10747883-AbhinavGupta-Surprisetest

Ans1.

First we will install docker

```
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.

C:\Users\10747883>cd Downloads

C:\Users\10747883\Downloads>ssh -i "mumbai.pem" ec2-user@ec2-13-235-99-169.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-13-235-99-169.ap-south-1.compute.amazonaws.com (13.235.99.169)' can't be established.
ED25519 key fingerprint is SHA256:qGU45KA36BaUAqozKo0yZ++5fMYvscytNVZlAsl/CUo.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-235-99-169.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-33-48 ~]$ sudo su
[root@ip-172-31-33-48 ec2-user]# cd
[root@ip-172-31-33-48 ~]# yum install docker* -y
```

Now start docker and enable it.

```
[root@ip-172-31-33-48 ~]# systemctl start docker
[root@ip-172-31-33-48 ~]# systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[root@ip-172-31-33-48 ~]# docker info
```

Now pull httpd

```
[root@ip-172-31-33-48 ~]# docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
a2318d6c47ec: Pull complete
62dd86107c65: Pull complete
4f4fb700ef54: Pull complete
22871f73faed: Pull complete
ca061a523d1f: Pull complete
509789394c2a: Pull complete
Digest: sha256:ae1124b8d23ee3fc35d49da35d5c748a2fce318d1f55ce59ccab889d612f8be8
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
[root@ip-172-31-33-48 ~]#
```

Now pull ubuntu

```
[root@ip-172-31-33-48 ~]# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
dafa2b0c44d2: Pull complete
Digest: sha256:dfc10878be8d8fc9c61cbff33166cb1d1fe44391539243703c72766894fa834a
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
[root@ip-172-31-33-48 ~]#
```

Check for containers available and as it is empty now created a webapp container and then started it and then attached it and inside webapp we will first update it and then

```
[root@ip-172-31-33-48 ~]# docker run -it --name webapp-02 -p 8080:80 ubuntu:latest /bin/bash
```

```
[root@ip-172-31-33-48 ~]# docker start  webapp-02
webapp-02
[root@ip-172-31-33-48 ~]# docker attach webapp-02
root@b7698475739a:/# apt update -y
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:3 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [442 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [360 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [331 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [469 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [19.3 MB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [13.7 kB]
Get:11 http://archive.ubuntu.com/ubuntu noble/main amd64 Packages [1808 kB]
Get:12 http://archive.ubuntu.com/ubuntu noble/restricted amd64 Packages [117 kB]
Get:13 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [446 kB]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [491 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [17.8 kB]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [669 kB]
Get:17 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [11.9 kB]
```

Then installed apache2

```
root@b7698475739a:/# apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adduser apache2-bin apache2-data apache2-utils ca-certificates krb5-locales li
  libaprutil1-ldap libaprutil1t64 libbrotli1 libcurl4t64 libexpat1 libgdbm-compa
  libicu74 libjansson4 libk5crypto3 libkeyutils1 libkrb5-3 libkrb5support0 libld
  libnghttp2-14 libperl5.38t64 libpsl5t64 librtmp1 libsasl2-2 libsasl2-modules l
  libssh-4 libssl3t64 libxml2 media-types netbase openssl perl perl-modules-5.38
Suggested packages:
  liblocale-gettext-perl cron quota ecryptfs-utils apache2-doc apache2-suexec-pr
```

Start apache2 and check its status and created a file

```
root@b7698475739a:/# service apache2 start
 * Starting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain n
erName' directive globally to suppress this message
 *
root@b7698475739a:/# service apache2 enable
Usage: apache2 {start|stop|graceful-stop|restart|reload|force-reload}
root@b7698475739a:/# service apache2 status
 * apache2 is running
root@b7698475739a:/# cs /var/www/html
bash: cs: command not found
root@b7698475739a:/# cd /var/www/html
root@b7698475739a:/var/www/html# echo "This is my surprise server"
```

```
root@b7698475739a:/# cs /var/www/html
bash: cs: command not found
root@b7698475739a:/# cd /var/www/html
root@b7698475739a:/var/www/html# echo "This is my surprise server">index.html
root@b7698475739a:/var/www/html# cat index.html
This is my surprise server
root@b7698475739a:/var/www/html#
```

Here we can see our server at port 80

```
[root@ip-172-31-33-48 ~]# curl 172.17.0.3:80
This is my surprise server
[root@ip-172-31-33-48 ~]#
```

Ans2.

```
root@ip-172-31-36-241:~# apt install unzip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 175 kB of archives.
After this operation, 384 kB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 unzip amd64 6.0-28ubuntu4 [175 kB]
Fetched 175 kB in 0s (9430 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 67741 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4) ...
Setting up unzip (6.0-28ubuntu4) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-36-241:~#
```

```
You can now run: /usr/local/bin/aws --version
root@ip-172-31-36-241:~# aws configure
AWS Access Key ID [None]: AKIATDMK5PO26RGZ2AV3
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
```

```
root@ip-172-31-36-241:~# curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_a
md64.tar.gz" | tar xz -C /tmp
root@ip-172-31-36-241:~# mv /tmp/eksctl /usr/local/bin
root@ip-172-31-36-241:~# eksctl version
0.190.0
root@ip-172-31-36-241:~# curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl https://storage.googleapis.com/kub
ernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100     7  100     7    0     0     21      0 --:--:-- --:--:-- --:--:--    21
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 53.7M  100 53.7M    0     0  9881k      0  0:00:05  0:00:05 --:--:-- 12.8M
root@ip-172-31-36-241:~# install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
root@ip-172-31-36-241:~# kubectl version --client
Client Version: v1.31.0
Kustomize Version: v5.4.2
root@ip-172-31-36-241:~# eksctl create cluster --name surprise-cluster --region ap-south-1 --version 1.29 --vpc-public-subnets subnet
-0ee841e1e73824361,subnet-0a428ceaf61f95f04 --without-nodegroup
2024-09-18 10:19:38 [ℹ]  eksctl version 0.190.0
2024-09-18 10:19:38 [ℹ]  using region ap-south-1
2024-09-18 10:19:39 [✔]  using existing VPC (vpc-04880a66c6cf9d586) and subnets (private:map[] public:map[ap-south-1a:{subnet-0ee841e
1e73824361 ap-south-1a 172.31.32.0/20 0 } ap-south-1c:{subnet-0a428ceaf61f95f04 ap-south-1c 172.31.16.0/20 0 }])
2024-09-18 10:19:39 [!]  custom VPC/subnets will be used; if resulting cluster doesn't function as expected, make sure to review the
configuration of VPC/subnets
2024-09-18 10:19:39 [ℹ]  using Kubernetes version 1.29
2024-09-18 10:19:39 [ℹ]  creating EKS cluster "surprise-cluster" in "ap-south-1" region with
```

```
uired instance types
root@ip-172-31-36-241:~#    eksctl create nodegroup \
  --cluster surprise-cluster \
  --region ap-south-1 \
  --name my-node-group \
  --node-ami-family Ubuntu2004 \
  --node-type t2.small \
  --subnet-ids subnet-0ee841e1e73824361,subnet-0fb603f894ee740b1  \
  --nodes 3 \
  --nodes-min 2 \
  --nodes-max 4 \
  --ssh-access \
  --ssh-public-key /root/.ssh/id_ed25519.pub
2024-09-18 10:35:22 [ℹ]  will use version 1.29 for new nodegroup(s) based on control plane version
2024-09-18 10:35:23 [ℹ]  nodegroup "my-node-group" will use "ami-011ab3d6667efecdb" [Ubuntu2004/1.29]
2024-09-18 10:35:23 [ℹ]  using SSH public key "/root/.ssh/id_ed25519.pub" as "eksctl-surprise-cluster-nodegroup-my-node-group-h
p3rJMtwgssNdP4mAaFruuPFsANcqlFcFOShs"
2024-09-18 10:35:23 [ℹ]  skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.smal
2024-09-18 10:35:23 [ℹ]  1 nodegroup (my-node-group) was included (based on the include/exclude rules)
2024-09-18 10:35:23 [ℹ]  will create a CloudFormation stack for each of 1 managed nodegroups in cluster "surprise-cluster"
2024-09-18 10:35:23 [ℹ]
2 sequential tasks: { fix cluster compatibility, 1 task: { 1 task: { create managed nodegroup "my-node-group" } }
}
2024-09-18 10:35:23 [ℹ]  checking cluster stack for missing resources
2024-09-18 10:35:24 [ℹ]  cluster stack has all required resources
2024-09-18 10:35:24 [ℹ]  building managed nodegroup stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:24 [ℹ]  skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.smal
2024-09-18 10:35:24 [ℹ]  deploying stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:24 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:54 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:36:50 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:37:39 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
```

```
root@ip-172-31-36-241:/home/ubuntu# cd
root@ip-172-31-36-241:~# vim surprise-deplo.yml
<ml" [New] 21L, 341B written
```

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: surp-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
        ports:
        - containerPort: 80
```

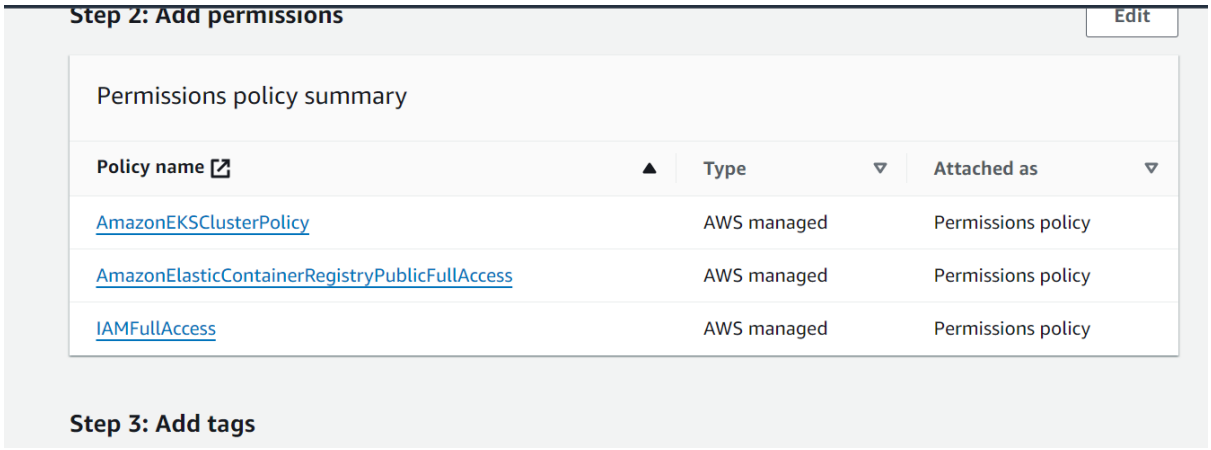
```
root@ip-172-31-36-241:~# kubectl apply -f surprise-deplo.
yml
deployment.apps/glenn-deployment created
root@ip-172-31-36-241:~# kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
frontend-4xrt4                      1/1     Running   0          24m
frontend-ctzbt                      1/1     Running   0          24m
frontend-gs7dl                      1/1     Running   0          26m
frontend-hg8rv                      1/1     Running   0          24m
frontend-mfxpw                      1/1     Running   0          26m
frontend-nl265                      1/1     Running   0          24m
frontend-p2rws                      1/1     Running   0          26m
frontend-pmj4r                      1/1     Running   0          24m
frontend-t5jjl                      1/1     Running   0          26m
glenn-deployment-86dcfdf4c6-nwxtd   1/1     Running   0          9s
glenn-deployment-86dcfdf4c6-pd427   1/1     Running   0          9s
glenn-deployment-86dcfdf4c6-r5mqj   1/1     Running   0          9s
root@ip-172-31-36-241:~# kubectl get rs
NAME                          DESIRED   CURRENT   READY   AGE
```

```
root@ip-172-31-36-241:~# kubectl get deployment
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
glenn-deployment   3/3     3            3           26s
root@ip-172-31-36-241:~# kubectl describe deployment glenn-deployment
Name:                   glenn-deployment
Namespace:              default
CreationTimestamp:      Wed, 18 Sep 2024 11:07:28 +0000
Labels:                 app=nginx
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=nginx
Replicas:               3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
   nginx:
    Image:         nginx:1.14.2
    Port:          80/TCP
    Host Port:     0/TCP
    Environment:   <none>
    Mounts:        <none>
  Volumes:         <none>
  Node-Selectors:  <none>
  Tolerations:     <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
```

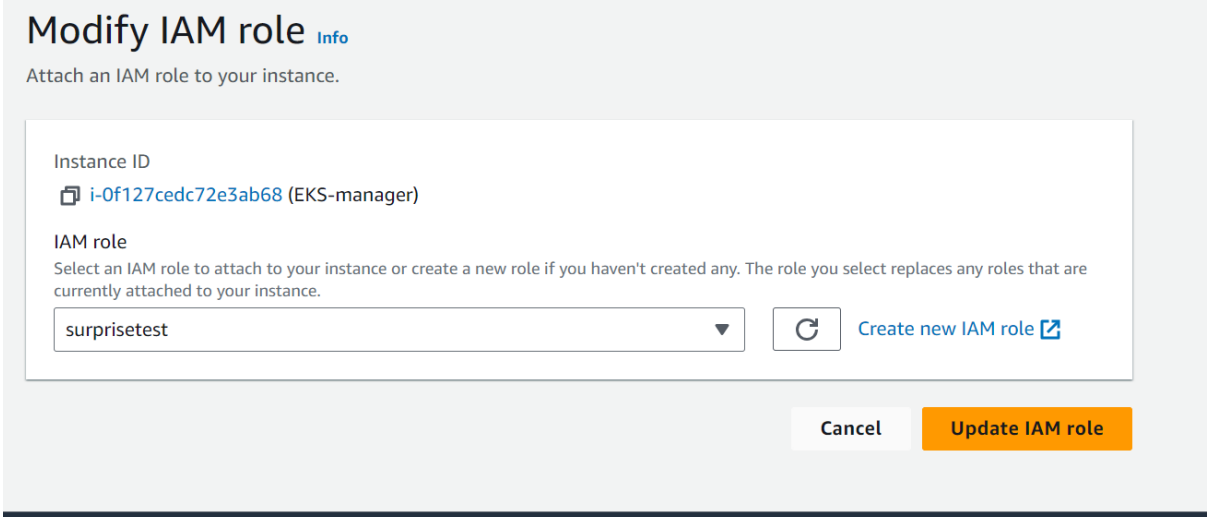
```
  ----            ------  ------
  Available       True    MinimumReplicasAvailable
  Progressing     True    NewReplicaSetAvailable
OldReplicaSets:   <none>
NewReplicaSet:    glenn-deployment-86dcfdf4c6 (3/3 replicas created)
Events:
  Type    Reason             Age   From
 Message
  ----    ------             ----  ----
 -------
  Normal  ScalingReplicaSet  34s   deployment-controller  Scaled up r
root@ip-172-31-36-241:~# kubectl autoscale deployment glenn-deploymen
horizontalpodautoscaler.autoscaling/glenn-deployment autoscaled
root@ip-172-31-36-241:~# kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
frontend-4xrt4                      1/1     Running   0          24m
frontend-ctzbt                      1/1     Running   0          24m
frontend-gs7dl                      1/1     Running   0          27m
frontend-hg8rv                      1/1     Running   0          24m
frontend-mfxpw                      1/1     Running   0          27m
frontend-nl265                      1/1     Running   0          24m
frontend-p2rws                      1/1     Running   0          27m
frontend-pmj4r                      1/1     Running   0          24m
frontend-t5jjl                      1/1     Running   0          27m
glenn-deployment-86dcfdf4c6-nwxtd   1/1     Running   0          49s
glenn-deployment-86dcfdf4c6-pd427   1/1     Running   0          49s
glenn-deployment-86dcfdf4c6-r5mqj   1/1     Running   0          49s
root@ip-172-31-36-241:~#
```

Ans3.

Create a new role with three permissions ElasticContainer full access , amazonkscluster policy and IAM full.



Attach it to the EKS server



Then install unzip

```
root@ip-172-31-36-241:~# apt install unzip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 175 kB of archives.
After this operation, 384 kB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 unzip amd64 6.0-28ubuntu4 [175 kB]
Fetched 175 kB in 0s (9430 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 67741 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4) ...
Setting up unzip (6.0-28ubuntu4) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-36-241:~#
```

Update aws configuration

```
You can now run: /usr/local/bin/aws --version
root@ip-172-31-36-241:~# aws configure
AWS Access Key ID [None]: AKIATDMK5PO26RGZ2AV3
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
```

Install EKS tools

Install kubectl

And created EKS cluster

```
root@ip-172-31-36-241:~# curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_a
md64.tar.gz" | tar xz -C /tmp
root@ip-172-31-36-241:~# mv /tmp/eksctl /usr/local/bin
root@ip-172-31-36-241:~# eksctl version
0.190.0
root@ip-172-31-36-241:~# curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl https://storage.googleapis.com/kub
ernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100     7  100     7    0     0     21      0 --:--:-- --:--:-- --:--:--    21
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 53.7M  100 53.7M    0     0  9881k      0  0:00:05  0:00:05 --:--:-- 12.8M
root@ip-172-31-36-241:~# install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
root@ip-172-31-36-241:~# kubectl version --client
Client Version: v1.31.0
Kustomize Version: v5.4.2
root@ip-172-31-36-241:~# eksctl create cluster --name surprise-cluster --region ap-south-1 --version 1.29 --vpc-public-subnets subnet
-0ee841e1e73824361,subnet-0a428ceaf61f95f04 --without-nodegroup
2024-09-18 10:19:38 [ℹ]  eksctl version 0.190.0
2024-09-18 10:19:38 [ℹ]  using region ap-south-1
2024-09-18 10:19:39 [✔]  using existing VPC (vpc-04880a66c6cf9d586) and subnets (private:map[] public:map[ap-south-1a:{subnet-0ee841e
1e73824361 ap-south-1a 172.31.32.0/20 0 } ap-south-1c:{subnet-0a428ceaf61f95f04 ap-south-1c 172.31.16.0/20 0 }])
2024-09-18 10:19:39 [!]  custom VPC/subnets will be used; if resulting cluster doesn't function as expected, make sure to review the
configuration of VPC/subnets
2024-09-18 10:19:39 [ℹ]  using Kubernetes version 1.29
2024-09-18 10:19:39 [ℹ]  creating EKS cluster "surprise-cluster" in "ap-south-1" region with
```

Create node3 group

```
uired instance types
root@ip-172-31-36-241:~#    eksctl create nodegroup \
  --cluster surprise-cluster \
  --region ap-south-1 \
  --name my-node-group \
  --node-ami-family Ubuntu2004 \
  --node-type t2.small \
  --subnet-ids subnet-0ee841e1e73824361,subnet-0fb603f894ee740b1  \
  --nodes 3 \
  --nodes-min 2 \
  --nodes-max 4 \
  --ssh-access \
  --ssh-public-key /root/.ssh/id_ed25519.pub
2024-09-18 10:35:22 [ℹ]  will use version 1.29 for new nodegroup(s) based on control plane version
2024-09-18 10:35:23 [ℹ]  nodegroup "my-node-group" will use "ami-011ab3d6667efecdb" [Ubuntu2004/1.29]
2024-09-18 10:35:23 [ℹ]  using SSH public key "/root/.ssh/id_ed25519.pub" as "eksctl-surprise-cluster-nodegroup-my-node-group-h
p3rJMtwgssNdP4mAaFruuPFsANcqlFcFOShs"
2024-09-18 10:35:23 [ℹ]  skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.smal
2024-09-18 10:35:23 [ℹ]  1 nodegroup (my-node-group) was included (based on the include/exclude rules)
2024-09-18 10:35:23 [ℹ]  will create a CloudFormation stack for each of 1 managed nodegroups in cluster "surprise-cluster"
2024-09-18 10:35:23 [ℹ]
2 sequential tasks: { fix cluster compatibility, 1 task: { 1 task: { create managed nodegroup "my-node-group" } }
}
2024-09-18 10:35:23 [ℹ]  checking cluster stack for missing resources
2024-09-18 10:35:24 [ℹ]  cluster stack has all required resources
2024-09-18 10:35:24 [ℹ]  building managed nodegroup stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:24 [ℹ]  skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.smal
2024-09-18 10:35:24 [ℹ]  deploying stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:24 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:35:54 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:36:50 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
2024-09-18 10:37:39 [ℹ]  waiting for CloudFormation stack "eksctl-surprise-cluster-nodegroup-my-node-group"
```

For replica set we have to create this file

```yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: frontend
  labels:
    app: guestbook
    tier: frontend
spec:
  replicas: 4
  selector:
    matchLabels:
      tier: frontend
  template:
    metadata:
      labels:
        tier: frontend
    spec:
      containers:
      - name: apache-app
        image: nginx
```

```
root@ip-172-31-36-241:~# vim replicaset.yml
root@ip-172-31-36-241:~# kubectl apply -f replicaset.yml
replicaset.apps/frontend created
root@ip-172-31-36-241:~# kubectl get pods
NAME             READY   STATUS              RESTARTS   AGE
frontend-gs7dl   1/1     Running             0          10s
frontend-mfxpw   0/1     ContainerCreating   0          10s
frontend-p2rws   0/1     ContainerCreating   0          10s
frontend-t5jjl   0/1     ContainerCreating   0          10s
```

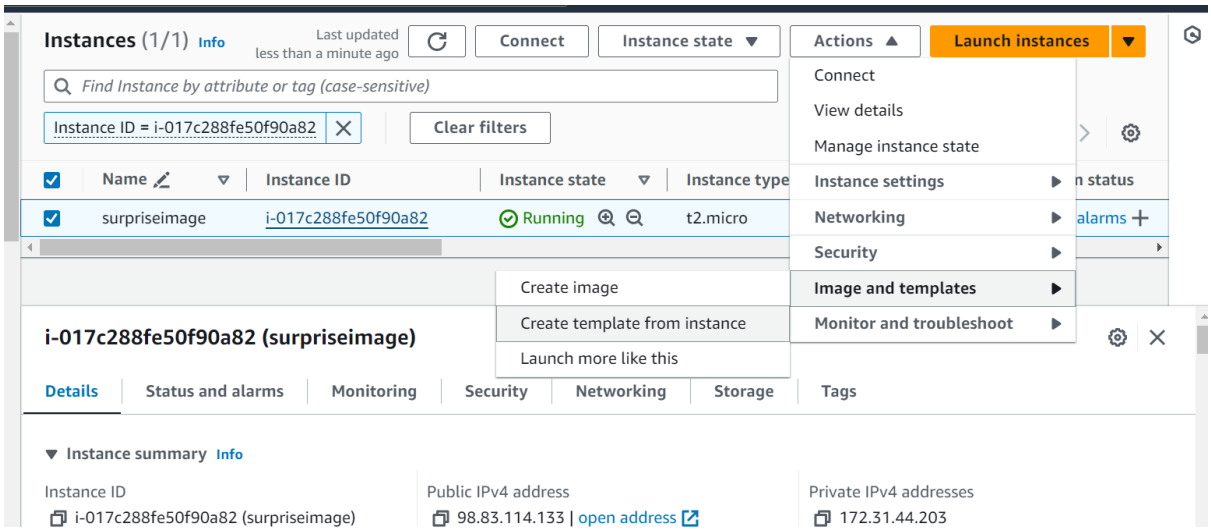
```
root@ip-172-31-36-241:~# kubectl describe pod frontend-gs7dl
Name:             frontend-gs7dl
Namespace:        default
Priority:         0
Service Account:  default
Node:             ip-172-31-12-154.ap-south-1.compute.internal/172.31.12.154
Start Time:       Wed, 18 Sep 2024 10:40:59 +0000
Labels:           tier=frontend
Annotations:      <none>
Status:           Running
IP:               172.31.8.193
IPs:
  IP:           172.31.8.193
Controlled By:  ReplicaSet/frontend
Containers:
  apache-app:
    Container ID:   containerd://7641b3b3b60b2fe482189da3186757f40e8b4195f0d7d778d4d63aa3f28209eb
```

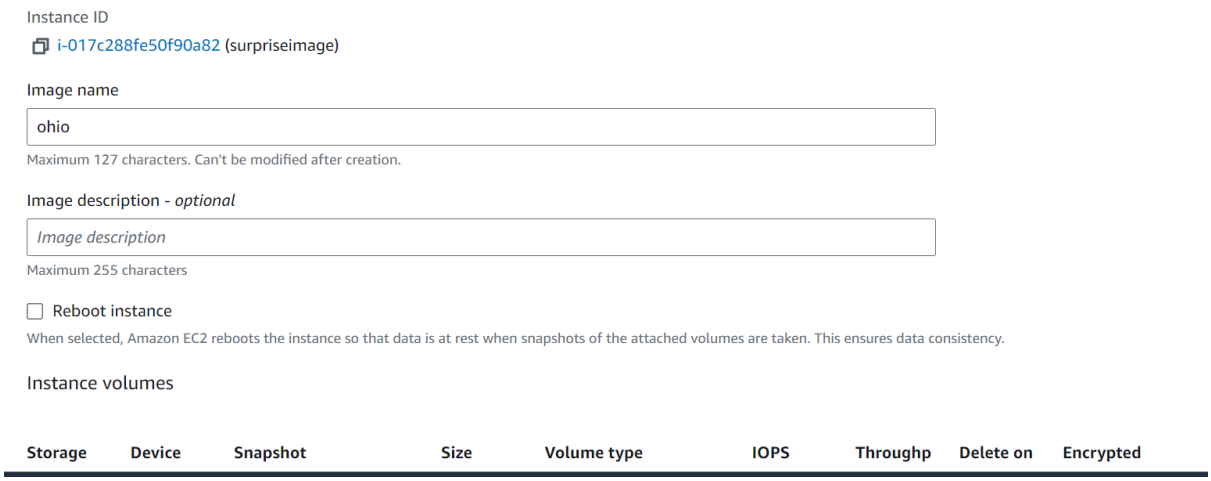
```
root@ip-172-31-36-241:~# kubectl get pod -o wide
NAME             READY   STATUS    RESTARTS   AGE    IP              NODE                                                NOMINATED NODE
READINESS GATES
frontend-gs7dl   1/1     Running   0          113s   172.31.8.193    ip-172-31-12-154.ap-south-1.compute.internal   <none>
<none>
frontend-mfxpw   1/1     Running   0          113s   172.31.36.143   ip-172-31-34-182.ap-south-1.compute.internal   <none>
<none>
frontend-p2rws   1/1     Running   0          113s   172.31.44.93    ip-172-31-40-144.ap-south-1.compute.internal   <none>
<none>
frontend-t5jjl   1/1     Running   0          113s   172.31.37.236   ip-172-31-34-182.ap-south-1.compute.internal   <none>
```

```
root@ip-172-31-36-241:~# kubectl autoscale rs frontend --max=10 --min=3
horizontalpodautoscaler.autoscaling/frontend autoscaled
root@ip-172-31-36-241:~# kubectl scale --replicas=9 -f replicaset.yml
replicaset.apps/frontend scaled
root@ip-172-31-36-241:~# kubectl get rs
NAME       DESIRED   CURRENT   READY   AGE
frontend   9         9         9       2m43s
```

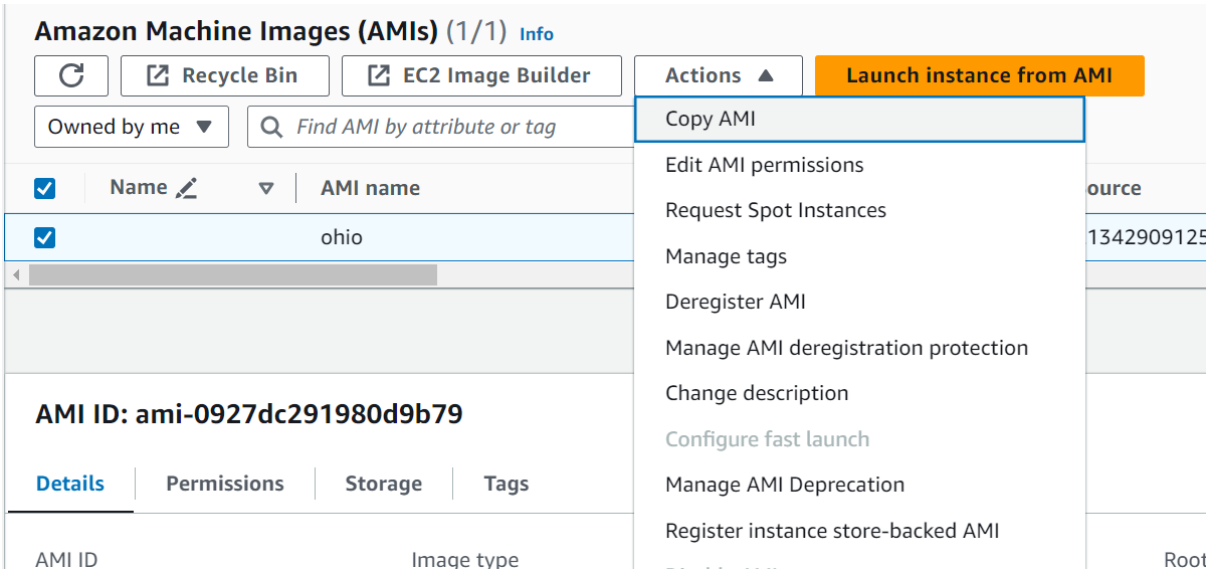
Ans4

Create one instance then create image then copy image

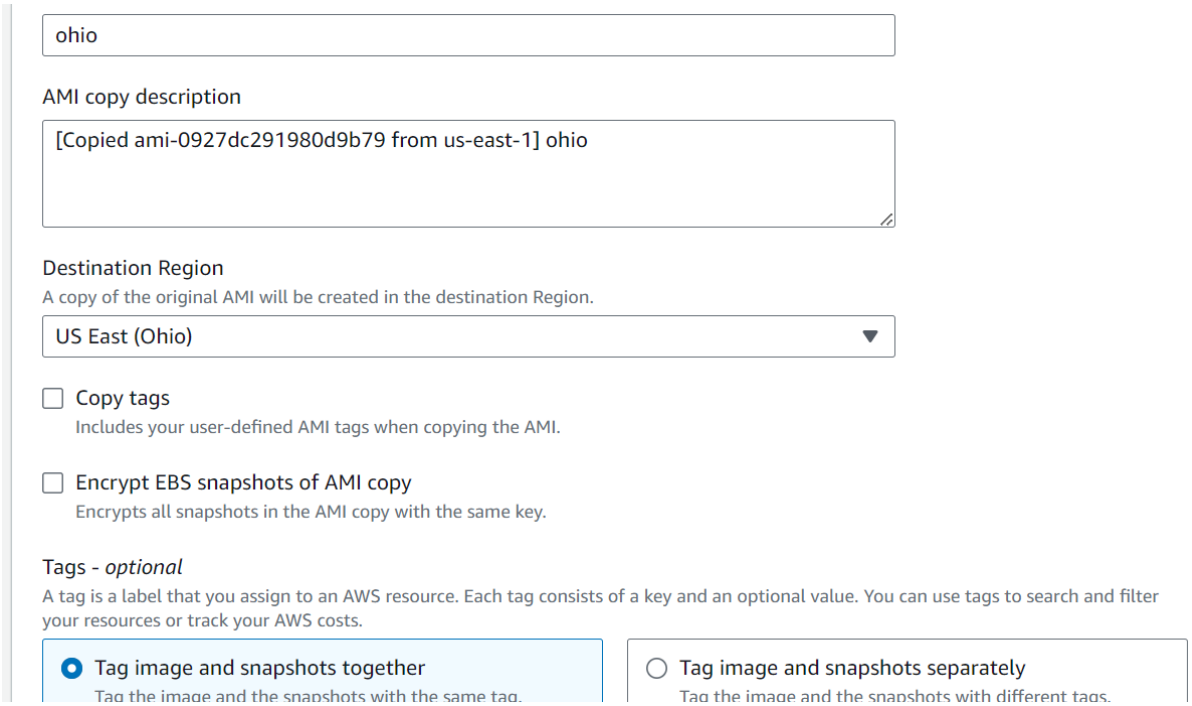




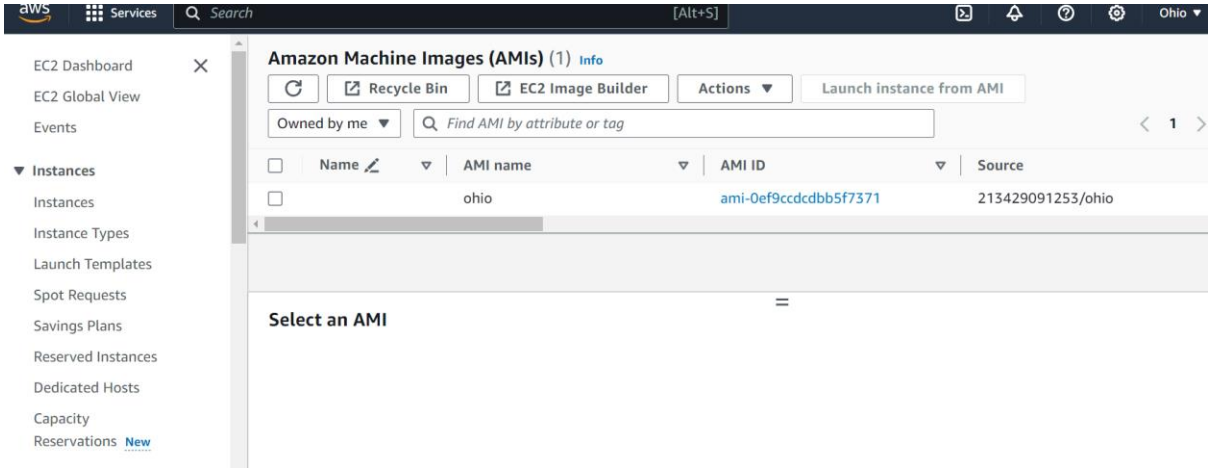
Copy image

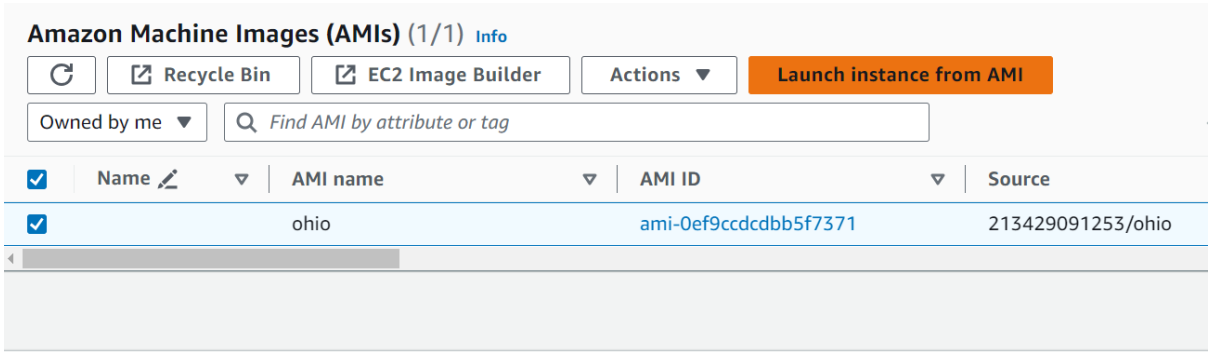


Choose ohio as destination

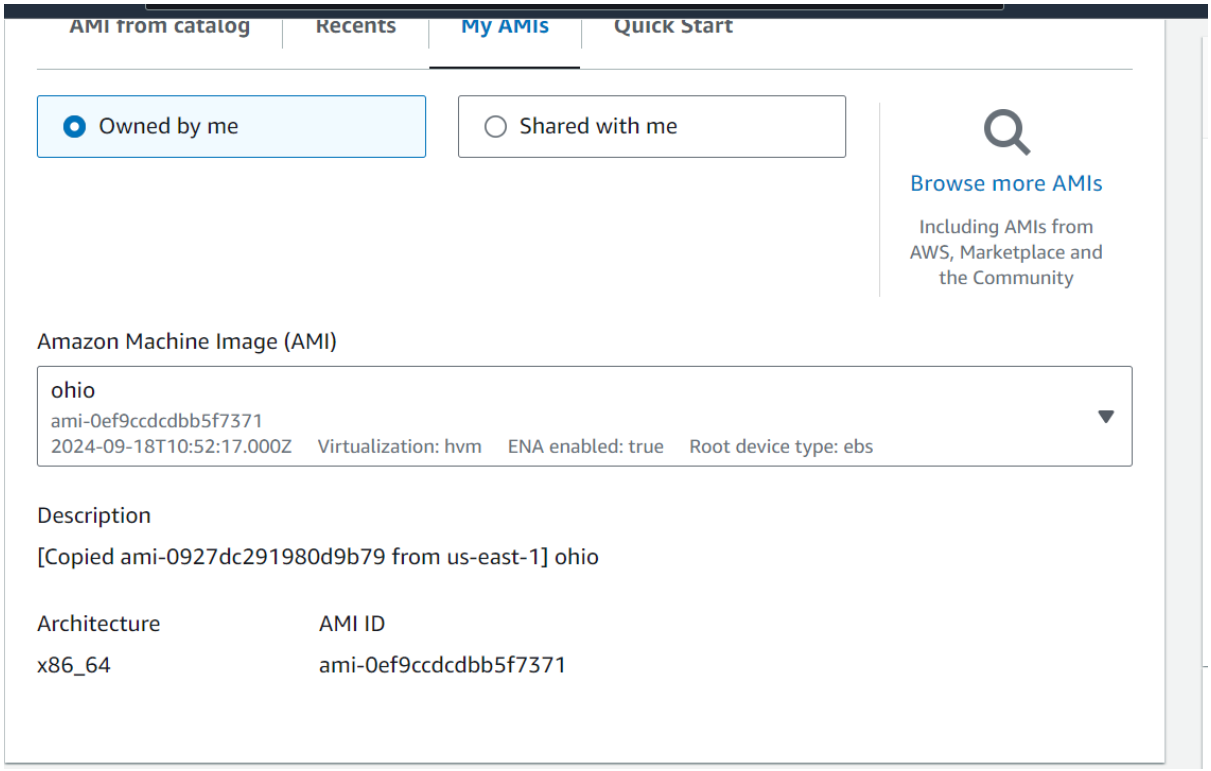


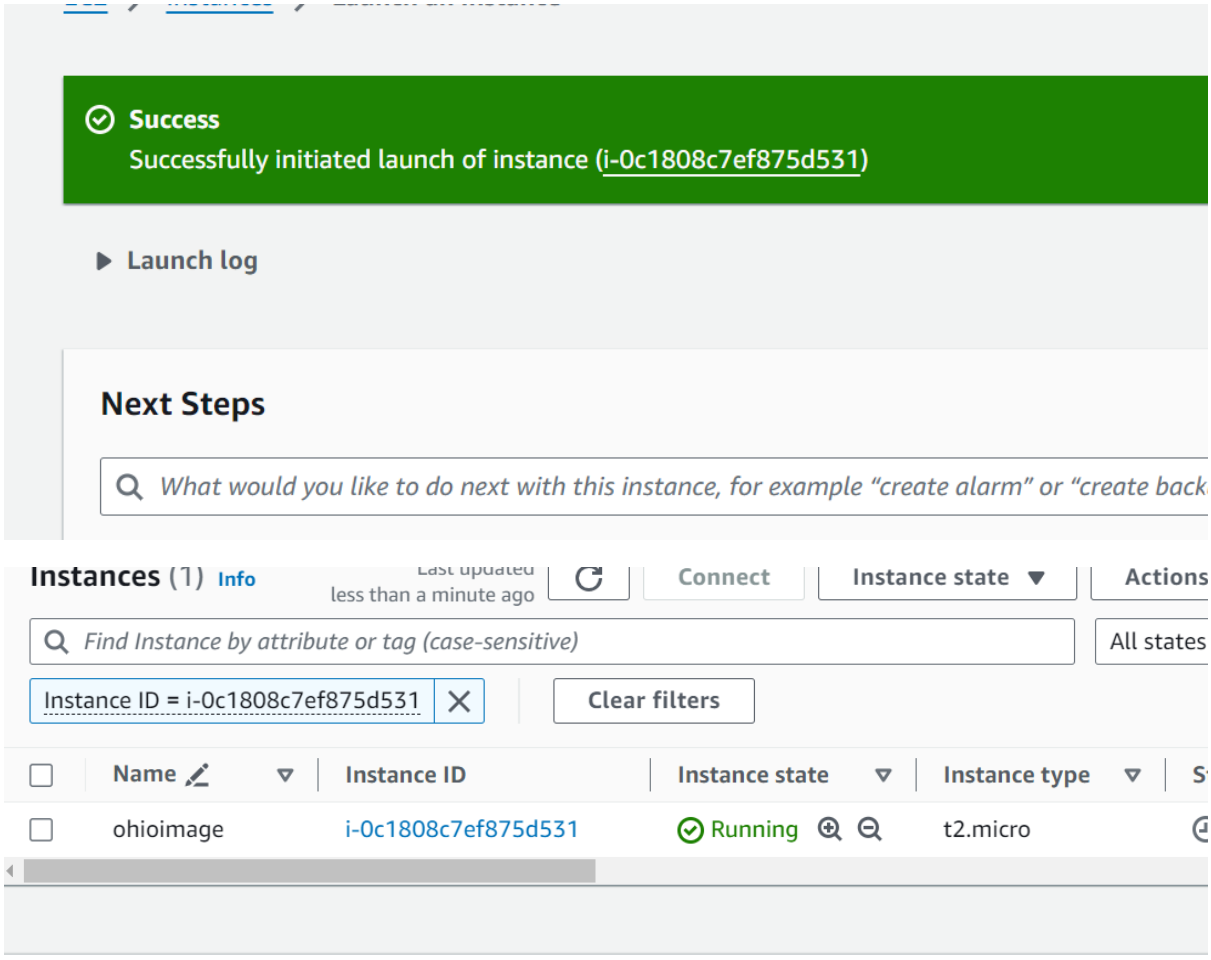
Now go to ohio regions and ami and launch the image





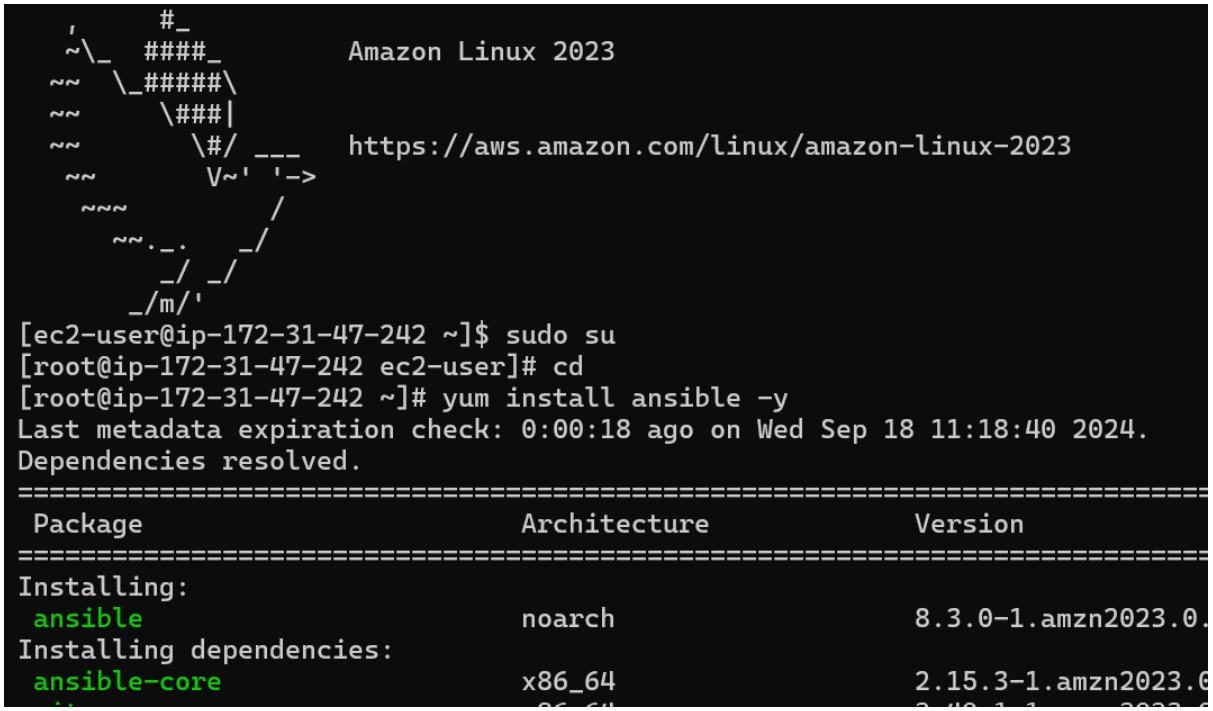
AMI ID: ami-0ef9ccdcdbb5f7371





Ans 5

First install ansible


,   #_
~_ ####_   Amazon Linux 2023
~~ _#####\
~~   \###|
~~   \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
~~   V~' '->
~~~   /
~~.~.   _/
_/ _/
_/m/'
[ec2-user@ip-172-31-47-242 ~]$ sudo su
[root@ip-172-31-47-242 ec2-user]# cd
[root@ip-172-31-47-242 ~]# yum install ansible -y
Last metadata expiration check: 0:00:18 ago on Wed Sep 18 11:18:40 2024.
Dependencies resolved.
Package Architecture Version
Installing:
ansible noarch 8.3.0-1.amzn2023.0.
Installing dependencies:
ansible-core x86_64 2.15.3-1.amzn2023.0
~~~