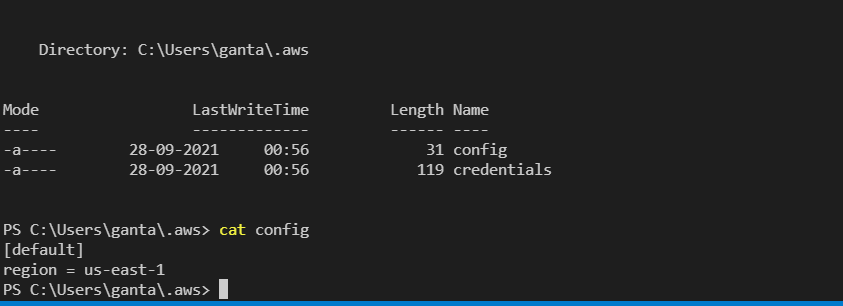
**UpGrad Course 5-Capstone Project**

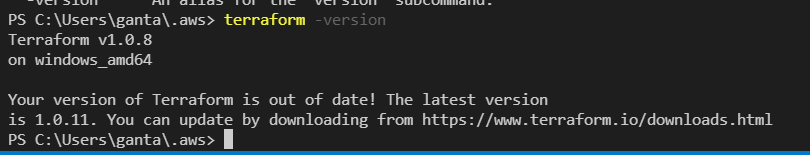
GitHub :<https://github.com/PrashanthGanta/DevOps_Kubernetes>

# **Task 1: Setup EKS Cluster**

AWS Configuration

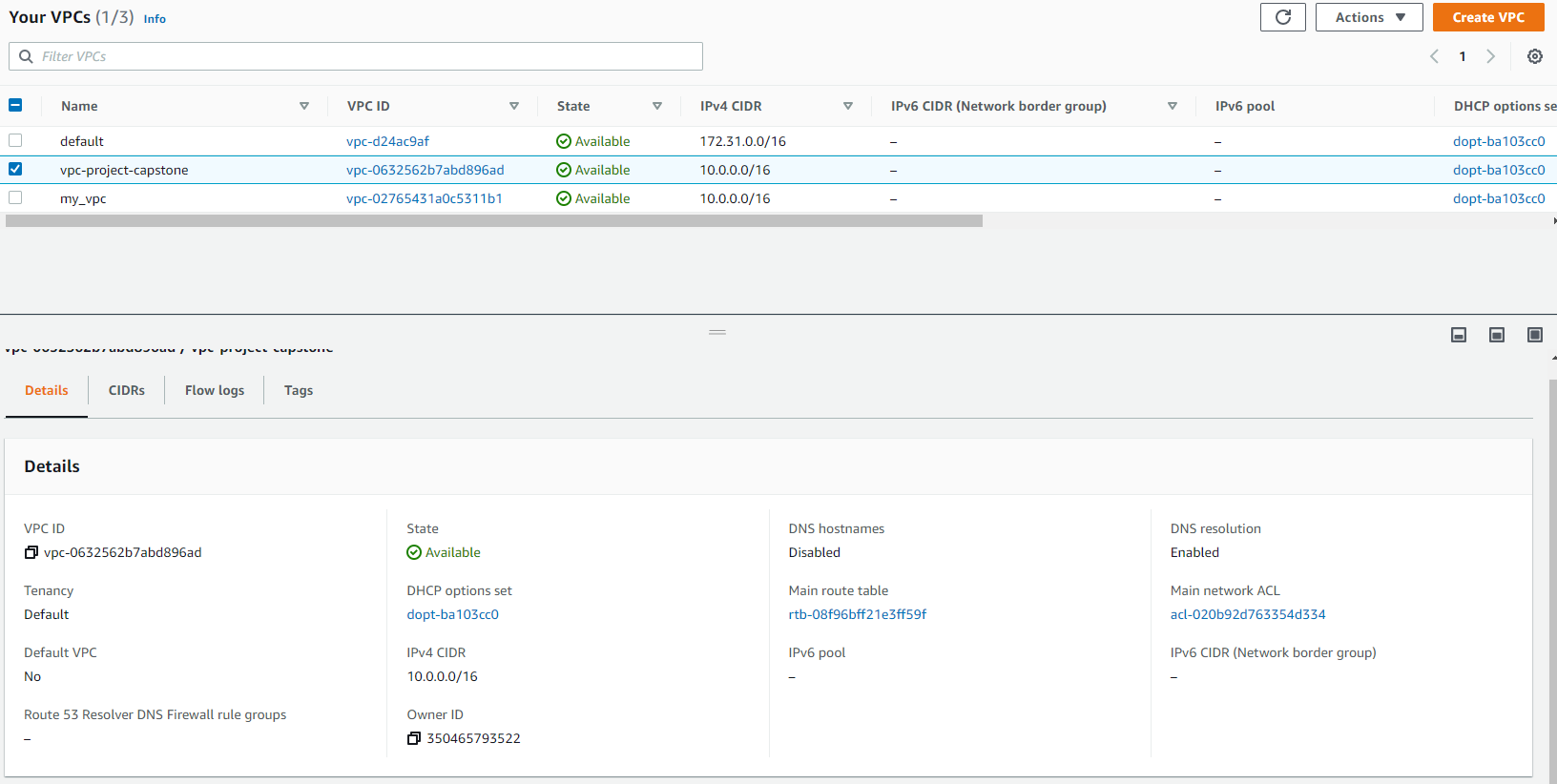


Terraform-configuration

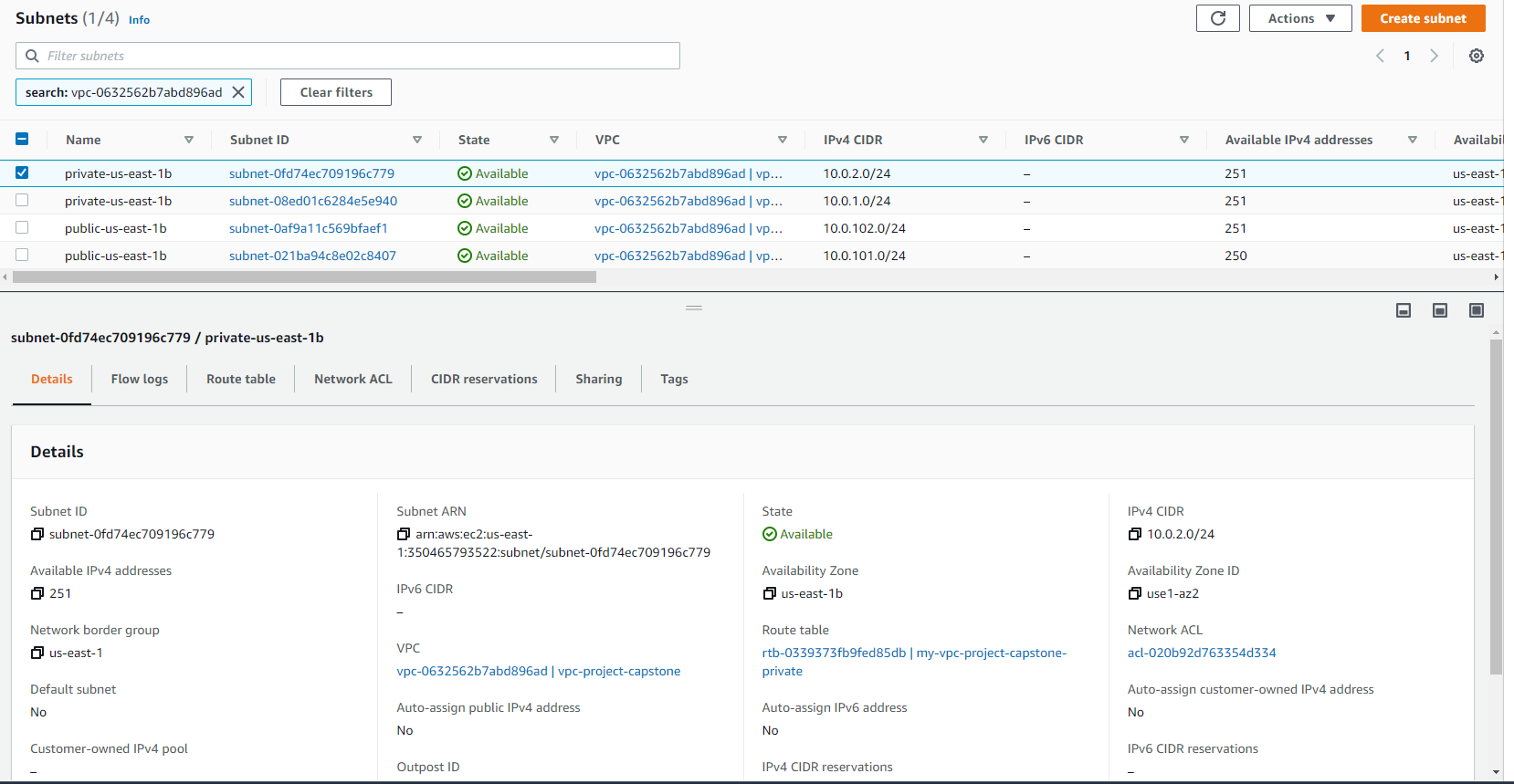


VPC- Creation

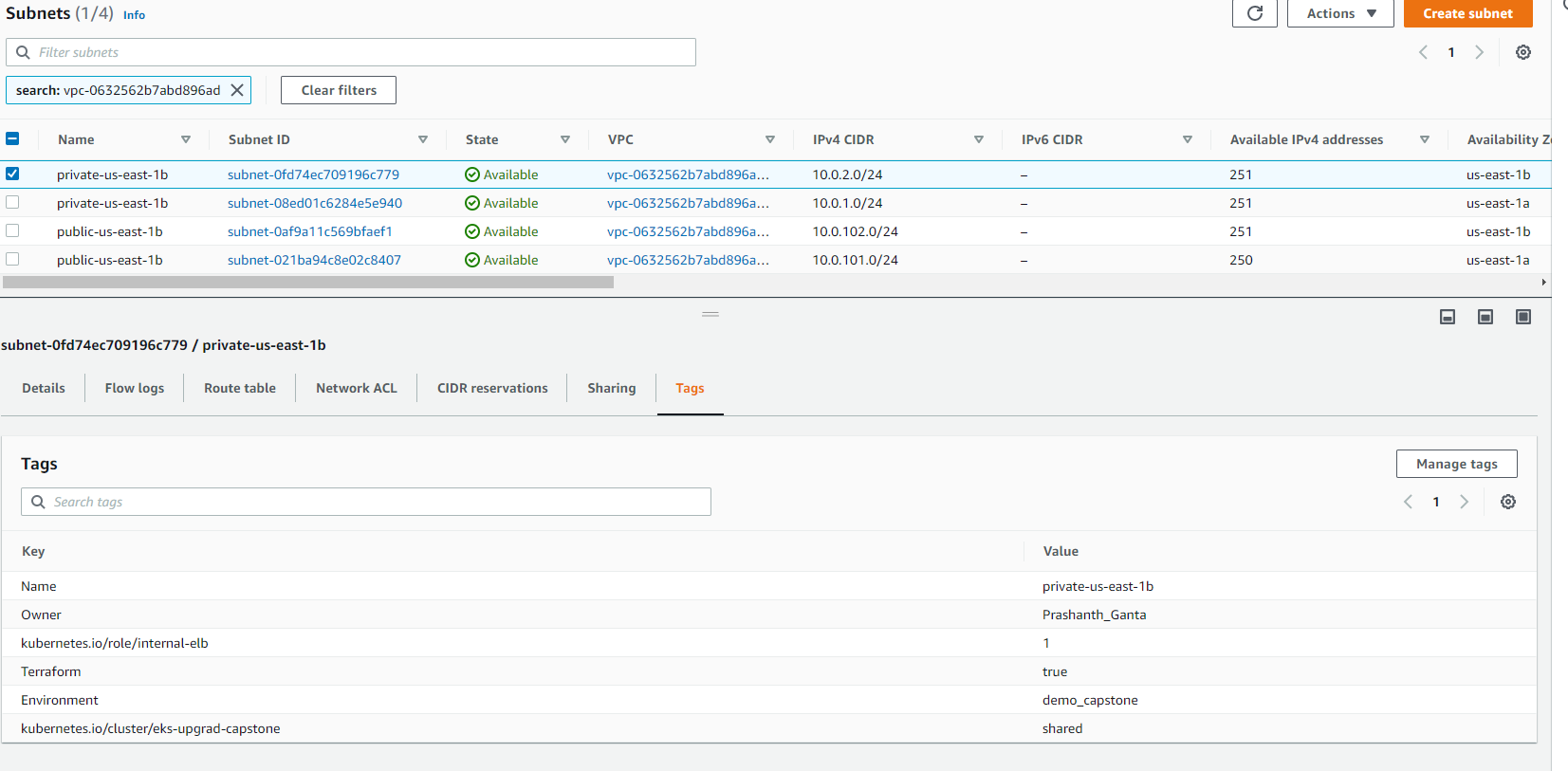
VPC



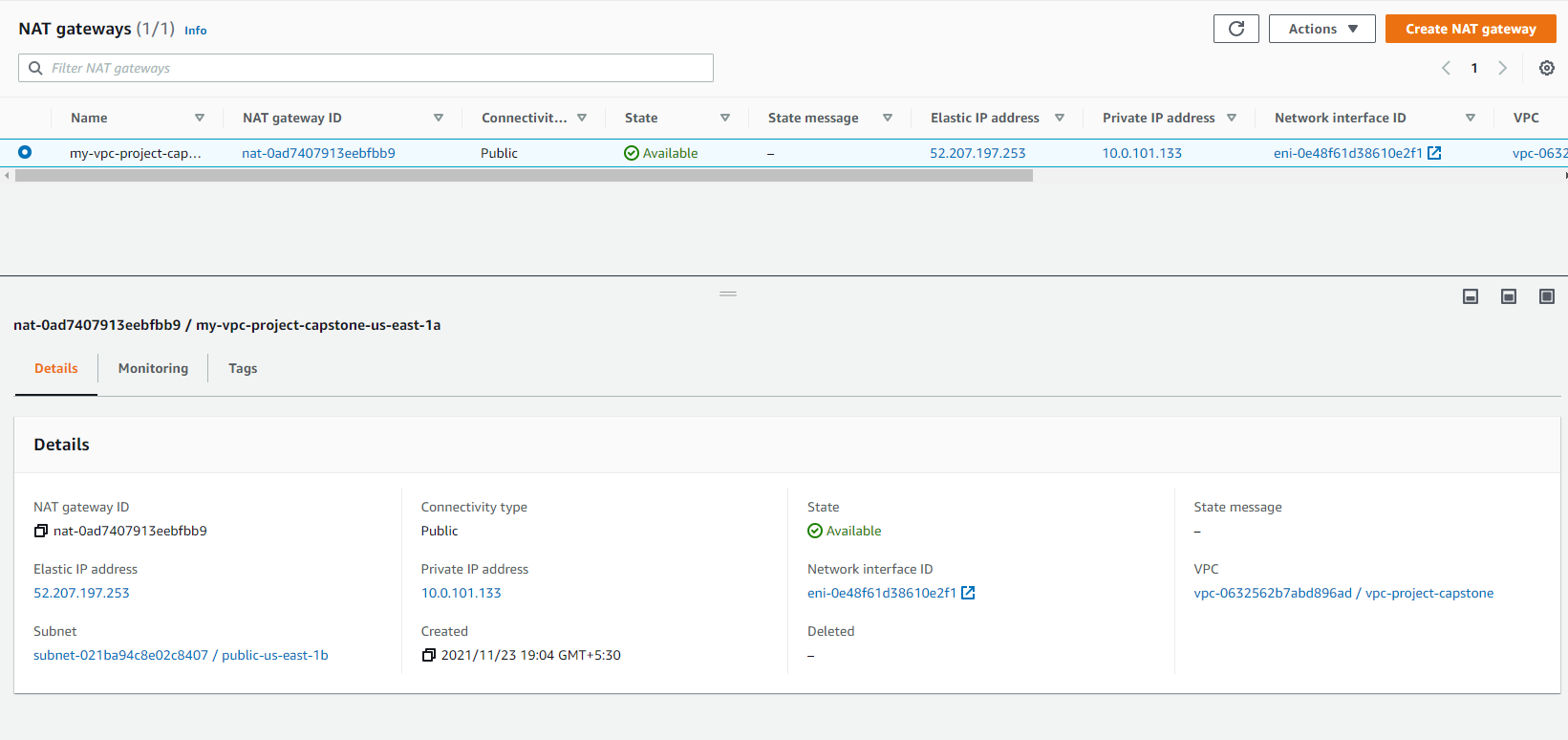
Subnets



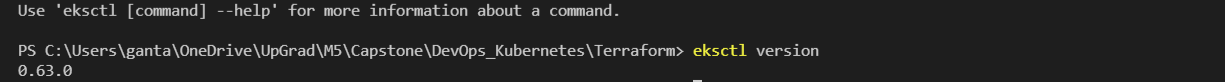
Subnets tags



Nat Gateway



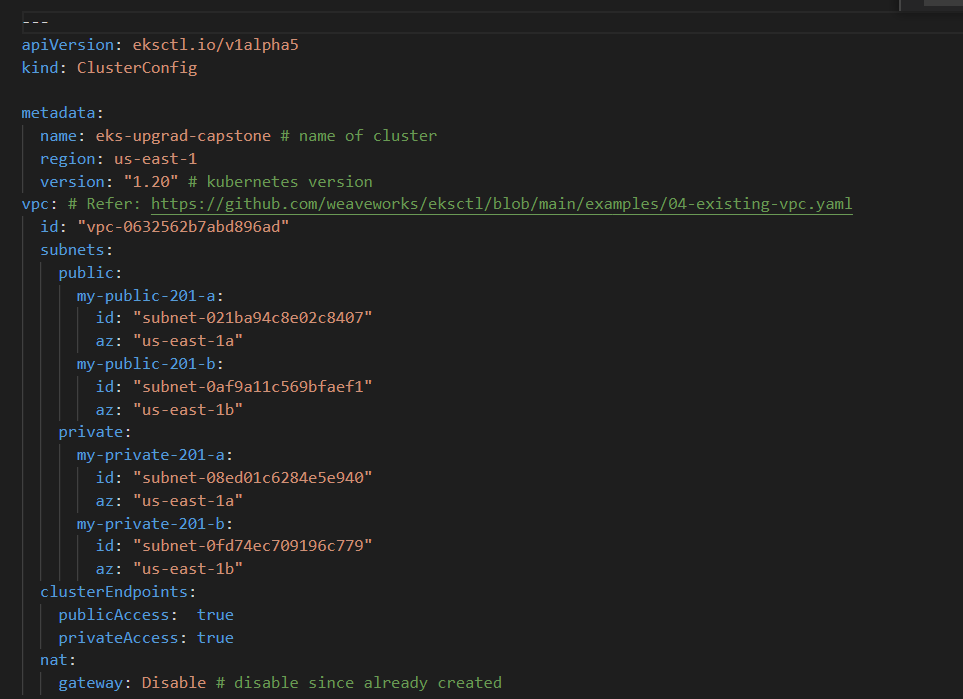
Eksctl setup



**Creating AWS EKS Cluster**

My-eks-conf.yaml

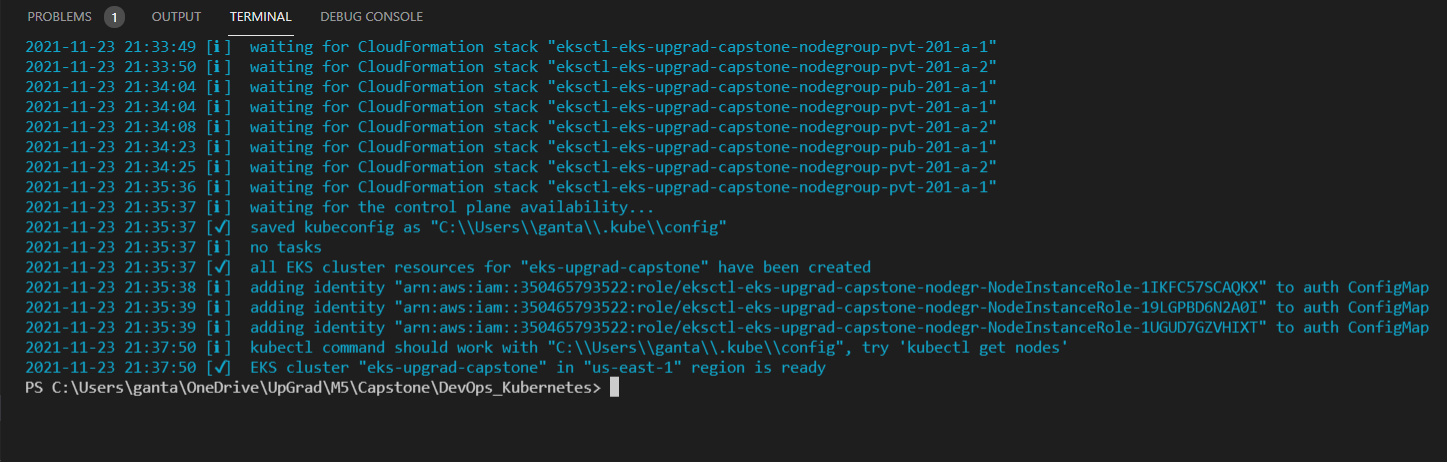
Provide the VPC and Subnets details created with VPC in cluster file according to requirements



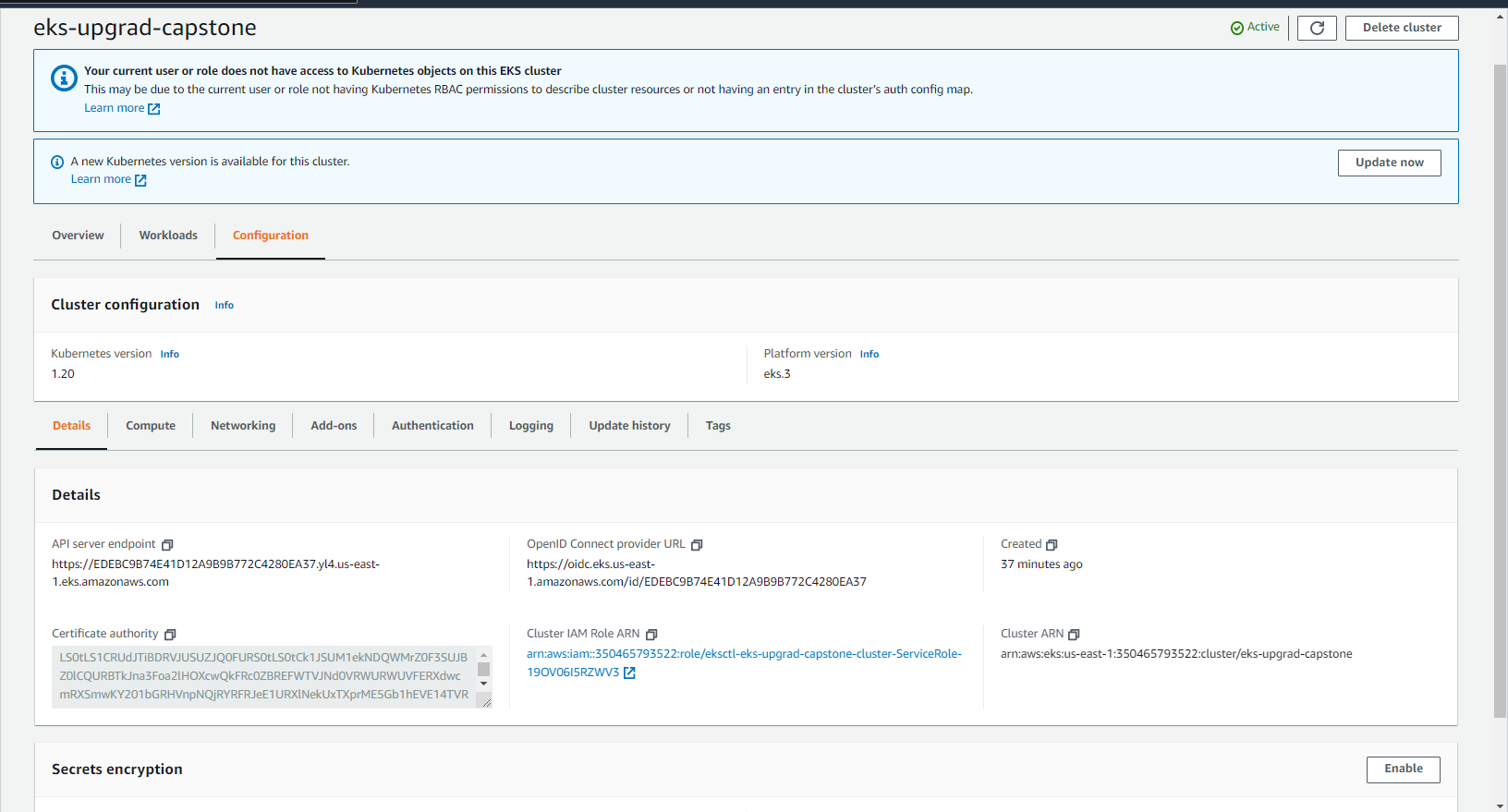
Run: eksctl create cluster -f my-eks-conf.yaml

Or run: make create\_cluster (this is the direct method of running the files with make command)

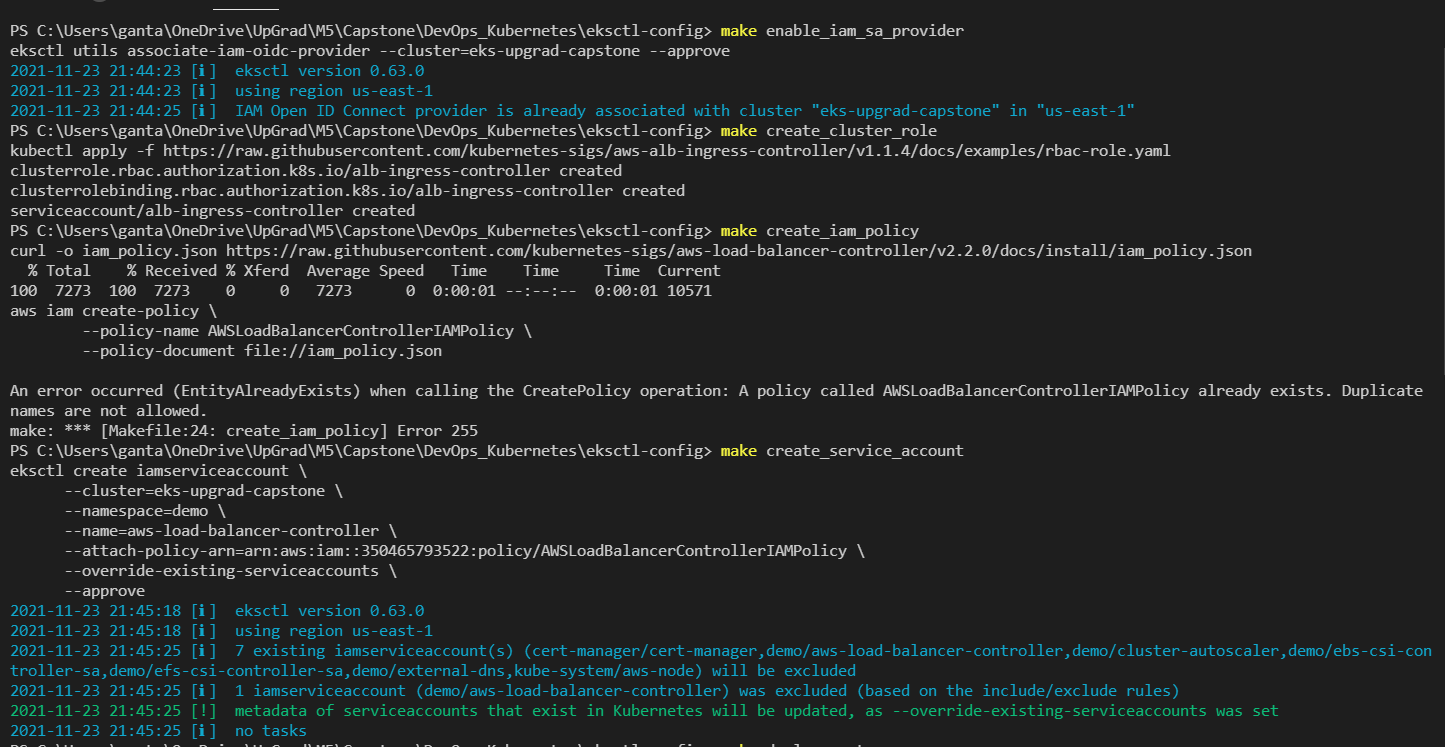
Wait for 15-20 minutes



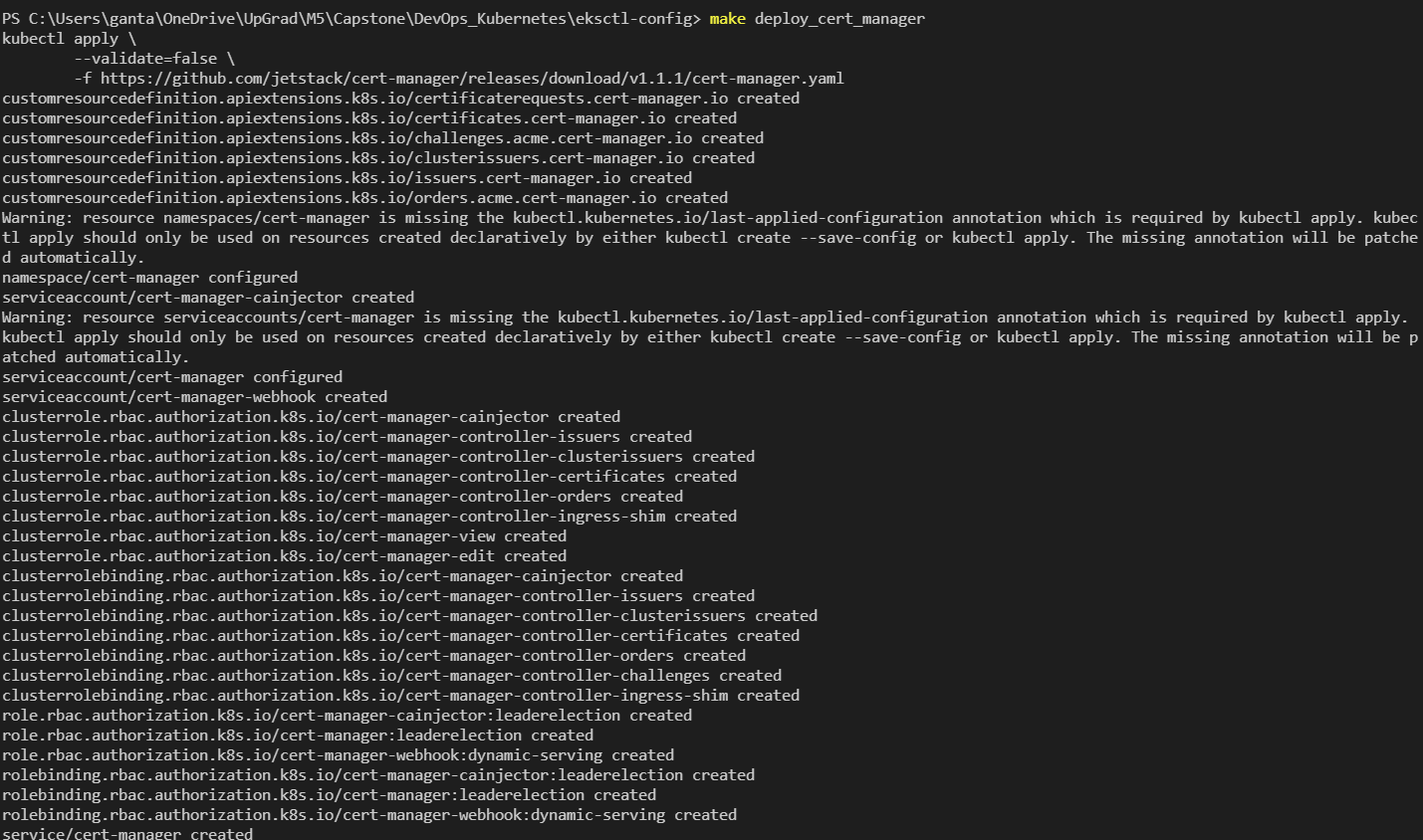
EKS cluster in console



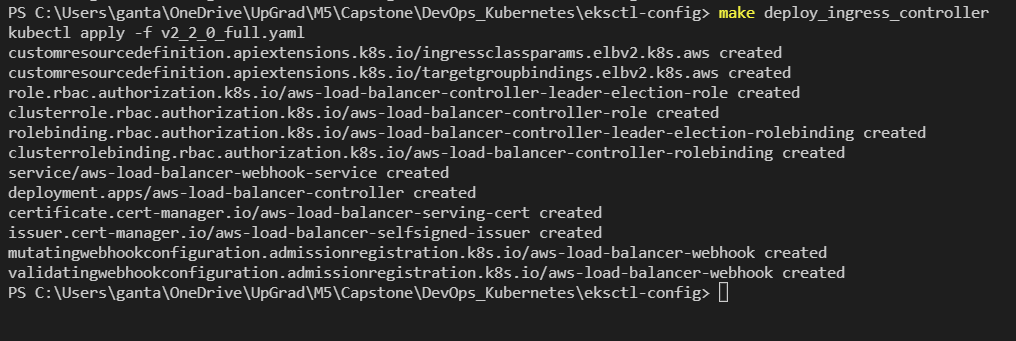
Install the following add ons to the EKS cluster:



Deploying Cert manager

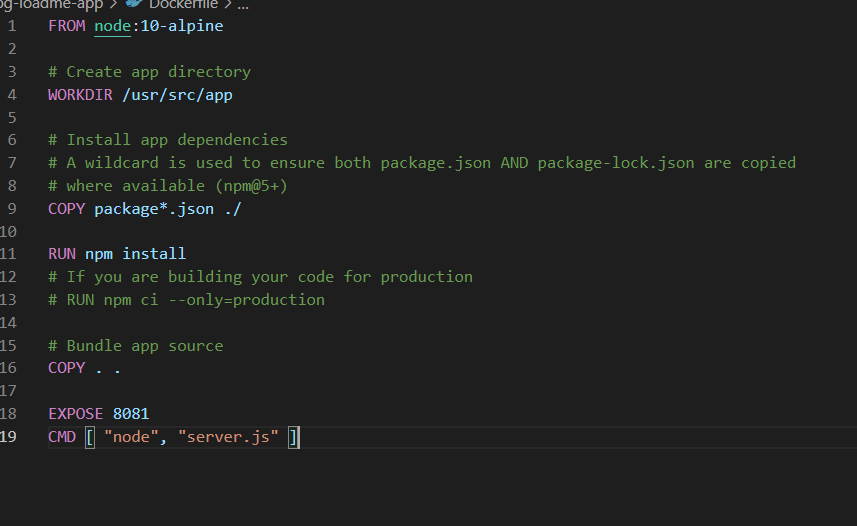


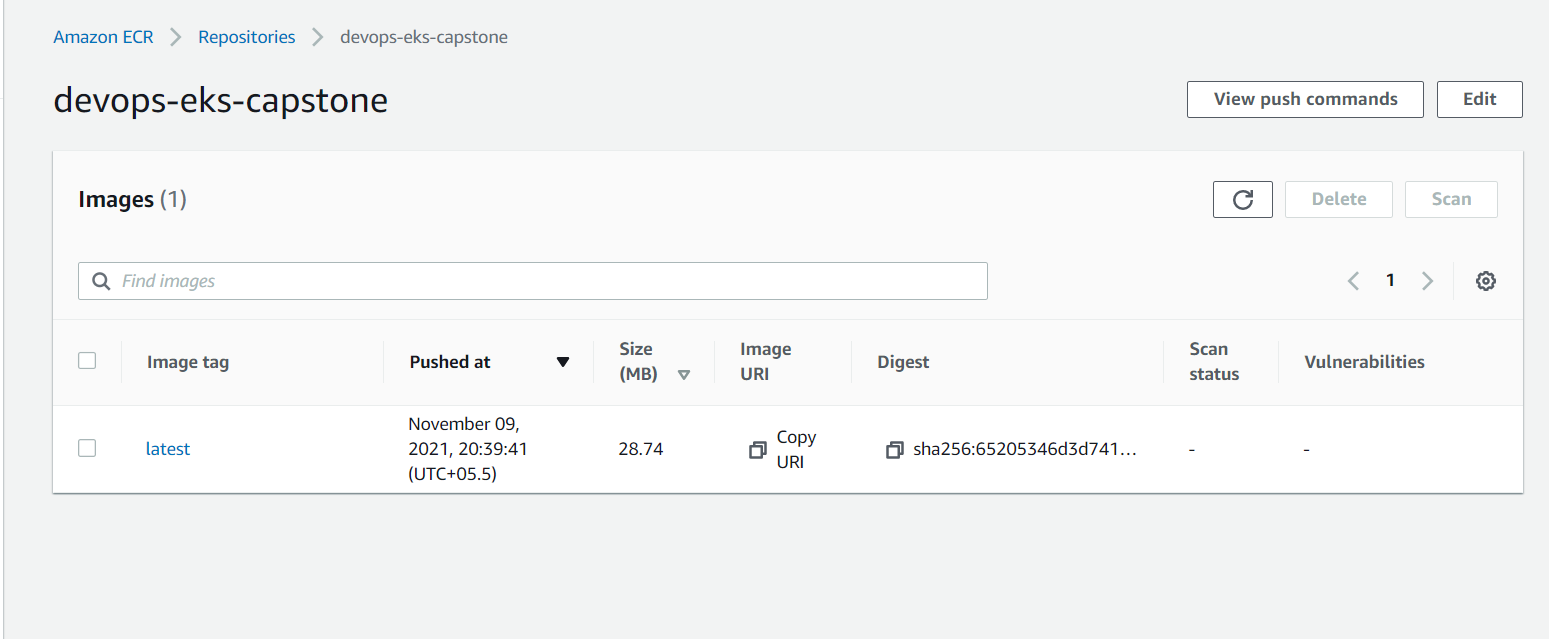
Deploying Ingress controller



**Task 2: Deployment of sample application**

Creating Docker file and deploying into ECR

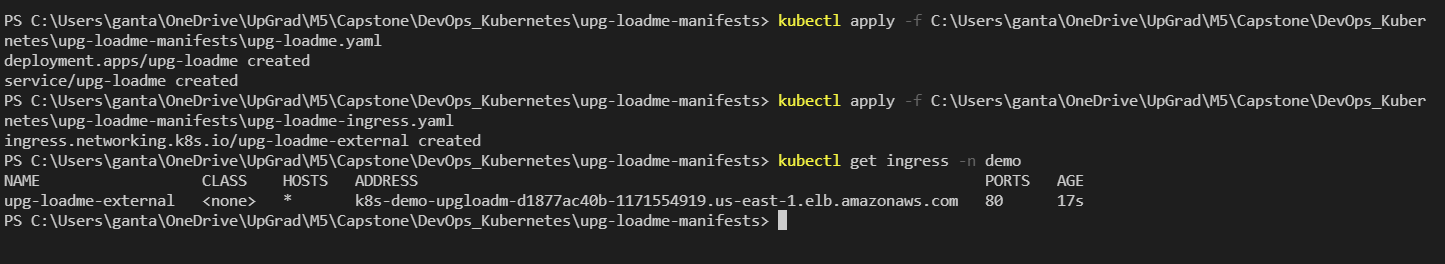




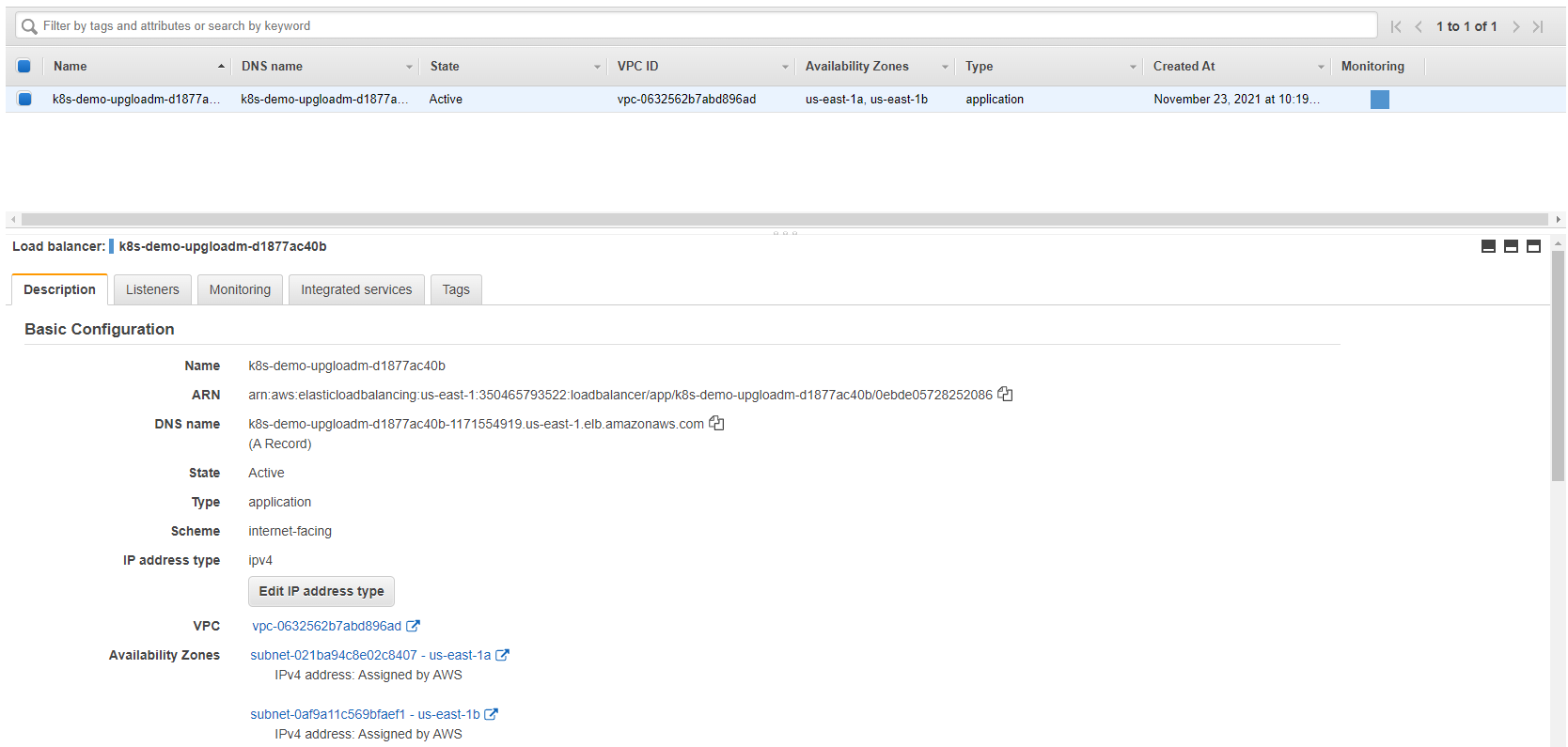
Deploying upg-loadme.yaml and upg-loadme-ingress.yaml files

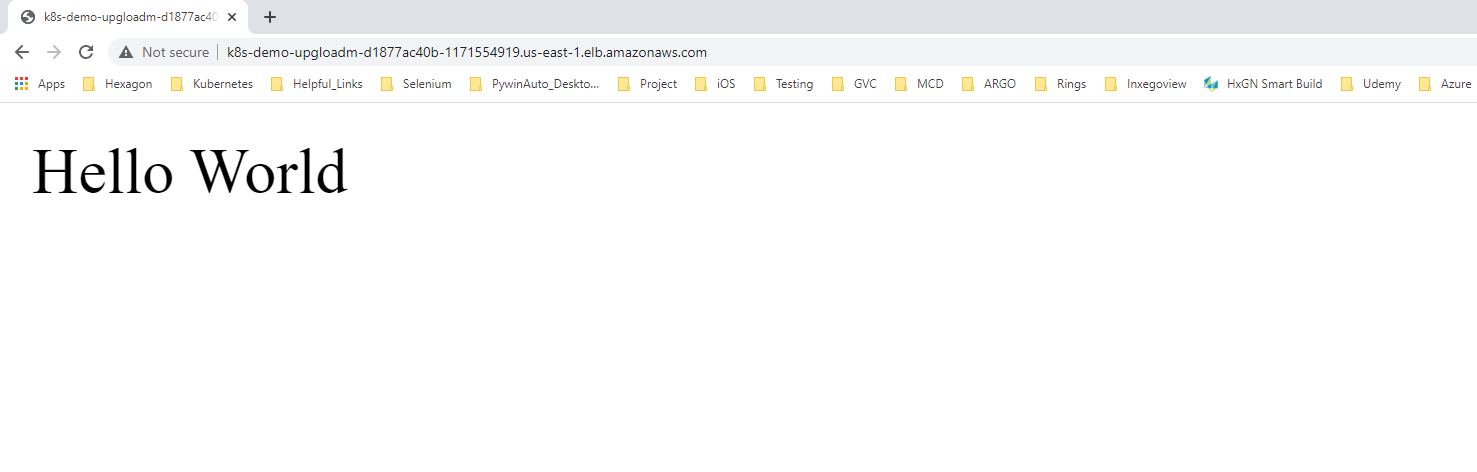
eksctl apply –f "upg-loadme.yaml”

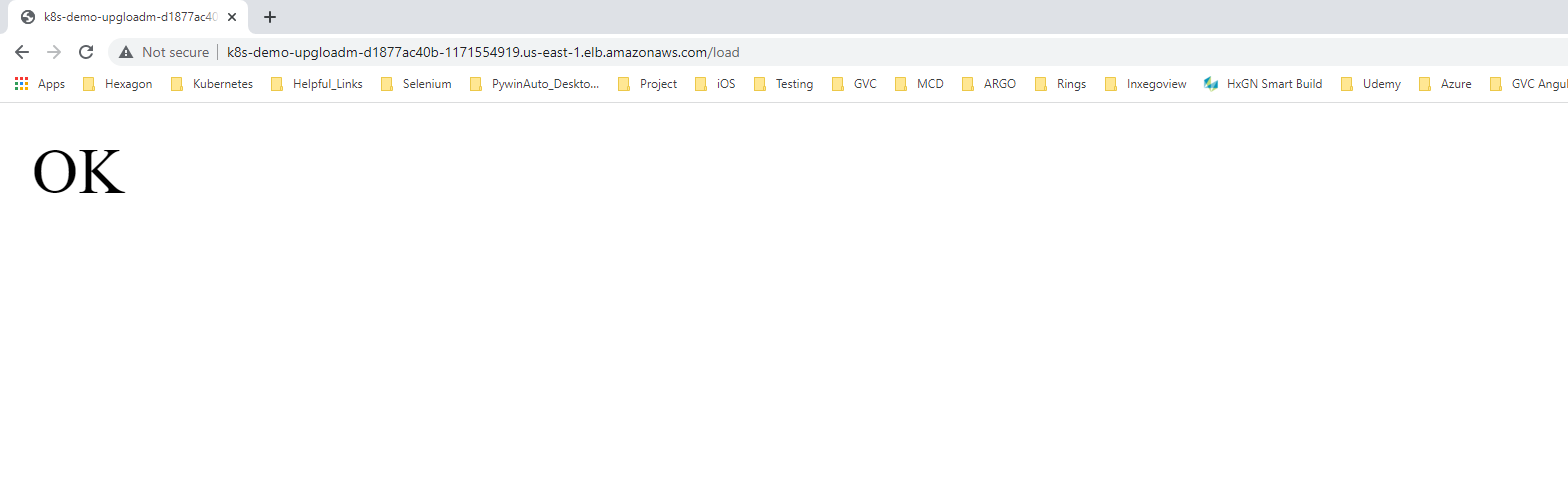
eksctl apply –f “upg-loadme-ingress.yaml”



Load-balancer end point:

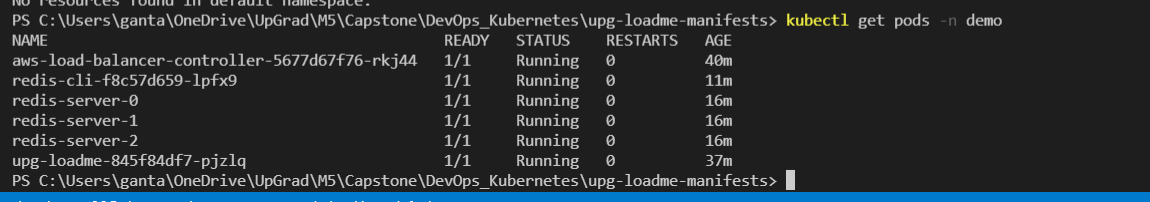




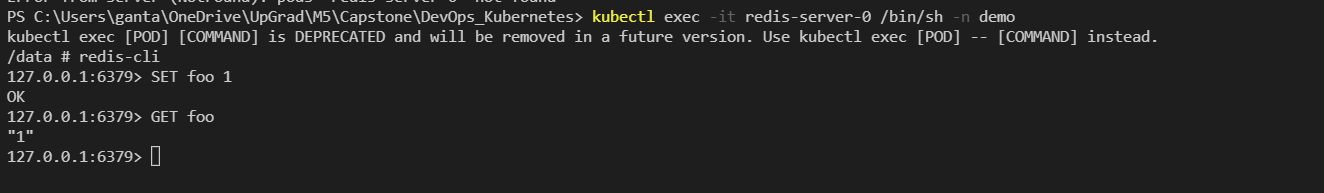


**Task 3: Deploy Redis server on Kubernetes**

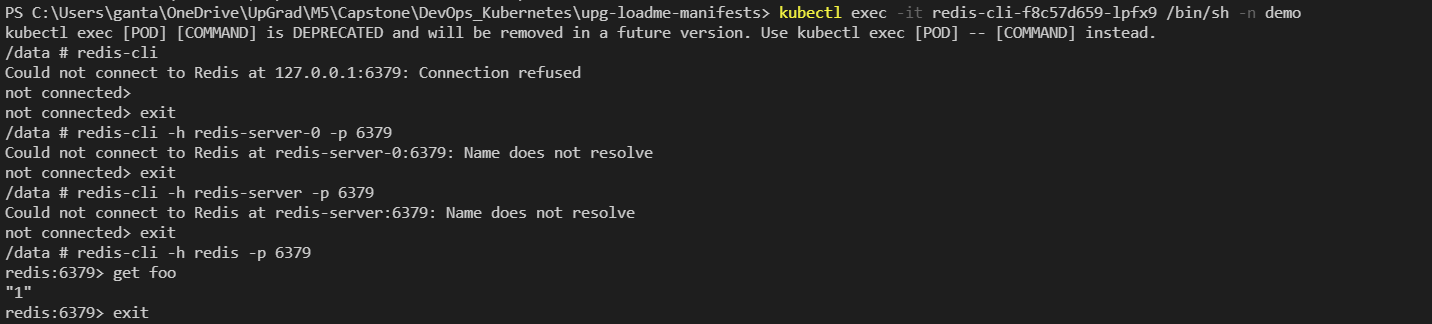
Redis-server:



Redis-server:

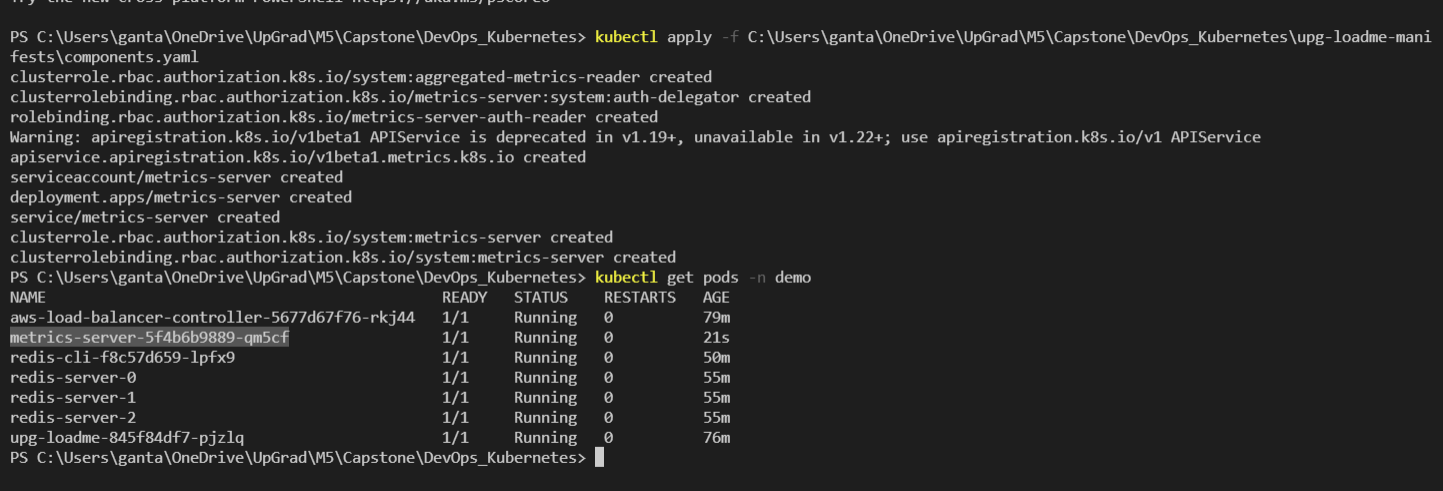


Redis-cli:



**Task 4: Test auto scaling of the application.**

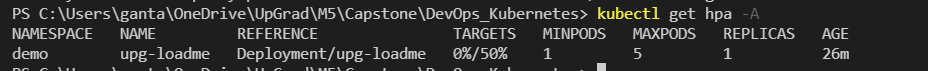
Metrics server deploying:



helm repo add metrics-server <https://kubernetes-sigs.github.io/metrics-server/>

helm upgrade --install metrics-server metrics-server/metrics-server --namespace demo

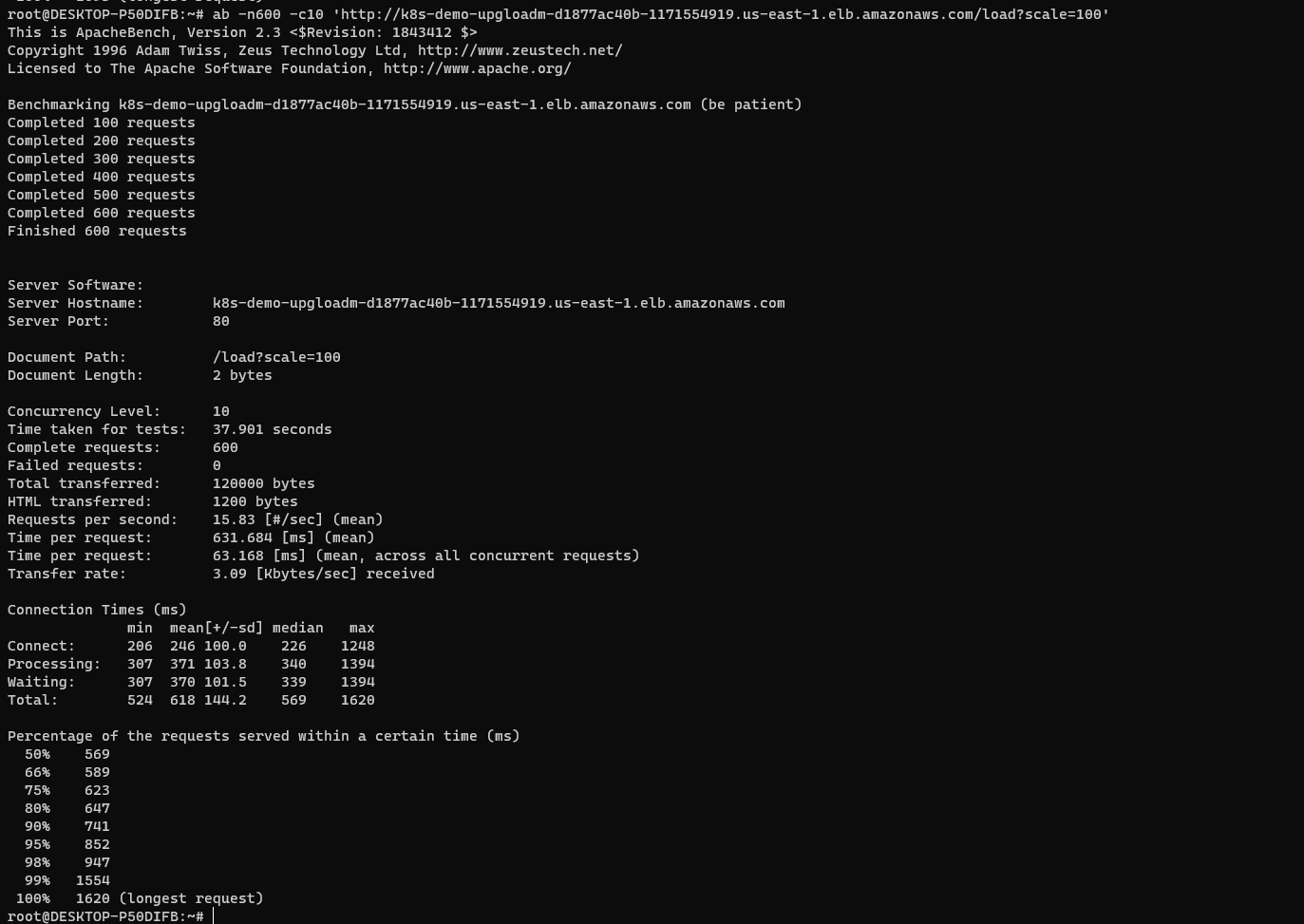
Metrics server



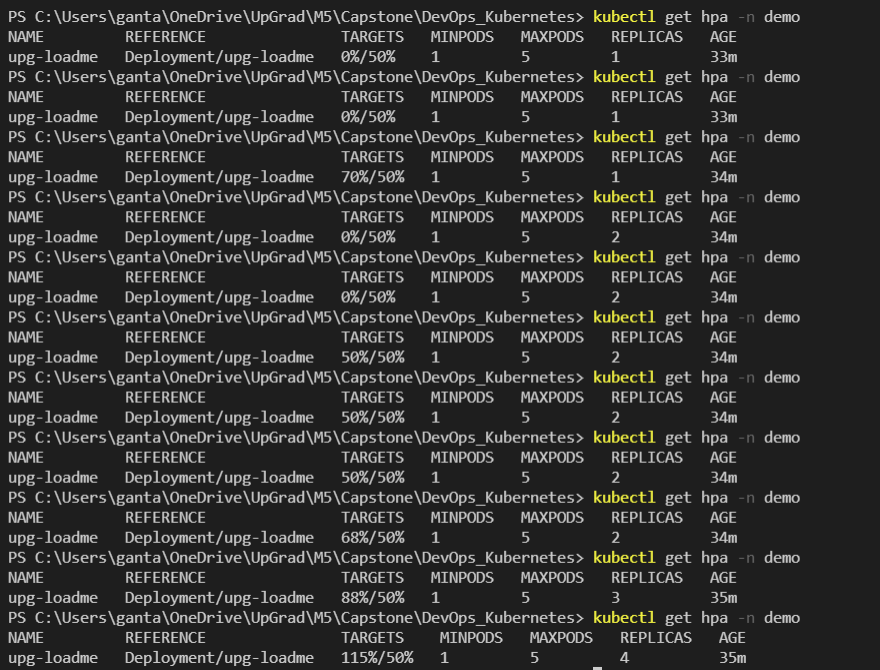
Followed the below link and installed apache bench load testing

<https://www.digitalocean.com/community/tutorials/how-to-use-apachebench-to-do-load-testing-on-an-ubuntu-13-10-vps>

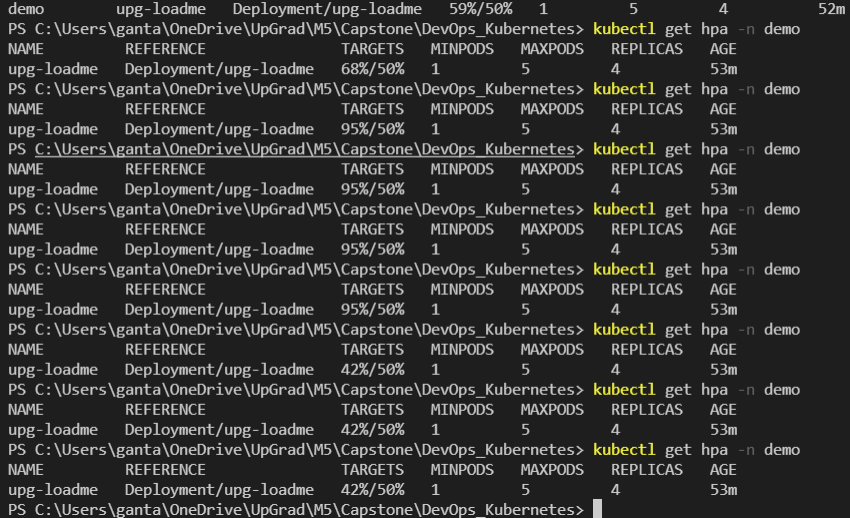
Load increase with ab  
ab -n100 -c10 'http://k8s-demo-upgloadm-d1877ac40b-1171554919.us-east-1.elb.amazonaws.com/load?scale=100'



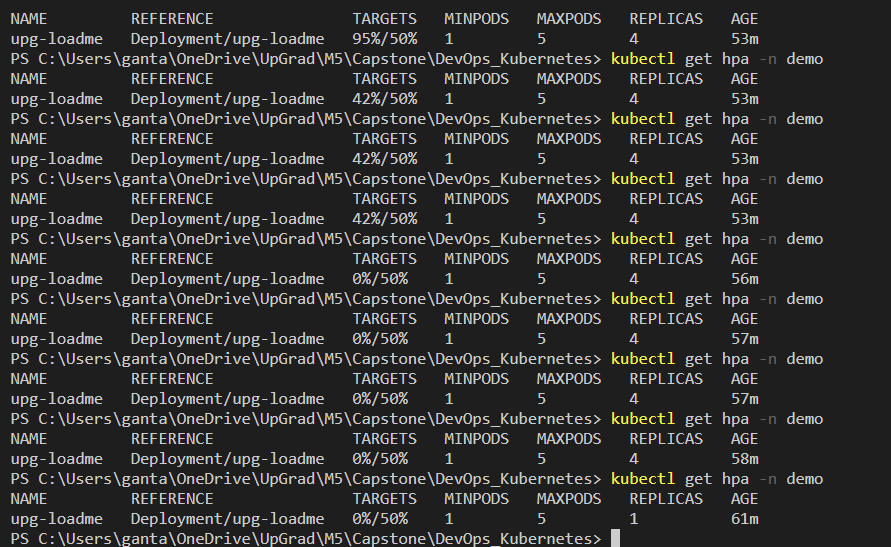
After increasing load, hpa scaling



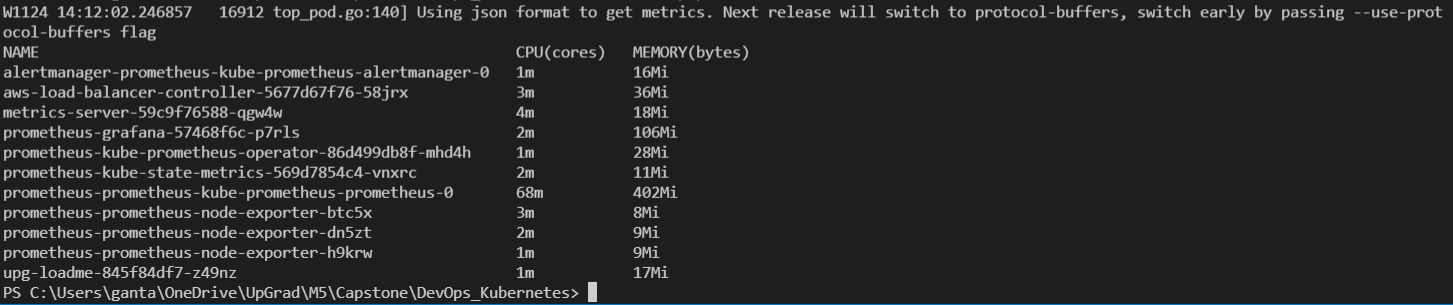
ab -n300 -c10 'http://k8s-demo-upgloadm-d1877ac40b-1171554919.us-east-1.elb.amazonaws.com/load?scale=300'



Load decreases according to the requests

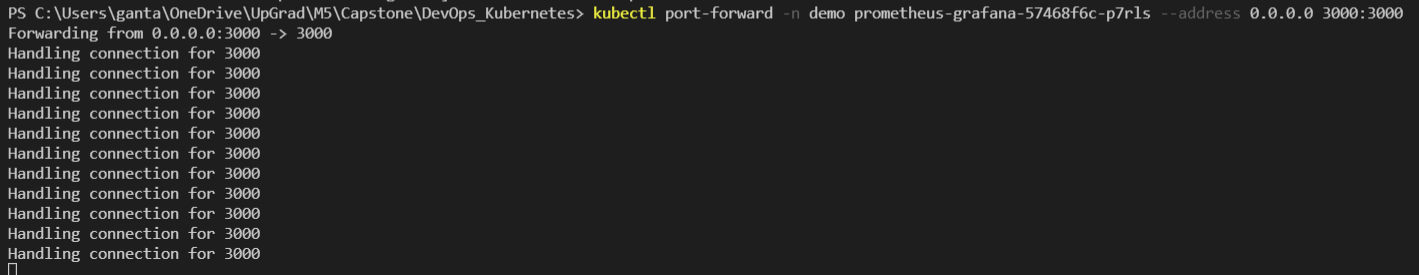


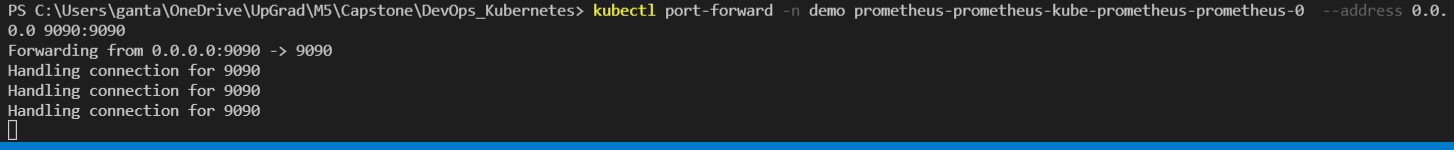
Top Pods in AWS -EKS



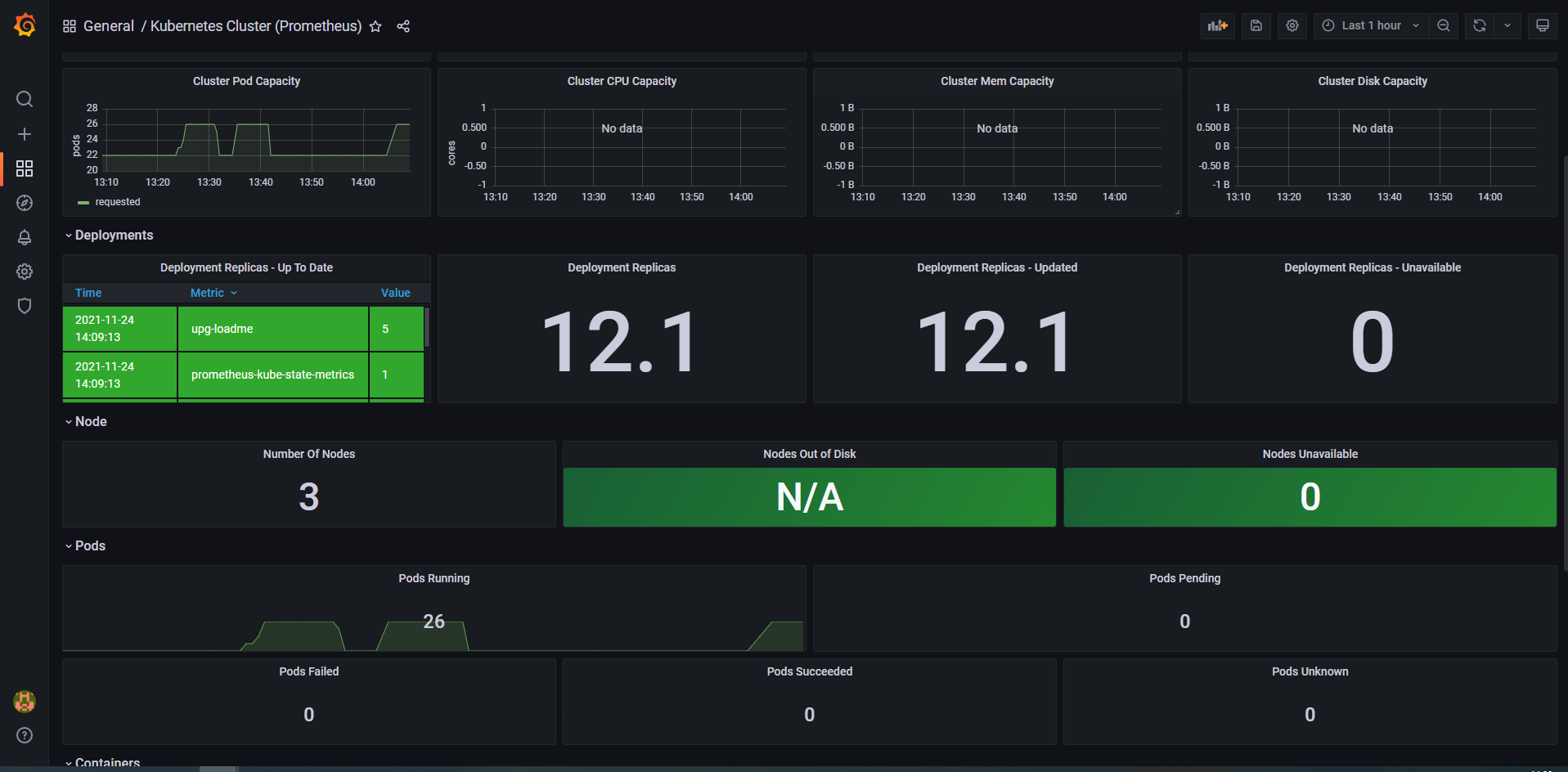
Port forwarding:

Forwarding 3000 and 9090 ports to access dashboards





Grafana-Prometheus-Dashboard



[Prometheus](http://localhost:9090/graph) Dashboard

