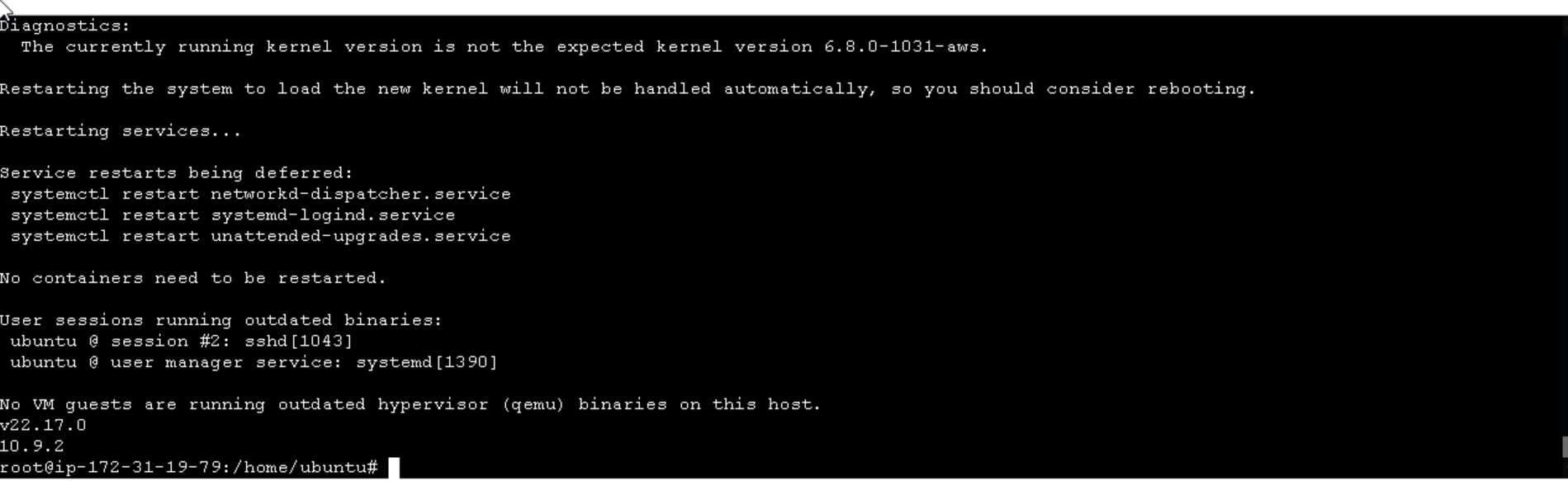
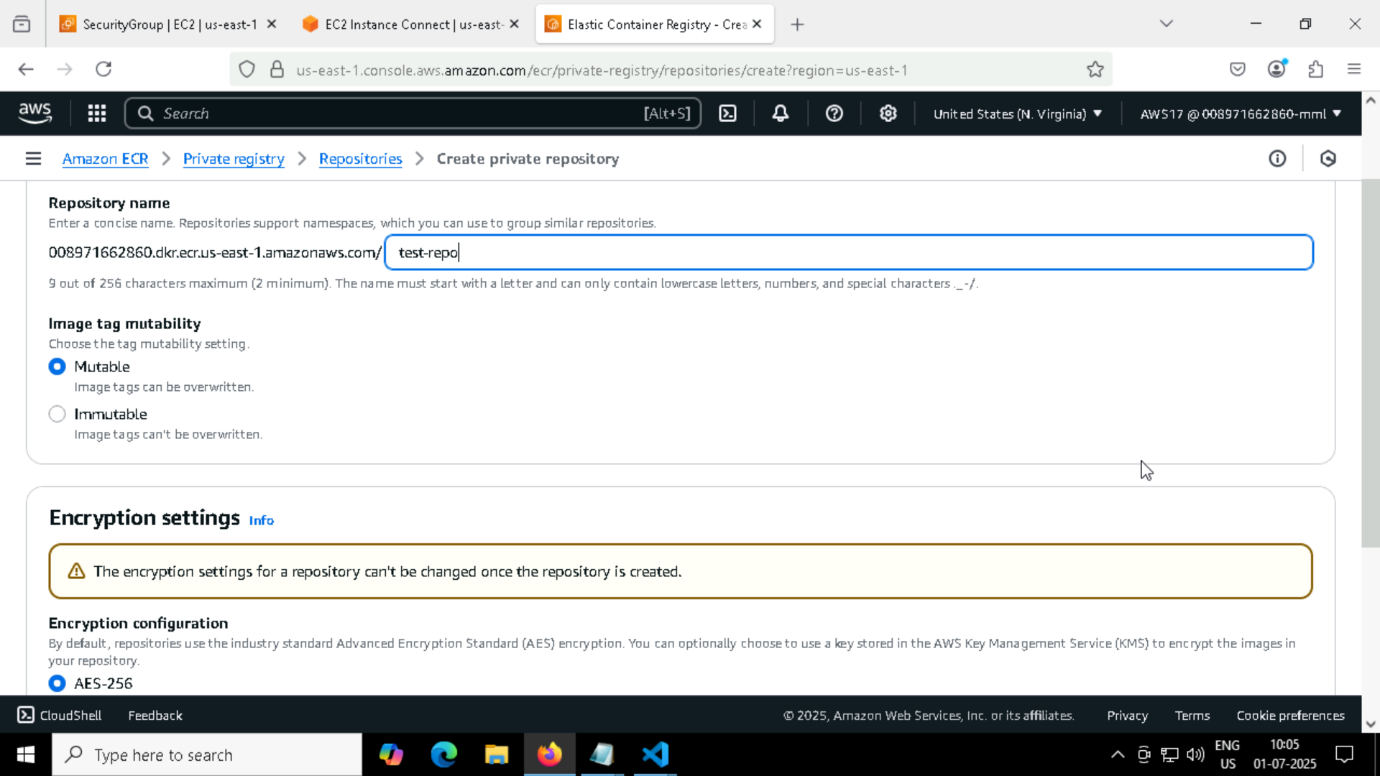
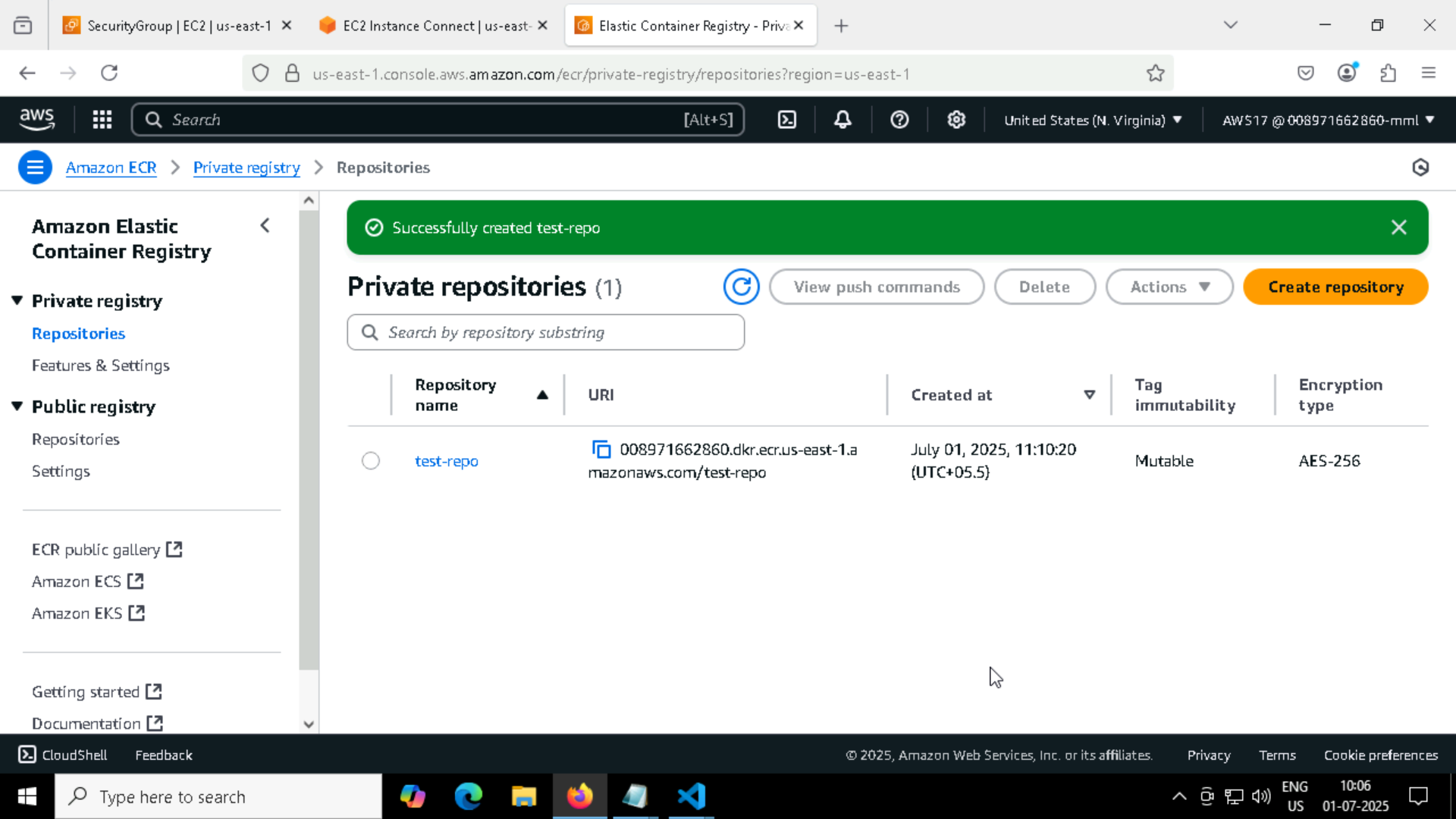
**AWS FINAL EXAM DOCUMENTATION**

**Nodejs installation**

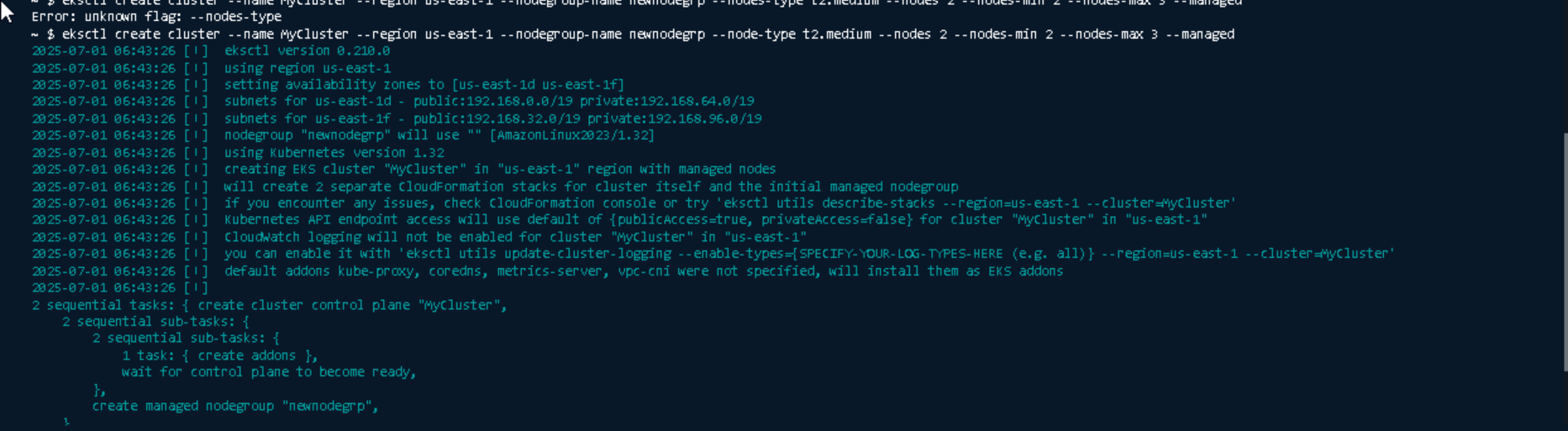
****

**ECR Repository Creation**

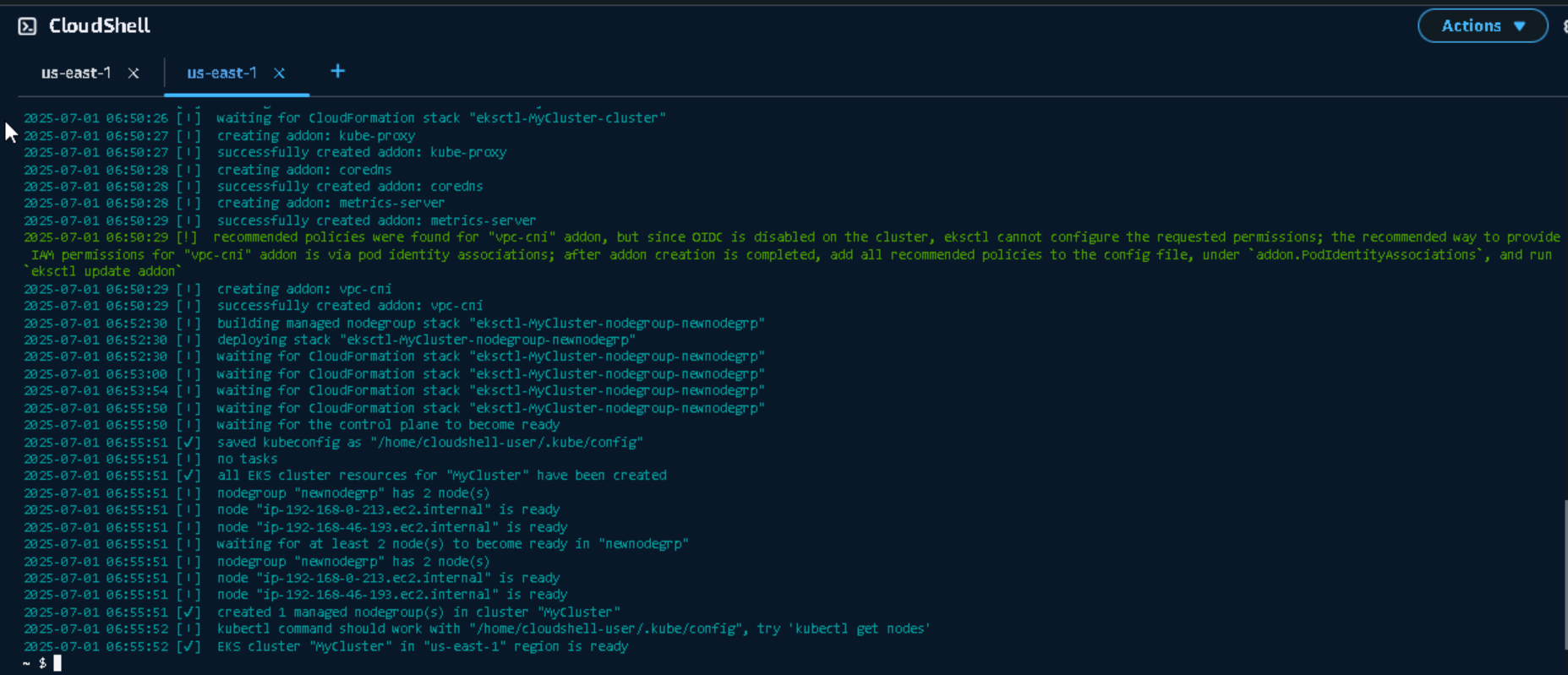
****



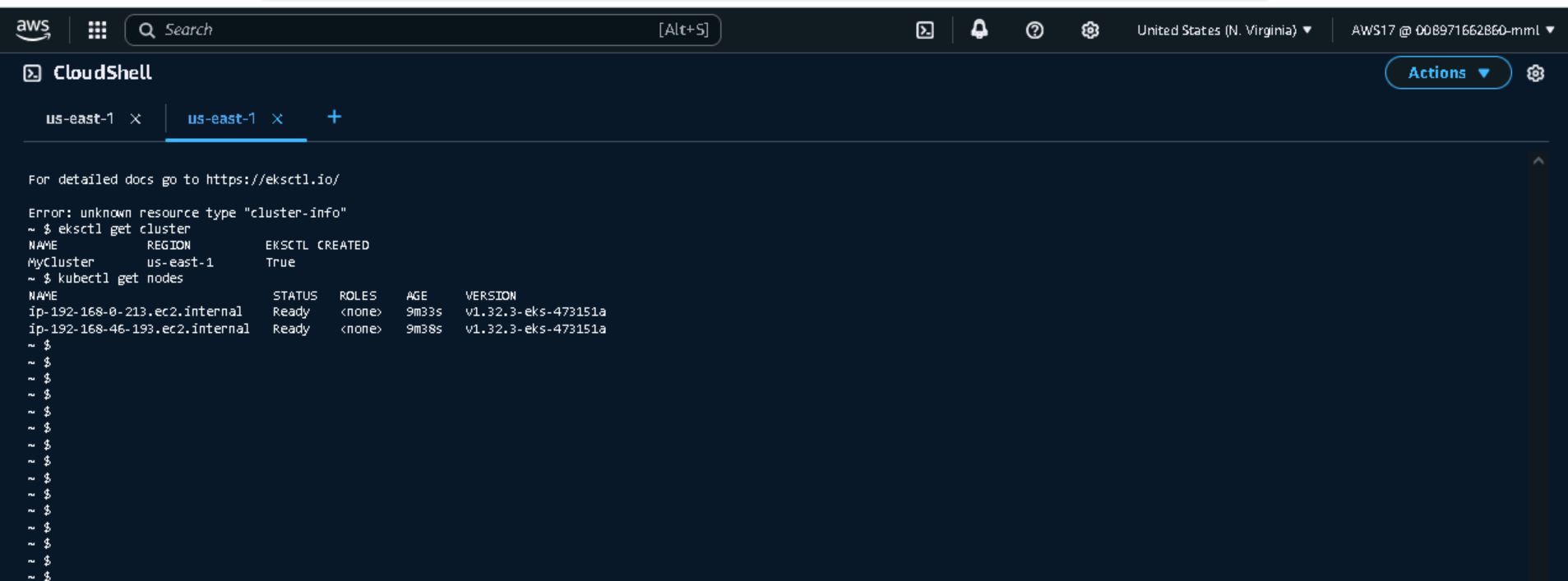
**2.Creating of EKS Cluster with worker node**

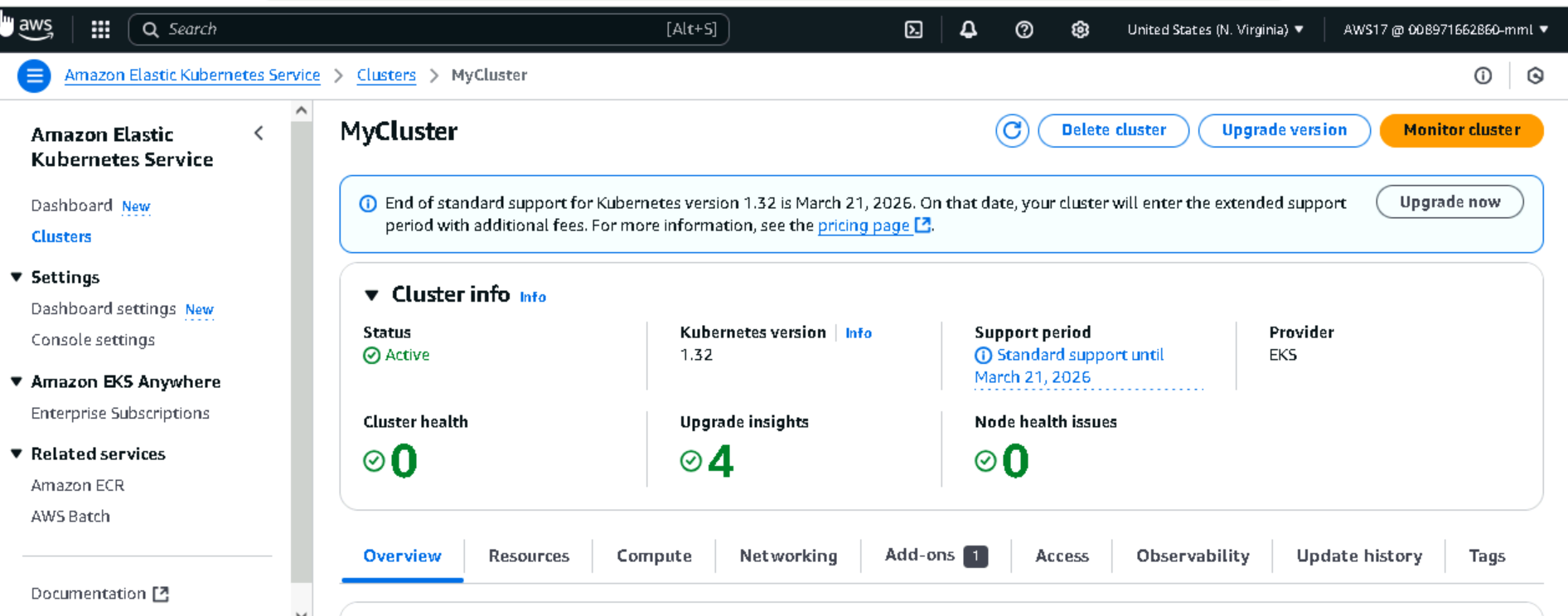
****

**EKS cluster is in Ready State:**

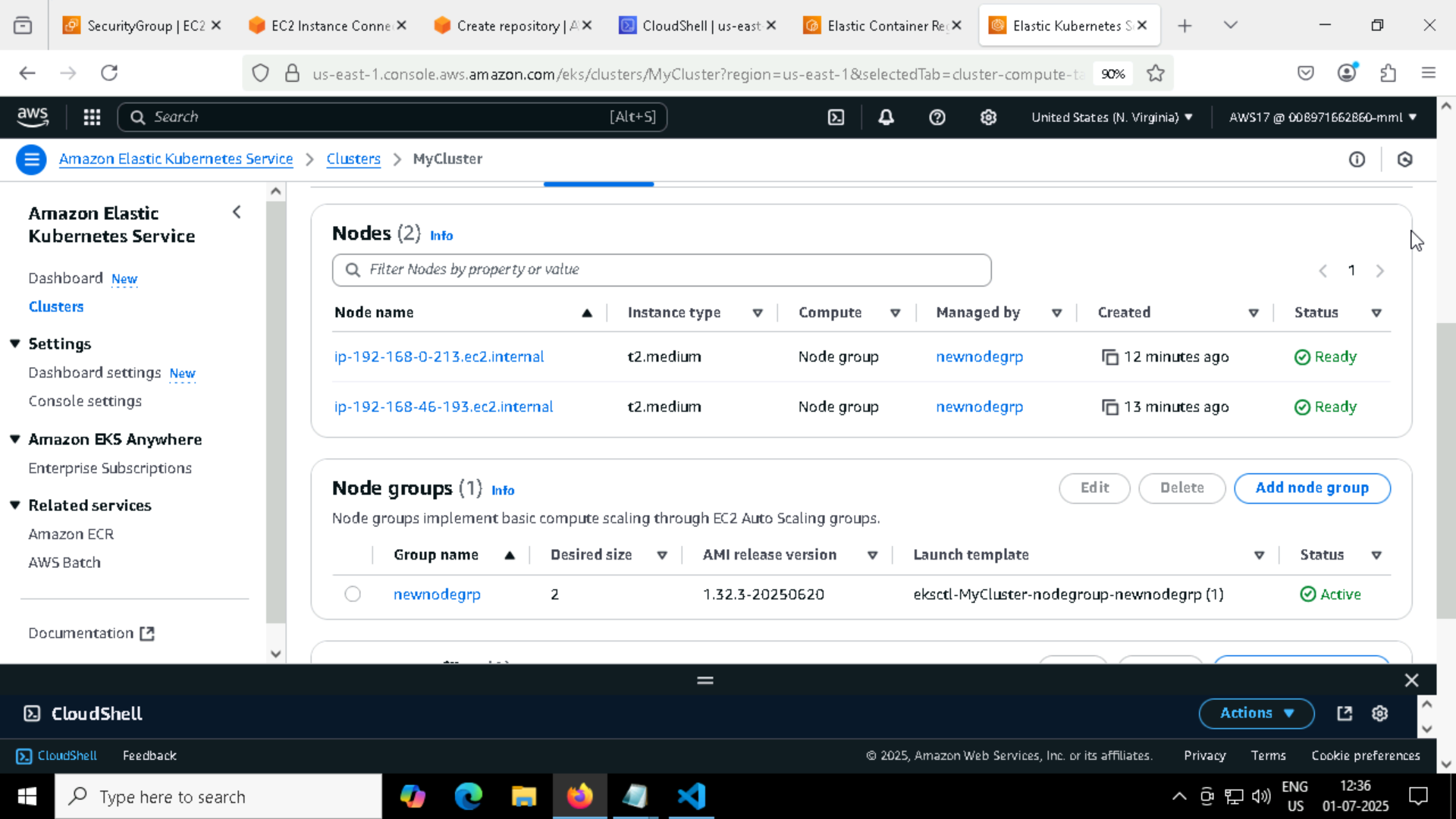
****

**Verifying cluster and worker nodes using cli commands and also with Console**

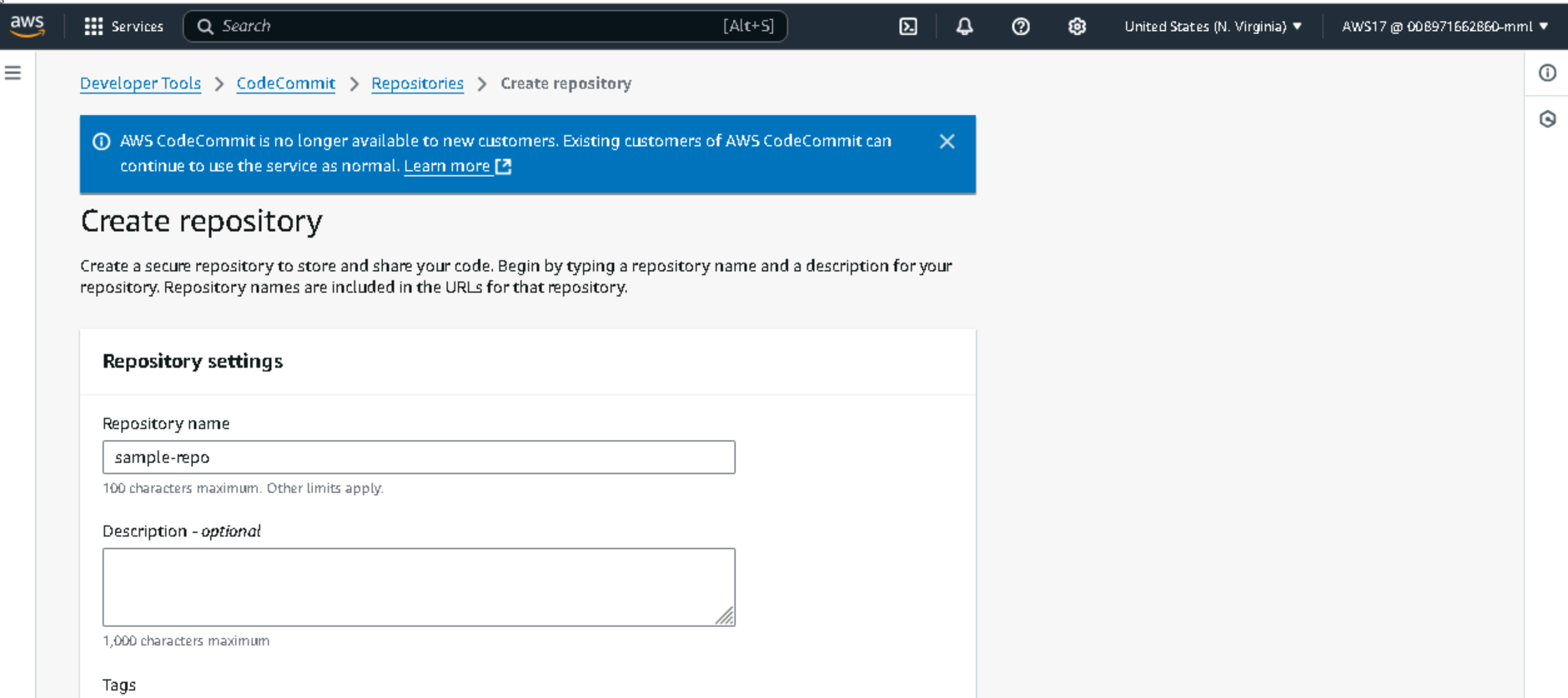
****

****

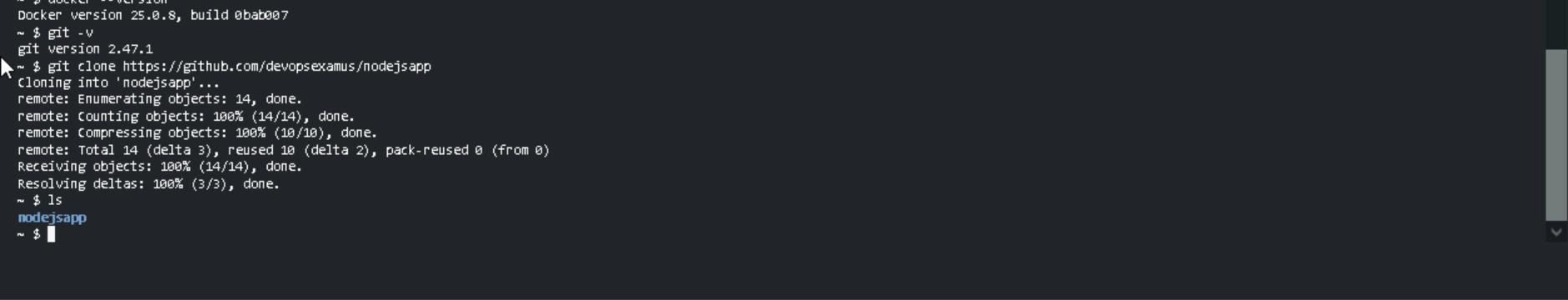
**Nodegroup and Nodes**



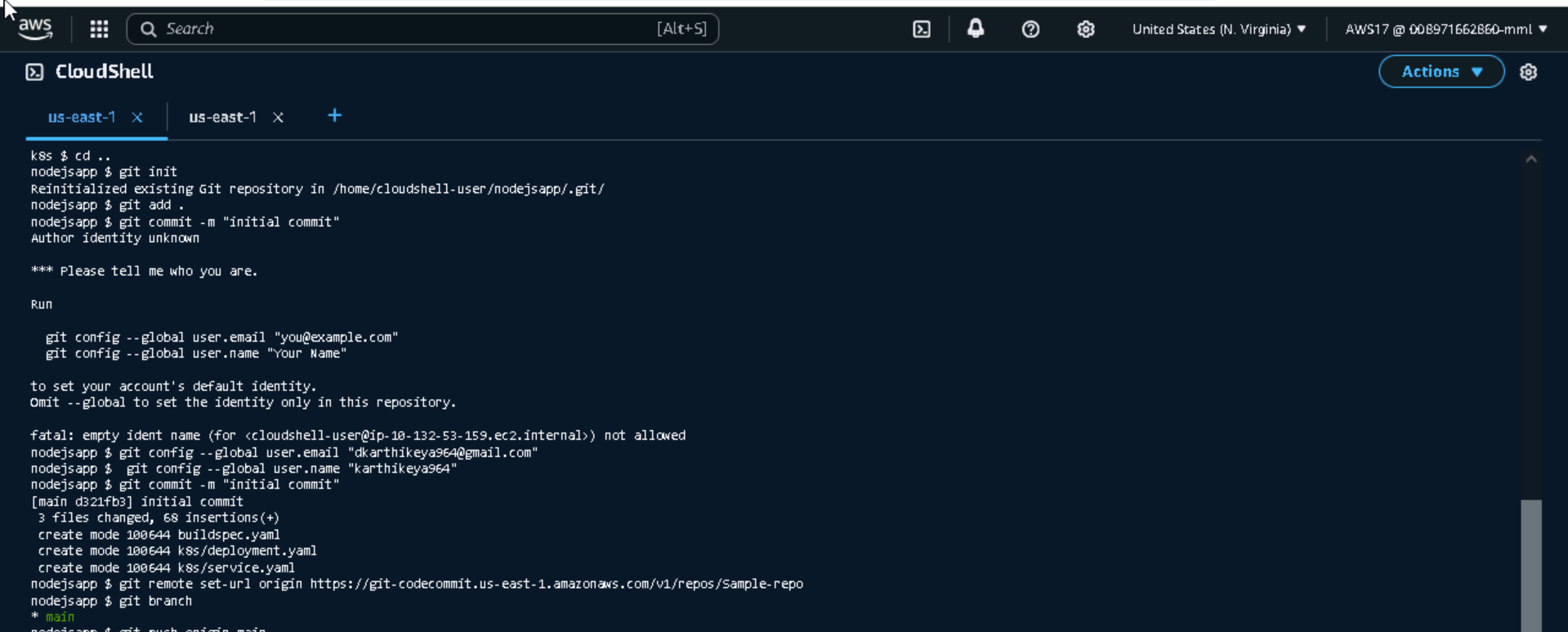
**Creating code commit repo**

****

**Cloning the repo using git clone command in Aws cli**

****

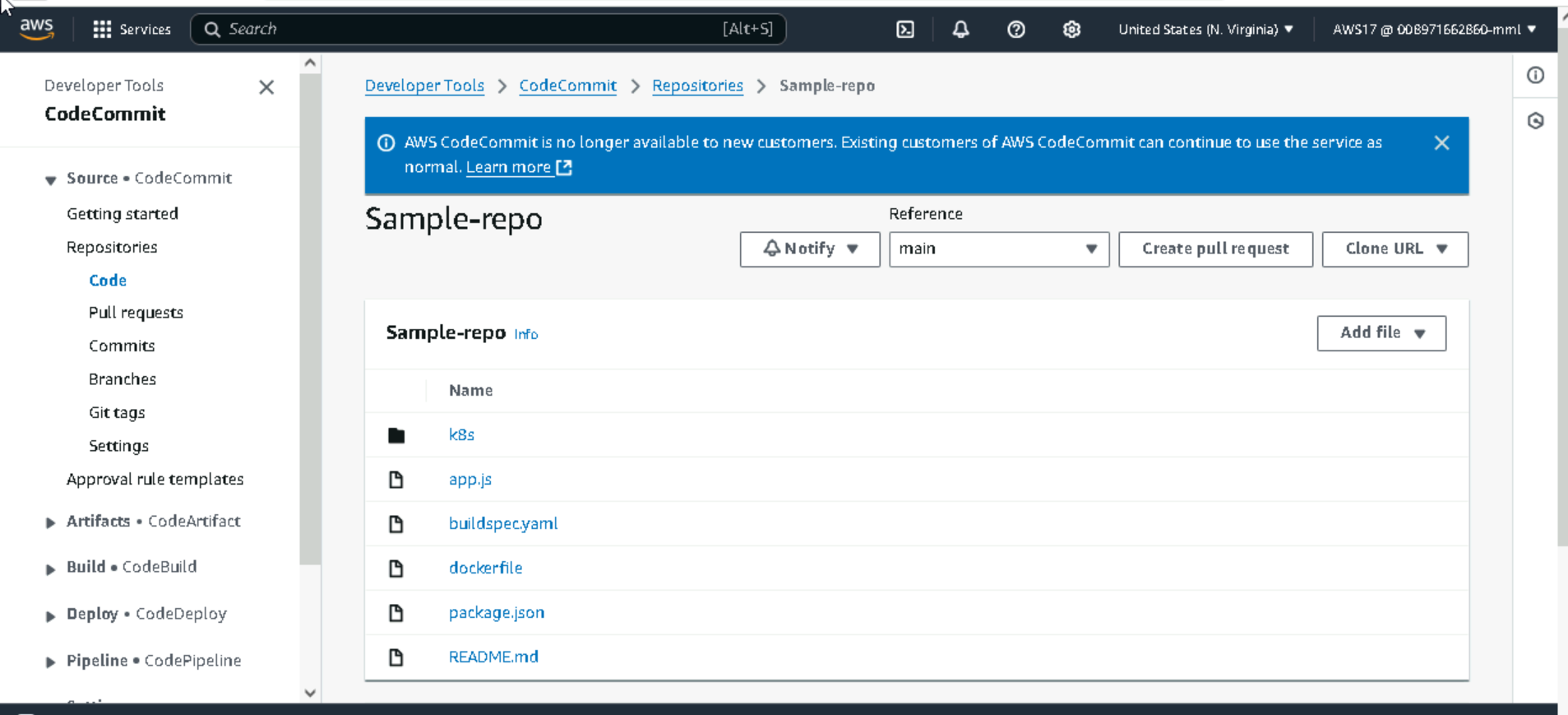
**Pushing all the file into Codecommit**

****

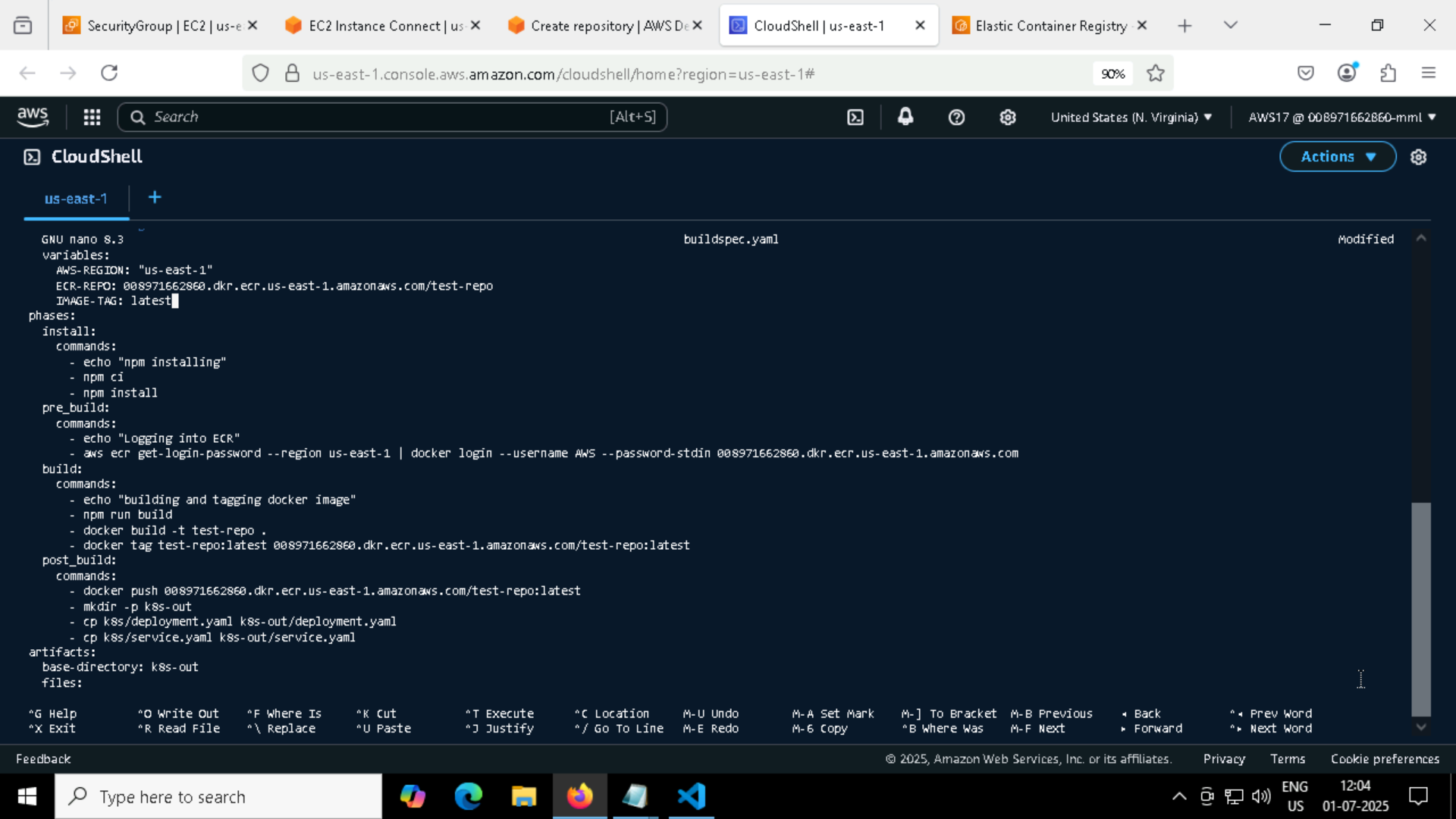
****

**Codecommit repo:**

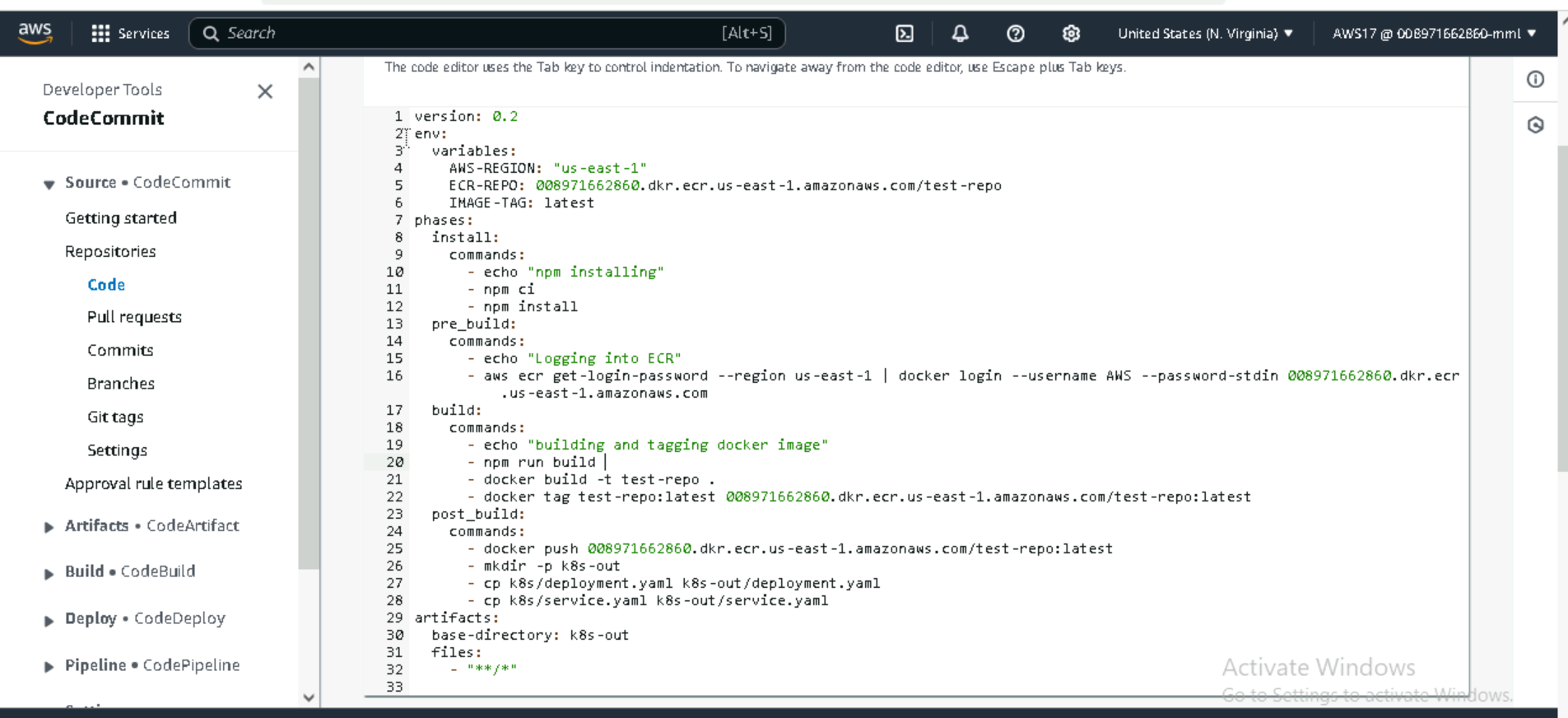
**All my files are inside the sample-repo**

****

**Buildspec.yml**



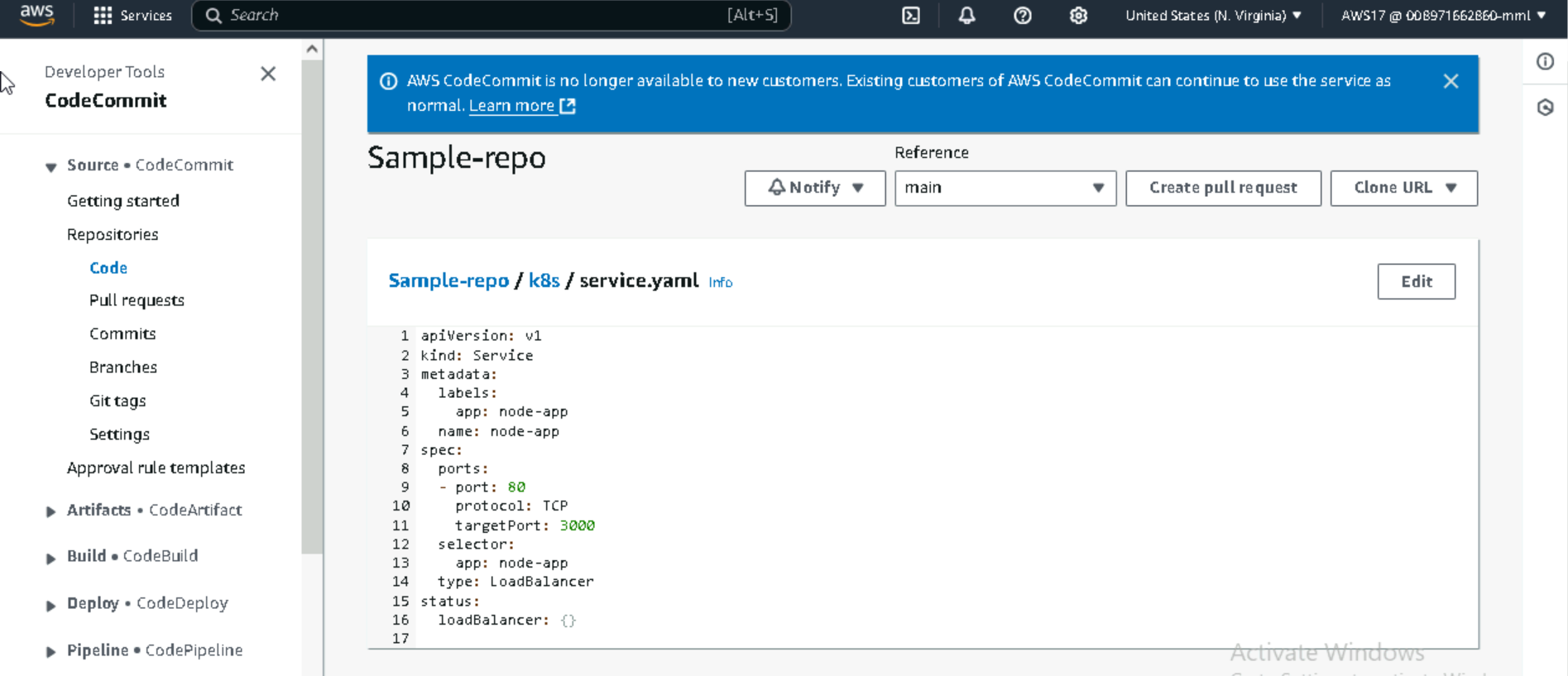
**Buildspec yaml through codecommit**

****

**Deployment.yaml :**

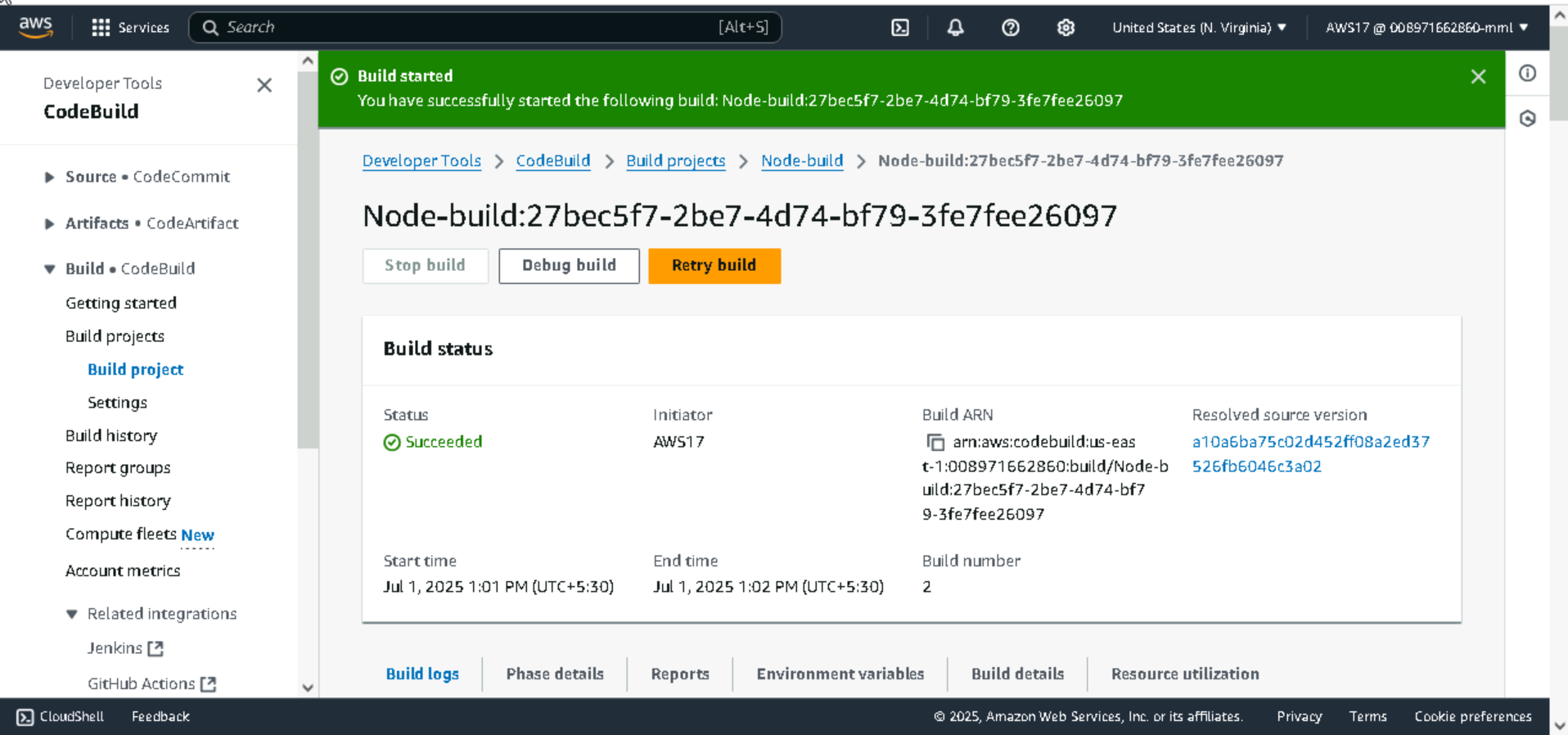
****

**Service.yaml**

****

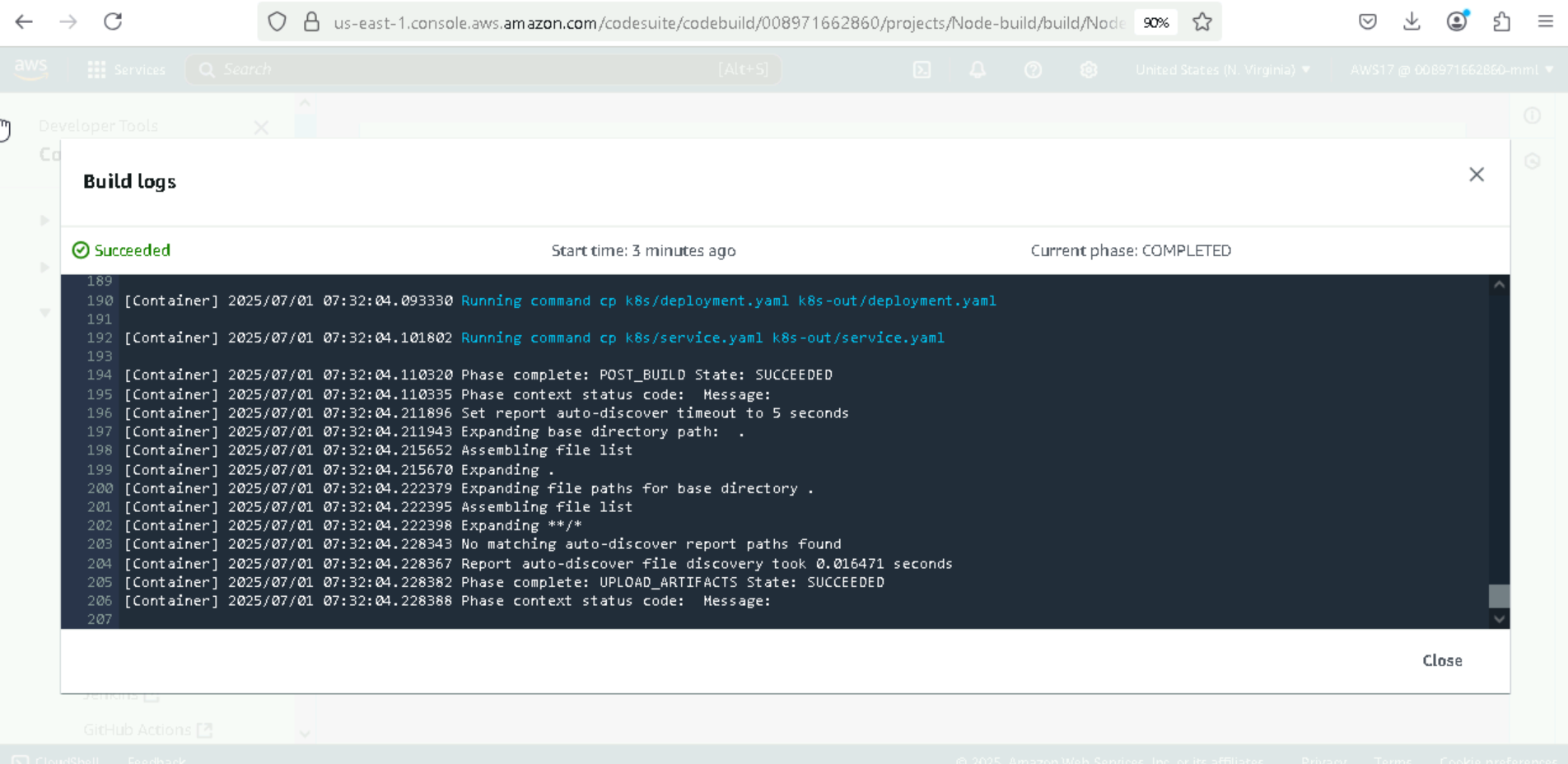
**CODE BUILD:**

**Build succeed**

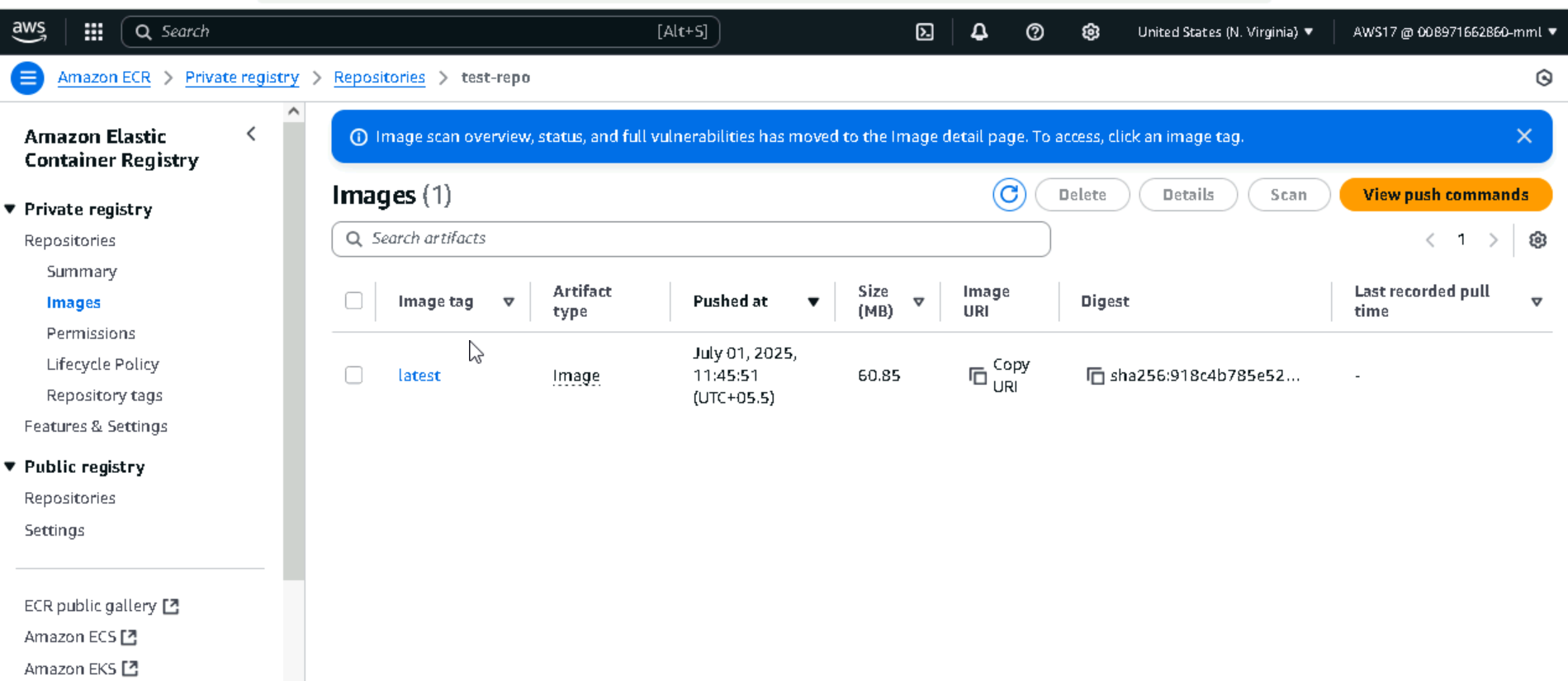
****

**Source I have taken from AwsCode Commit service in codebuild**

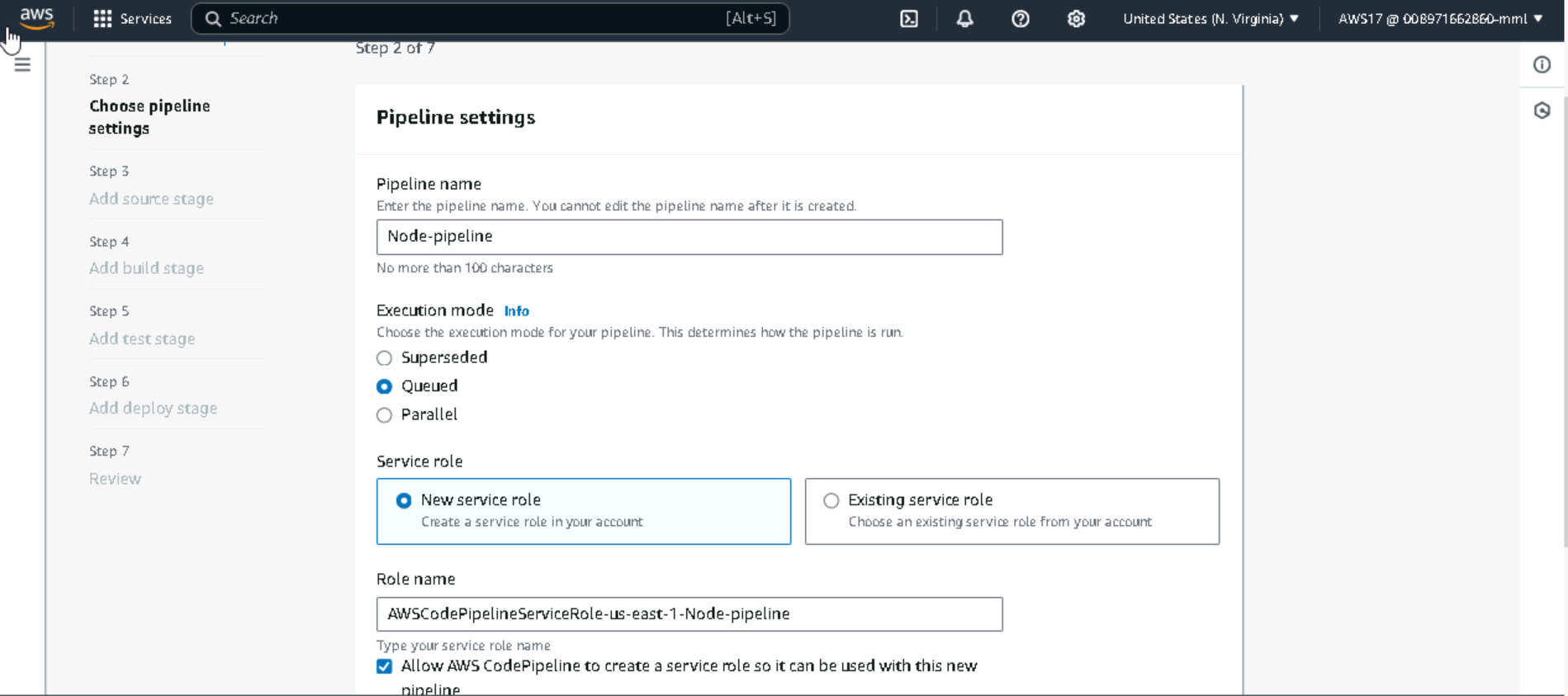
**Build logs:**

****

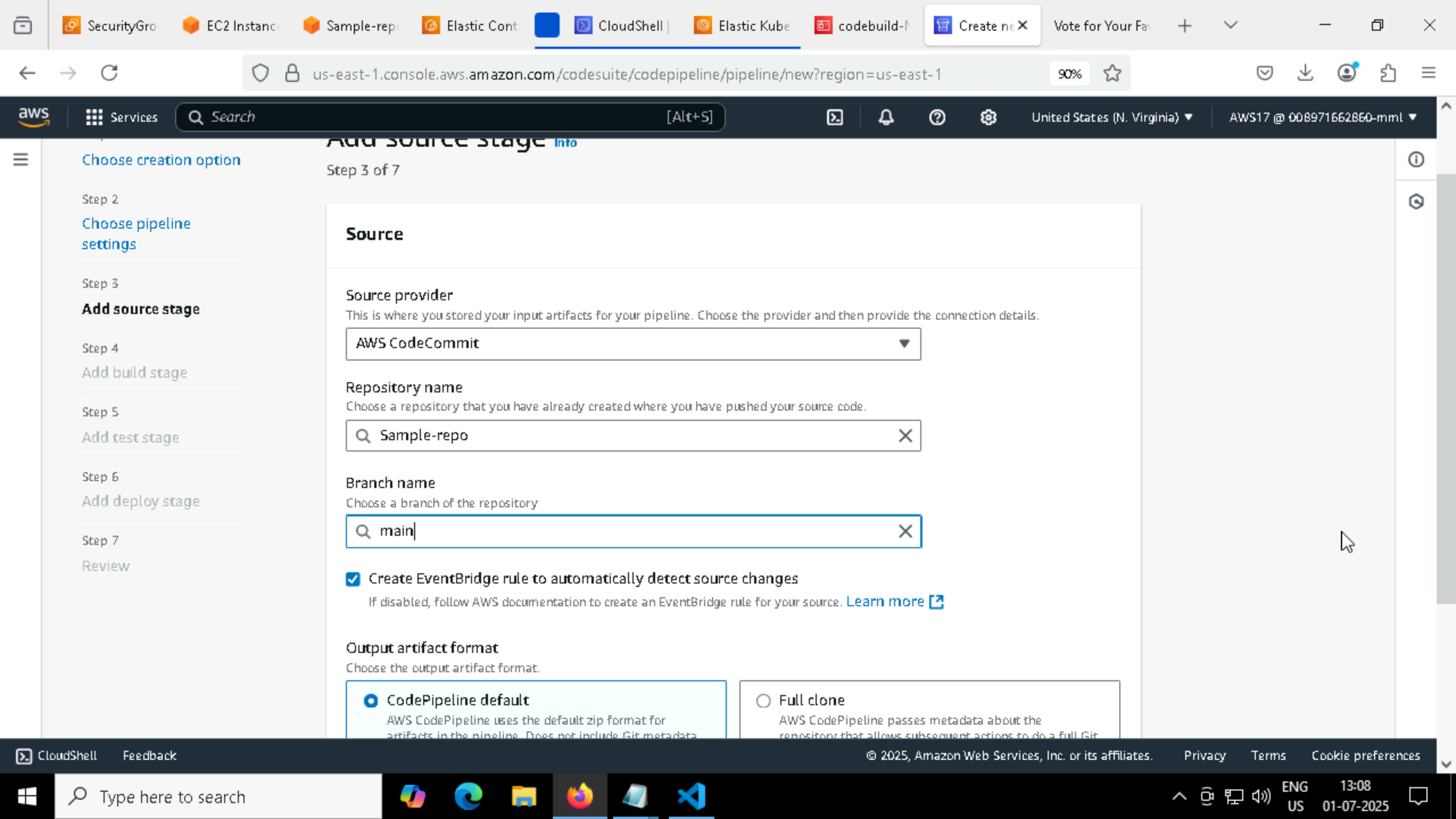
**Image is pushed in ECR repo**

****

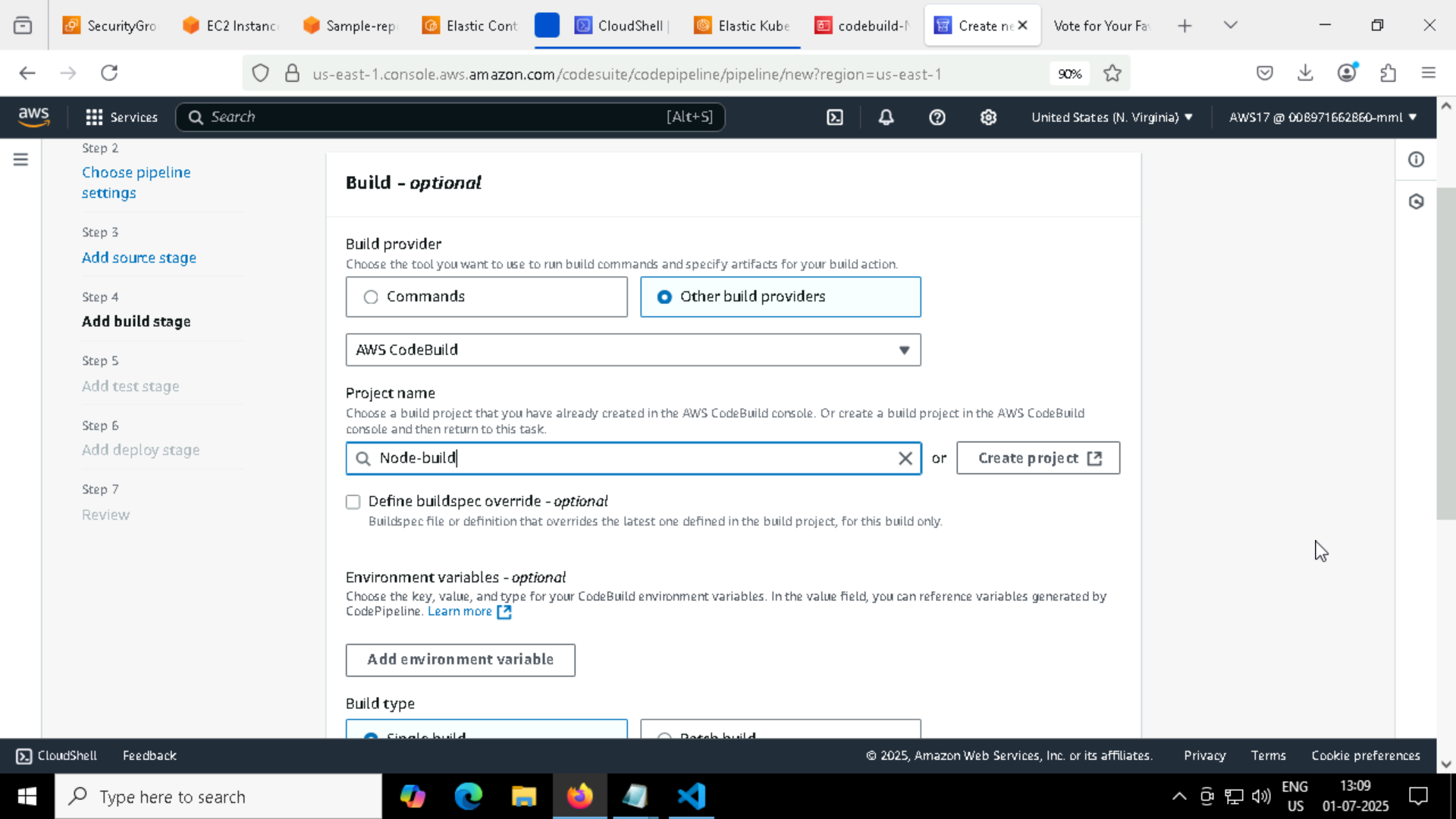
**Node pipline Creation:**

****

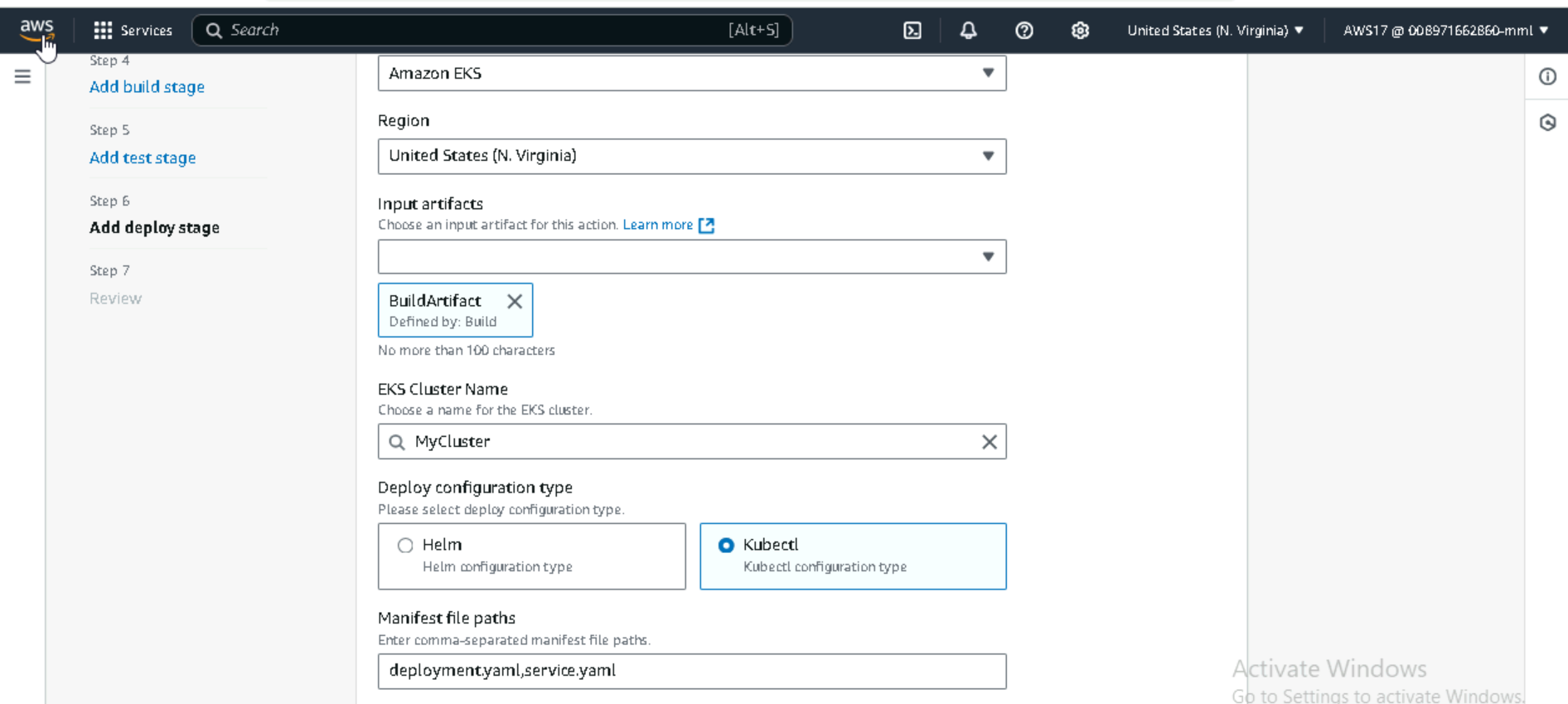
**Source : Aws codecommit**



**Giving Codebuild for pipline**

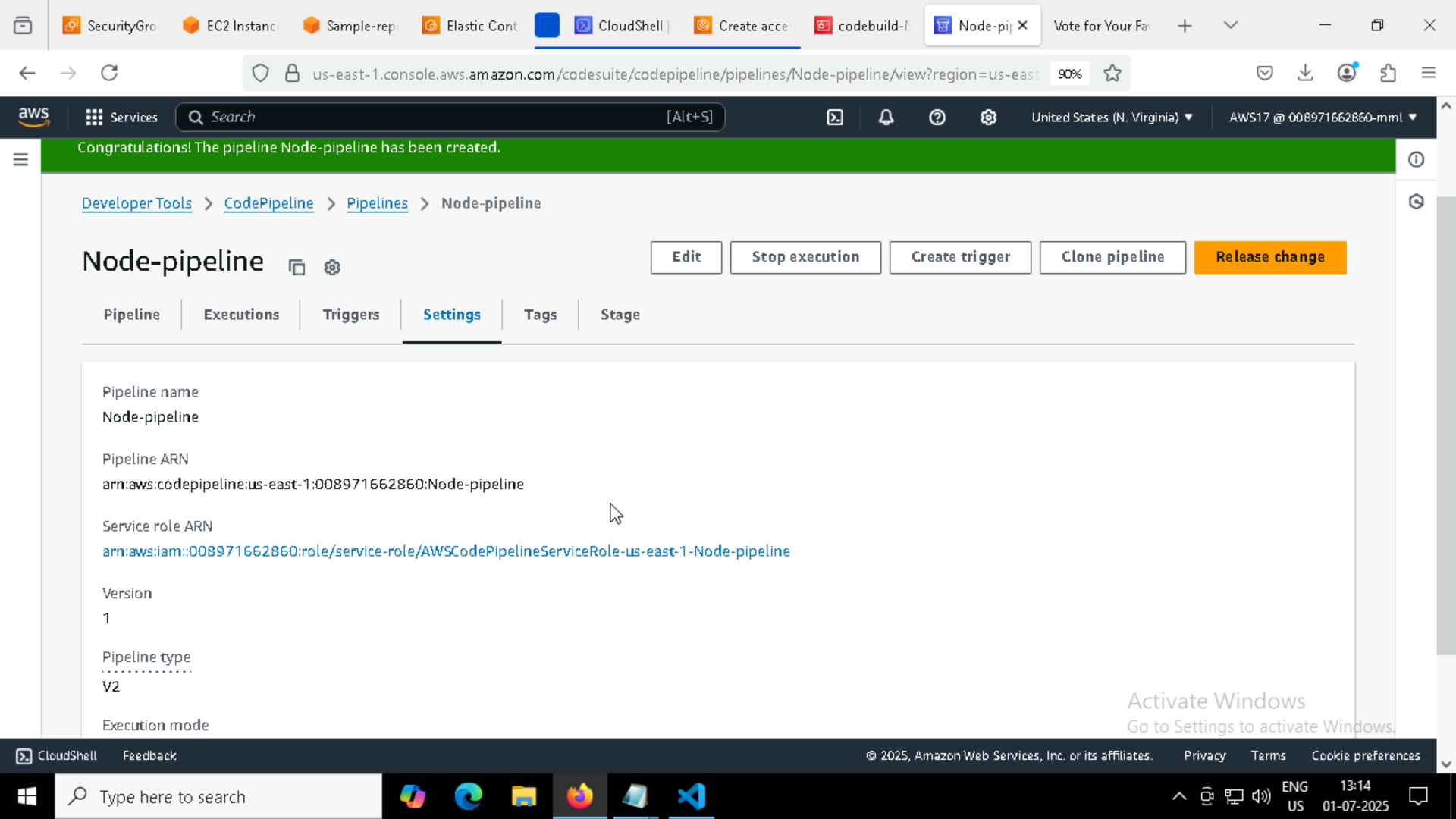


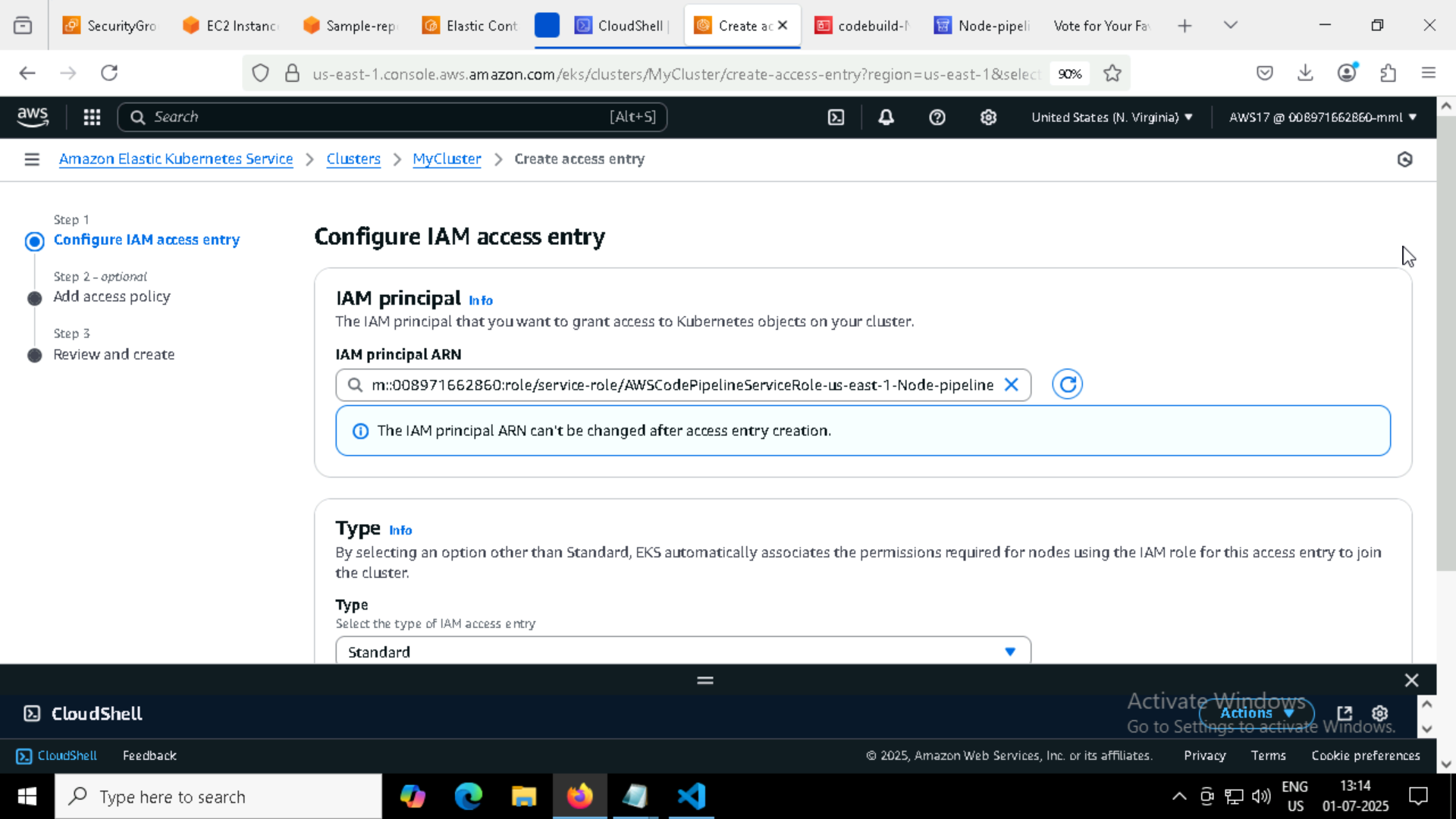
**Deploying in EKS cluster in codepipeline**

****

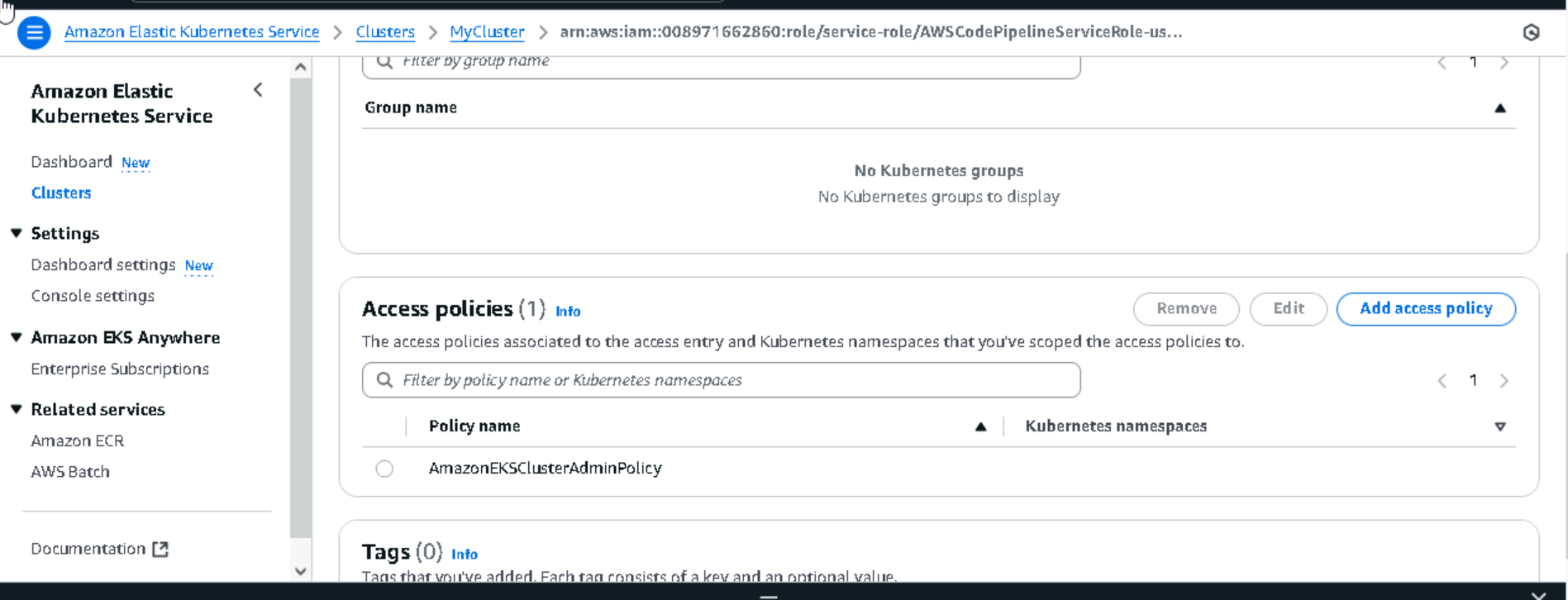
**Giving EKS Cluster Service Role of Our Node-pipeline**

**Kubernetes Service Connection:**

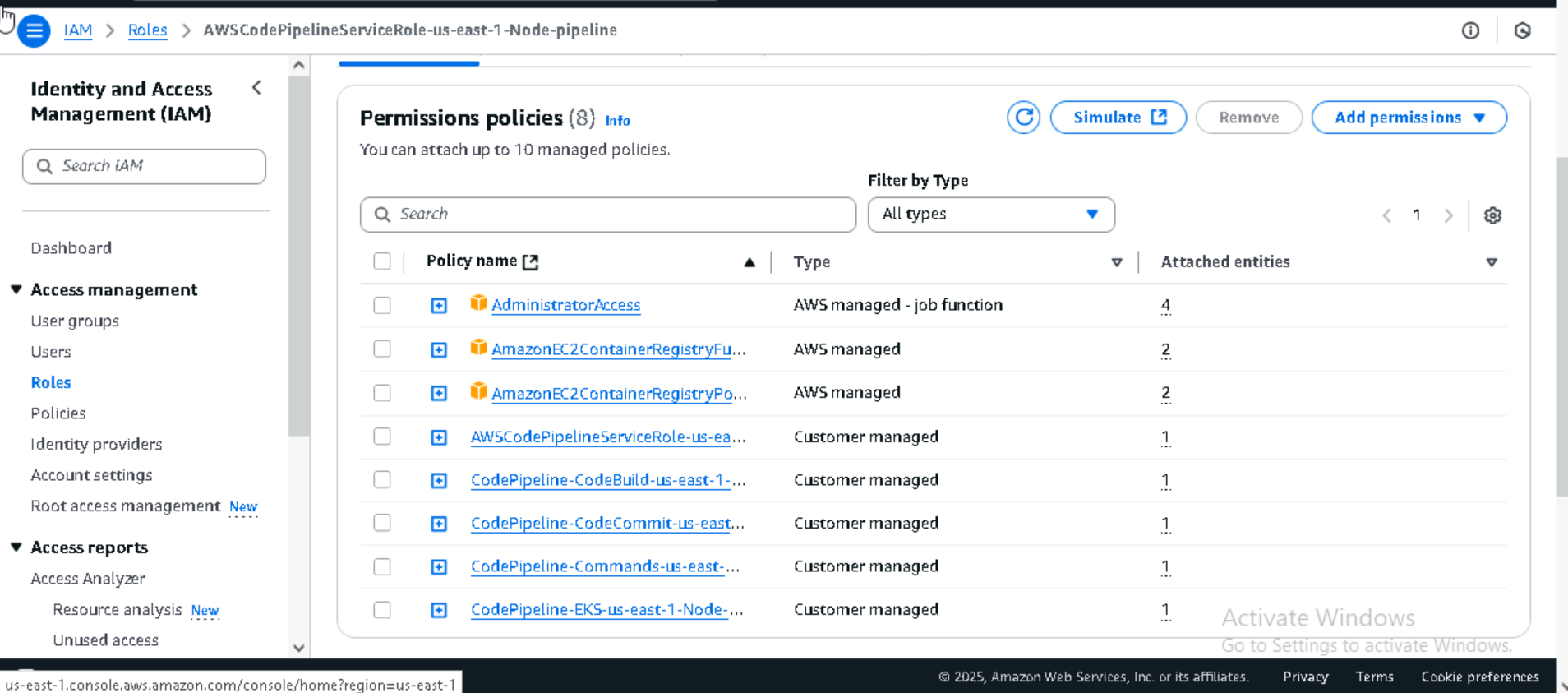




**Giving ClusterAdmin policy to our EKS cluster**

****

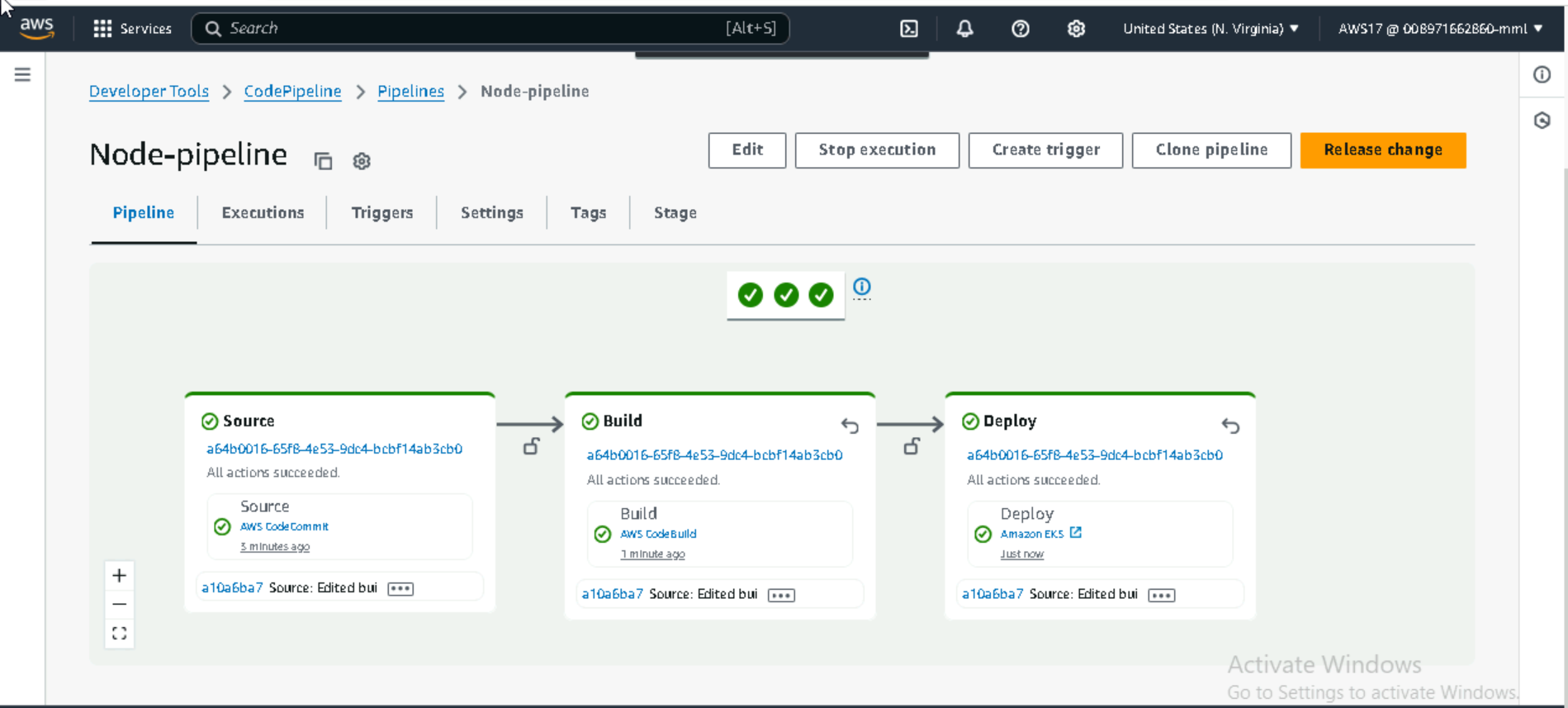
**ECR-EKR Proper integration by Providing Proper IAM Role for Our pipeline**

****

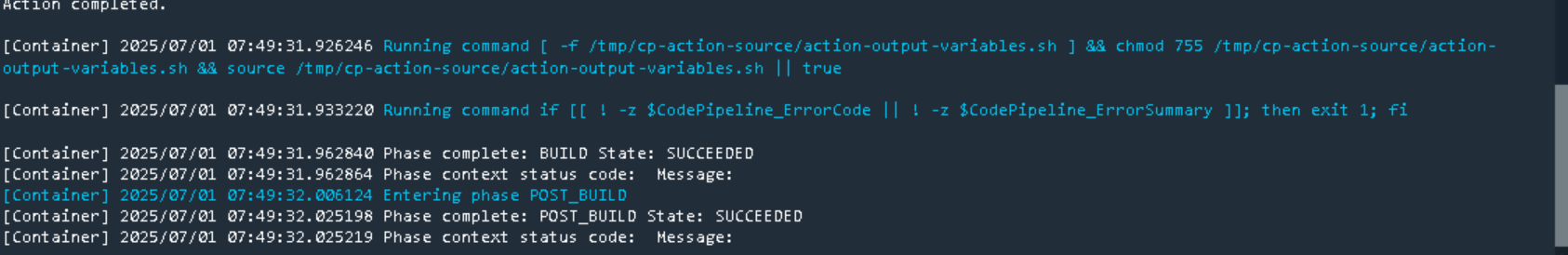
Giving access to EC2 container registry for full access to pull the image from ECR

**Pipeline deployment Succeded**

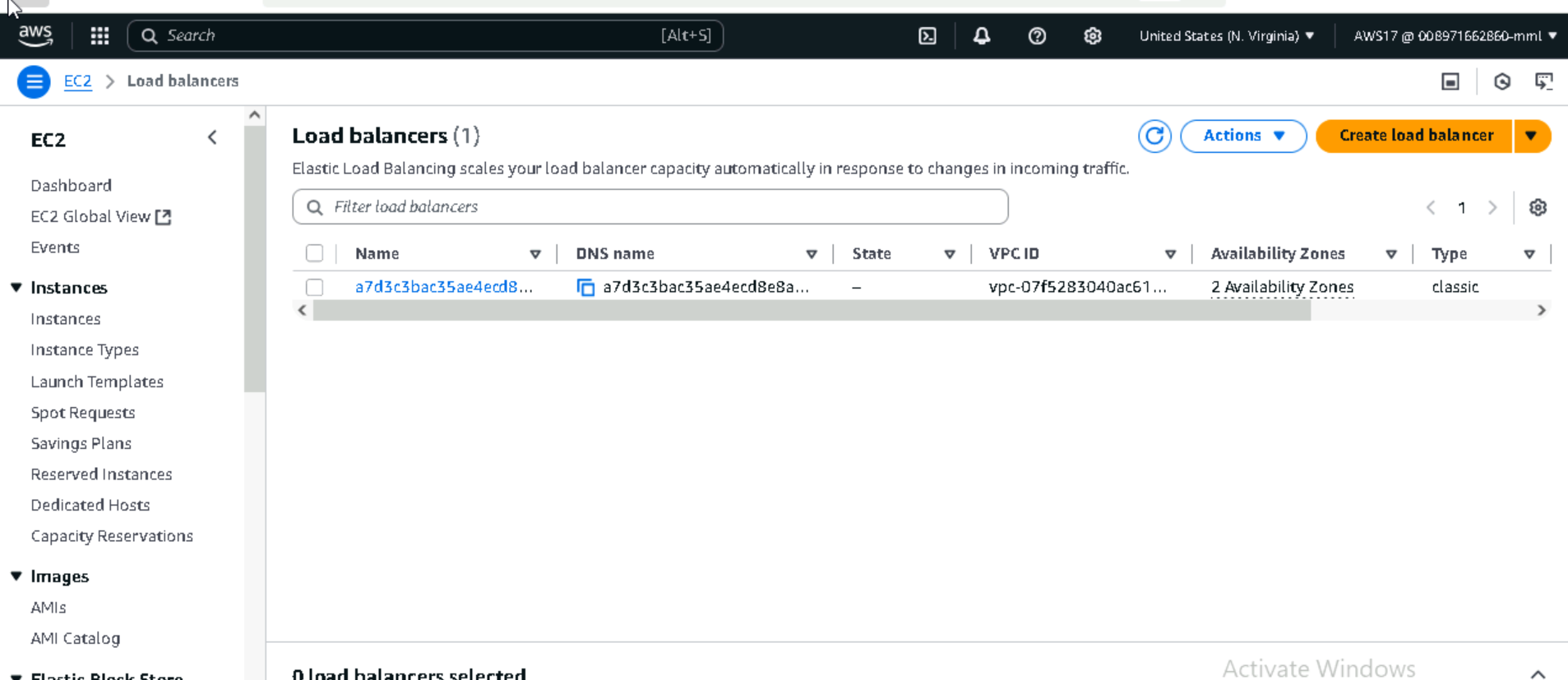
**Source I used as codecommit and Deployed my application in EKS cluster as you can see in below image**

****

**Pipeline logs**

****

**Checking the Loadbalancer**

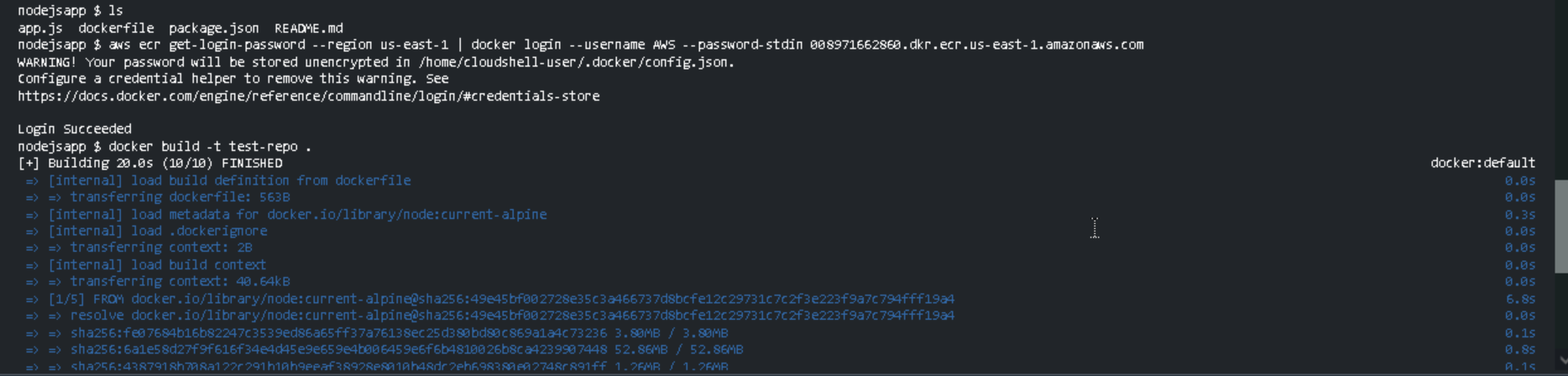
****

**Now lets Check if our nodejs application is running by accessing loadbalancer dns**

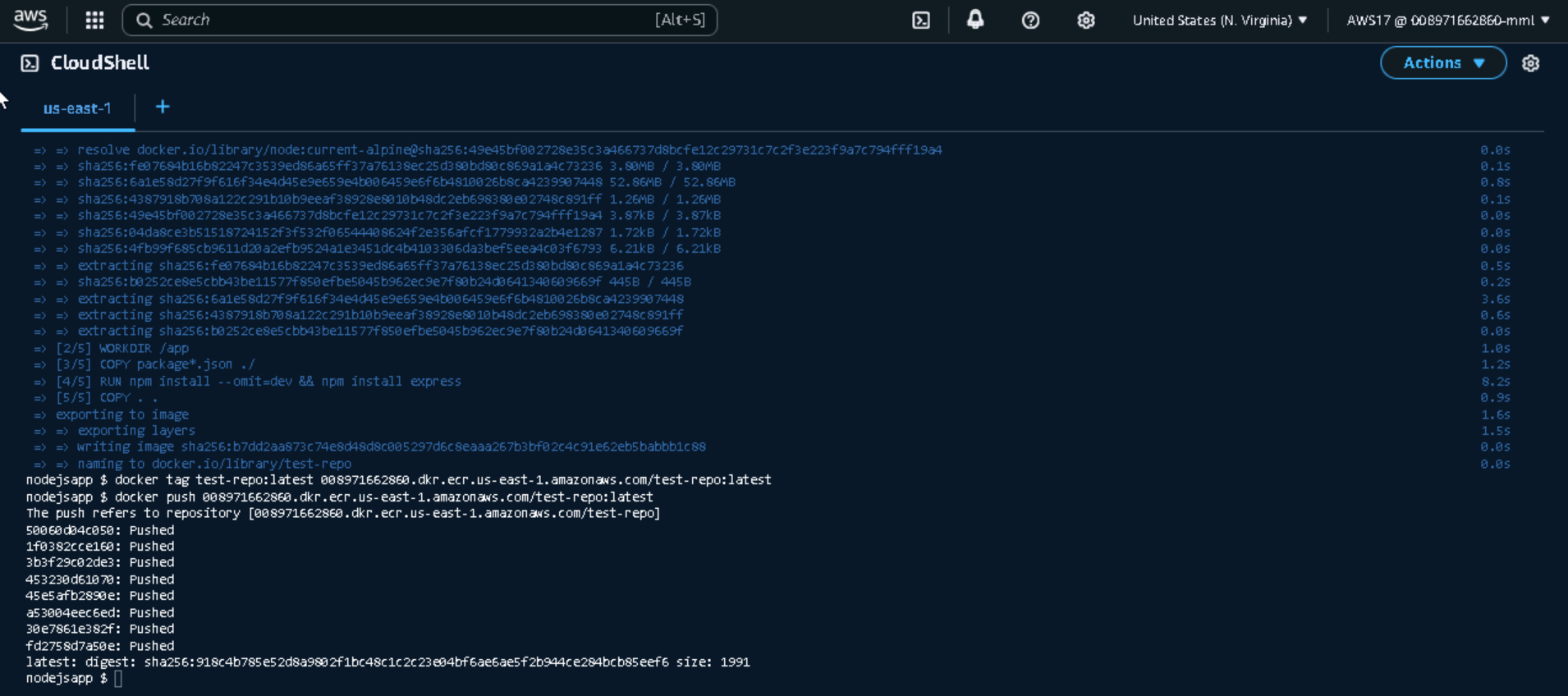
****

**Now doing manually and verifying if everything is working fine**

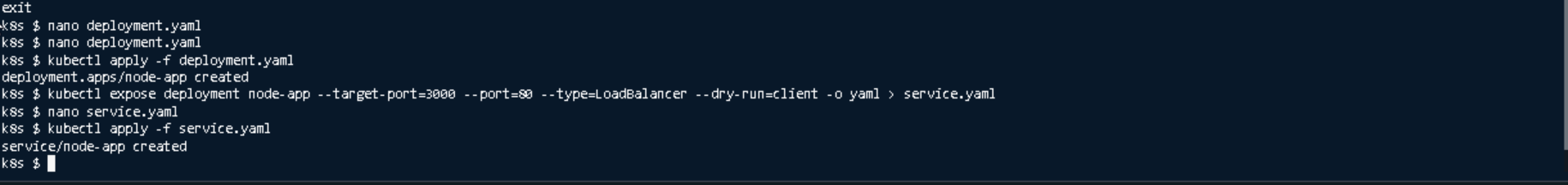
**Aws Ecr Login and Docker build :**

****

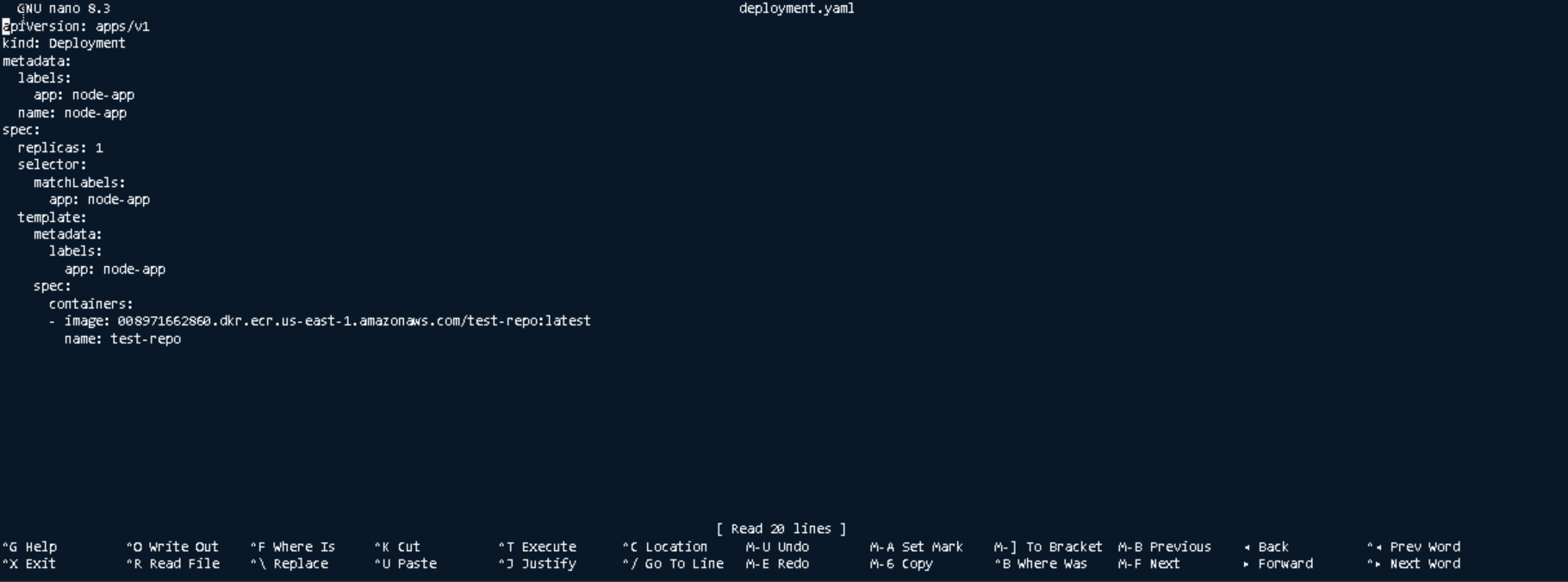
**Tagging and pushing the image**

****

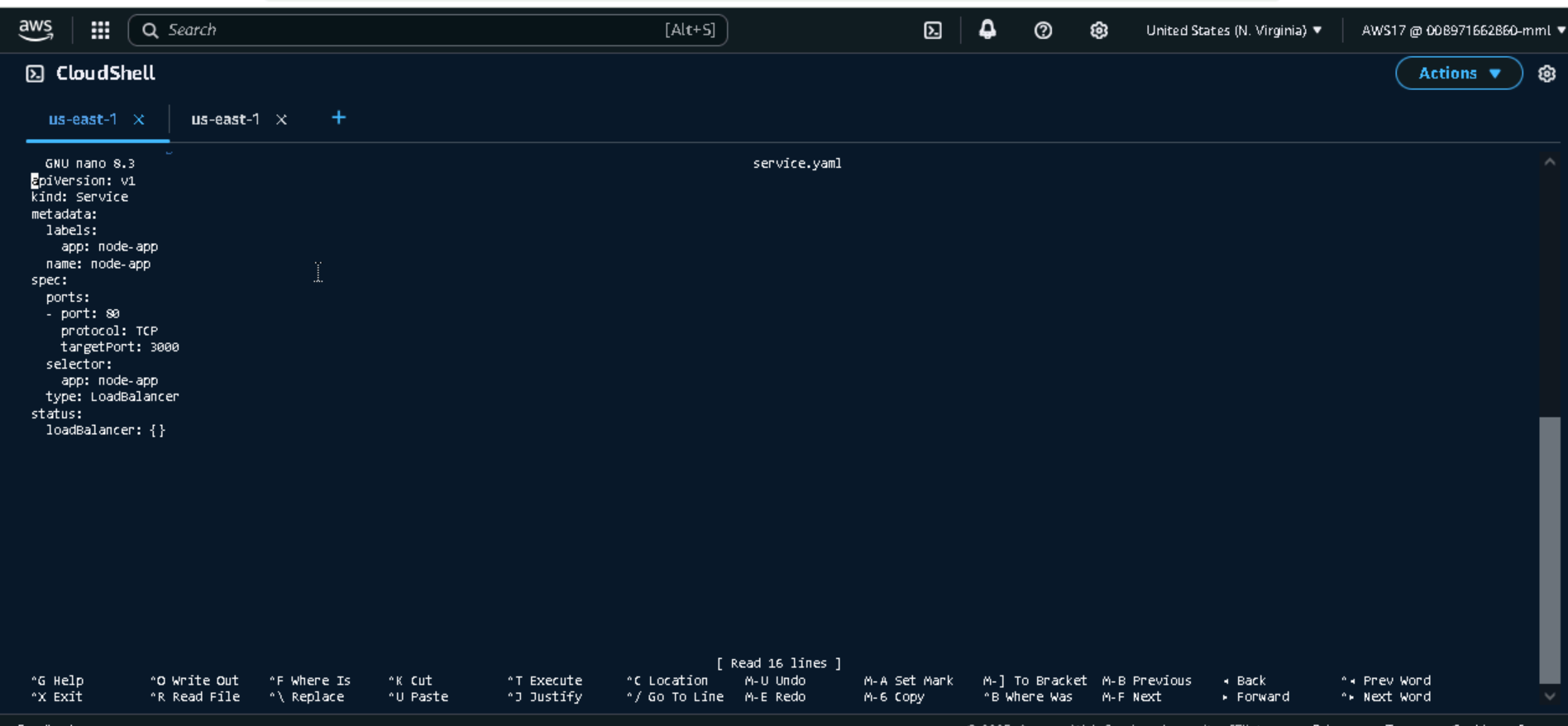
**Creating deployment.yaml and service.yaml then applying both**

****

**Deployment.yaml file**

****

**Service.yaml file**

****

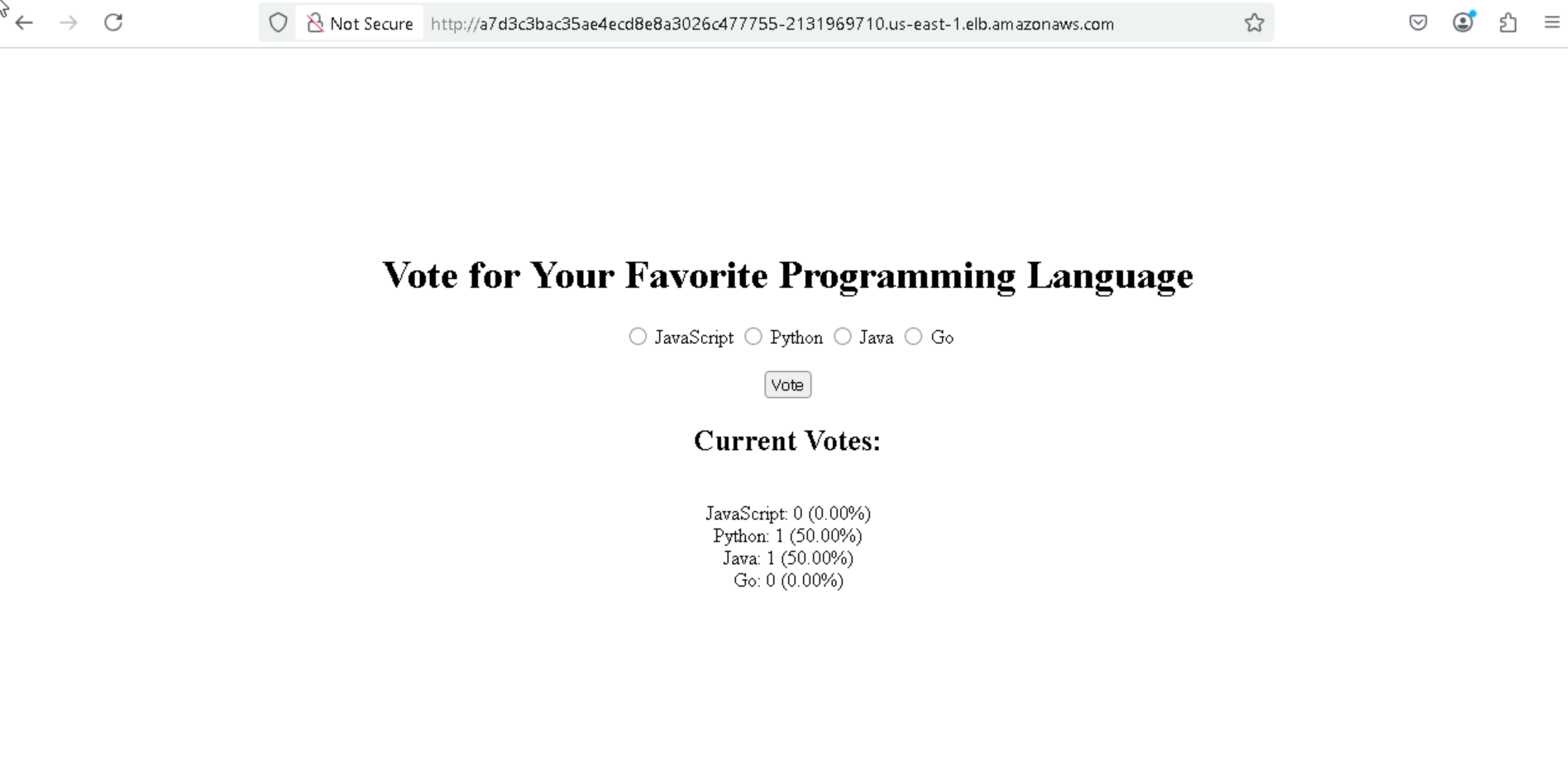
**Verifying the deployment and service yaml files**

****

**All my pods are up and in running state and loadbalancer is also created**

**Accessing Loadbalancer through Browser**

**http://<loadbalancer-IP>**

****

**Application is Up and running successfully**