**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.

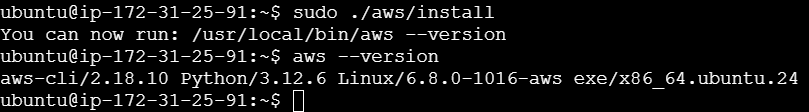
**Step 1 – Install aws cli in an ec2 instance:**

* Create ec2 instance

Commands to install aws cli :

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

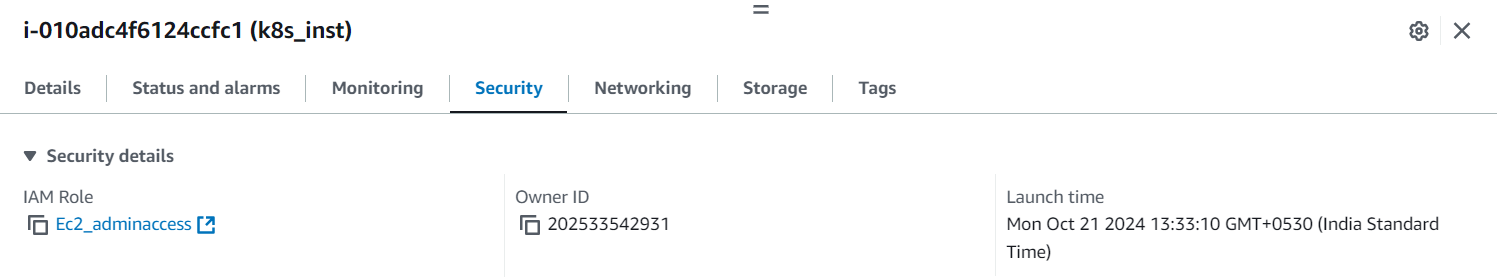
**Output:**



**Step 2 – Create IAM role & attach with ec2 instance:**

* Create IAM role with administrator access and attach to the ec2 instance.

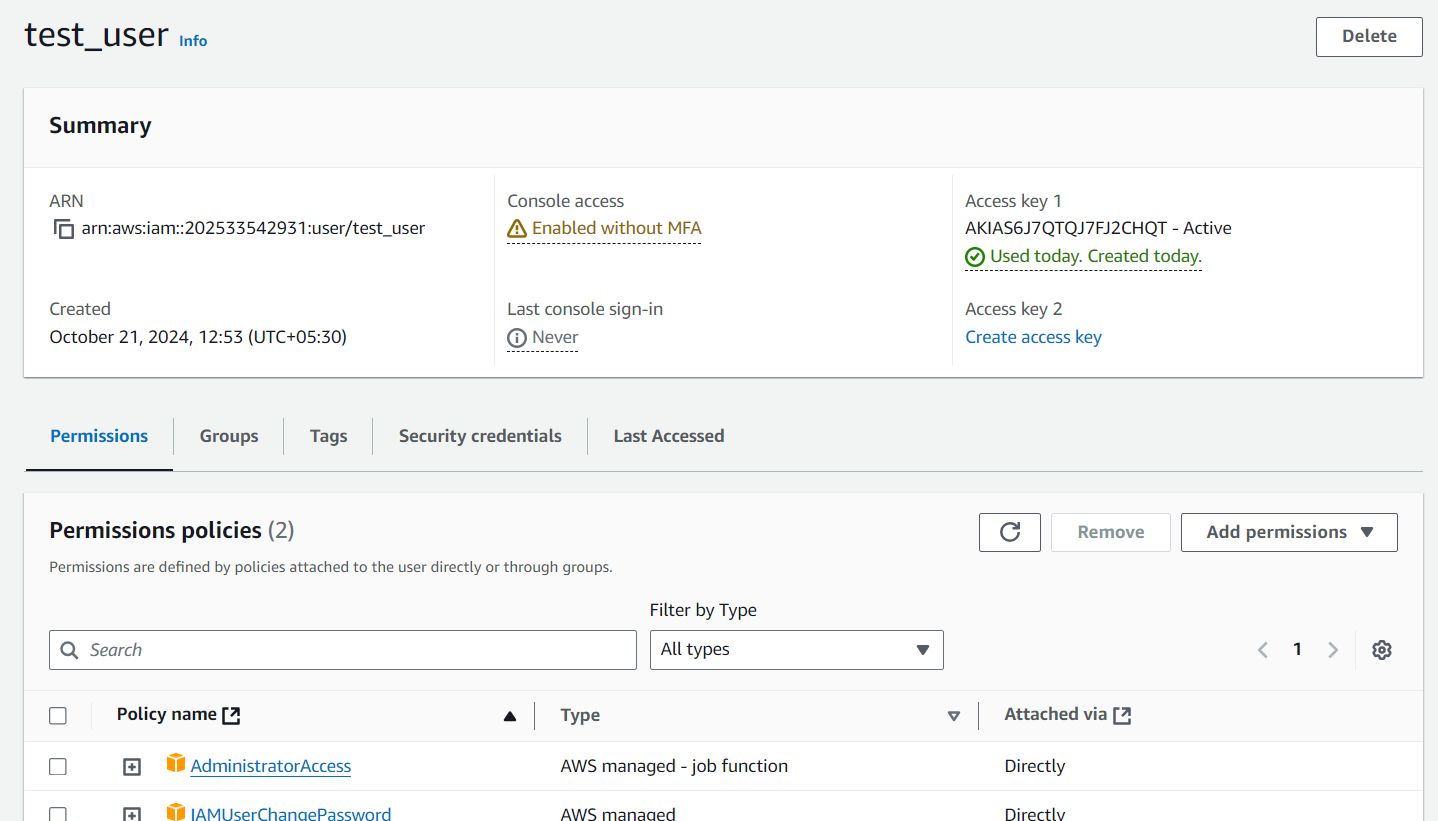
**Output:**



**Step 3 – Create a user:**

* Create a user with administrator access
* In cli – aws configure
* Provide AWS Access Key ID & Secret Access Key & Region name

**Output:**

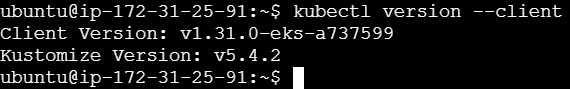


**Step 4 – Install kubectl in instance:**

Commands to install kubectl

* curl -O <https://s3.us-west-2.amazonaws.com/amazon-eks/1.31.0/2024-09-12/bin/linux/amd64/kubectl>
* chmod +x ./kubectl
* sudo mv kubectl /usr/local/bin
* kubectl version –client

**Output:**

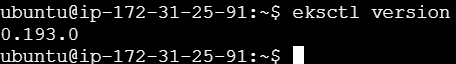


**Step 5 – Install eksctl in instance:**

Commands to install eksctl

* 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

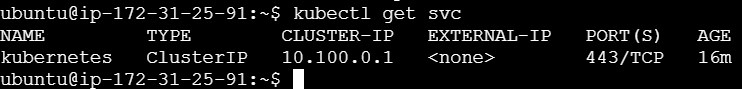
**Output:**

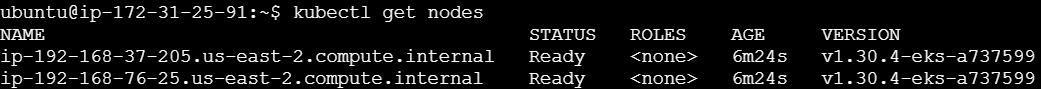


**Step 6 – Create EKS Cluster:**

* eksctl create cluster guvi-cluster --region us-east-2 --node-type t2.micro
* To confirm the cluster has been created - **kubectl get svc**
* To view the nodes - **kubectl get nodes**

**Output:**



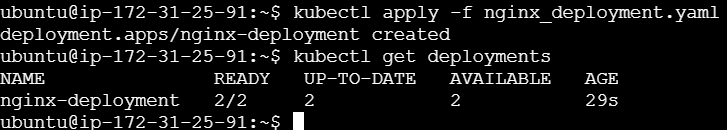


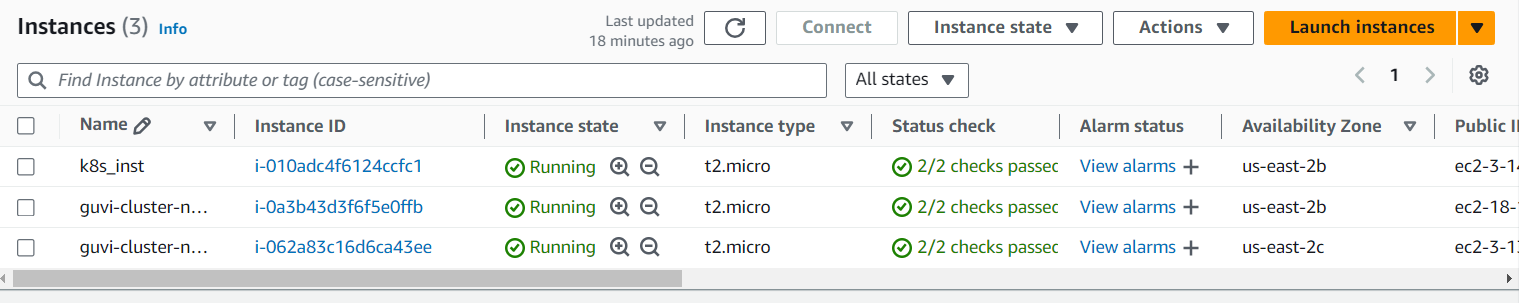
**Step 7 – Create nginx\_deployment yaml file:**

* Create a yaml file and write a code to deploy Nginx application
* Apply the deployment - **kubectl apply -f nginx\_deployment.yaml**
* Check the status of the deployment - **kubectl get deployments**

**Output:**



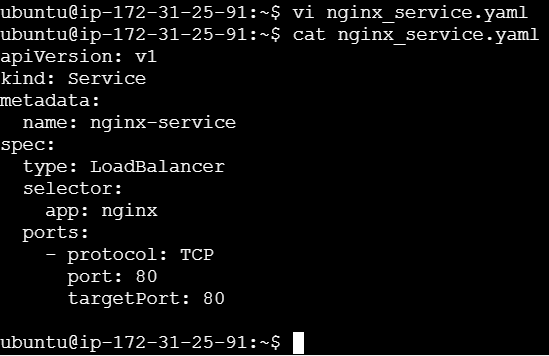




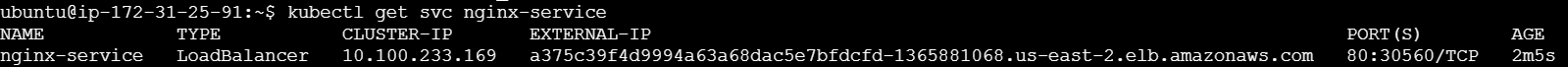
**Step 8 - Expose Nginx Outside the Cluster:**

* Create a Service to expose the deployment
* Apply the service - **kubectl apply -f nginx\_service.yaml**
* Get the External IP of the LoadBalancer - **kubectl get svc nginx\_service**

**Output:**







**Result:**

Access Nginx from webpage through public ip or DNS name.

