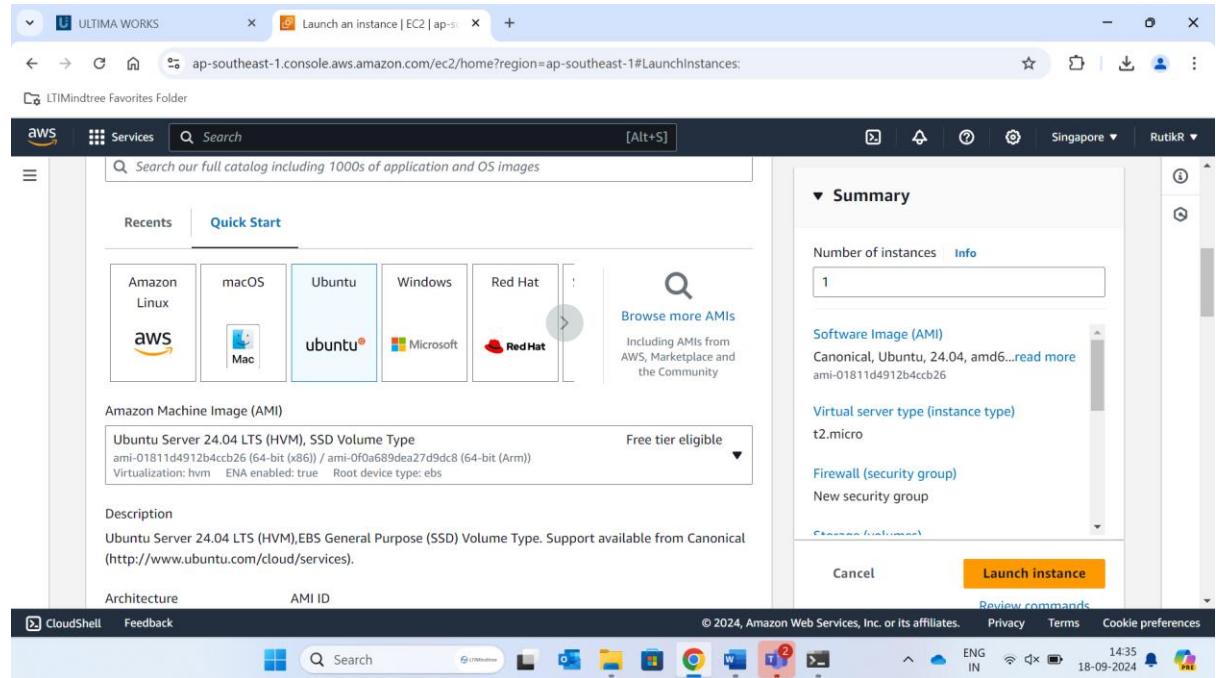
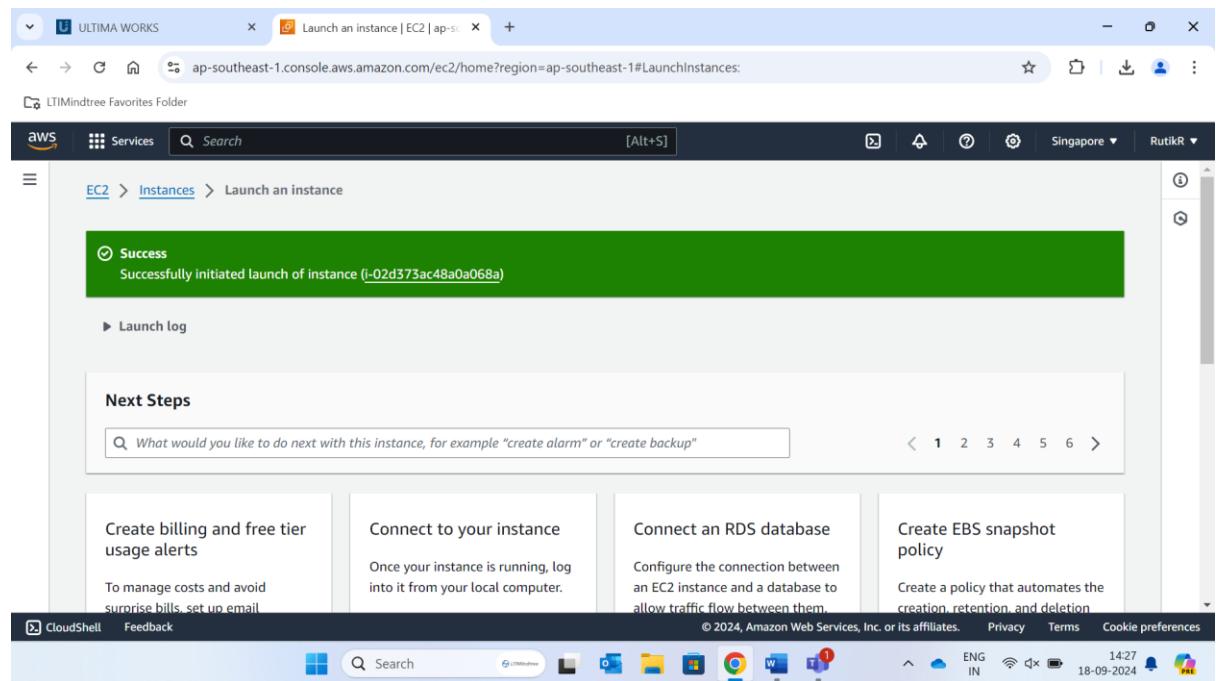


- 1) Created instance using below specifications. Used rhel to install docker



- 2) Created instance for a docker



- 3) Installing Docker on the created instance.

```
root@docker: ~
Get:16 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [616 B]
Get:17 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1124 kB]
Get:18 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [261 kB]
Get:19 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [26.1 kB]
Get:20 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [43.3 kB]
Get:21 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.8 kB]
Get:22 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.1 kB]
Get:23 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.8 kB]
Get:24 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [28.8 kB]
Get:25 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:26 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:27 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.8 kB]
Get:28 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:29 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [672 B]
Get:30 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1839 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [298 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2431 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [418 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [903 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [177 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [19.3 kB]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [228 B]
Fetched 33.8 MB in 5s (7196 kB/s)
Reading package lists... Done
root@docker:~#
```

4) Installing docker on the created instance

```
root@ip-172-31-20-74:~
File
Paste
Clipboard
root@ip-172-31-20-74:~ Transaction test succeeded.
Running transaction
Preparing : 1/1
Installing  : fuse3libs-3.10.2-8.el9.x86_64
Running scriptlet: container-selinux-3:2.229.0-1.el9_3.noarch 2/15
Installing  : container-selinux-3:2.229.0-1.el9_3.noarch 2/15
Running scriptlet: container-selinux-3:2.229.0-1.el9_3.noarch 2/15
Installing  : docker-compose-plugin-2.29.2-1.el9.x86_64 3/15
Running scriptlet: docker-compose-plugin-2.29.2-1.el9.x86_64 3/15
Installing  : containerd.io-1.7.22-3.1.el9.x86_64 4/15
Running scriptlet: containerd.io-1.7.22-3.1.el9.x86_64 4/15
Installing  : libnftnl-1.2.6-4.el9_4.x86_64 5/15
Installing  : iptables-nft-1.8.10-4.el9_4.x86_64 6/15
Running scriptlet: iptables-nft-1.8.10-4.el9_4.x86_64 6/15
Installing  : fuse-common-3.10.2-8.el9.x86_64 7/15
Installing  : fuse3-3.10.2-8.el9.x86_64 8/15
Installing  : fuse-overlayfs-1.13-1.el9.x86_64 9/15
Running scriptlet: fuse-overlayfs-1.13-1.el9.x86_64 9/15
Installing  : libslirp-4.4.0-7.el9.x86_64 10/15
Installing  : slirp4netns-1.2.3-1.el9.x86_64 11/15
Installing  : docker-buildx-plugin-0.16.2-1.el9.x86_64 12/15
Running scriptlet: docker-buildx-plugin-0.16.2-1.el9.x86_64 12/15
Installing  : docker-ce-cli-1:27.2.1-1.el9.x86_64 13/15
Running scriptlet: docker-ce-cli-1:27.2.1-1.el9.x86_64 13/15
Installing  : docker-ce-rootless-extras-27.2.1-1.el9.x86_64 14/15
Running scriptlet: docker-ce-rootless-extras-27.2.1-1.el9.x86_64 14/15
Installing  : docker-ce-3:27.2.1-1.el9.x86_64 15/15
Running scriptlet: docker-ce-3:27.2.1-1.el9.x86_64 15/15
Running scriptlet: container-selinux-3:2.229.0-1.el9_3.noarch 15/15
```

5) Started and enabled docker .

```
Verifying : docker-ce-rootless-extras-27.2.1-1.el9.x86_64 5/15
Verifying : docker-compose-plugin-2.29.2-1.el9.x86_64 6/15
Verifying : libslirp-4.4.0-7.el9.x86_64 7/15
Verifying : container-selinux-3:2.229.0-1.el9.noarch 8/15
Verifying : fuse3-libs-3.10.2-8.el9.x86_64 9/15
Verifying : fuse-overlayfs-1.13-1.el9.x86_64 10/15
Verifying : fuse3-3.10.2-8.el9.x86_64 11/15
Verifying : slirp4netns-1.2.3-1.el9.x86_64 12/15
Verifying : fuse-common-3.10.2-8.el9.x86_64 13/15
Verifying : libnftnl-1.2.6-4.el9_4.x86_64 14/15
Verifying : iptables-nft-1.8.10-4.el9_4.x86_64 15/15
Installed products updated.

Installed:
  container-selinux-3:2.229.0-1.el9.noarch
  docker-buildx-plugin-0.16.2-1.el9.x86_64
  docker-ce-cli-1:27.2.1-1.el9.x86_64
  docker-compose-plugin-2.29.2-1.el9.x86_64
  fuse-overlayfs-1.13-1.el9.x86_64
  fuse3-libs-3.10.2-8.el9.x86_64
  libnftnl-1.2.6-4.el9_4.x86_64
  slirp4netns-1.2.3-1.el9.x86_64
  containerd.io-1.7.22-3.1.el9.x86_64
  docker-ce-3:27.2.1-1.el9.x86_64
  docker-ce-rootless-extras-27.2.1-1.el9.x86_64
  fuse-common-3.10.2-8.el9.x86_64
  fuse3-3.10.2-8.el9.x86_64
  iptables-nft-1.8.10-4.el9_4.x86_64
  libslirp-4.4.0-7.el9.x86_64

Complete!
[root@docker ~]# sudo systemctl start docker
[root@docker ~]# systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[root@docker ~]# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
[root@docker ~]#
```

Page 2 of 2 32 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 15:05 18-09-2024

6) Pulling image from docker hub:

Pulling ubuntu image from docker using command docker pull ubuntu.

```
Verifying : fuse3-libs-3.10.2-8.el9.x86_64 9/15
Verifying : fuse-overlayfs-1.13-1.el9.x86_64 10/15
Verifying : fuse3-3.10.2-8.el9.x86_64 11/15
Verifying : slirp4netns-1.2.3-1.el9.x86_64 12/15
Verifying : fuse-common-3.10.2-8.el9.x86_64 13/15
Verifying : libnftnl-1.2.6-4.el9_4.x86_64 14/15
Verifying : iptables-nft-1.8.10-4.el9_4.x86_64 15/15
Installed products updated.

Installed:
  container-selinux-3:2.229.0-1.el9.noarch
  docker-buildx-plugin-0.16.2-1.el9.x86_64
  docker-ce-cli-1:27.2.1-1.el9.x86_64
  docker-compose-plugin-2.29.2-1.el9.x86_64
  fuse-overlayfs-1.13-1.el9.x86_64
  fuse3-libs-3.10.2-8.el9.x86_64
  libnftnl-1.2.6-4.el9_4.x86_64
  slirp4netns-1.2.3-1.el9.x86_64
  containerd.io-1.7.22-3.1.el9.x86_64
  docker-ce-3:27.2.1-1.el9.x86_64
  docker-ce-rootless-extras-27.2.1-1.el9.x86_64
  fuse-common-3.10.2-8.el9.x86_64
  fuse3-3.10.2-8.el9.x86_64
  iptables-nft-1.8.10-4.el9_4.x86_64
  libslirp-4.4.0-7.el9.x86_64

Complete!
[root@docker ~]# sudo systemctl start docker
[root@docker ~]# systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[root@docker ~]# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
[root@docker ~]# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
dafa2bdc44d2: Pulling fs layer
[root@docker ~]#
```

Page 3 of 3 53 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 15:07 18-09-2024

7)

Created container to use image ubuntu . And installed requirements:

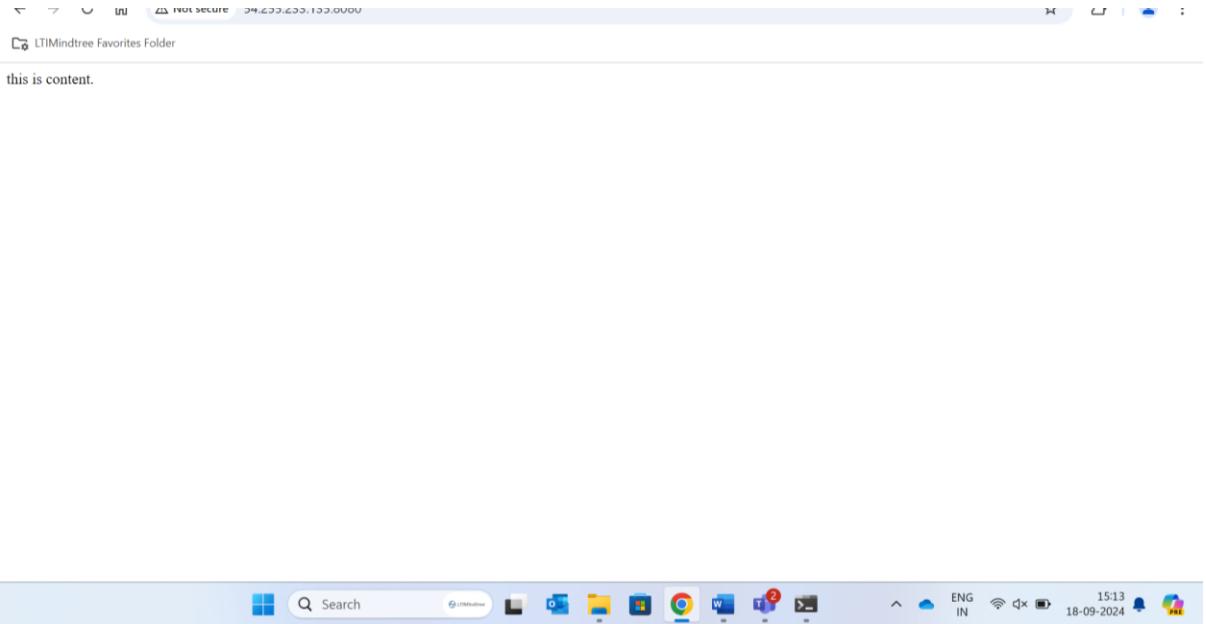
```
root@97244e963cc6:/ 
File 
Enabling module authn_file. 
Enabling module authz_user. 
Enabling module alias. 
Enabling module dir. 
Enabling module autoindex. 
Enabling module env. 
Enabling module mime. 
Enabling module negotiation. 
Enabling module setenvif. 
Enabling module filter. 
Enabling module deflate. 
Enabling module status. 
Enabling module reqtimeout. 
Enabling conf charset. 
Enabling conf localized-error-pages. 
Enabling conf other-vhosts-access-log. 
Enabling conf security. 
Enabling conf serve-cgi-bin. 
Enabling site 000-default. 
invoke-rc.d: could not determine current runlevel 
invoke-rc.d: policy-rc.d denied execution of start. 
Processing triggers for libc-bin (2.39-0ubuntu8.3) ... 
Processing triggers for ca-certificates (20240203) ... 
Updating certificates in /etc/ssl/certs... 
0 added, 0 removed; done. 
Running hooks in /etc/ca-certificates/update.d... 
done. 
root@97244e963cc6:/#
```

Page 3 of 3 54 words English (India) Text Predictions: On Accessibility: Investigate Focus Search ENG IN 15:09 18-09-2024

7) Successfully hosted web-app which is available globally.

```
root@ip-172-31-20-74:~ 
File 
valid_lft forever preferred_lft forever 
inet6 ::1/128 scope host 
    valid_lft forever preferred_lft forever 
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000 
    link/ether 02:20:a5:6a:d6:bb brd ff:ff:ff:ff:ff:ff 
    altname enX0 
    inet 172.31.20.74/20 brd 172.31.31.255 scope global dynamic noprefixroute eth0 
        valid_lft 3056sec preferred_lft 3056sec 
        inet6 fe80::20:a5ff:fe6a:d6bb/64 scope link 
            valid_lft forever preferred_lft forever 
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default 
    link/ether 02:42:a6:0:f1:bf brd ff:ff:ff:ff:ff:ff 
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0 
        valid_lft forever preferred_lft forever 
        inet6 fe80::42:a6ff:fed0:f1bf/64 scope link 
            valid_lft forever preferred_lft forever 
5: vethadc8bf1@if4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default 
    link/ether 66:c9:a8:19:e8:1b brd ff:ff:ff:ff:ff:ff link-netnsid 0 
    inet6 fe80::64c9:a8ff:fe19:e81b/64 scope link 
        valid_lft forever preferred_lft forever 
[root@docker ~]# curl http://172.17.0.1:8080 
this is content. 
[root@docker ~]# history 
1 rpmquery docker 
2 yum update -y 
3 sudo yum install -y yum-utils 
4 sudo yum-config-manager --add-repo 
5 https://download.docker.com/linux/rhel/docker-ce.repo 
6 sudo yum-config-manager --add-repo 
7 https://download.docker.com/linux/rhel/docker-ce.repo
```

Page 4 of 4 64 words English (India) Text Predictions: On Accessibility: Investigate Focus Search ENG IN 15:12 18-09-2024



Q2) Deploy a Nginx application on your Kubernetes cluster. and it should be available across of the cluster on port NO. 80.

Ans:

So created instance to create eks cluster.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. The main area displays a table of instances:

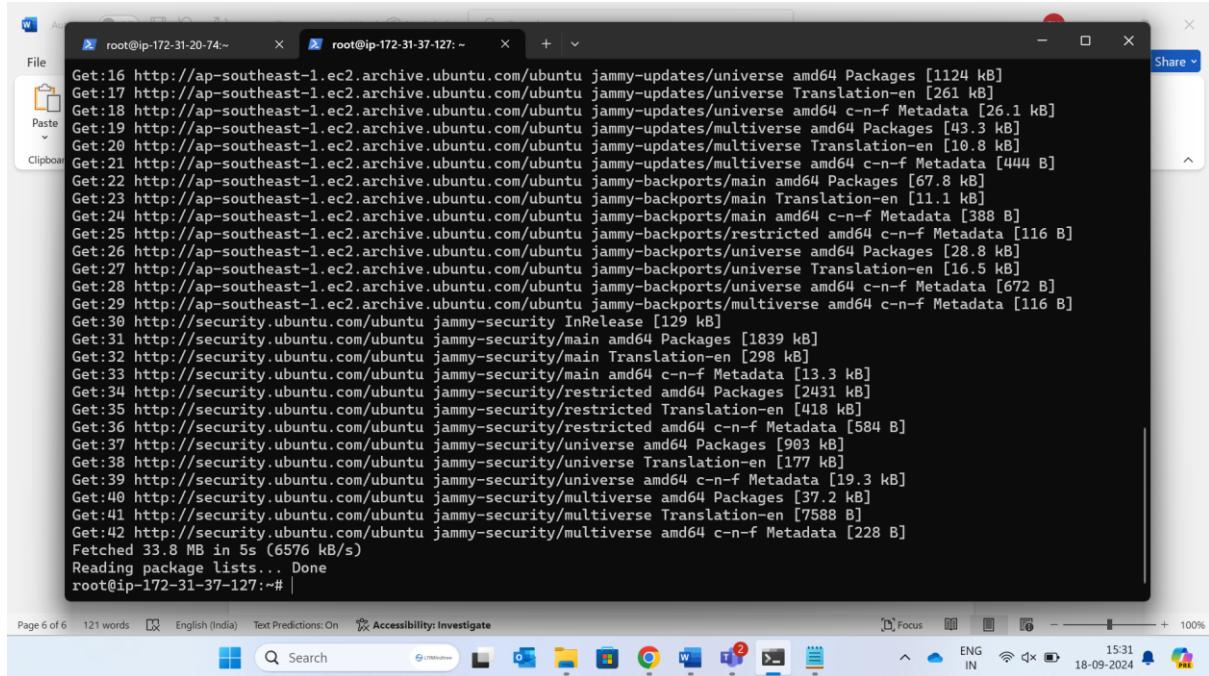
Name	Instance ID	Instance state	Instance type	Status check	Alarm status
docker	i-02d373ac48a0a068a	Running	t2.micro	2/2 checks passed	View alarms
eks-cluster	i-0224c523c768ff7f9	Pending	t2.medium	-	View alarms
jenkins	i-0e5e70003beb5a1d9	Running	t2.medium	2/2 checks passed	View alarms
docker2	i-09f240846d569ed5a	Running	t2.micro	2/2 checks passed	View alarms

Below the table, there's a detailed view for the instance **i-0224c523c768ff7f9 (eks-cluster)**. It shows the following details:

- Details** tab selected.
- Instance summary**:
 - Instance ID: i-0224c523c768ff7f9 (eks-cluster)
 - Public IPv4 address: 54.254.184.84 | [open address](#)
 - Private IPv4 addresses: 172.31.39.4
 - Public IPv4 DNS
- Networking**, **Storage**, and **Tags** tabs are also present.

Setting up the eks cluster on the created instance

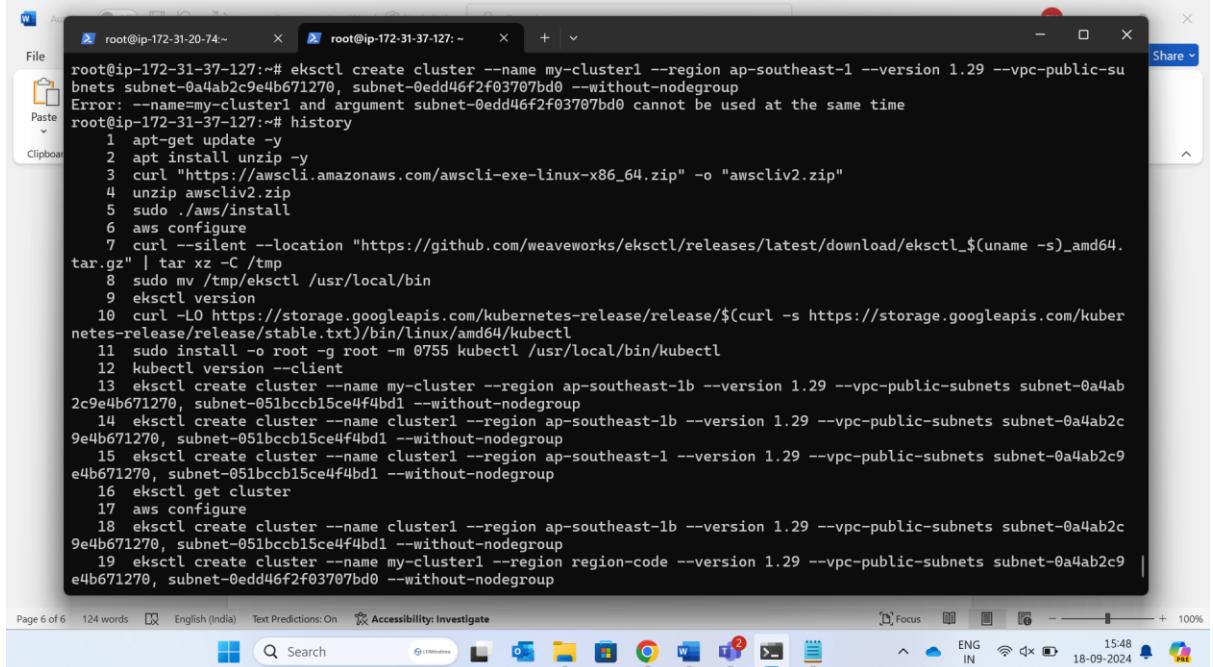
1) Created image and attached an policies to the instance.



```
Get:16 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1124 kB]
Get:17 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [261 kB]
Get:18 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe and64 c-n-f Metadata [26.1 kB]
Get:19 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [43.3 kB]
Get:20 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.8 kB]
Get:21 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [444 B]
Get:22 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.8 kB]
Get:23 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.1 kB]
Get:24 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:25 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:26 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.8 kB]
Get:27 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:28 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [672 B]
Get:29 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:30 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1839 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [298 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2431 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [418 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [903 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [177 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [19.3 kB]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [228 B]
Fetched 33.8 MB in 5s (6576 kB/s)
Reading package lists... Done
root@ip-172-31-37-127:~# |
```

Updated the system

2) Created a cluster using following commands



```
root@ip-172-31-37-127:~# eksctl create cluster --name my-cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-0edd46f2f03707bd0 --without-nodegroup
Error: --name=my-cluster1 and argument subnet-0edd46f2f03707bd0 cannot be used at the same time
root@ip-172-31-37-127:~# history
 1 apt-get update -y
 2 apt install unzip -y
 3 curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
 4 unzip awscliv2.zip
 5 sudo ./aws/install
 6 aws configure
 7 curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_amd64.tar.gz" | tar xz -C /tmp
 8 sudo mv /tmp/eksctl /usr/local/bin
 9 eksctl version
10 curl -L0 https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
11 sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
12 kubectl version --client
13 eksctl create cluster --name my-cluster --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
14 eksctl create cluster --name cluster1 --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
15 eksctl create cluster --name cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
16 eksctl get cluster
17 aws configure
18 eksctl create cluster --name cluster1 --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
19 eksctl create cluster --name my-cluster1 --region region-code --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-0edd46f2f03707bd0 --without-nodegroup
root@ip-172-31-37-127:~# |
```

Cluster creation is taking place

```
root@ip-172-31-20-74:~# eksctl create cluster --name my-cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subs
bnets subnet-0edd46f2f03707bd0,subnet-0a4ab2c9e4b671270 --without-nodegroup
2024-09-18 10:21:05 [!] eksctl version 0.190.0
2024-09-18 10:21:05 [!] using region ap-southeast-1
2024-09-18 10:21:05 [!] using existing VPC (vpc-00a1742b758b543b0) and subnets (private:map[] public:map[ap-southeast-1b:{subnet-0a4ab2c9e4b671270 ap-southeast-1b 172.31.32.0/20 0 } ap-southeast-1c:{subnet-0edd46f2f03707bd0 ap-southeast-1c 172.31.0.0/20 0 }])
2024-09-18 10:21:05 [!] 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 10:21:05 [!] using Kubernetes version 1.29
2024-09-18 10:21:05 [!] creating EKS cluster "my-cluster1" in "ap-southeast-1" region with
2024-09-18 10:21:05 [!] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=ap-southeast-1 --cluster=my-cluster1'
2024-09-18 10:21:05 [!] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "my-cluster1" in "ap-southeast-1"
2024-09-18 10:21:05 [!] CloudWatch logging will not be enabled for cluster "my-cluster1" in "ap-southeast-1"
2024-09-18 10:21:05 [!] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=ap-southeast-1 --cluster=my-cluster1'
2024-09-18 10:21:05 [!] default addons vpc-cni, kube-proxy, coredns were not specified, will install them as EKS addons
2024-09-18 10:21:05 [!]
2 sequential tasks: { create cluster control plane "my-cluster1",
  2 sequential sub-tasks: {
    1 task: { create addons },
    wait for control plane to become ready,
  }
}
2024-09-18 10:21:05 [!] building cluster stack "eksctl-my-cluster1-cluster"
2024-09-18 10:21:06 [!] deploying stack "eksctl-my-cluster1-cluster"
```

Page 6 of 7 131 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 15:51 18-09-2024

Cluster creation is successful:

```
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro

System information as of Wed Sep 18 10:38:30 UTC 2024
System load: 0.0          Processes:           113
Usage of /: 33.9% of 7.57GB  Users logged in: 0
Memory usage: 5%          IPv4 address for eth0: 172.31.37.127
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

78 updates can be applied immediately.
43 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Sep 18 10:00:06 2024 from 167.103.3.70
ubuntu@ip-172-31-37-127:~$ sudo su -
root@ip-172-31-37-127:~# eksctl get cluster
NAME      REGION      EKSCTL CREATED
my-cluster1  ap-southeast-1  True
root@ip-172-31-37-127:~#
```

Page 8 of 8 166 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 16:08 18-09-2024

Created a node groups.

The screenshot shows a terminal window in AWS CloudShell with two tabs open. The left tab is titled 'root@ip-172-31-20-74:' and the right tab is titled 'root@ip-172-31-37-127:~'. The terminal output shows the execution of the 'eksctl create nodegroup' command:

```
E *.*.+ + o
. .o+= = + .
. O.*.+ + o
B.+$*.+
. +*o===
. o.++ .
.
.
[SHA256]-----
root@ip-172-31-37-127:~# eksctl create nodegroup \
--cluster my-cluster1 \
--region ap-southeast-1 \
--name my-node-group \
--node-ami-family Ubuntu2004 \
--node-type t2.small \
--subnet-ids subnet-0edd46f2f03707bd0,subnet-0a4ab2c9e4b671270 \
--nodes 3 \
--nodes-min 2 \
--nodes-max 4 \
--ssh-access \
--ssh-public-key /root/.ssh/id_rsa.pub
2024-09-18 10:58:08 [i] will use version 1.29 for new nodegroup(s) based on control plane version
2024-09-18 10:58:08 [i] nodegroup "my-node-group" will use "ami-09c3a3b747509e547" [Ubuntu2004/1.29]
2024-09-18 10:58:08 [i] using SSH public key "/root/.ssh/id_rsa.pub" as "eksctl-my-cluster1-nodegroup-my-node-group-ec:6a:ce:ca:08:f8:e8:8f:7f:a6:26:d2:ed:62:7f:ad"
2024-09-18 10:58:09 [i] 1 nodegroup (my-node-group) was included (based on the include/exclude rules)
2024-09-18 10:58:09 [i] will create a CloudFormation stack for each of 1 managed nodegroups in cluster "my-cluster1"
2024-09-18 10:58:09 [i] 2 sequential tasks: { fix cluster compatibility, 1 task: { create managed nodegroup "my-node-group" } }
```

The terminal also displays several informational messages from eksctl about the node group creation process.

Created deployment using nginx to host web-app

The screenshot shows a terminal window in AWS CloudShell with two tabs open. The left tab is titled 'root@ip-172-31-20-74:' and the right tab is titled 'root@ip-172-31-37-127:~'. The terminal output shows the execution of various commands to update the system, get cluster information, and create a deployment:

```
File
78 updates can be applied immediately.
43 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Sep 18 10:00:06 2024 from 167.103.3.70
ubuntu@ip-172-31-37-127:~$ sudo su -
root@ip-172-31-37-127:~# eksctl get cluster
NAME      REGION      EKSCTL CREATED
my-cluster1  ap-southeast-1  True
root@ip-172-31-37-127:~# vim replica.yml
root@ip-172-31-37-127:~# vim repli.yml

root@ip-172-31-37-127:~# kubectl get pods
No resources found in default namespace.
root@ip-172-31-37-127:~# kubectl apply -f repli.yml
pod/web-app created
root@ip-172-31-37-127:~# kubectl get pods
NAME      READY  STATUS   RESTARTS  AGE
web-app  0/1    Pending  0          6s
root@ip-172-31-37-127:~# vim b.yml
root@ip-172-31-37-127:~# kubectl apply -f b.yml
deployment.apps/nginx-deployment created
root@ip-172-31-37-127:~#
```

The terminal shows the user performing system updates, checking the cluster configuration, and then creating a deployment named 'nginx-deployment' using an NGINX pod template.

Deployment is exposed on the port 80.



```
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

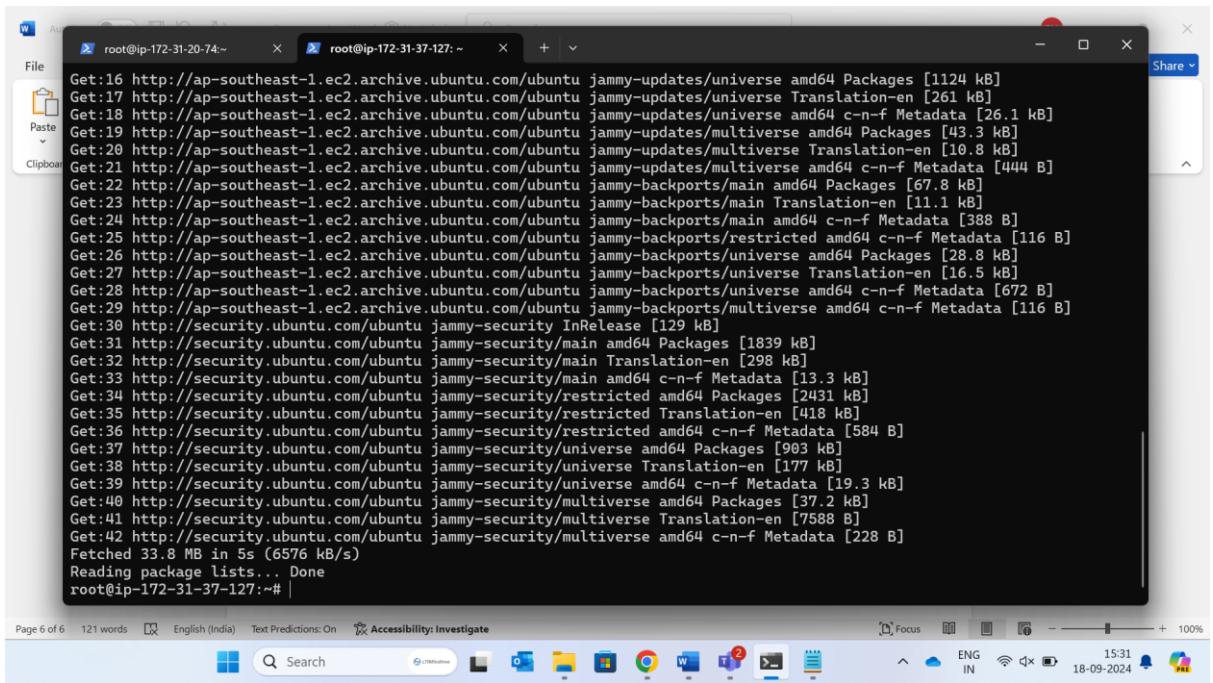
Last login: Wed Sep 18 10:00:06 2024 from 167.103.3.70
ubuntu@ip-172-31-37-127:~$ sudo su -
root@ip-172-31-37-127:~# eksctl get cluster
NAME      REGION      EKSCTL CREATED
my-cluster1  ap-southeast-1  True
root@ip-172-31-37-127:~# vim replica.yml
root@ip-172-31-37-127:~# vim repli.yml

root@ip-172-31-37-127:~# kubectl get pods
No resources found in default namespace.
root@ip-172-31-37-127:~# kubectl apply -f repli.yml
pod/web-app created
root@ip-172-31-37-127:~# kubectl get pods
NAME      READY  STATUS    RESTARTS   AGE
web-app  0/1    Pending   0          6s
root@ip-172-31-37-127:~# vim b.yml
root@ip-172-31-37-127:~# kubectl apply -f b.yml
deployment.apps/nginx-deployment created
root@ip-172-31-37-127:~# kubectl expose deployment/nginx-deployment --type="LoadBalancer" --port=80
service/nginx-deployment exposed
root@ip-172-31-37-127:~# |
```

Q3)

Setting up the eks cluster on the created instance

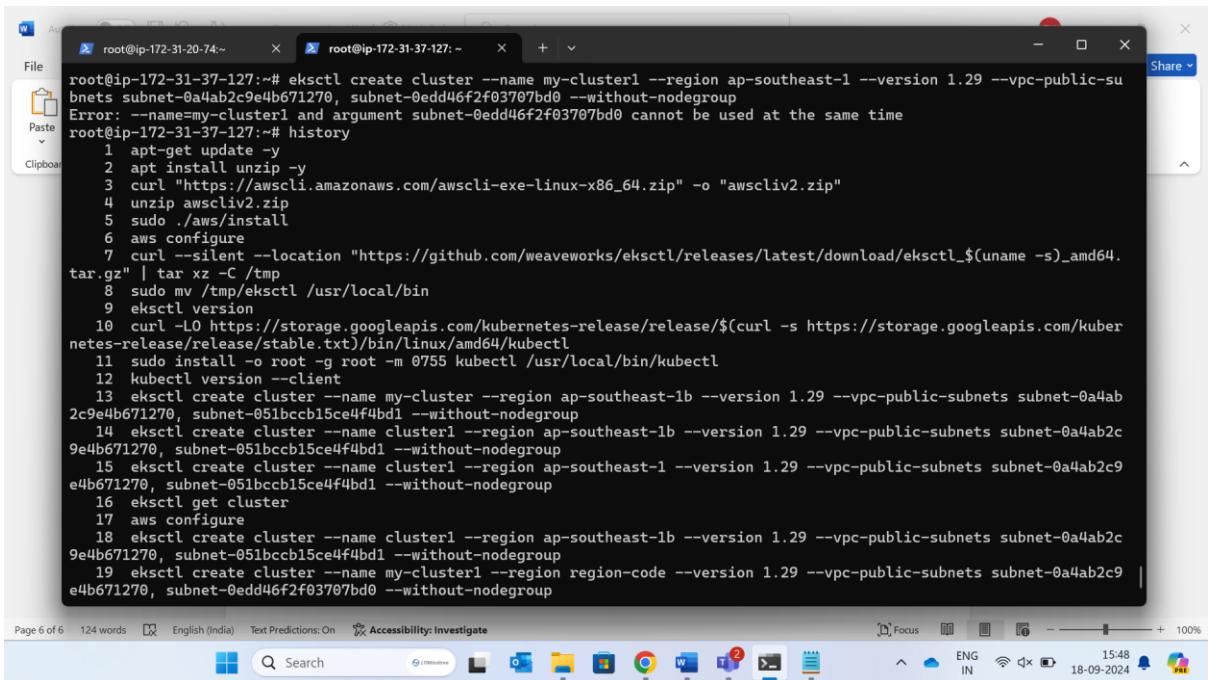
- 1) Created image and attached an policies to the instance.



```
Get:16 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1124 kB]
Get:17 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [261 kB]
Get:18 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [26.1 kB]
Get:19 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [43.3 kB]
Get:20 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.8 kB]
Get:21 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [444 kB]
Get:22 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.8 kB]
Get:23 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.1 kB]
Get:24 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:25 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:26 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.8 kB]
Get:27 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:28 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [672 B]
Get:29 http://ap-southeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:30 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1839 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [298 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2431 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [418 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [903 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [177 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [19.3 kB]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [228 B]
Fetched 33.8 MB in 5s (6576 kB/s)
Reading package lists... Done
root@ip-172-31-37-127:~# |
```

Updated the system

2) Created a cluster using following commands



```
root@ip-172-31-37-127:~# eksctl create cluster --name my-cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-0edd46f2f03707bd0 --without-nodegroup
Error: --name=my-cluster1 and argument subnet-0edd46f2f03707bd0 cannot be used at the same time
root@ip-172-31-37-127:~# history
 1  apt-get update -y
 2  apt install unzip -y
 3  curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
 4  unzip awscliv2.zip
 5  sudo ./aws/install
 6  aws configure
 7  curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_amd64.tar.gz" | tar xz -C /tmp
 8  sudo mv /tmp/eksctl /usr/local/bin
 9  eksctl version
10  curl -L0 https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
11  sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
12  kubectl version --client
13  eksctl create cluster --name my-cluster --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
14  eksctl create cluster --name cluster1 --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
15  eksctl create cluster --name cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
16  eksctl get cluster
17  aws configure
18  eksctl create cluster --name cluster1 --region ap-southeast-1b --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-051bccb15ce4f4bd1 --without-nodegroup
19  eksctl create cluster --name my-cluster1 --region region-code --version 1.29 --vpc-public-subnets subnet-0a4ab2c9e4b671270, subnet-0edd46f2f03707bd0 --without-nodegroup
root@ip-172-31-37-127:~# |
```

Cluster creation is taking place

```
File 22 history
root@ip-172-31-20-74:~# eksctl create cluster --name my-cluster1 --region ap-southeast-1 --version 1.29 --vpc-public-subs
bnets subnet-0edd46f2f03707bd0,subnet-0a4ab2c9e4b671270 --without-nodegroup
2024-09-18 10:21:05 [!] eksctl version 0.190.0
2024-09-18 10:21:05 [!] using region ap-southeast-1
2024-09-18 10:21:05 [!] using existing VPC (vpc-00a1742b758b543b0) and subnets (private:map[] public:map[ap-southeast-1b:{subnet-0a4ab2c9e4b671270 ap-southeast-1b 172.31.32.0/20 0 } ap-southeast-1c:{subnet-0edd46f2f03707bd0 ap-southeast-1c 172.31.0.0/20 0 }])
2024-09-18 10:21:05 [!] 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 10:21:05 [!] using Kubernetes version 1.29
2024-09-18 10:21:05 [!] creating EKS cluster "my-cluster1" in "ap-southeast-1" region with
2024-09-18 10:21:05 [!] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=ap-southeast-1 --cluster=my-cluster1'
2024-09-18 10:21:05 [!] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "my-cluster1" in "ap-southeast-1"
2024-09-18 10:21:05 [!] CloudWatch logging will not be enabled for cluster "my-cluster1" in "ap-southeast-1"
2024-09-18 10:21:05 [!] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=ap-southeast-1 --cluster=my-cluster1'
2024-09-18 10:21:05 [!] default addons vpc-cni, kube-proxy, coredns were not specified, will install them as EKS addons
2024-09-18 10:21:05 [!]
2 sequential tasks: { create cluster control plane "my-cluster1",
  2 sequential sub-tasks: {
    1 task: { create addons },
    wait for control plane to become ready,
  }
}
2024-09-18 10:21:05 [!] building cluster stack "eksctl-my-cluster1-cluster"
2024-09-18 10:21:06 [!] deploying stack "eksctl-my-cluster1-cluster"
```

Page 6 of 7 131 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 15:51 18-09-2024

Cluster creation is successful:

```
File * Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
Clipboard System information as of Wed Sep 18 10:38:30 UTC 2024
System load: 0.0 Processes: 113
Usage of /: 33.9% of 7.57GB Users logged in: 0
Memory usage: 5% IPv4 address for eth0: 172.31.37.127
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

78 updates can be applied immediately.
43 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Sep 18 10:00:06 2024 from 167.103.3.70
ubuntu@ip-172-31-37-127:~$ sudo su -
root@ip-172-31-37-127:~# eksctl get cluster
NAME REGION EKSCTL CREATED
my-cluster1 ap-southeast-1 True
root@ip-172-31-37-127:~#
```

Page 8 of 8 166 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 16:08 18-09-2024

```

root@ip-172-31-20-74:~ % aws lambda create-function --function-name LTM --runtime python3.10 --role arn:aws:iam::123456789012:lambda-execution-role --handler lambda_function.lambda_handler --code S3Bucket=LambdaFunctionBucket,S3Key=lambda_function.zip
{
    "FunctionName": "LTM",
    "FunctionArn": "arn:aws:lambda:ap-southeast-1:123456789012:function:LTM"
}

```

Now create the replicaset using yml file.

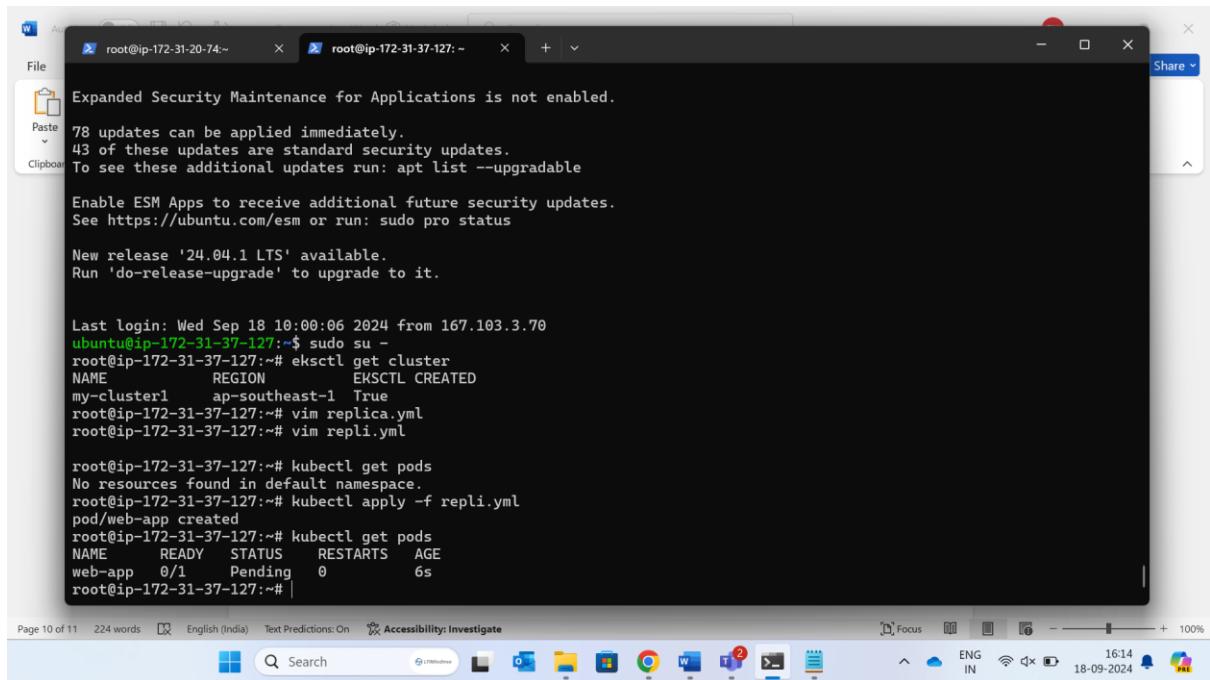
```

root@ip-172-31-20-74:~ % kubectl apply -f replica.yml
replicaset "replica" created

```

Currently there are no pods

So create pods

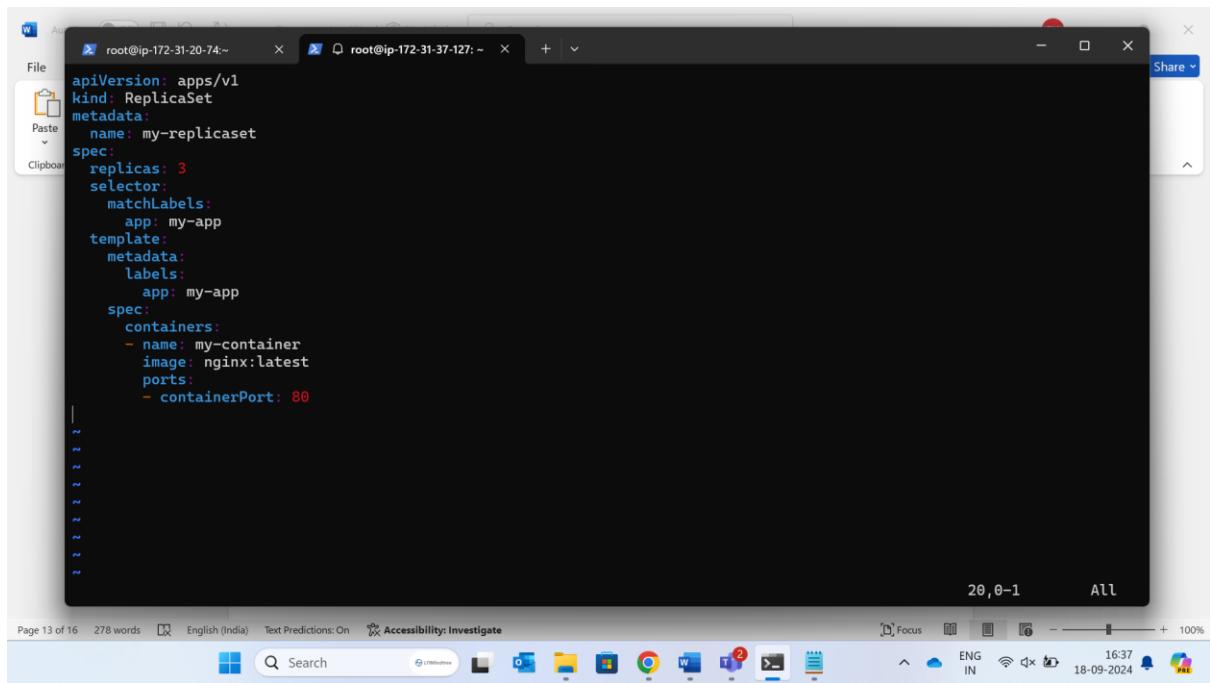


```
Expanded Security Maintenance for Applications is not enabled.  
78 updates can be applied immediately.  
43 of these updates are standard security updates.  
To see these additional updates run: apt list --upgradable  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
New release '24.04.1 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.  
  
Last login: Wed Sep 18 10:00:06 2024 from 167.103.3.70  
ubuntu@ip-172-31-37-127:~$ sudo su -  
root@ip-172-31-37-127:~# eksctl get cluster  
NAME REGION EKSCTL CREATED  
my-cluster1 ap-southeast-1 True  
root@ip-172-31-37-127:~# vim replica.yml  
root@ip-172-31-37-127:~# vim repli.yml  
  
root@ip-172-31-37-127:~# kubectl get pods  
No resources found in default namespace.  
root@ip-172-31-37-127:~# kubectl apply -f repli.yml  
pod/web-app created  
root@ip-172-31-37-127:~# kubectl get pods  
NAME READY STATUS RESTARTS AGE  
web-app 0/1 Pending 0 6s
```

Page 10 of 11 224 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 16:14 18-09-2024

Created web-app and also created pod.

Creating replicaset :



```
apiVersion: apps/v1  
kind: ReplicaSet  
metadata:  
  name: my-replicaset  
spec:  
  replicas: 3  
  selector:  
    matchLabels:  
      app: my-app  
  template:  
    metadata:  
      labels:  
        app: my-app  
    spec:  
      containers:  
      - name: my-container  
        image: nginx:latest  
        ports:  
        - containerPort: 80
```

Page 13 of 16 278 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 16:37 18-09-2024

\

Created replica sets.

```
root@ip-172-31-20-74:~  x root@ip-172-31-37-127:~ + 
File Paste Clipboard
2024-09-18 10:58:09 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 10:58:39 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 10:59:10 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:01:00 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:01:31 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:02:23 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:02:23 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:02:23 [1] no tasks
2024-09-18 11:02:23 [1] created 0 nodegroup(s) in cluster "my-cluster1"
2024-09-18 11:02:23 [1] nodegroup "my-node-group" has 3 node(s)
2024-09-18 11:02:23 [1] node "ip-172-31-12-110.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-33-180.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-45-96.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] waiting for at least 2 node(s) to become ready in "my-node-group"
2024-09-18 11:02:23 [1] nodegroup "my-node-group" has 3 node(s)
2024-09-18 11:02:23 [1] node "ip-172-31-12-110.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-33-180.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-45-96.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] created 1 managed nodegroup(s) in cluster "my-cluster1"
2024-09-18 11:02:23 [1] checking security group configuration for all nodegroups
2024-09-18 11:02:23 [1] all nodegroups have up-to-date cloudformation templates
root@ip-172-31-37-127:~#
root@ip-172-31-37-127:~#
root@ip-172-31-37-127:~# vim aaa.yml
root@ip-172-31-37-127:~# kubectl apply -f aaa.yml
replicaset.apps/my-replicaset created
root@ip-172-31-37-127:~# kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
my-replicaset 3         3         1       7s
nginx-deployment-55f6fc556d 2         2         0       16m
root@ip-172-31-37-127:~# |
```

Page 13 of 16 278 words English (India) Text Predictions: On Accessibility: Investigate 16:37 18-09-2024

Replica set is scaled using below command

```
root@ip-172-31-20-74:~  x root@ip-172-31-37-127:~ + 
File Paste Clipboard
2024-09-18 10:59:10 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:01:00 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:01:31 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:02:23 [1] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-18 11:02:23 [1] no tasks
2024-09-18 11:02:23 [1] created 0 nodegroup(s) in cluster "my-cluster1"
2024-09-18 11:02:23 [1] nodegroup "my-node-group" has 3 node(s)
2024-09-18 11:02:23 [1] node "ip-172-31-12-110.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-33-180.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-45-96.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] waiting for at least 2 node(s) to become ready in "my-node-group"
2024-09-18 11:02:23 [1] nodegroup "my-node-group" has 3 node(s)
2024-09-18 11:02:23 [1] node "ip-172-31-12-110.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-33-180.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] node "ip-172-31-45-96.ap-southeast-1.compute.internal" is ready
2024-09-18 11:02:23 [1] created 1 managed nodegroup(s) in cluster "my-cluster1"
2024-09-18 11:02:23 [1] checking security group configuration for all nodegroups
2024-09-18 11:02:23 [1] all nodegroups have up-to-date cloudformation templates
root@ip-172-31-37-127:~#
root@ip-172-31-37-127:~#
root@ip-172-31-37-127:~# vim aaa.yml
root@ip-172-31-37-127:~# kubectl apply -f aaa.yml
replicaset.apps/my-replicaset created
root@ip-172-31-37-127:~# kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
my-replicaset 3         3         1       7s
nginx-deployment-55f6fc556d 2         2         0       16m
root@ip-172-31-37-127:~# kubectl scale rs/my-replicaset --replicas=5
replicaset.apps/my-replicaset scaled
root@ip-172-31-37-127:~# |
```

Page 14 of 17 282 words English (India) Text Predictions: On Accessibility: Investigate 16:39 18-09-2024

Q4) a)

Creating image from Singapore region using actions > images > create image >

The screenshot shows the AWS EC2 console in the Singapore region. The left sidebar shows navigation options like Instances, Instance Types, Launch Templates, etc. The main area displays the 'Amazon Machine Images (AMIs)' page with one item listed:

Name	AMI name	AMI ID	Source
image1	image1	ami-033815b334da2da05	039612874025/image1

Below the table, the details for AMI ID 'ami-033815b334da2da05' are shown:

AMI ID	Image type	Platform details	Root device type
ami-033815b334da2da05	machine	Linux/UNIX	EBS

Other tabs include Permissions, Storage, and Tags.

Image created and it is available in amis in ssingapore region

Copy image to ohio region

The screenshot shows the 'Copy Amazon Machine Image (AMI)' dialog. It asks to create a copy of an AMI in a different region. The fields are filled as follows:

- Original AMI ID: ami-033815b334da2da05
- AMI copy name: image1
- AMI copy description: copy-from-singa
- Destination Region: US East (Ohio)
- Copy tags: Copy tags (unchecked)

Storage is full so not able to do it .

LTMindtree Favorites Folder

Services Search [Alt+S] Singapore ▾ RutikR ▾

Includes your user-defined AMI tags when copying the AMI.

Encrypt EBS snapshots of AMI copy
Encrypts all snapshots in the AMI copy with the same key.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Tag image and snapshots together
Tag the image and the snapshots with the same tag.

Tag image and snapshots separately
Tag the image and the snapshots with different tags.

No tags associated with the resource.

Add new tag You can add up to 50 more tags.

Failed to copy ami-03d08470b5677b4cd
The storage for the ami is not available in the source region.

Cancel Copy AMI

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 16:18 18-09-2024

I tried from virginia region still same error

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CopyImage:imageId=ami-0efbb62254d3b9d89

LTMindtree Favorites Folder

Services Search [Alt+S] N. Virginia ▾ RutikR ▾

Includes your user-defined AMI tags when copying the AMI.

Encrypt EBS snapshots of AMI copy
Encrypts all snapshots in the AMI copy with the same key.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Tag image and snapshots together
Tag the image and the snapshots with the same tag.

Tag image and snapshots separately
Tag the image and the snapshots with different tags.

No tags associated with the resource.

Add new tag You can add up to 50 more tags.

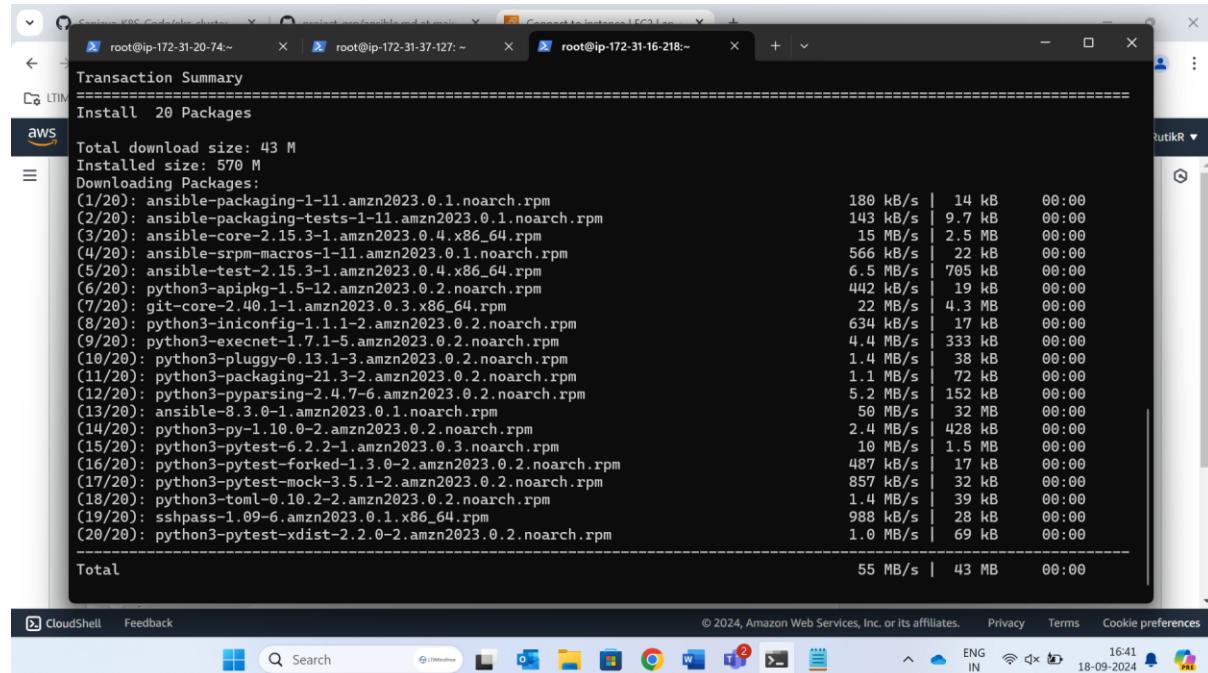
Failed to copy ami-0efbb62254d3b9d89
The storage for the ami is not available in the source region.

Diagnose with Amazon Q Cancel Copy AMI

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 16:32 18-09-2024

4)b) creating an instance to configure ansible on it.

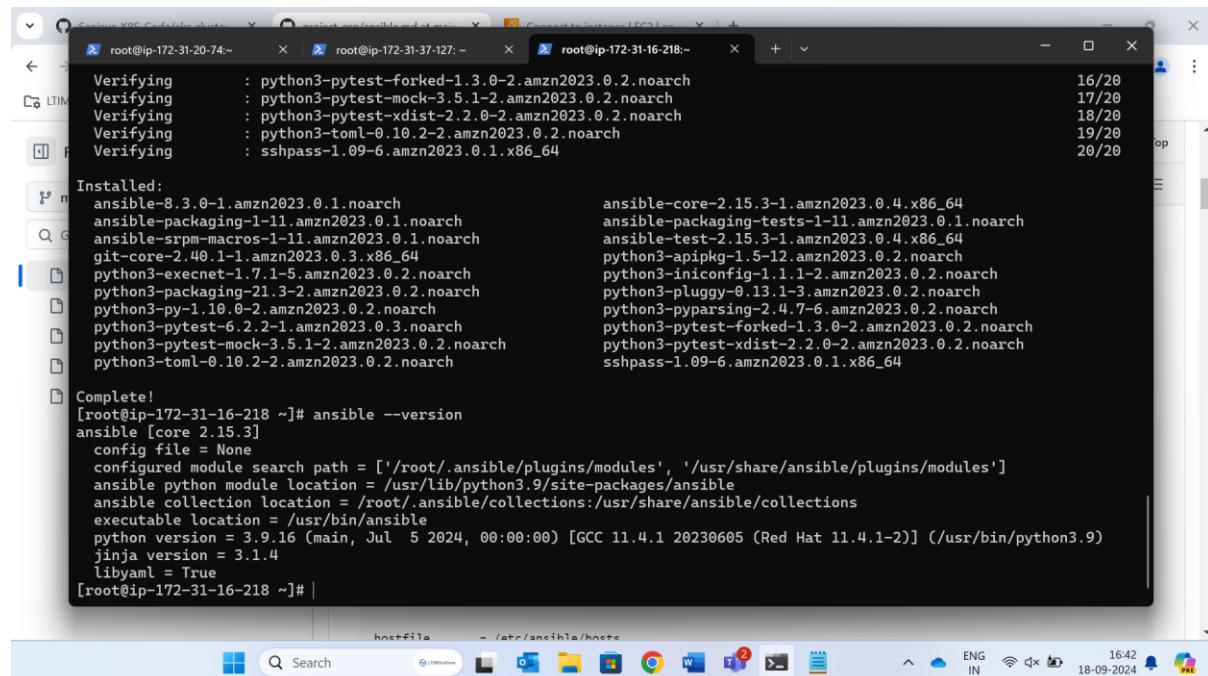
Setting ansible on an instance.



The screenshot shows a terminal window in AWS CloudShell with three tabs open. The current tab displays the output of an `aws` command followed by a list of package installations. The output shows the download and install process for 20 packages, including various Ansible components and dependencies. The total download size is 43 MB, and the total installed size is 570 M. The packages listed include `ansible`, `python3-apt`, `git`, `python3-pip`, and several Ansible modules like `ansible-test`, `ansible-core`, and `python3-ansible`.

```
aws
Transaction Summary
=====
Install 20 Packages
Total download size: 43 M
Installed size: 570 M
Downloading Packages:
(1/20): ansible-packaging-1-11.amzn2023.0.1.noarch.rpm          180 kB/s | 14 kB   00:00
(2/20): ansible-packaging-tests-1-11.amzn2023.0.1.noarch.rpm    143 kB/s | 9.7 kB   00:00
(3/20): ansible-core-2.15.3-1.amzn2023.0.4.x86_64.rpm           15 MB/s | 2.5 MB   00:00
(4/20): ansible-srpm-macros-1-11.amzn2023.0.1.noarch.rpm        566 kB/s | 22 kB   00:00
(5/20): ansible-test-2.15.3-1.amzn2023.0.4.x86_64.rpm            6.5 MB/s | 705 kB   00:00
(6/20): python3-apidkg-1.5-12.amzn2023.0.2.noarch.rpm           442 kB/s | 19 kB   00:00
(7/20): git-core-2.40.1-1.amzn2023.0.3.x86_64.rpm                22 MB/s | 4.3 MB   00:00
(8/20): python3-iniconfig-1.1.1-2.amzn2023.0.2.noarch.rpm        634 kB/s | 17 kB   00:00
(9/20): python3-execnet-1.7.1-5.amzn2023.0.2.noarch.rpm         4.4 MB/s | 333 kB   00:00
(10/20): python3-pluggy-0.13.1-3.amzn2023.0.2.noarch.rpm        1.4 MB/s | 38 kB   00:00
(11/20): python3-packaging-21.3-2.amzn2023.0.2.noarch.rpm       1.1 MB/s | 72 kB   00:00
(12/20): python3-pyparsing-2.4.7-6.amzn2023.0.2.noarch.rpm      5.2 MB/s | 152 kB   00:00
(13/20): ansible-8.3.0-1.amzn2023.0.1.noarch.rpm                50 MB/s | 32 MB   00:00
(14/20): python3-py-1.10.0-2.amzn2023.0.2.noarch.rpm             2.4 MB/s | 428 kB   00:00
(15/20): python3-pytest-6.2.2-1.amzn2023.0.3.noarch.rpm          10 MB/s | 1.5 MB   00:00
(16/20): python3-pytest-forked-1.3.0-2.amzn2023.0.2.noarch.rpm   487 kB/s | 17 kB   00:00
(17/20): python3-pytest-mock-3.5.1-2.amzn2023.0.2.noarch.rpm     857 kB/s | 32 kB   00:00
(18/20): python3-toml-0.10.2-2.amzn2023.0.2.noarch.rpm          1.4 MB/s | 39 kB   00:00
(19/20): sshpass-1.09-6.amzn2023.0.1.x86_64.rpm                 988 kB/s | 28 kB   00:00
(20/20): python3-pytest-xdist-2.2.0-2.amzn2023.0.2.noarch.rpm    1.0 MB/s | 69 kB   00:00
Total                                         55 MB/s | 43 MB   00:00
```

Successfully installed ansible



The screenshot shows the continuation of the terminal session. It displays the verification of the installed packages, followed by a summary of the installed packages and their versions. Finally, it shows the completion of the installation process and the execution of the `ansible --version` command, which outputs the version information for Ansible and its dependencies.

```
Verifying : python3-pytest-forked-1.3.0-2.amzn2023.0.2.noarch          16/20
Verifying : python3-pytest-mock-3.5.1-2.amzn2023.0.2.noarch          17/20
Verifying : python3-pytest-xdist-2.2.0-2.amzn2023.0.2.noarch          18/20
Verifying : python3-toml-0.10.2-2.amzn2023.0.2.noarch          19/20
Verifying : sshpass-1.09-6.amzn2023.0.1.x86_64                    20/20

Installed:
ansible-8.3.0-1.amzn2023.0.1.noarch
ansible-packaging-1-11.amzn2023.0.1.noarch
ansible-srpm-macros-1-11.amzn2023.0.1.noarch
git-core-2.40.1-1.amzn2023.0.3.x86_64
python3-execnet-1.7.1-5.amzn2023.0.2.noarch
python3-packaging-21.3-2.amzn2023.0.2.noarch
python3-py-1.10.0-2.amzn2023.0.2.noarch
python3-pytest-6.2.2-1.amzn2023.0.3.noarch
python3-pytest-mock-3.5.1-2.amzn2023.0.2.noarch
python3-toml-0.10.2-2.amzn2023.0.2.noarch

Complete!
[root@ip-172-31-16-218 ~]# ansible --version
ansible [core 2.15.3]
  config file = None
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3.9/site-packages/ansible
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.9.16 (main, Jul  5 2024, 00:00:00) [GCC 11.4.1 20230605 (Red Hat 11.4.1-2)] (/usr/bin/python3.9)
  jinja version = 3.1.4
  libyaml = True
[root@ip-172-31-16-218 ~]#
```

Created an hosts file.

```
Verifying : python3-toml-0.10.2-2.amzn2023.0.2.noarch  
Verifying : sshpass-1.09-6.amzn2023.0.1.x86_64  
  
Installed:  
ansible-8.3.0-1.amzn2023.0.1.noarch  
ansible-packaging-1-11.amzn2023.0.1.noarch  
ansible-srpm-macros-1-11.amzn2023.0.1.noarch  
git-core-2.40.1-1.amzn2023.0.3.x86_64  
python3-execnet-1.7.1-5.amzn2023.0.2.noarch  
python3-packaging-21.3-2.amzn2023.0.2.noarch  
python3-py-1.10.0-2.amzn2023.0.2.noarch  
python3-pytest-6.2.2-1.amzn2023.0.3.noarch  
python3-pytest-mock-3.5.1-2.amzn2023.0.2.noarch  
python3-toml-0.10.2-2.amzn2023.0.2.noarch  
  
ansible-core-2.15.3-1.amzn2023.0.4.x86_64  
ansible-packaging-tests-1-11.amzn2023.0.1.noarch  
ansible-test-2.15.3-1.amzn2023.0.4.x86_64  
python3-apipkg-1.5-12.amzn2023.0.2.noarch  
python3-iniconfig-1.1.1-2.amzn2023.0.2.noarch  
python3-pluggy-0.13.1-3.amzn2023.0.2.noarch  
python3-pyparsing-2.4.7-6.amzn2023.0.2.noarch  
python3-pytest-forked-1.3.0-2.amzn2023.0.2.noarch  
python3-pytest-xdist-2.2.0-2.amzn2023.0.2.noarch  
sshpass-1.09-6.amzn2023.0.1.x86_64  
  
Complete!  
[root@ip-172-31-16-218 ~]# ansible --version  
ansible [core 2.15.3]  
  config file = None  
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']  
  ansible python module location = /usr/lib/python3.9/site-packages/ansible  
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections  
  executable location = /usr/bin/ansible  
  python version = 3.9.16 (main, Jul 5 2024, 00:00:00) [GCC 11.4.1 20230605 (Red Hat 11.4.1-2)] (/usr/bin/python3.9)  
  jinja version = 3.1.4  
  libyaml = True  
[root@ip-172-31-16-218 ~]# cd /etc/ansible  
[root@ip-172-31-16-218 ansible]# vim ansible.cfg  
[root@ip-172-31-16-218 ansible]# vim hosts  
[root@ip-172-31-16-218 ansible]# |
```

Hosts are visible

```
Installed:  
ansible-8.3.0-1.amzn2023.0.1.noarch  
ansible-packaging-1-11.amzn2023.0.1.noarch  
ansible-srpm-macros-1-11.amzn2023.0.1.noarch  
git-core-2.40.1-1.amzn2023.0.3.x86_64  
python3-execnet-1.7.1-5.amzn2023.0.2.noarch  
python3-packaging-21.3-2.amzn2023.0.2.noarch  
python3-py-1.10.0-2.amzn2023.0.2.noarch  
python3-pytest-6.2.2-1.amzn2023.0.3.noarch  
python3-pytest-mock-3.5.1-2.amzn2023.0.2.noarch  
python3-toml-0.10.2-2.amzn2023.0.2.noarch  
  
ansible-core-2.15.3-1.amzn2023.0.4.x86_64  
ansible-packaging-tests-1-11.amzn2023.0.1.noarch  
ansible-test-2.15.3-1.amzn2023.0.4.x86_64  
python3-apipkg-1.5-12.amzn2023.0.2.noarch  
python3-iniconfig-1.1.1-2.amzn2023.0.2.noarch  
python3-pluggy-0.13.1-3.amzn2023.0.2.noarch  
python3-pyparsing-2.4.7-6.amzn2023.0.2.noarch  
python3-pytest-forked-1.3.0-2.amzn2023.0.2.noarch  
python3-pytest-xdist-2.2.0-2.amzn2023.0.2.noarch  
sshpass-1.09-6.amzn2023.0.1.x86_64  
  
Complete!  
[root@ip-172-31-16-218 ~]# ansible --version  
ansible [core 2.15.3]  
  config file = None  
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']  
  ansible python module location = /usr/lib/python3.9/site-packages/ansible  
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections  
  executable location = /usr/bin/ansible  
  python version = 3.9.16 (main, Jul 5 2024, 00:00:00) [GCC 11.4.1 20230605 (Red Hat 11.4.1-2)] (/usr/bin/python3.9)  
  jinja version = 3.1.4  
  libyaml = True  
[root@ip-172-31-16-218 ~]# cd /etc/ansible  
[root@ip-172-31-16-218 ansible]# vim ansible.cfg  
[root@ip-172-31-16-218 ansible]# vim hosts  
[root@ip-172-31-16-218 ansible]# ansible all --list-hosts  
  hosts (1):  
    172.31.16.218  
[root@ip-172-31-16-218 ansible]# |
```

Hosts are connected successfully

```
[root@ip-172-31-20-74 ~]# root@ip-172-31-37-127:~ x root@ip-172-31-16-218:~ x root@ip-172-31-16-218:~ x + - Share [File Paste Clipboard Share] -n: dry run    -- no keys are actually copied -s: use sftp    -- use sftp instead of executing remote-commands. Can be useful if the remote only allows sftp -h|-?: print this help [root@ip-172-31-16-218 ~]# ssh-copy-id root@172.31.16.218 /usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub" The authenticity of host '172.31.16.218 (172.31.16.218)' can't be established. ED25519 key fingerprint is SHA256:hj5ASr8daHxNesUhUP7fhVZ4ERw4sDHhC09SHFdFjw. This key is not known by any other names Are you sure you want to continue connecting (yes/no/[fingerprint])? yes /usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed /usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys root@172.31.16.218's password: Number of key(s) added: 1 Now try logging into the machine, with: "ssh 'root@172.31.16.218'" and check to make sure that only the key(s) you wanted were added. [root@ip-172-31-16-218 ~]# ansible all -m ping [WARNING]: Platform linux on host 172.31.16.218 is using the discovered Python interpreter at /usr/bin/python3.9, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information. 172.31.16.218 | SUCCESS => { "ansible_facts": { "discovered_interpreter_python": "/usr/bin/python3.9" }, "changed": false, "ping": "pong" } [root@ip-172-31-16-218 ~]#
```

Done the changes at host side

```
File Paste Clipboard Share >
root@ip-172-31-20-74:~| x root@ip-172-31-37-127:~| x root@ip-172-31-16-218:/etc| x root@ip-172-31-16-218:~| + | - | x
[root@ip-172-31-16-218 ansible]# cd /etc/fstab
[root@ip-172-31-16-218 ~]# vim a.yml
[root@ip-172-31-16-218 ~]# cd /etc/ansible
[root@ip-172-31-16-218 ansible]# vim a.yml
[root@ip-172-31-16-218 ansible]# vim a.yml
[root@ip-172-31-16-218 ansible]# ansible-playbook a.yml
PLAY [HTTPD and copy fstab] ****
TASK [Gathering Facts] ****
[WARNING]: Platform linux on host 172.31.16.218 is using the discovered Python interpreter at /usr/bin/python3.9, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.16.218]

TASK [Install httpd package] ****
changed: [172.31.16.218]

TASK [Start and enable httpd service] ****
changed: [172.31.16.218]

TASK [Copy fstab file to /tmp] ****
changed: [172.31.16.218]

PLAY RECAP ****
172.31.16.218 : ok=4    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
[root@ip-172-31-16-218 ansible]# |
```

Page 19 of 19 308 words English (India) Text Predictions: On Accessibility: Investigate Focus ENG IN 16:53 18-09-2024