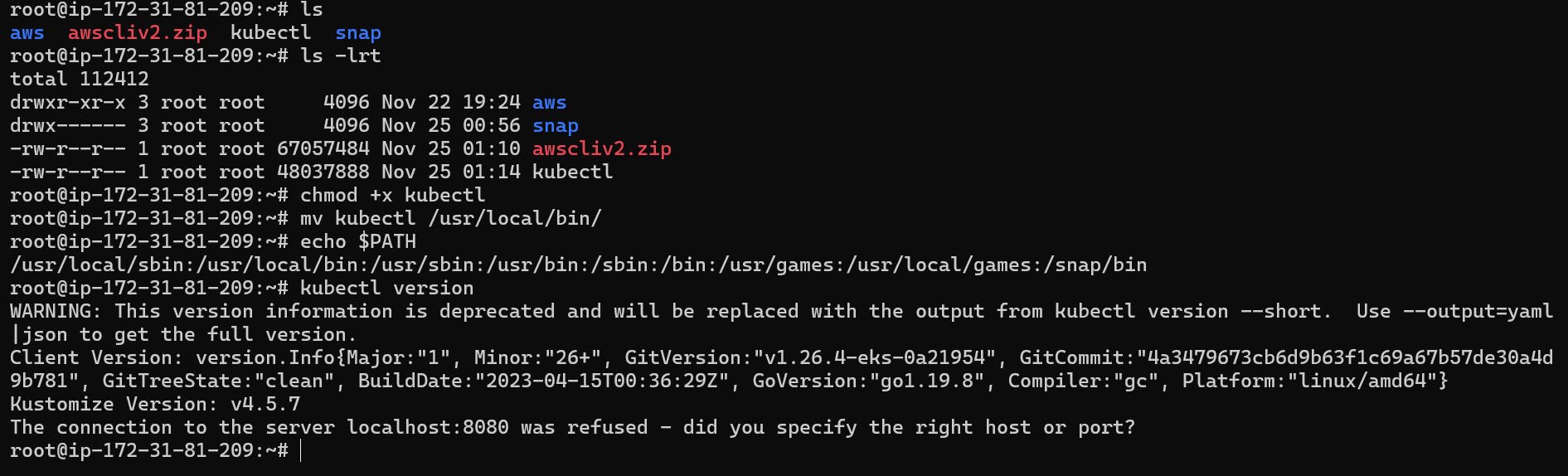


Download the kubectl:

* To use the below url to download the kubectl.

| curl -O https://s3.us-west-2.amazonaws.com/amazon-eks/1.26.4/2023-05-11/bin/linux/amd64/kubectl |
| --- |

* Once downloaded the file needs to provide the execution permission and move to /usr/local/bin folder.



Setup to eksctl:

* To use the below url to download the eksctl and move to usr/local/bin folder.

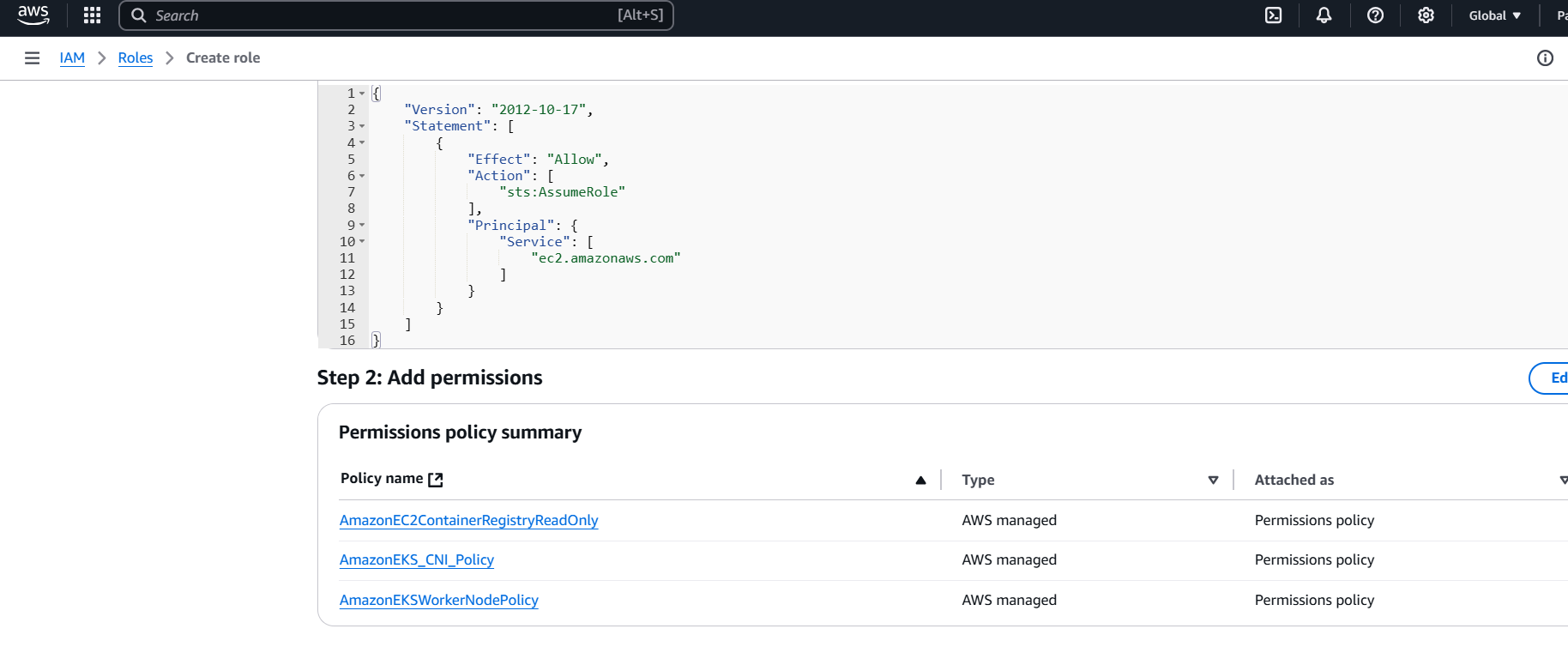
| curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp  sudo mv /tmp/eksctl /usr/local/bin  eksctl version |
| --- |



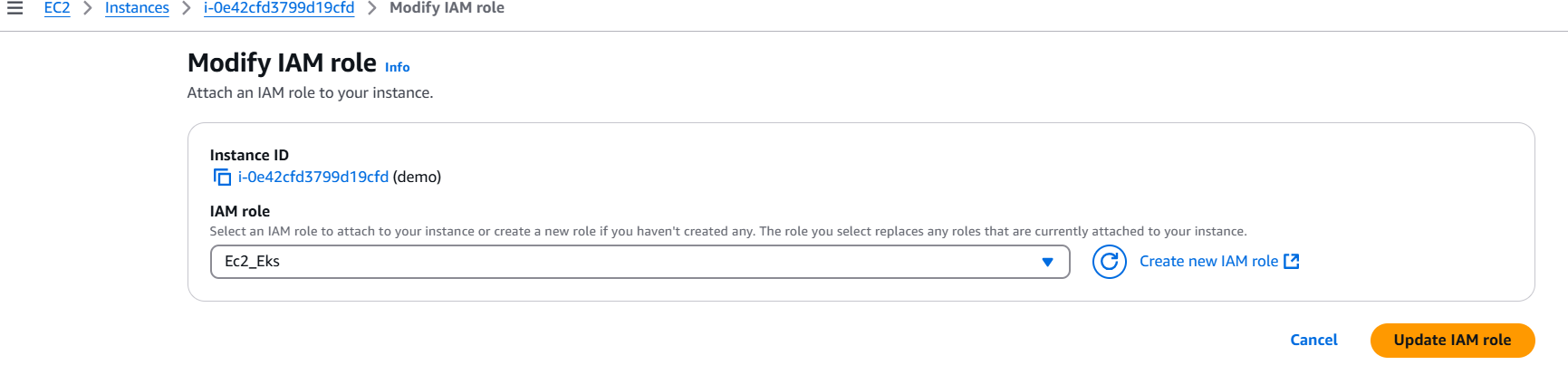
3. **Create the IAM role and attach to Ec2Instance**

* Created the IAM role to provide the permission to create the eks cluster and attached to ec2Instance.

| * + AmazonEC2ContainerRegistryReadOnly   + AmazonEKSWorkerNodePolicy   + AmazonEKS\_CNI\_Policy |
| --- |



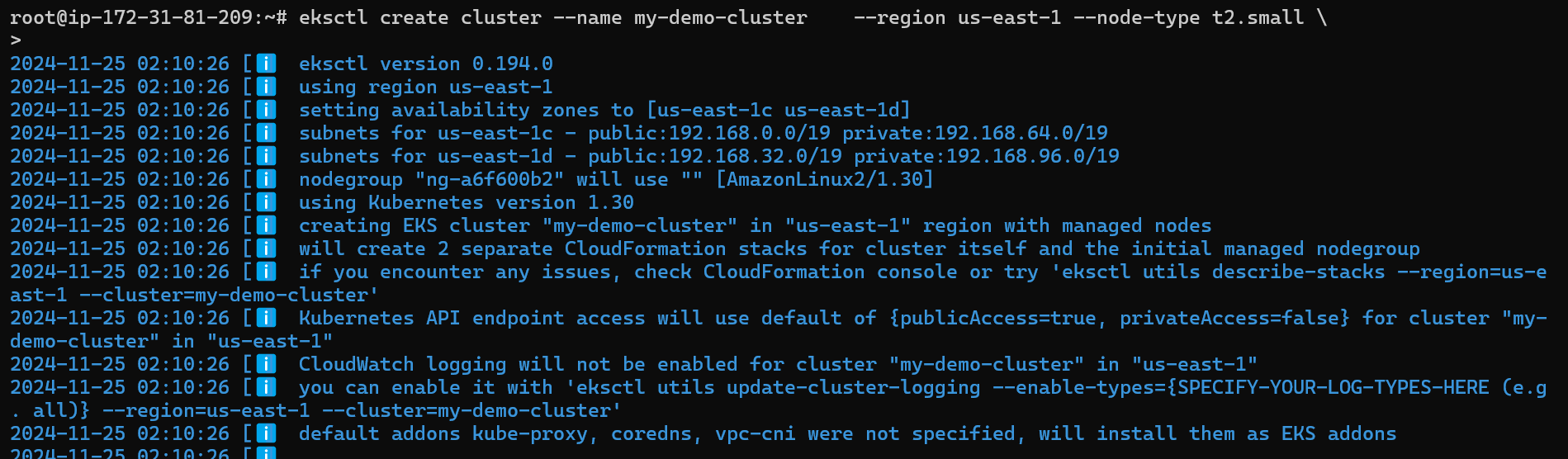
* Attached the IAM role to Ec2Instance.

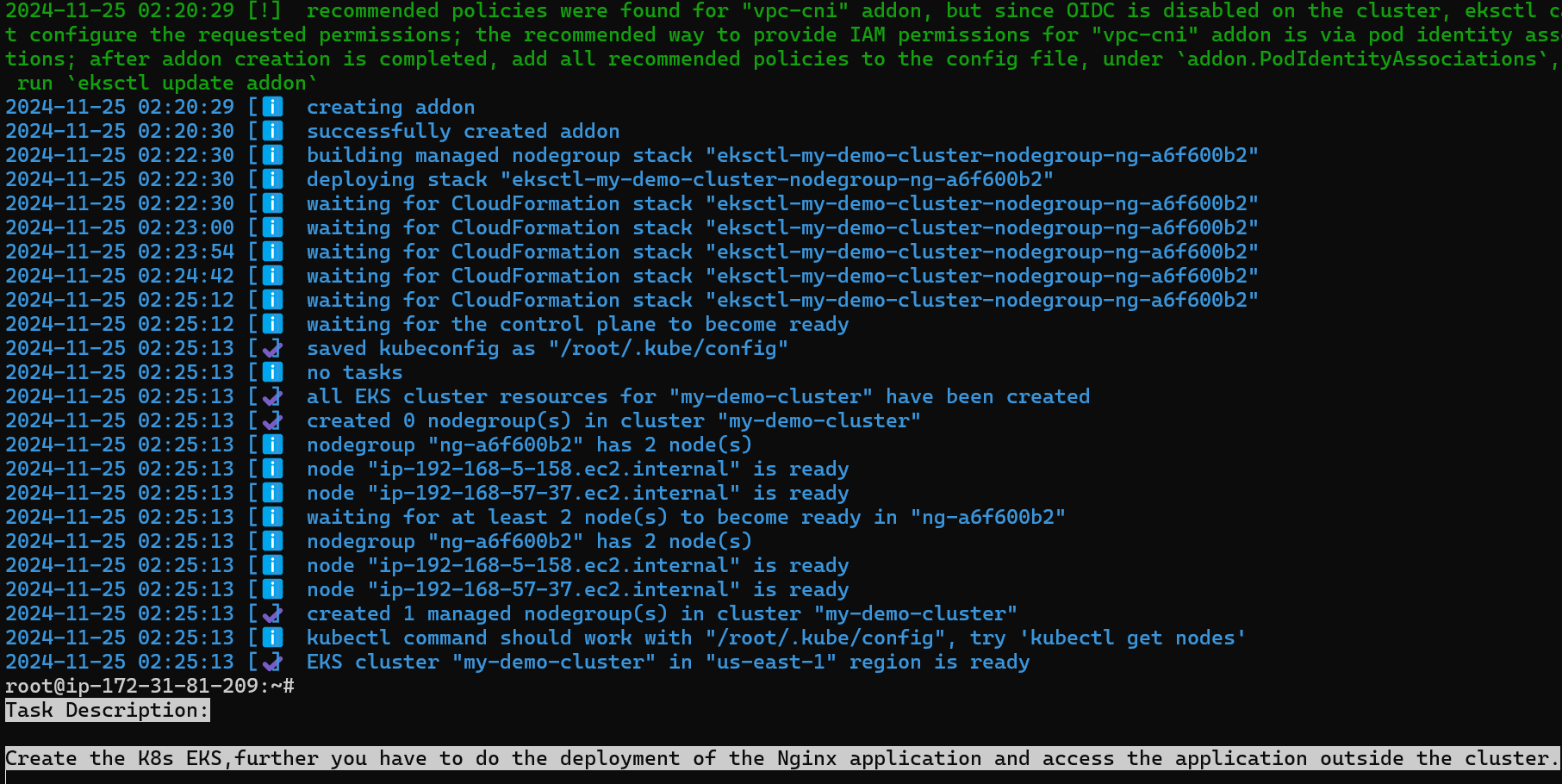


4.Creating the EKS cluster:

* Before execute the command need to confirm that aws configured or not.
* Once aws configured, then execute the eksctl command to create the eks cluster.

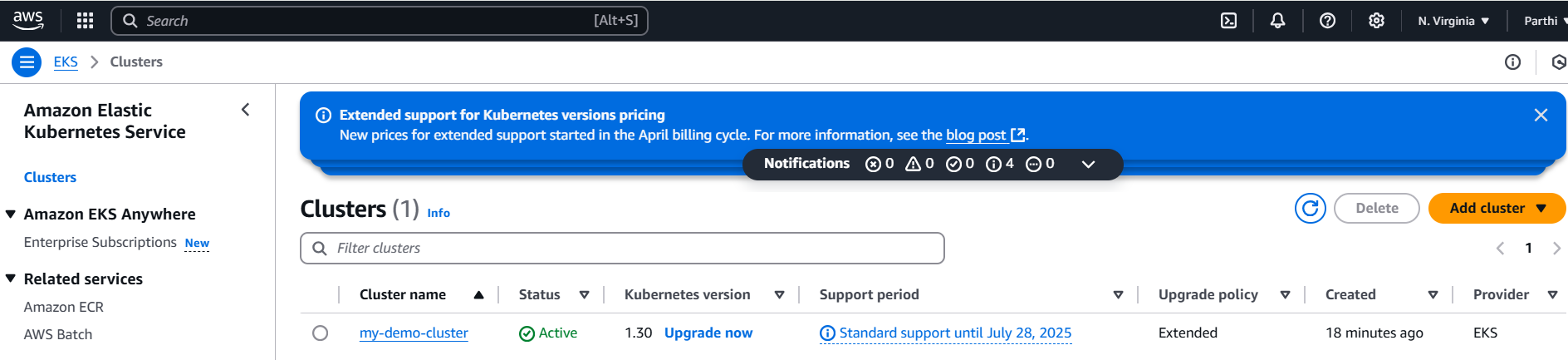
| eksctl create cluster --name my-demo-cluster \  --region us-east-1 \  --node-type t2.small \ |
| --- |

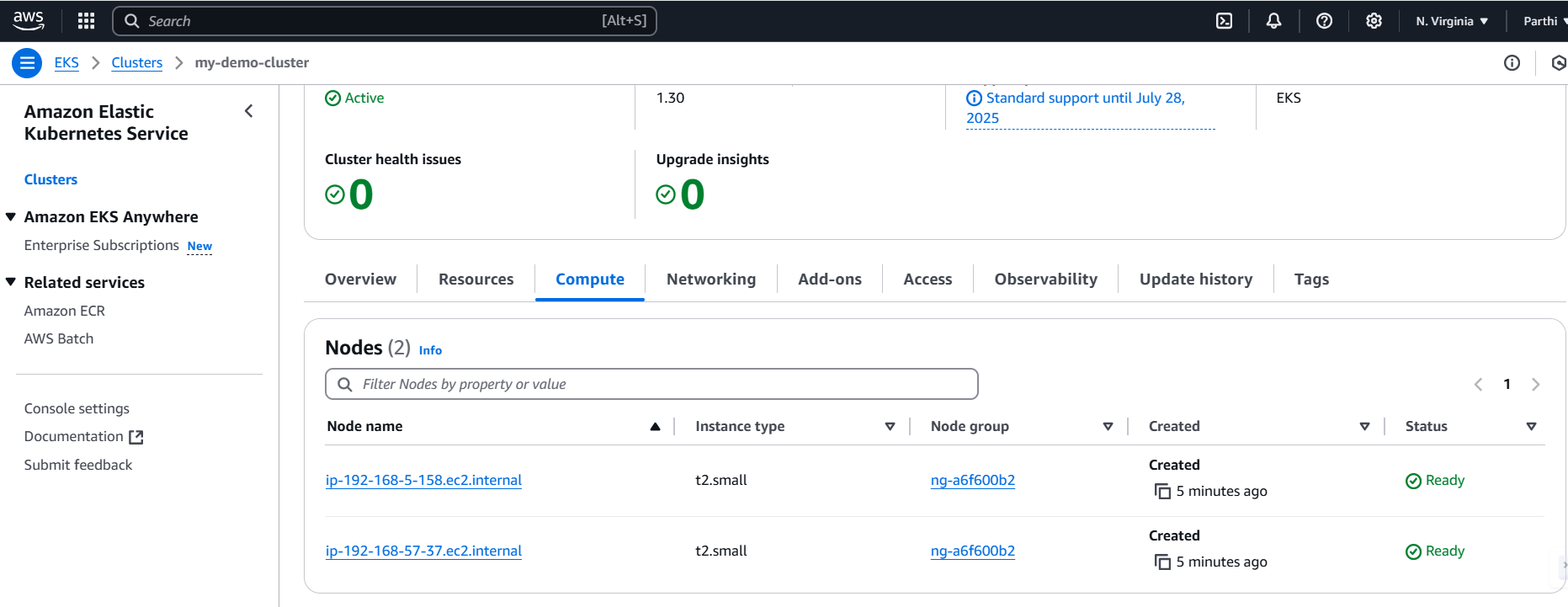


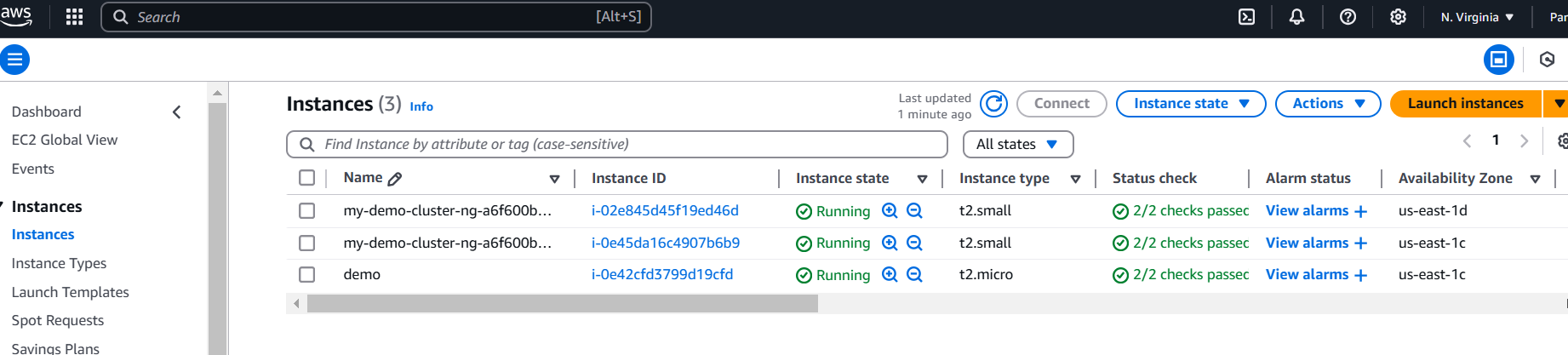


* Successfully created the eks cluster and also verified that two node

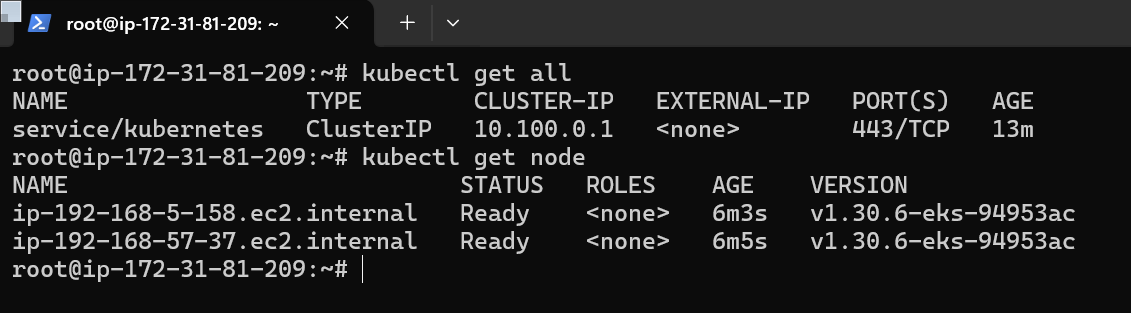
created.







* To verify cluster details using the kubectl command.

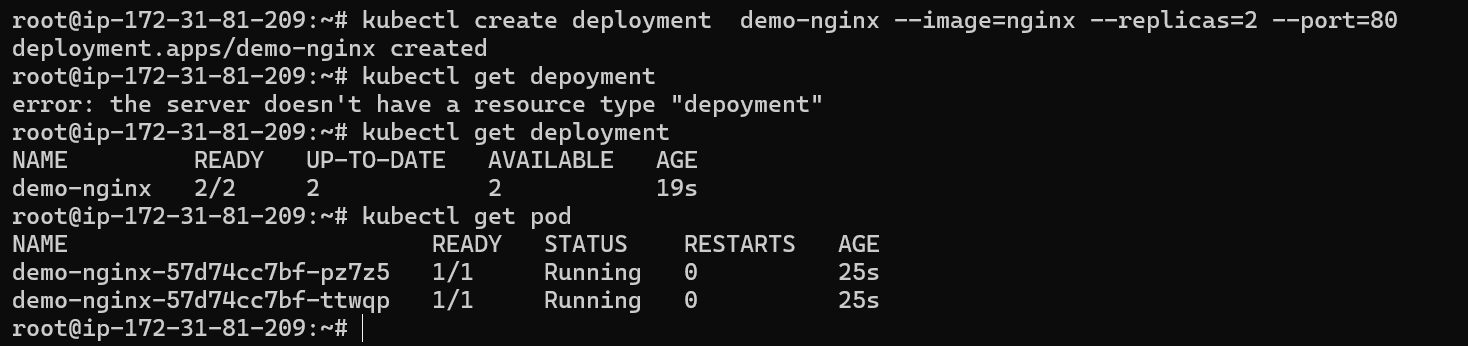


5. **Creating the deployment and service**

* Now create the nginx deployment using the below command

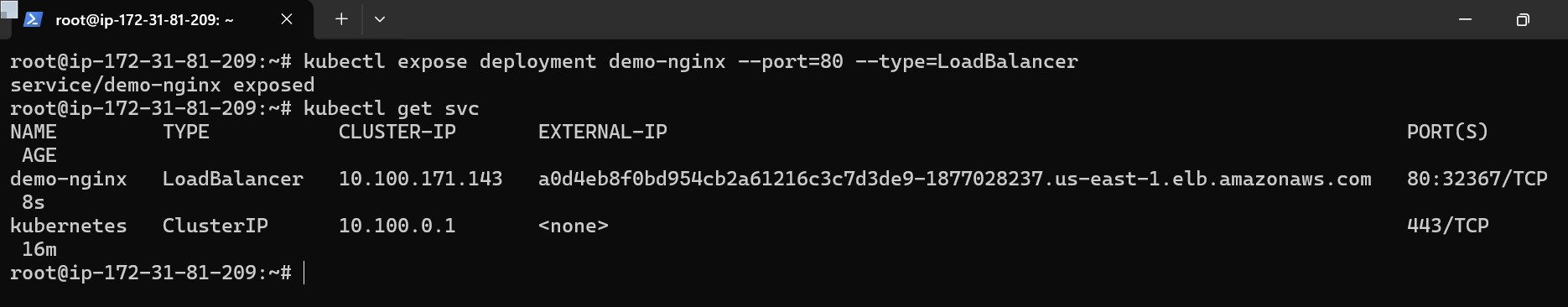
| kubectl create deployment demo-nginx --image=nginx --replicas=2 --port=80 |
| --- |

* Now create the deployment with two replicas and verified with kubectl command.
* Now deployment is up and running.

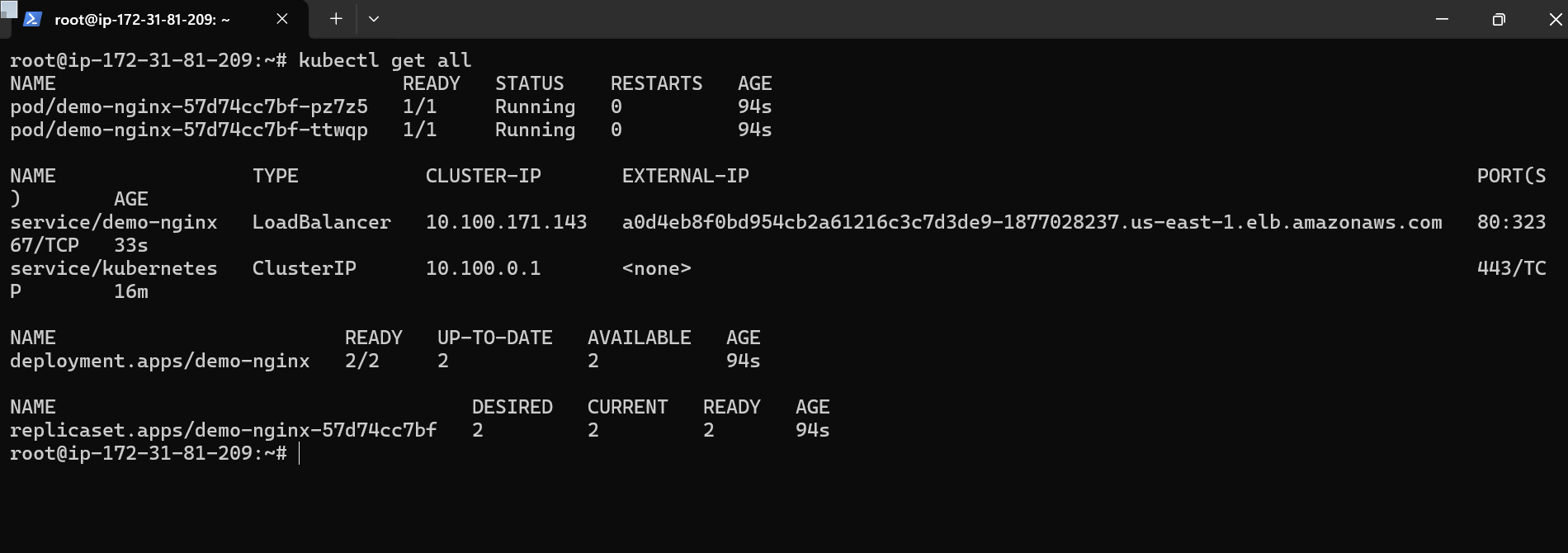


* Now I need to expose my deployment to the browser, so I created the LoadBalance service using the following command.

| kubectl expose deployment demo-nginx --port=80 --type=LoadBalancer |
| --- |



* Now successfully created the Load Balancer.



6.**Expose the application in the website**

* My application is able to the output in the browser using the LoadBalancer External-ip address.



7**. Delete the cluster**

* Need to delete the cluster using the below command.

| eksctl delete cluster my-demo-cluster --region us-east-1 |
| --- |

