

# Adv. Devops Experiment no. 3

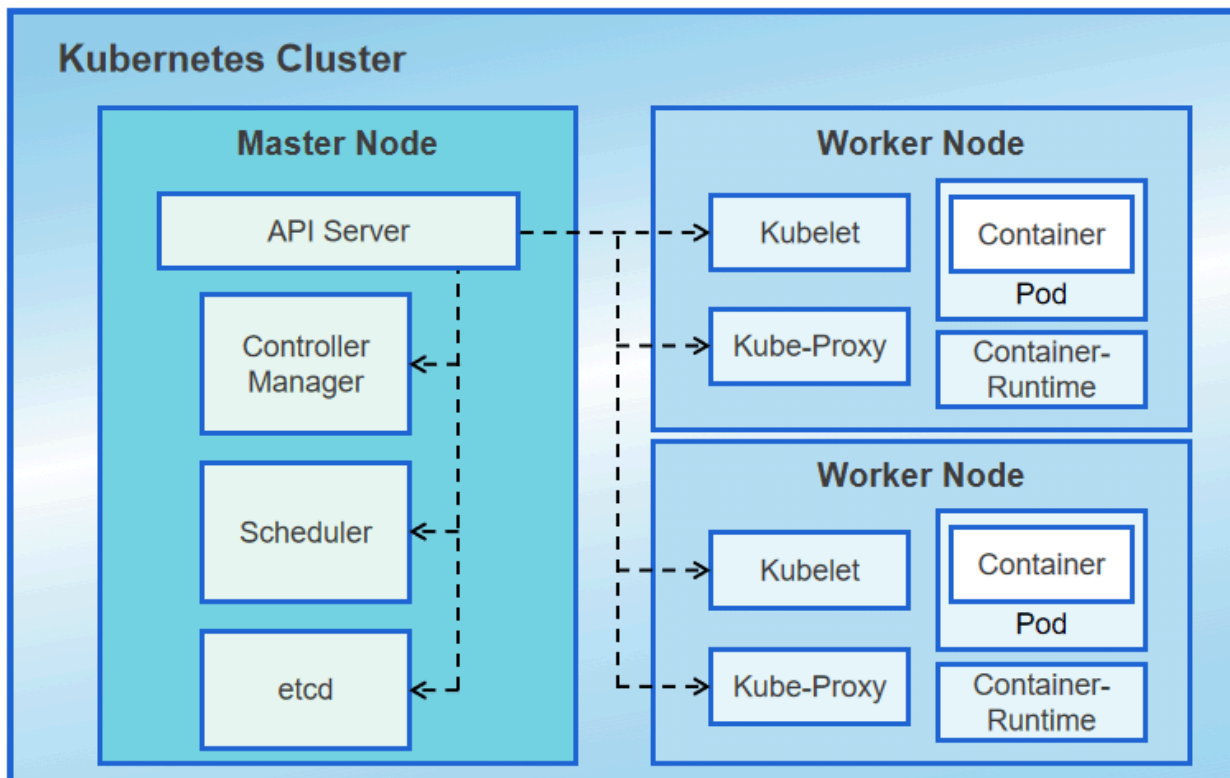
Name: Rohan Lalchandani

Class: D15A Roll no.: 25

**Aim:** To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

## Theory:

Kubernetes cluster architecture is designed to manage containerized applications efficiently. It consists of two main components: the **Control Plane** and the **Worker Nodes**.



## 1. Control Plane

The control plane is responsible for managing the overall state of the cluster. It makes global decisions about the cluster, such as scheduling applications, and detects/responds to cluster events (like a container failure).

- **API Server:** The entry point for all administrative tasks in the cluster. It exposes the Kubernetes API, acting as the front end of the control plane.
- **etcd:** A distributed key-value store that stores all the cluster's data. It is a highly reliable store for all cluster state and configuration.
- **Controller Manager:** Monitors the cluster state and performs routine tasks like handling node failures, maintaining the correct number of replicas for pods, and balancing the load across the cluster.
- **Scheduler:** Assigns newly created pods to nodes based on resource availability, affinity rules, and other policies.

## 2. Worker Nodes

Worker nodes run the containerized applications (pods). Each node has a set of components that communicate with the control plane to receive and execute tasks.

- **Kubelet:** The agent that runs on every node in the cluster. It ensures that containers are running in pods and communicates with the API server to get the desired state of the node.
- **Container Runtime:** The software responsible for running containers, such as Docker, containerd, or CRI-O. It manages the lifecycle of containers on the node.
- **Kube Proxy:** Ensures networking rules are applied to allow communication between the different components in the cluster, handling services and load balancing.

## 3. Cluster Networking

Kubernetes provides an abstracted networking layer for communication between pods. Key features include:

- **Pod-to-Pod Communication:** Every pod in a cluster can communicate with every other pod without using NAT (Network Address Translation).

- **Service Abstraction:** Services provide a stable endpoint to access a group of pods. This decouples the frontend from backend pods, offering load balancing and failure recovery.

## 4. Storage

Kubernetes uses persistent storage systems like persistent volumes (PVs) and persistent volume claims (PVCs) to manage data storage needs.

### Implementation:

1. Create three EC2 Ubuntu Instances - Master, Worker 1 and Worker 2.

The screenshot displays the AWS Management Console interface for launching an EC2 instance. The top navigation bar includes the AWS logo, 'Services', a search bar, and the user's account information. The main content area is titled 'Launch an instance' and provides a brief overview of Amazon EC2. The 'Name and tags' section has a text input field for the instance name, which is currently set to 'Master'. The 'Application and OS Images (Amazon Machine Image)' section is expanded, showing a search bar and a list of available AMIs. The 'Summary' section on the right provides a overview of the instance configuration, including the number of instances (1), the software image (AMI), the virtual server type (instance type), the firewall (security group), and the storage (volumes). The 'Launch instance' button is prominently displayed in orange.

**Launch an instance** [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags** [Info](#)

Name

Master [Add additional tags](#)

**Application and OS Images (Amazon Machine Image)** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

**Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...[read more](#)  
ami-0e86e20dae9224db8

Virtual server type (instance type)

t2.medium

Firewall (security group)

New security group

Storage (volumes)

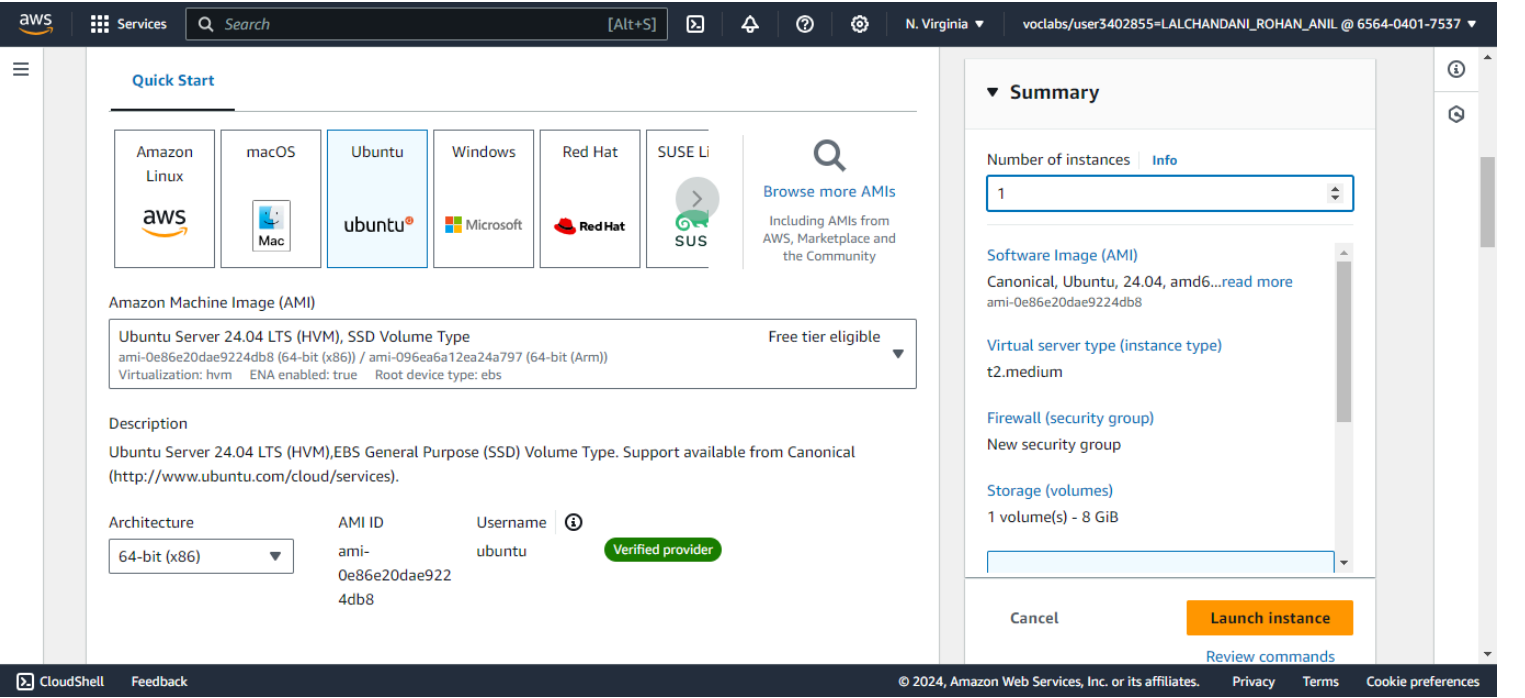
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#)

[Review commands](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

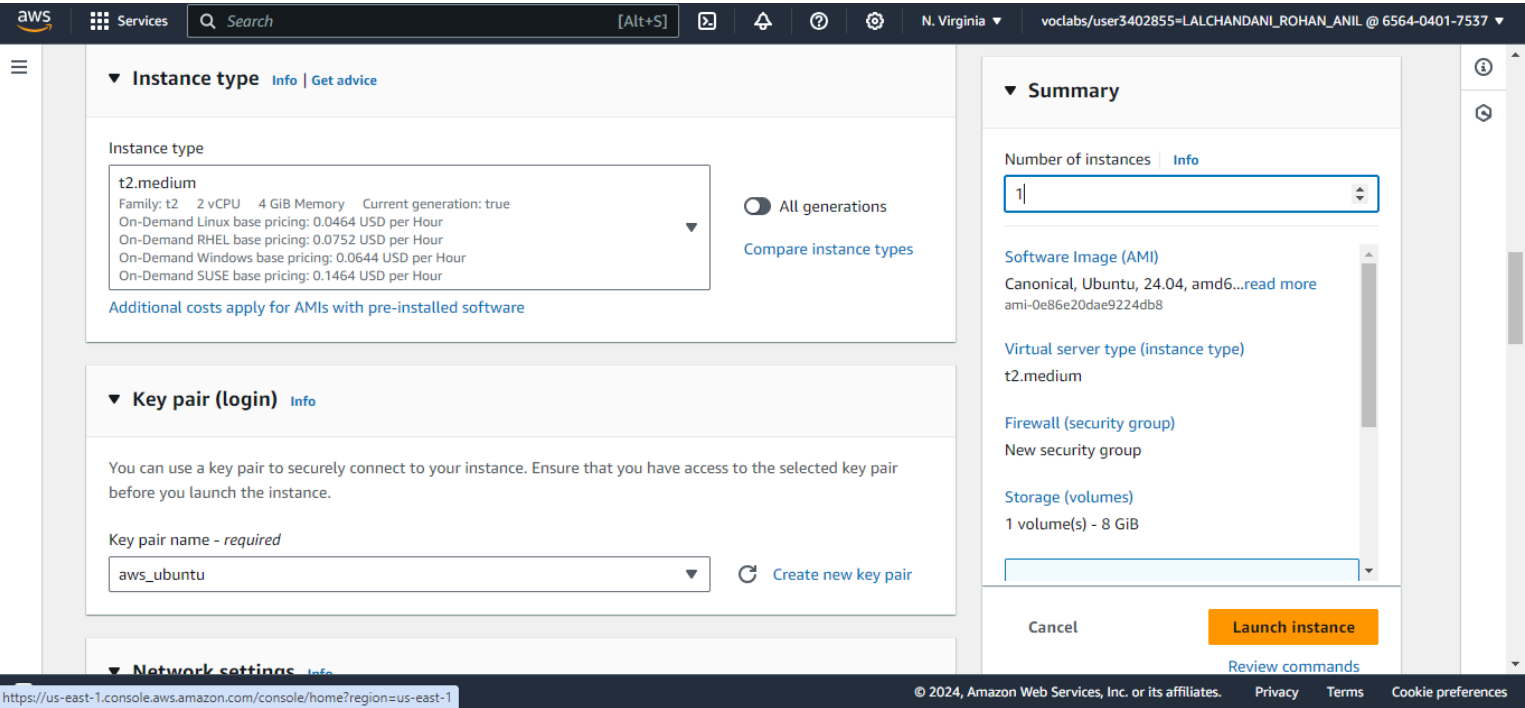
2. Select the following AMI image.



3. Select t2.medium in instance type.

Why?

**t2.medium** has 2 vCPUs and 4 GB of RAM, which allows it to run more pods and handle larger or more resource-intensive containers compared to **t2.micro** (1 vCPU, 1 GB of RAM).



#### 4. Create security group

In it type the group name and description as my-master-sg

The screenshot shows the 'Create instance' page in the AWS console, specifically the 'Network settings' tab. The 'VPC' is set to 'vpc-0c73ad858cebf5faa' (default). The 'Subnet' is set to 'No preference'. 'Auto-assign public IP' is set to 'Enable'. Under 'Firewall (security groups)', the 'Create security group' radio button is selected. The 'Security group name' is 'my-master-sg'. The 'Summary' tab on the right shows 'Number of instances' as 1, 'Software Image (AMI)' as 'Canonical, Ubuntu, 24.04, amd64...', 'Virtual server type (instance type)' as 't2.medium', and 'Storage (volumes)' as '1 volume(s) - 8 GiB'. The 'Launch instance' button is visible at the bottom right.

#### 5. Edit the inbound rules and add a new rule to accept “All traffic” as shown below.

The screenshot shows the 'Create instance' page in the AWS console, specifically the 'Network settings' tab, with the 'Firewall (security groups)' section expanded. It shows two inbound security group rules. Rule 1 is for 'ssh' (Type), 'TCP' (Protocol), '22' (Port range), 'Anywhere' (Source type), and '0.0.0.0/0' (Source). Rule 2 is for 'All traffic' (Type), 'All' (Protocol), 'All' (Port range), 'Custom' (Source type), and '0.0.0.0/0' (Source). The 'Summary' tab on the right shows the same instance configuration as before. The 'Launch instance' button is visible at the bottom right.

6. Create Worker 1 and Worker 2 also with same settings and select the existing security group of “my-master-sg” in them. Connect all the instances.

EC2 Dashboard

EC2 Global View

Events

Console-to-Code

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Instances (3) Info

Last updated less than a minute ago

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public I
<input type="checkbox"/>	Worker 1	i-04a3cf028e1290254	Running	t2.medium	Initializing	View alarms	us-east-1c	ec2-35-
<input type="checkbox"/>	Worker 2	i-0ee6249a6f40632c6	Running	t2.medium	Initializing	View alarms	us-east-1c	ec2-35-
<input type="checkbox"/>	Master	i-093cde3353624675c	Running	t2.medium	2/2 checks pass	View alarms	us-east-1c	ec2-34-

Select an instance

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

7. Now run “sudo su” to execute commands as root user.  
“cd” to change directory  
then “apt-get update”

Services

Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/\*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo\_root" for details.

ubuntu@ip-172-31-91-125:~\$ sudo su  
root@ip-172-31-91-125:/home/ubuntu# cd  
root@ip-172-31-91-125:~# apt-get updates  
E: Invalid operation updates  
root@ip-172-31-91-125:~# apt-get update  
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease  
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]  
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [377 kB]  
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]  
Get:8 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [81.4 kB]  
Get:9 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4516 B]  
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [269 kB]  
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [113 kB]

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 8. Install docker by “apt-get install docker.io -y”

```
aws Services Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537
root@ip-172-31-91-125:~# apt-get install docker.io -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-buildx docker-compose-v2 docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 139 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-1ubuntu2 [33.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dns-root-data all 2023112702-willsync1 [4450 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dnsmasq-base amd64 2.90-2build2 [375 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 docker.io amd64 24.0.7-0ubuntu4.1 [29.1 MB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 76.8 MB in 1s (81.4 MB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 67741 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.8-1_amd64.deb ...
Unpacking pigz (2.8-1) ...

i-093cde3353624675c (Master)
PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws Services Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537
Setting up runc (1.1.12-0ubuntu3.1) ...
Setting up dns-root-data (2023112702-willsync1) ...
Setting up bridge-utils (1.7.1-1ubuntu2) ...
Setting up pigz (2.8-1) ...
Setting up containerd (1.7.12-0ubuntu4.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /usr/lib/systemd/system/ubuntu-fan.service.
Setting up docker.io (24.0.7-0ubuntu4.1) ...
info: Selecting GID from range 100 to 999 ...
info: Adding group 'docker' (GID 113) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Processing triggers for dbus (1.14.10-4ubuntu4) ...
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.

i-093cde3353624675c (Master)
PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

## 9. enable, start and check status of docker if it's active.

```
aws Services [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-91-125:~# service docker restart
root@ip-172-31-91-125:~# systemctl enable docker
root@ip-172-31-91-125:~# systemctl start docker
root@ip-172-31-91-125:~# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Sun 2024-09-22 13:59:43 UTC; 1min 5s ago
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 2479 (dockerd)
      Tasks: 9
     Memory: 24.6M (peak: 24.8M)
        CPU: 222ms
    CGroup: /system.slice/docker.service
            └─2479 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 22 13:59:42 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:42.921053877Z" level=info msg="Starting up"
Sep 22 13:59:42 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:42.922042636Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved,
Sep 22 13:59:42 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:42.974206157Z" level=info msg="[graphdriver] using prior storage driver: overlay2"
Sep 22 13:59:42 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:42.974206157Z" level=info msg="Loading containers: start."
Sep 22 13:59:43 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:43.138929302Z" level=info msg="Default bridge (docker0) is assigned with an IP address 172
Sep 22 13:59:43 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:43.193826483Z" level=info msg="Loading containers: done."
Sep 22 13:59:43 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:43.213251021Z" level=info msg="Docker daemon" commit=24.0.7-0ubuntu4.1 graphdriver=overlay
Sep 22 13:59:43 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:43.213311717Z" level=info msg="Daemon has completed initialization"
Sep 22 13:59:43 ip-172-31-91-125 dockerd[2479]: time="2024-09-22T13:59:43.241077478Z" level=info msg="API listen on /run/docker.sock"
Sep 22 13:59:43 ip-172-31-91-125 systemd[1]: Started docker.service - Docker Application Container Engine.

i-093cde3353624675c (Master)
PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

10. Now we have to install kubernetes, for that we will refer the documentation at <https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/>

Got to the “Debian-based distributions” and follow the instructions below to install kubeadm

```
sudo apt-get install -y apt-transport-https ca-certificates curl gpg
sudo mkdir -p -m 755 /etc/apt/keyrings
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.31/deb/Release.key | sudo gpg
--dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]
https://pkgs.k8s.io/core:/stable:/v1.31/deb/ ' | sudo tee
/etc/apt/sources.list.d/kubernetes.list
sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
sudo systemctl enable --now kubelet
systemctl restart kubelet
```



aws

Services

Q Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
root@ip-172-31-91-125:~# sudo apt-get install -y apt-transport-https ca-certificates curl gpg
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
gpg is already the newest version (2.4.4-2ubuntu17).
gpg set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3t64-gnutls libcurl4t64
3 upgraded, 1 newly installed, 0 to remove and 136 not upgraded.
Need to get 904 kB of archives.
After this operation, 38.9 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.4 [227 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4t64 amd64 8.5.0-2ubuntu10.4 [341 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl3t64-gnutls amd64 8.5.0-2ubuntu10.4 [333 kB]
Fetched 904 kB in 0s (24.7 MB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 68108 files and directories currently installed.)
Preparing to unpack .../apt-transport-https 2.7.14build2_all.deb ...
Unpacking apt-transport-https (2.7.14build2) ...
Preparing to unpack .../curl 8.5.0-2ubuntu10.4 amd64.deb ...
Unpacking curl (8.5.0-2ubuntu10.4) over (8.5.0-2ubuntu10.1) ...
Preparing to unpack .../libcurl4t64 8.5.0-2ubuntu10.4 amd64.deb ...
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Q Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
Preparing to unpack .../libcurl4t64 8.5.0-2ubuntu10.4 amd64.deb ...
Unpacking libcurl4t64:amd64 (8.5.0-2ubuntu10.4) over (8.5.0-2ubuntu10.1) ...
Preparing to unpack .../libcurl3t64-gnutls 8.5.0-2ubuntu10.4 amd64.deb ...
Unpacking libcurl3t64-gnutls:amd64 (8.5.0-2ubuntu10.4) over (8.5.0-2ubuntu10.1) ...
Setting up apt-transport-https (2.7.14build2) ...
Setting up libcurl4t64:amd64 (8.5.0-2ubuntu10.4) ...
Setting up libcurl3t64-gnutls:amd64 (8.5.0-2ubuntu10.4) ...
Setting up curl (8.5.0-2ubuntu10.4) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.2) ...
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-172-31-91-125:~# sudo mkdir -p -m 755 /etc/apt/keyrings
root@ip-172-31-91-125:~# curl -fsSL https://pkgs.k8s.io/core:stable/v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
root@ip-172-31-91-125:~# echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:stable/v1.31/deb/ ' | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:stable/v1.31/deb/ /
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.31/deb/ /
root@ip-172-31-91-125:~# sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb InRelease [1186 B]
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb Packages [4865 B]
Fetched 132 kB in 0s (288 kB/s)
Reading package lists... Done
root@ip-172-31-91-125:~# sudo apt-get install -y kubelet kubeadm kubectl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  conntrack cri-tools kubernetes-cni
The following NEW packages will be installed:
  conntrack cri-tools kubeadm kubectl kubelet kubernetes-cni
0 upgraded, 6 newly installed, 0 to remove and 136 not upgraded.
Need to get 87.4 MB of archives.
After this operation, 314 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 conntrack amd64 1:1.4.8-1ubuntu1 [37.9 kB]
Get:2 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb cri-tools 1.31.1-1.1 [15.7 MB]
Get:3 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb kubeadm 1.31.1-1.1 [11.4 MB]
Get:4 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb kubectl 1.31.1-1.1 [11.2 MB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb kubernetes-cni 1.5.1-1.1 [33.9 MB]
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.31/deb kubelet 1.31.1-1.1 [15.2 MB]
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
Selecting previously unselected package conntrack.
(Reading database ... 68112 files and directories currently installed.)
Preparing to unpack .../0-conntrack_1:1.4.8-1ubuntu1_amd64.deb ...
Unpacking conntrack (1:1.4.8-1ubuntu1) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.31.1-1.1_amd64.deb ...
Unpacking cri-tools (1.31.1-1.1) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../2-kubeadm_1.31.1-1.1_amd64.deb ...
Unpacking kubeadm (1.31.1-1.1) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../3-kubectl_1.31.1-1.1_amd64.deb ...
Unpacking kubectl (1.31.1-1.1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../4-kubernetes-cni_1.5.1-1.1_amd64.deb ...
Unpacking kubernetes-cni (1.5.1-1.1) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Setting up conntrack (1:1.4.8-1ubuntu1) ...
Setting up kubectl (1.31.1-1.1) ...
Setting up cri-tools (1.31.1-1.1) ...
Setting up kubernetes-cni (1.5.1-1.1) ...
Setting up kubeadm (1.31.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
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-172-31-91-125:~# sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
root@ip-172-31-91-125:~# sudo systemctl enable --now kubelet
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Now only in **Master Node**, execute the “kubeadm init” to initialize the cluster.

```
aws Services Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.91.125:6443 --token uue2q7.i3wahpw8072hh8ms \
--discovery-token-ca-cert-hash sha256:64b48d7aa808c0dafaeb2879e1ae9a55e5a58504c9b2dc0871c4af03910b494
root@ip-172-31-91-125:~# sudo apt-get install -y socat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  socat
0 upgraded, 1 newly installed, 0 to remove and 136 not upgraded.

i-093cde3353624675c (Master)
PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125
```

11. Now copy the 3 commands from the regular user and 1 command from root user and one by one execute them.

Execute the same set of steps from 1 to 11 in Worker 1 and Worker 2 except the kubeadm init command.

```
aws Services Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537

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-172-31-91-125:~# mkdir -p $HOME/.kube
root@ip-172-31-91-125:~# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
root@ip-172-31-91-125:~# sudo chown $(id -u):$(id -g) $HOME/.kube/config
root@ip-172-31-91-125:~# export KUBECONFIG=/etc/kubernetes/admin.conf
root@ip-172-31-91-125:~# systemctl restart kubelet
```

12. Execute “kubectl get nodes” in master to get a list of nodes present. Initially there will be no nodes present.

```
aws Services Q Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537
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-172-31-91-125:~# mkdir -p $HOME/.kube
root@ip-172-31-91-125:~# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
root@ip-172-31-91-125:~# sudo chown $(id -u):$(id -g) $HOME/.kube/config
root@ip-172-31-91-125:~# export KUBECONFIG=/etc/kubernetes/admin.conf
root@ip-172-31-91-125:~# systemctl restart kubelet
root@ip-172-31-91-125:~# kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
ip-172-31-91-125    NotReady  control-plane  4m8s  v1.31.1
```

Copy the join command from Master and execute it in Worker 1 and Worker 2 to join them to the cluster.

```
aws Services Q Search [Alt+S] N. Virginia voclabs/user3402855=LALCHANDANI_ROHAN_ANIL @ 6564-0401-7537
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-172-31-84-210:~#
kubeadm join 172.31.91.125:6443 --token uue2q7.i3wahpw8072hh8ms \
--discovery-token-ca-cert-hash sha256:64b48d7aa808c0dafaeb2879e1ae9a55e5a58504c9b2dc0871c4af03910b494
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-check] Waiting for a healthy kubelet at http://127.0.0.1:10248/healthz. This can take up to 4m0s
[kubelet-check] The kubelet is healthy after 500.852828ms
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap

This node has joined the cluster:
* Certificate signing request was sent to apiservert and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

root@ip-172-31-84-210:~#
```

i-04a3cf028e1290254 (Worker 1)

PublicIPs: 35.174.111.130 PrivateIPs: 172.31.84.210

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Q Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.

kubelet set on hold.

kubeadm set on hold.

kubectl set on hold.

root@ip-172-31-92-94:~#

kubeadm join 172.31.91.125:6443 --token uue2q7.i3wahpw8072hh8ms \

--discovery-token-ca-cert-hash sha256:64b48d7aa808c0dafebea2879e1ae9a55e5a58504c9b2dc0871c4af03910b494

[preflight] Running pre-flight checks

[preflight] Reading configuration from the cluster...

[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'

[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"

[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"

[kubelet-start] Starting the kubelet

[kubelet-check] Waiting for a healthy kubelet at http://127.0.0.1:10248/healthz. This can take up to 4m0s

[kubelet-check] The kubelet is healthy after 501.859381ms

[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap

This node has joined the cluster:

\* Certificate signing request was sent to apiserver and a response was received.

\* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

root@ip-172-31-92-94:~#

i-Oee6249a6f40632c6 (Worker 2)

PublicIPs: 35.171.163.251 PrivateIPs: 172.31.92.94

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Again execute kubectl get nodes to see if they have joined. You can see the status of nodes to be Not Ready, we have to make them Ready.

aws

Services

Q Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

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-172-31-91-125:~# mkdir -p \$HOME/.kube

root@ip-172-31-91-125:~# sudo cp -i /etc/kubernetes/admin.conf \$HOME/.kube/config

root@ip-172-31-91-125:~# sudo chown \$(id -u):\$(id -g) \$HOME/.kube/config

root@ip-172-31-91-125:~# export KUBECONFIG=/etc/kubernetes/admin.conf

root@ip-172-31-91-125:~# systemctl restart kubelet

root@ip-172-31-91-125:~# kubectl get nodes

NAME STATUS ROLES AGE VERSION

ip-172-31-91-125 NotReady control-plane 4m8s v1.31.1

root@ip-172-31-91-125:~# kubectl get nodes

NAME STATUS ROLES AGE VERSION

ip-172-31-84-210 NotReady <none> 21s v1.31.1

ip-172-31-91-125 NotReady control-plane 20m v1.31.1

ip-172-31-92-94 NotReady <none> 13s v1.31.1

root@ip-172-31-91-125:~#

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

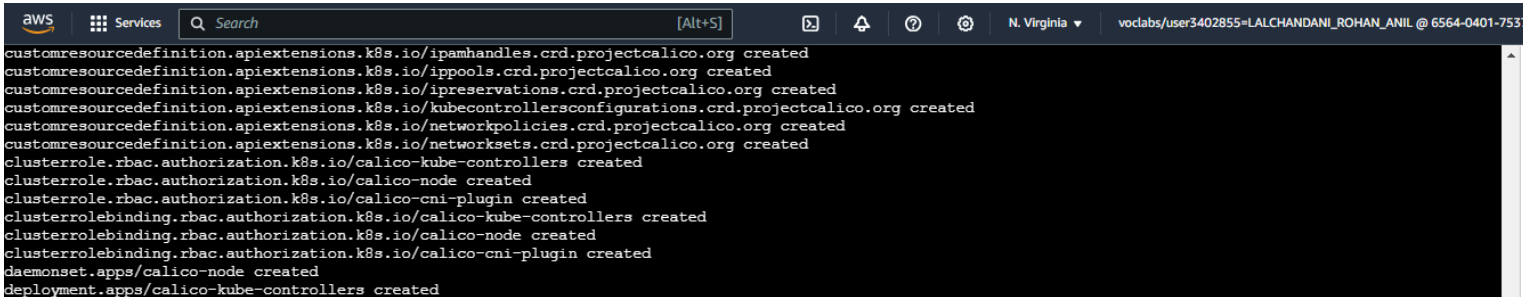
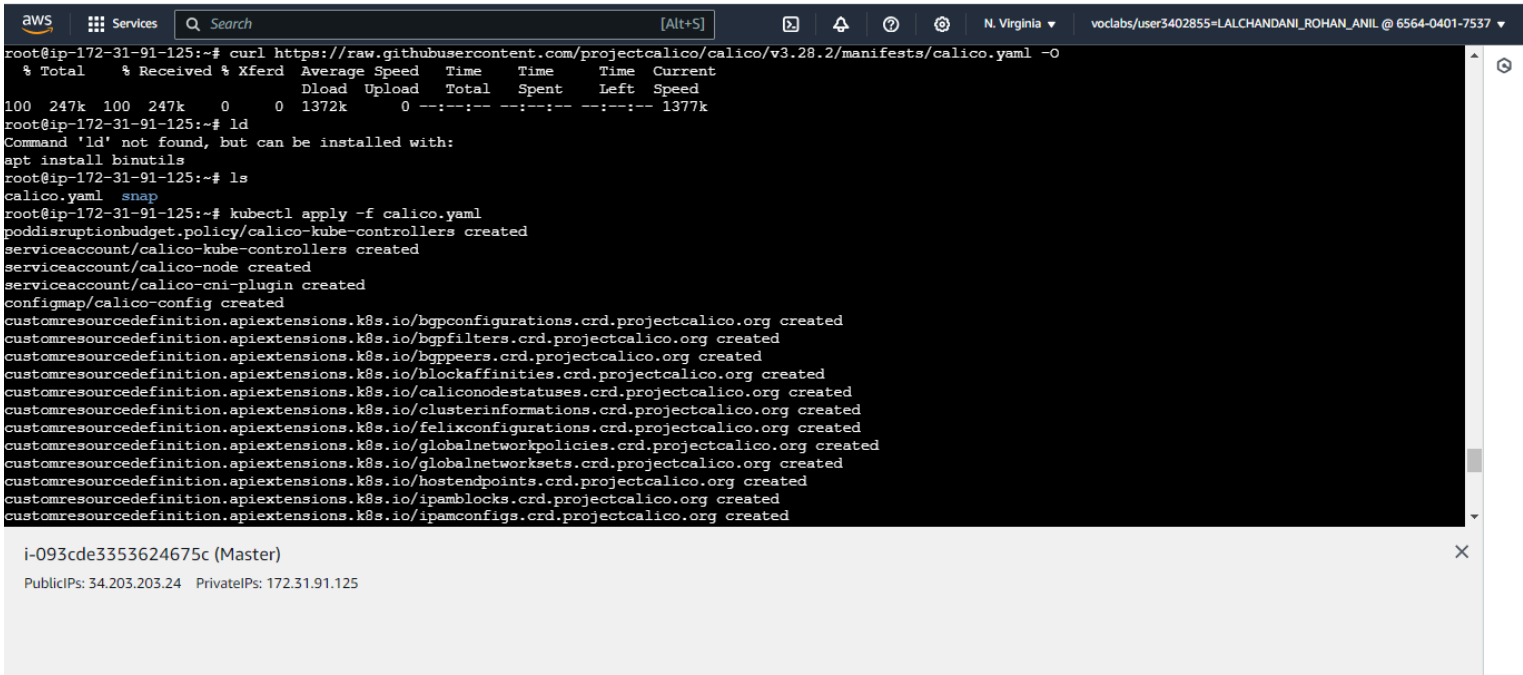
CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Now, we have to manifest “calico.yaml” file to make the status ready for that execute:

```
curl https://raw.githubusercontent.com/projectcalico/calico/v3.28.2/manifests/calico.yaml -O
ls
kubectl apply -f calico.yaml
```



Now the status becomes ready.

**EXTRA:** Renaming the role names:

```
kubectl label nodes <node ip address> node-role.kubernetes.io/<nodename>=<nodename>
```

Replace node ip address with your worker 1 or 2 ip address.

Type the name of the node in nodename.

aws

Services

Search

[Alt+S]

N. Virginia

voclabs/user3402855=LALCHANDANI\_ROHAN\_ANIL @ 6564-0401-7537

```
root@ip-172-31-91-125:~# kubectl get nodes
NAME                                STATUS    ROLES    AGE   VERSION
ip-172-31-84-210                    Ready    <none>   29m   v1.31.1
ip-172-31-91-125                    Ready    control-plane 49m   v1.31.1
ip-172-31-92-94                    Ready    <none>   29m   v1.31.1
root@ip-172-31-91-125:~# kubectl label nodes ip-172-31-84-210 node-role.kubernetes.io/worker1=Worker1
node/ip-172-31-84-210 labeled
root@ip-172-31-91-125:~# kubectl label nodes ip-172-31-92-94 node-role.kubernetes.io/worker2=Worker2
node/ip-172-31-92-94 labeled
root@ip-172-31-91-125:~# kubectl get nodes
NAME                                STATUS    ROLES    AGE   VERSION
ip-172-31-84-210                    Ready    worker1   29m   v1.31.1
ip-172-31-91-125                    Ready    control-plane 49m   v1.31.1
ip-172-31-92-94                    Ready    worker2   29m   v1.31.1
root@ip-172-31-91-125:~#
```

i-093cde3353624675c (Master)

PublicIPs: 34.203.203.24 PrivateIPs: 172.31.91.125

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences