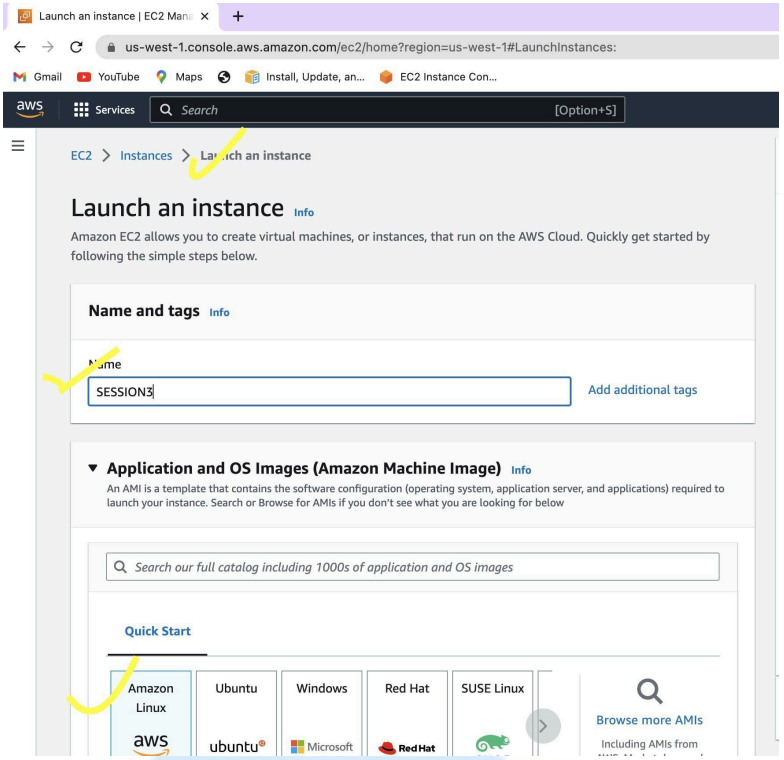
AWS EKS PROJECT SETUP CLOUD

NOTE – COST WILL BE INCURRED FOR AWS EKS SETUP SO MAKE SURE YOU ARE DOING THE PROJECT ON YOUR ON INTEREST AND ALSO YOU CAN DO THE SAME PROJECT WITH MINIKUBE SETUP

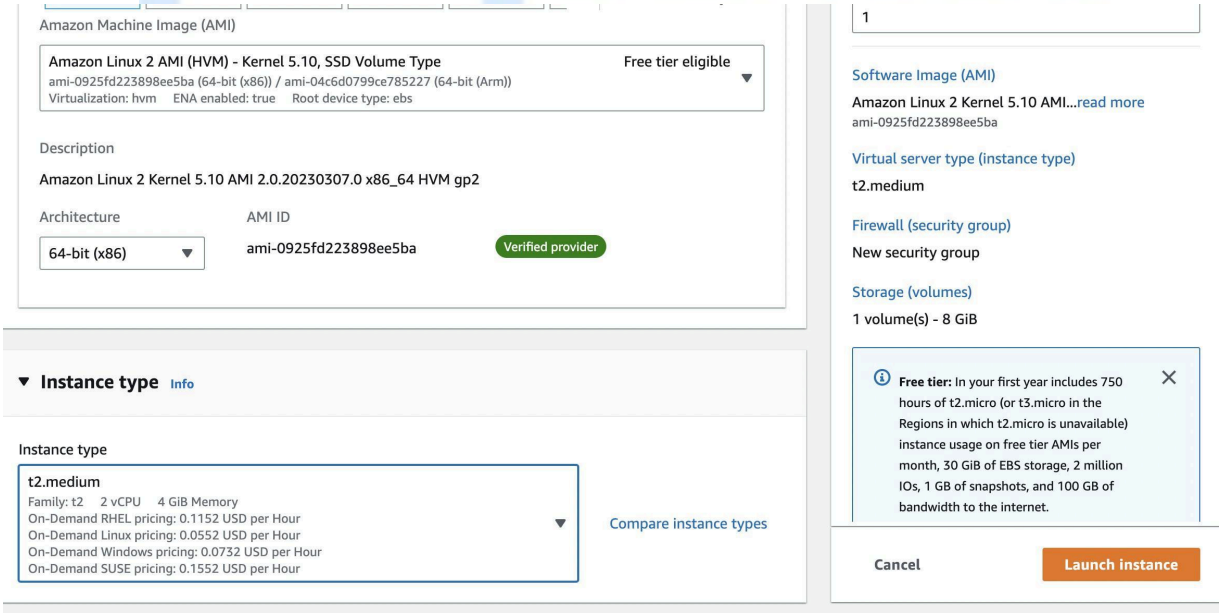
[REFER MINIKUBE SETUP Session 6 (Live 21st July 9am) Kubernetes HANDS-ON PART1 [BASICS] project

Step 1 – LOGIN to the AWS Console https://aws.amazon.com/console/

Step 2 – Select EC2 and create T2.MEDIUM INSTANCE IN US WEST1



STEP3 – Select AMAZON LINUX 2 AMI 30 GB



STEP3.1 – Make sure to create the PEM file in the EC2 instance with name key-test

Step 4 – Install all tools Pre-requisites

- Install Git

yum install git -y

Step 5 – INSTALL SETUP FOR EKS

- Install kubectl

```
curl -o kubectl
https://amazon-eks.s3-us-west-2.amazonaws.com/1.14.6/2019-08-22/bin/linux/amd64/kubectl
chmod +x ./kubectl
mkdir -p $HOME/bin
cp ./kubectl $HOME/bin/kubectl
export PATH=$HOME/bin:$PATH
echo 'export PATH=$HOME/bin:$PATH' >> ~/.bashrc
source $HOME/.bashrc
kubectl version --short –client
```

OR

```
curl -O
https://s3.us-west-2.amazonaws.com/amazon-eks/1.24.11/2023-03-17/bin/linux
/ amd64/kubectl.sha256
chmod +x ./kubectl
mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export
PATH=$PATH:$HOME/bin
echo 'export PATH=$PATH:$HOME/bin' >> ~/.bashrc
kubectl version --short –client
```

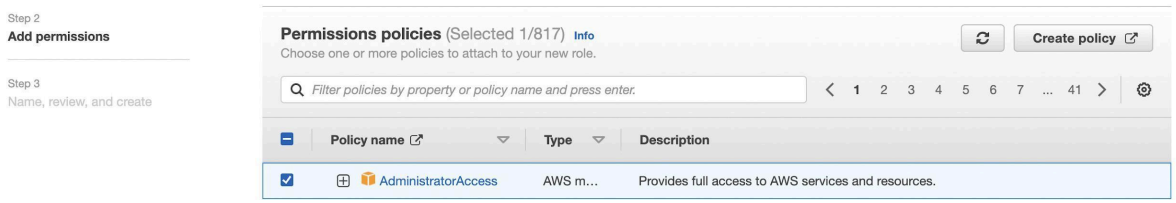
- Install eksctl

```
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/bin
eksctl version
```

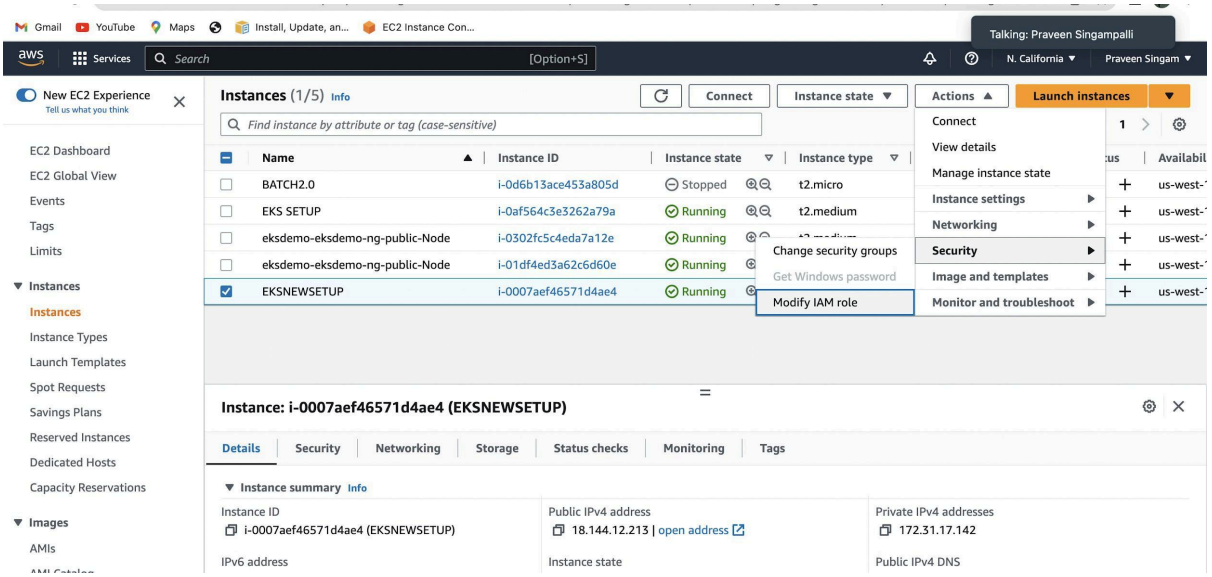
Step6 - ATTACH THE IAM ROLE

Go to IAM -> CLICK CREATE NEW IAM ROLE -> SELECT EC2 -> CLICK ON ADMINISTRATOR ACCESS -> CREATE ROLE



STEP 7 –

Go toEC2 instance you have created -> Click on ACTIONS -> SECURITY -> MODIFY IAM ROLE -> ATTACH YOUR NEW ROLE



Step 8 -

- **MASTER Cluster creation [Change the master cluster name** eksdemo **as per your wish and select region as us-west-1]**

```
eksctl create cluster --name=eksdemo \
        --region=us-west-1 \
        --zones=us-west-1b,us-west-1c \
        --without-nodegroup
```

- **Add Iam-Oidc-Providers**

```
eksctl utils associate-iam-oidc-provider \
  --region us-west-1 \
  --cluster eksdemo \
  --approve
```

- **WORKER NODE Create node-group [** Change the PEM key ssh-public-key to your key created in step 3.1]

```
eksctl create nodegroup --cluster=eksdemo \
        --region=us-west-1 \
        --name=eksdemo-ng-public \
        --node-type=t2.micro \
        --nodes=2 \
        --nodes-min=1 \
        --nodes-max=2 \
        --ssh-access \
        --ssh-public-key=key-test \
        --node-volume-size=10 \
        --managed \
        --asg-access \
        --external-dns-access \
        --full-ecr-access \
        --appmesh-access \
        --alb-ingress-access
```

STEP 9 –

** Clone the repository

https://github.com/praveen1994dec/Custom_Resource_Definition.git

** cd Custom_Resource_Definition/ and hit the below command

kubectl apply -f crd.yml

** Once the CRD is registered, verify that by running the

kubectl api-resources | grep myplatform

** Creating the custom resource

kubectl apply -f cr.yml

** Hit the below command

kubectl get myp

STEP 10 – DELETE NODE AND THEN THE CLUSTER

```
eksctl delete nodegroup --cluster=eksdemo
--region=us-west-1 --name=eksdemo-ng-public
```

```
eksctl delete cluster --name=eksdemo
--region=us-west-1
```