**\*\*Lab 4: EKS Networking with VPC and Load Balancers\*\***

**\*\*Objective\*\*: Explore network configurations within EKS.**

**\*\*Tasks\*\*:**

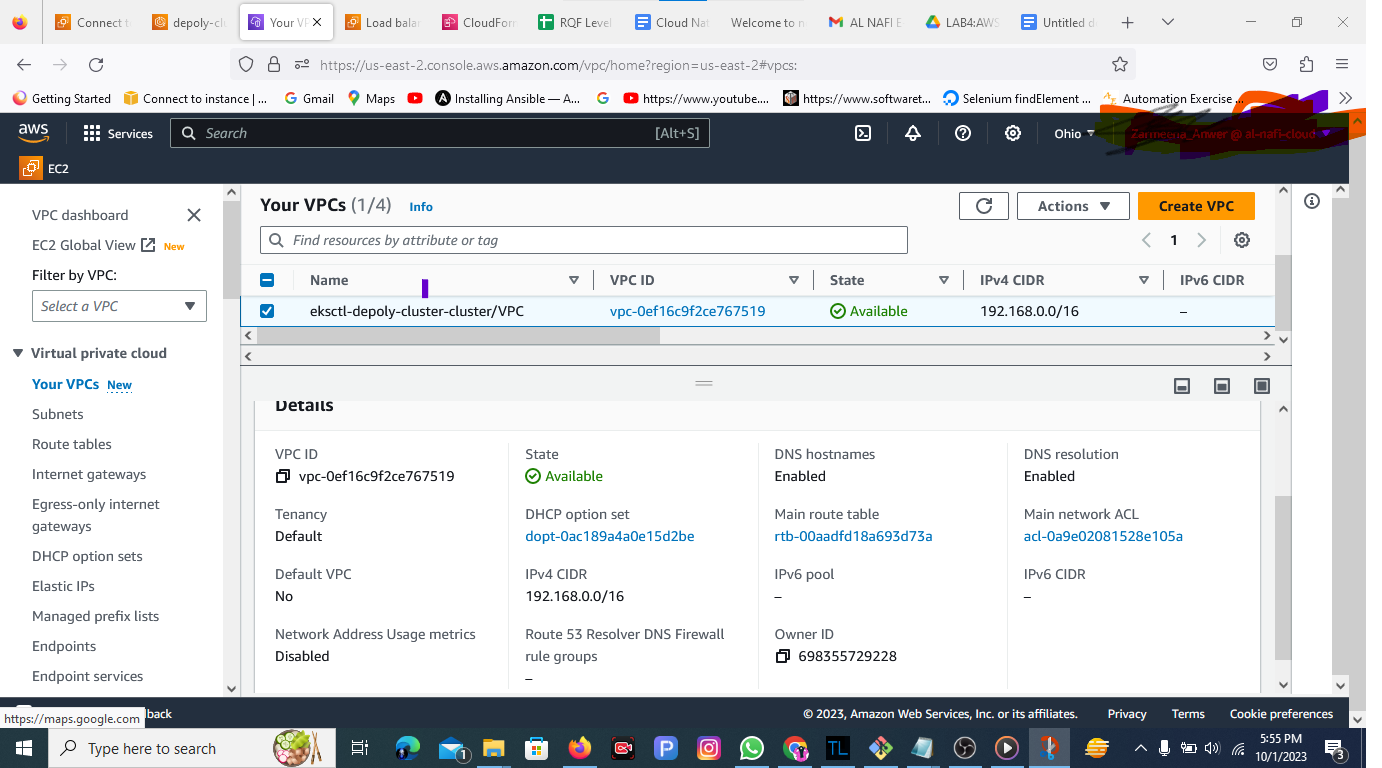
**1. Set up a VPC for EKS.**

**2. Deploy a service with an AWS Load Balancer.**

**3. Test connectivity and load balancing.**

**Task 1: Set up a VPC for EKS**

| **eksctl create cluster --name depoly-cluster --version 1.27 --region us-east-2 --nodegroup-name App-nodes --node-type t3.micro --nodes 2 --nodes-min 1 --nodes-max 4 --managed** |
| --- |



**Deploy your application to the cluster using Kubernetes manifests:**

| # Example deployment.yaml for a sample application apiVersion: apps/v1 kind: Deployment metadata:  name: sample-app spec:  replicas: 3  selector:  matchLabels:  app: sample-app  template:  metadata:  labels:  app: sample-app  spec:  containers:  - name: sample-app  image: <your-container-image>  ports:  - containerPort: 80 --- apiVersion: v1 kind: Service metadata:  name: sample-app-service spec:  selector:  app: sample-app  ports:  - protocol: TCP  port: 80  targetPort: 80  type: LoadBalancer |
| --- |

**Apply the deployment and service:**

| **kubectl apply -f deployment.yaml** |
| --- |

**Task 3: Test Connectivity and Load Balancing**

| **kubectl get svc sample-app-service** |
| --- |

Test connectivity by accessing the external IP in your web browser or via curl:

| curl http://<external-ip> |
| --- |

