**Cprime**

**Step :1 Deploy a web application to Azure. You can write this web application, or it can be an open-source project.**

Small Application we I created.

import uvicorn

from fastapi import FastAPI

app=FastAPI()

@app.get('/')

def sample():

    return 'Hello, Cprime Team! This is sample web application deployed on Azure'

if \_\_name\_\_=='\_\_main\_\_':

    uvicorn.run(app, host='0.0.0.0', port=8000)

**Step :2   
Deploy the web application as a Docker Container.**Docker file

FROM python:3.9-slim

WORKDIR /Cprime

COPY requirements.txt /Cprime/

RUN pip install --no-cache-dir -r requirements.txt

COPY . /Cprime/

EXPOSE 8000

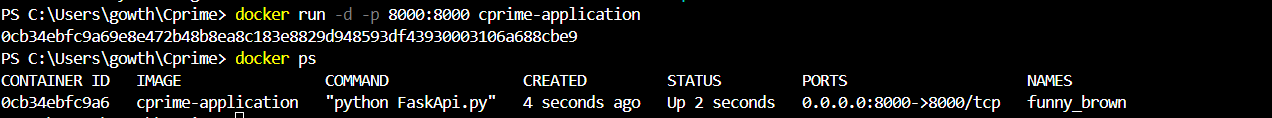
CMD ["python","FaskApi.py"]

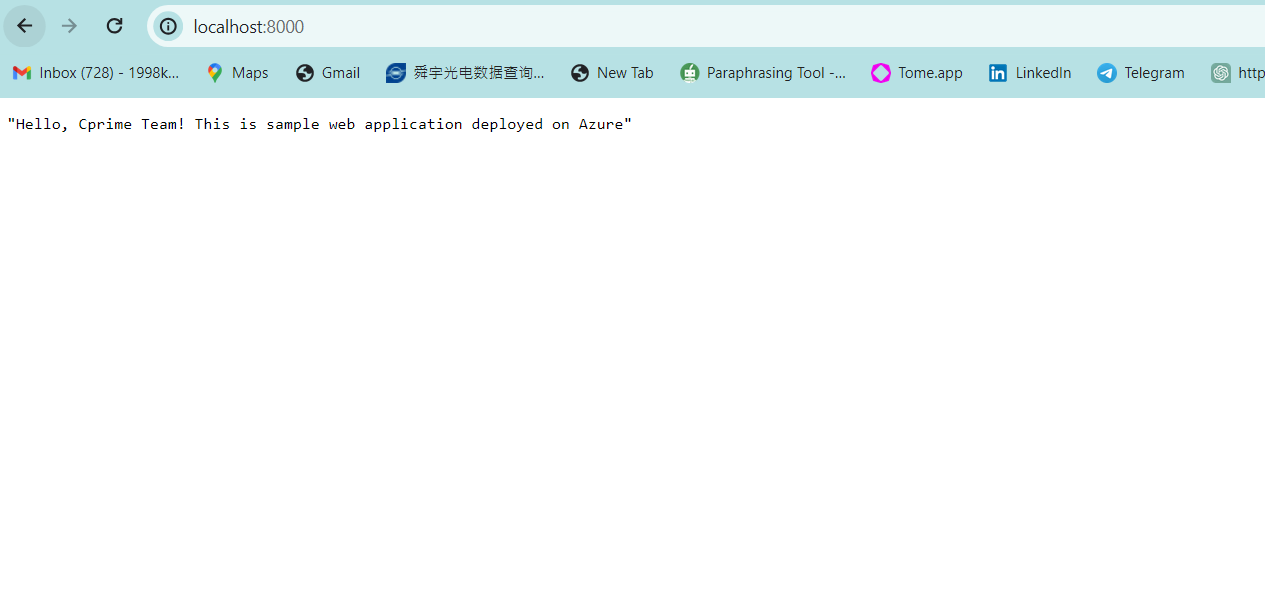
Requirement.txt

fastapi

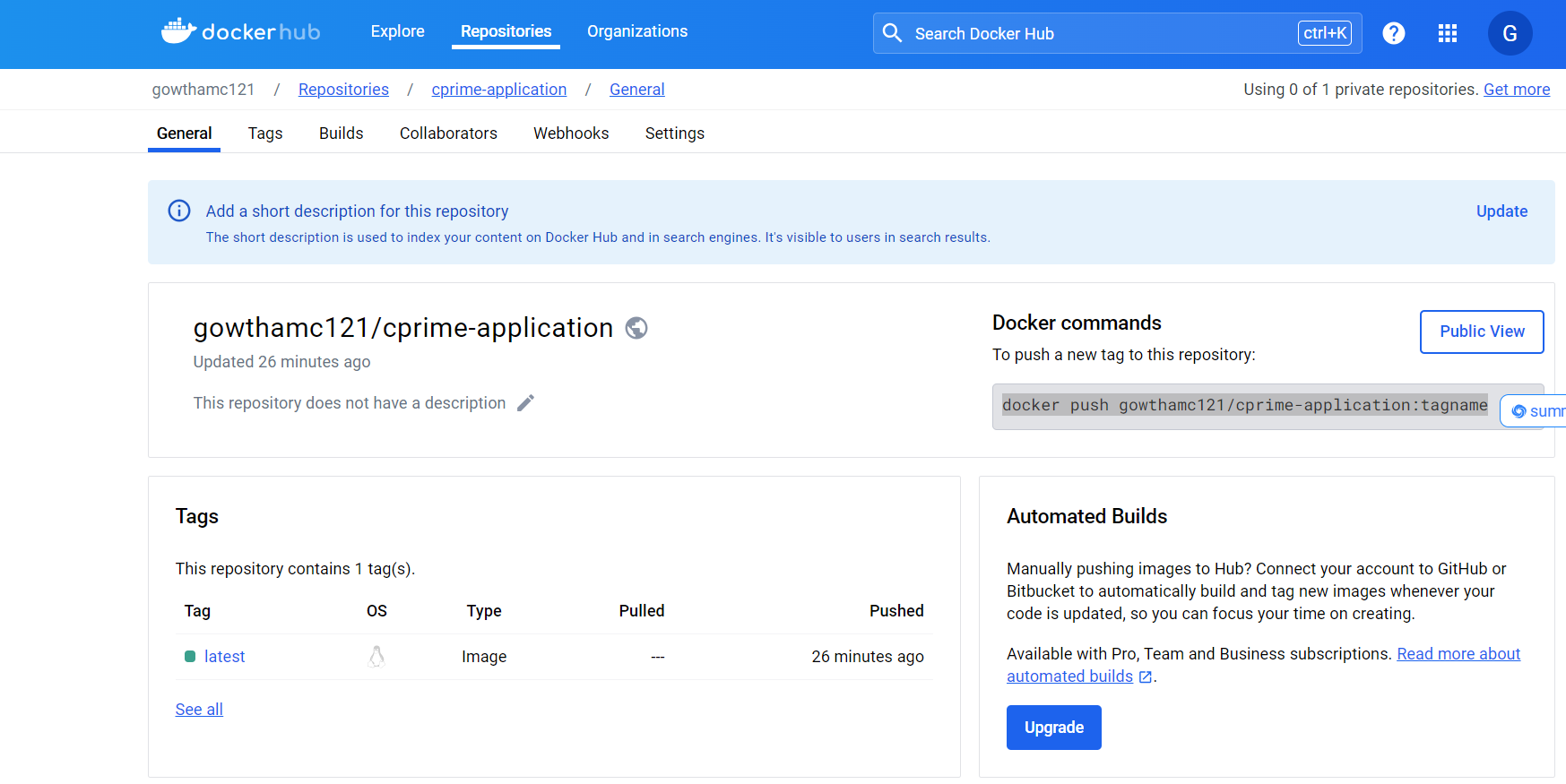
uvicorn

docker image and container created

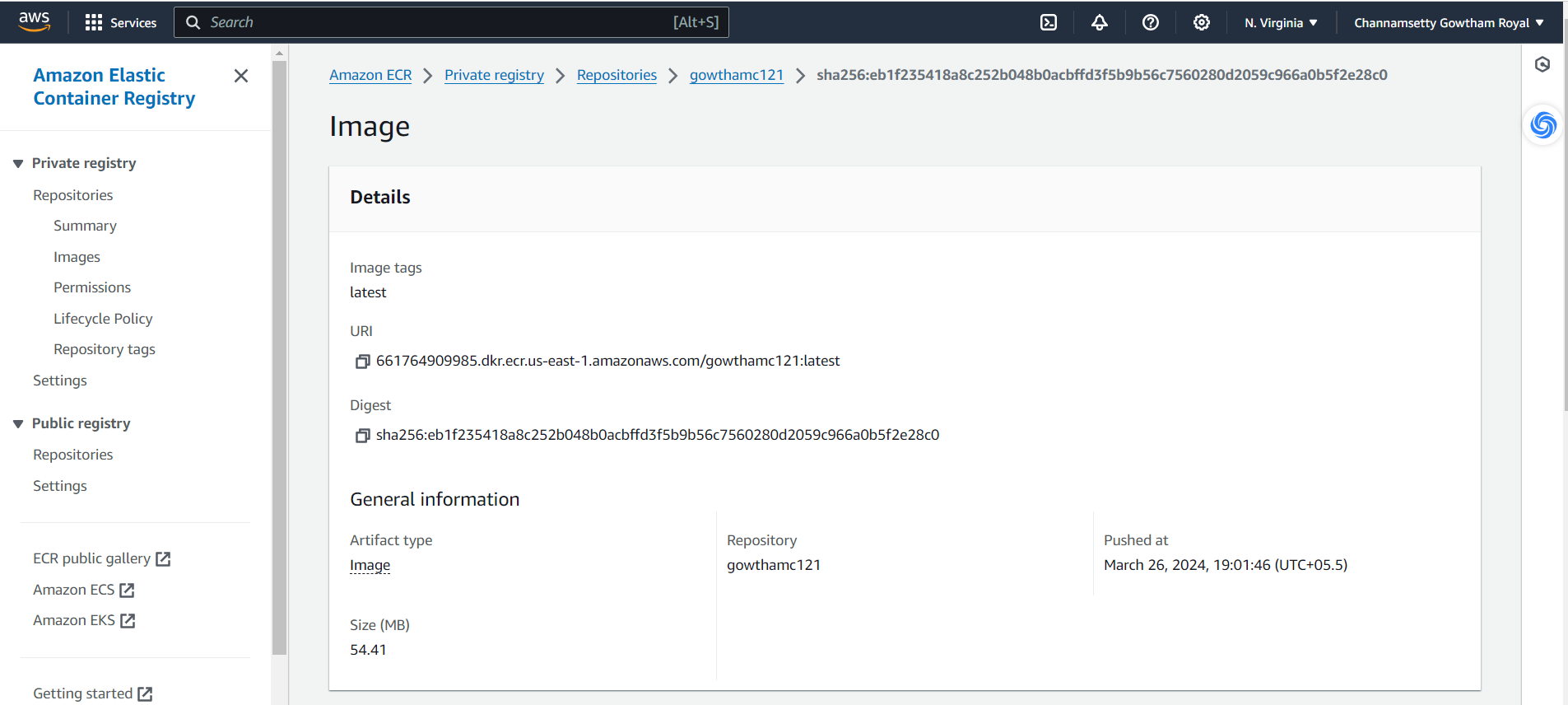




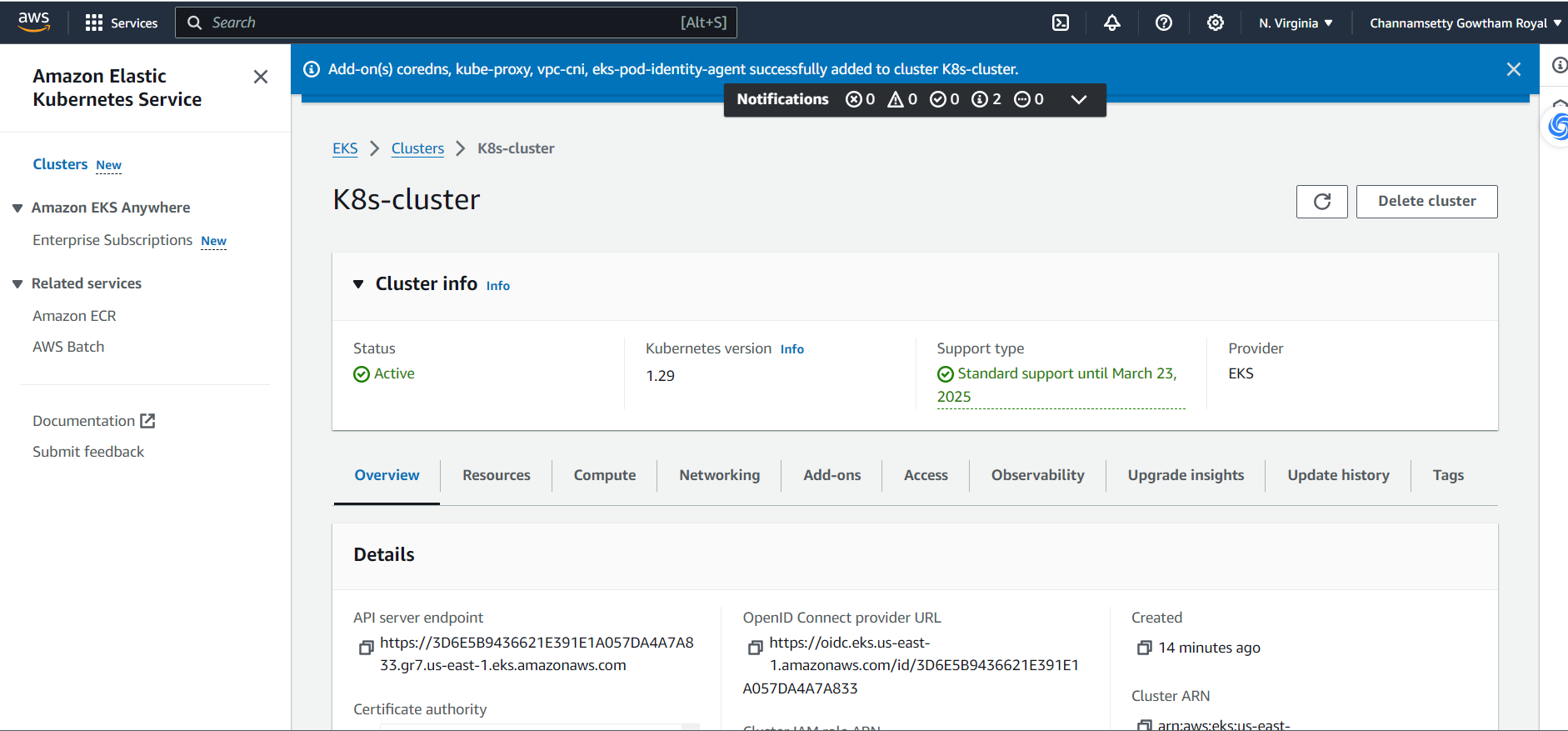
I pushed the image in Docker HUB

docker push gowthamc121/cprime-application:tagname  
  


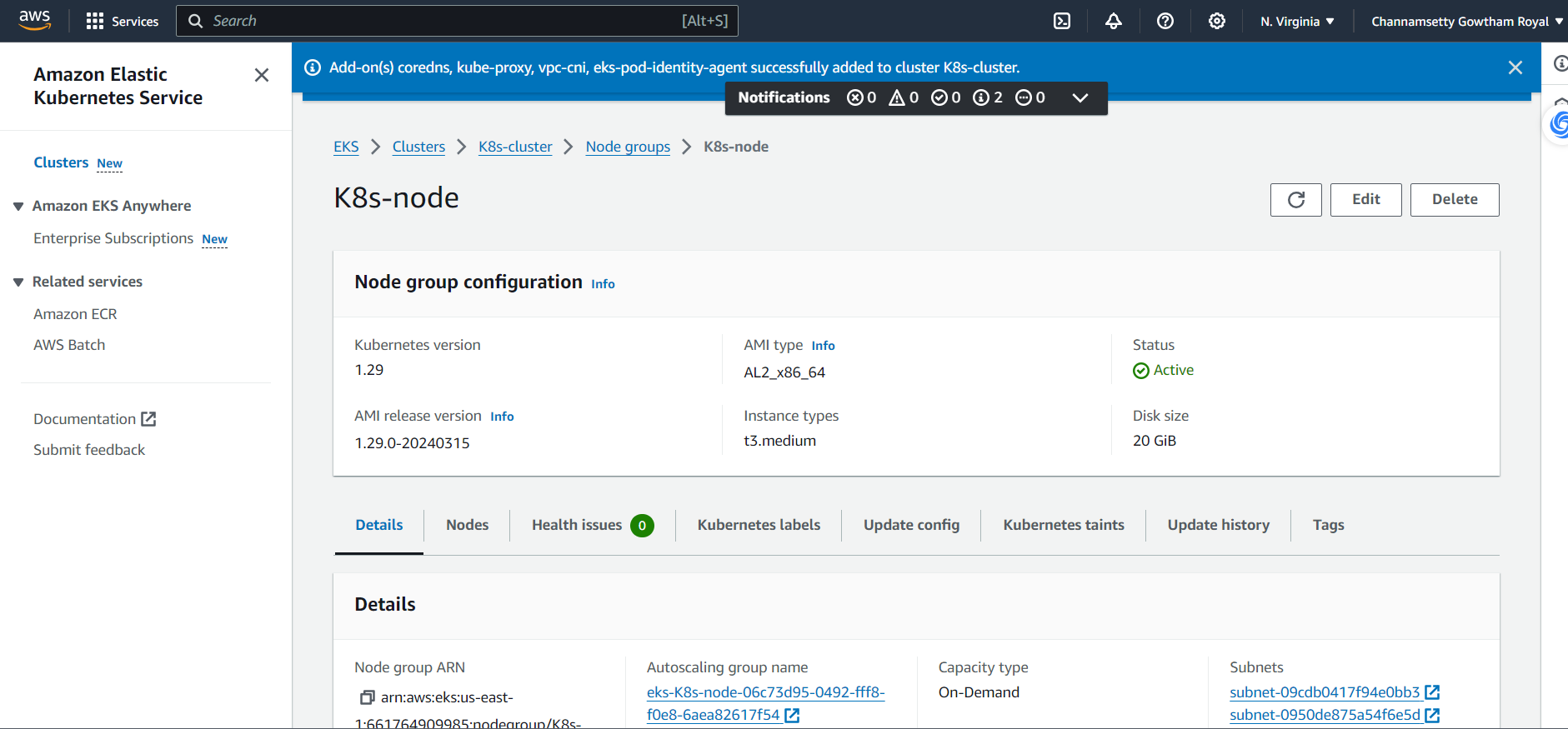
**I pushed the image in AWS ECR as well.**

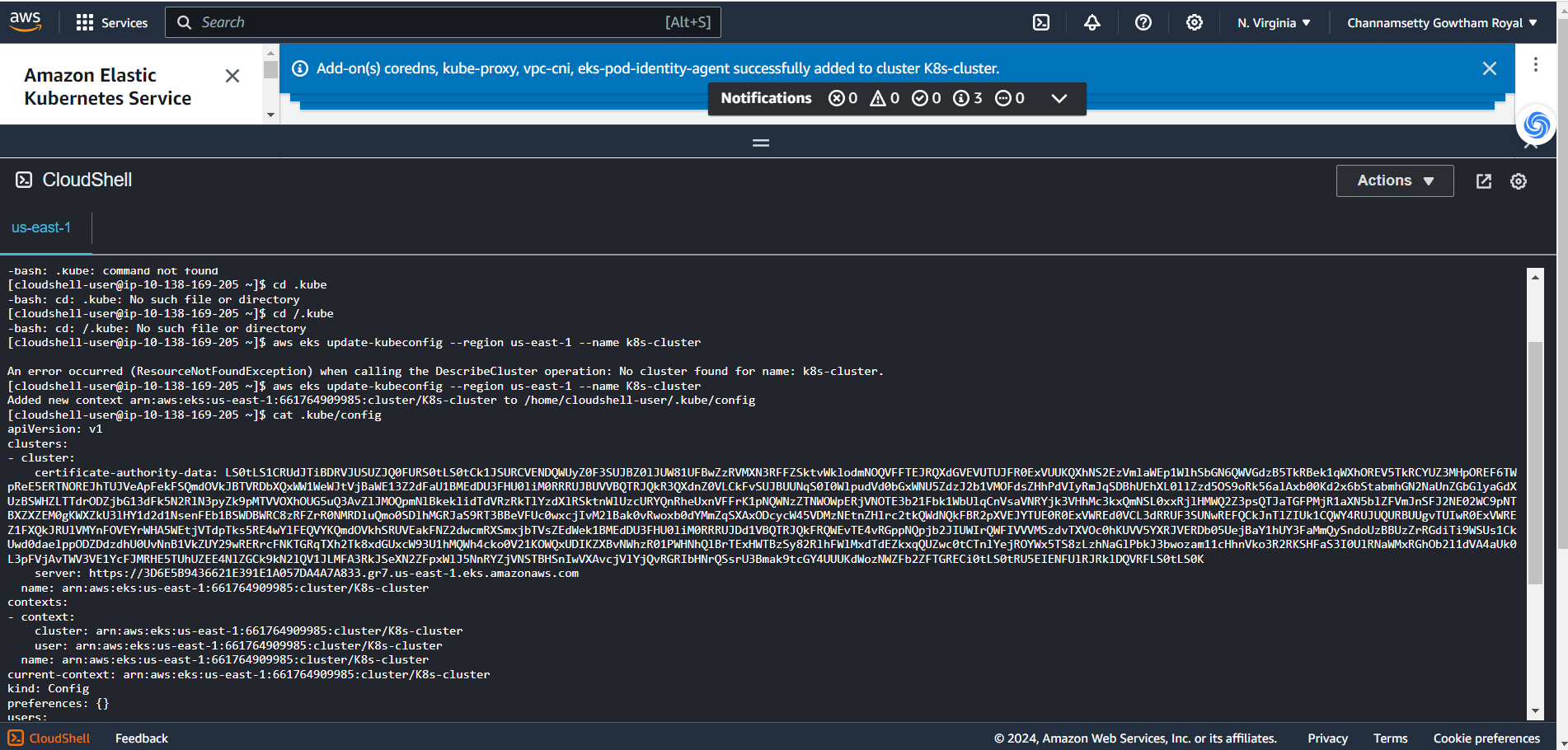


Created the EKS Cluster  
  
I created the VPC, Subnets, Security groups and also I created the Roles for cluster and node level



Created the Node



Set the ./kube/config file  
  


Created the Pod.yaml

apiVersion: v1

kind: Pod

metadata:

  name: cprime-application-pod

  lables:

    app: cprime-application

spec:

  containers:

  - name: cprime-application-container

    image: 661764909985.dkr.ecr.us-east-1.amazonaws.com/gowthamc121:latest

    ports:

    - containerPort: 8000

Created the Service.yaml

apiVersion: v1

kind: Service

metadata:

  name: cprime-application-service

spec:

  selector:

    app: cprime-application

  ports:

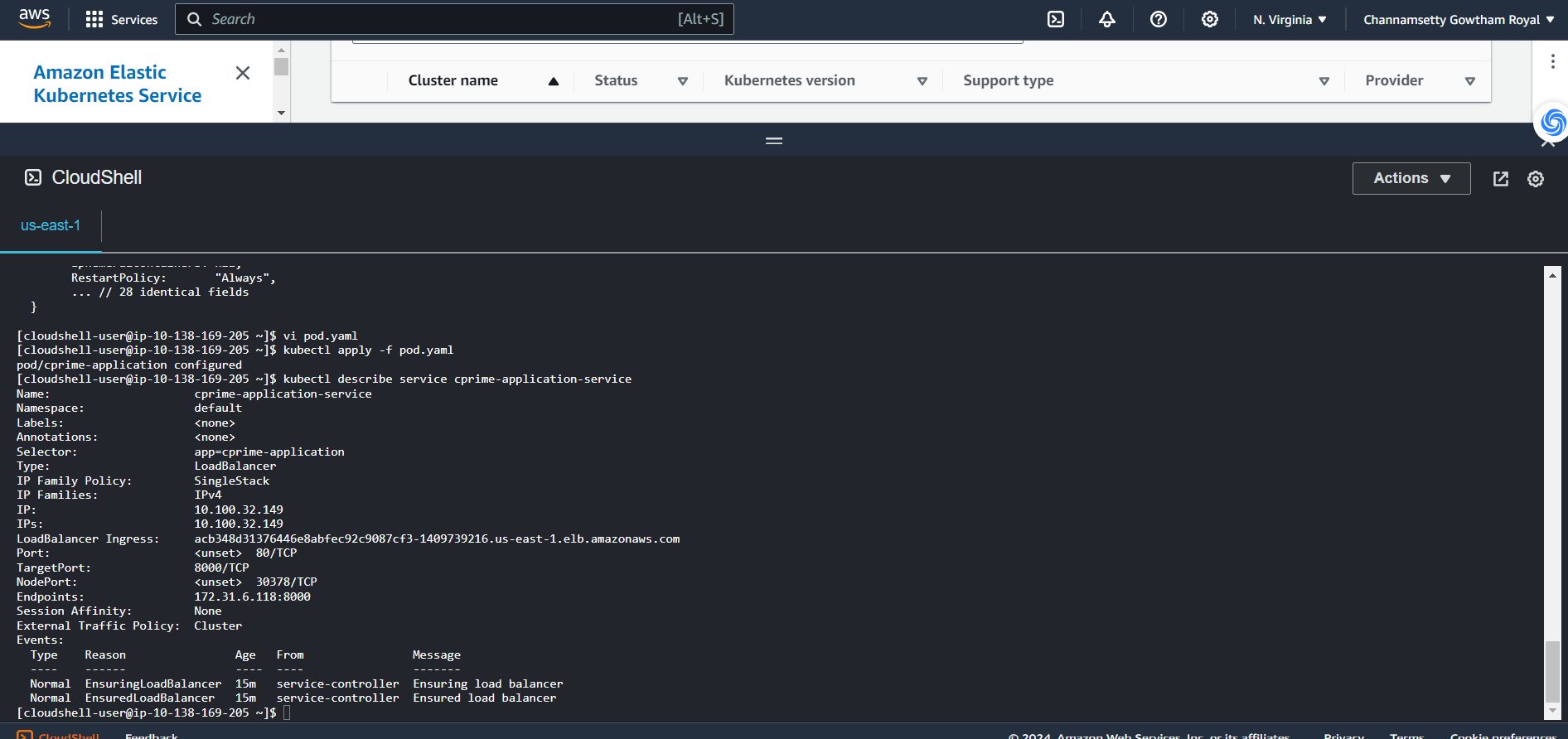
  - protocol: TCP

    port: 80

    targetPort: 8000

  type: LoadBalancer

LoadBalancer Created

  
  
I expose by LB  
  
