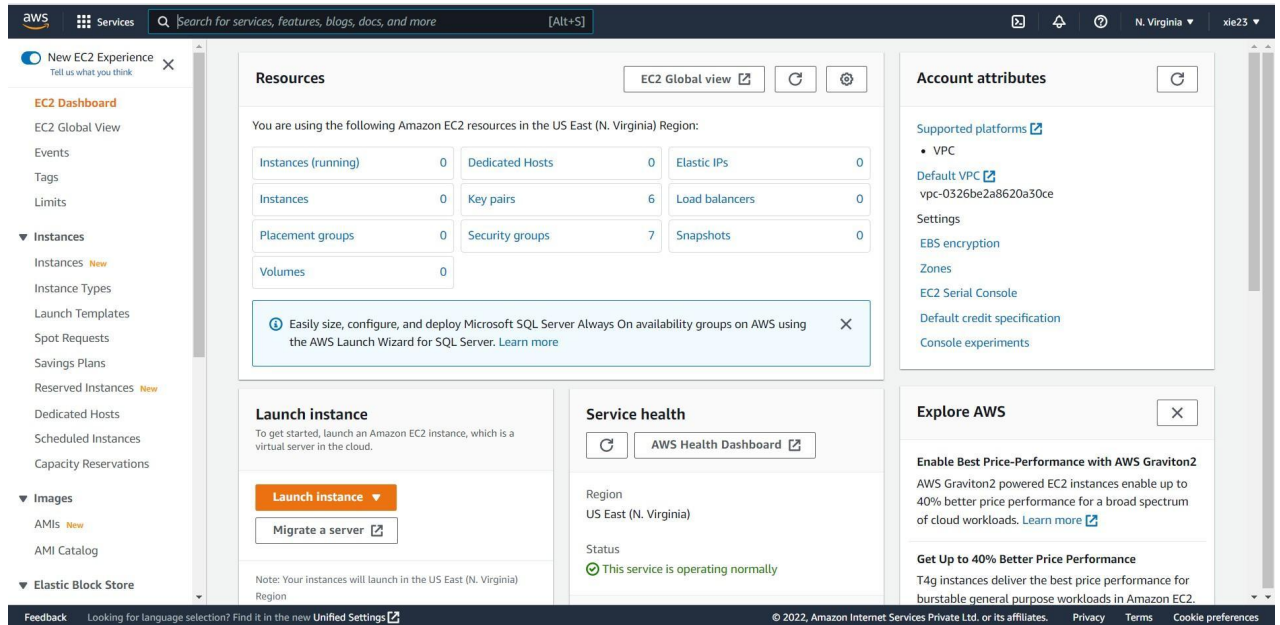


# Experiment No.

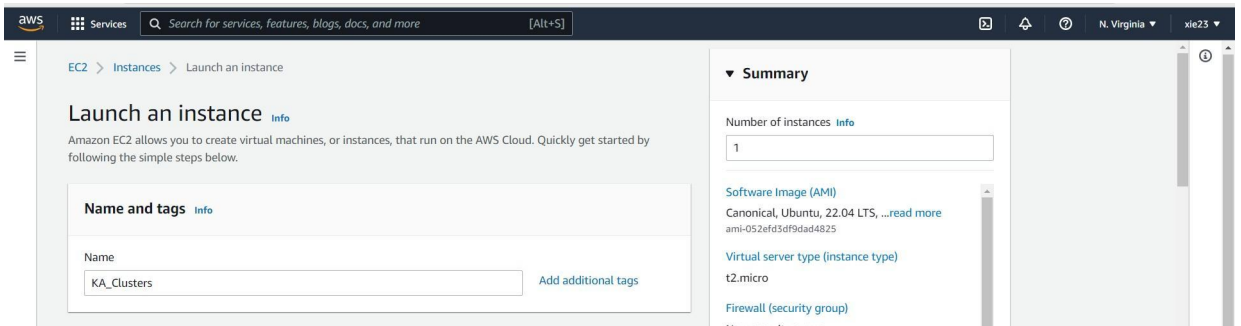
## Steps & Screenshots:

### Step1: Create a new Instance

1. Goto EC2 console and click on Launch Instance



2. Give the name to your instance



3. Choose Ubuntu OS

▼ 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

Q Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

S

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

Free tier eligible

ami-052efd3df9dad4825 (64-bit (x86)) / ami-070650c005cce4203 (64-bit (Arm))

Virtualization: hvm    ENA enabled: true    Root device type: ebs

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2022-06-09

Architecture

AMI ID

64-bit (x86)

ami-052efd3df9dad4825

Verified provider

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...[read more](#)

ami-052efd3df9dad4825

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 3 million IOPS, 1 TB of snapshots, and

Cancel

Launch instance

#### 4. Click on Create new key pair

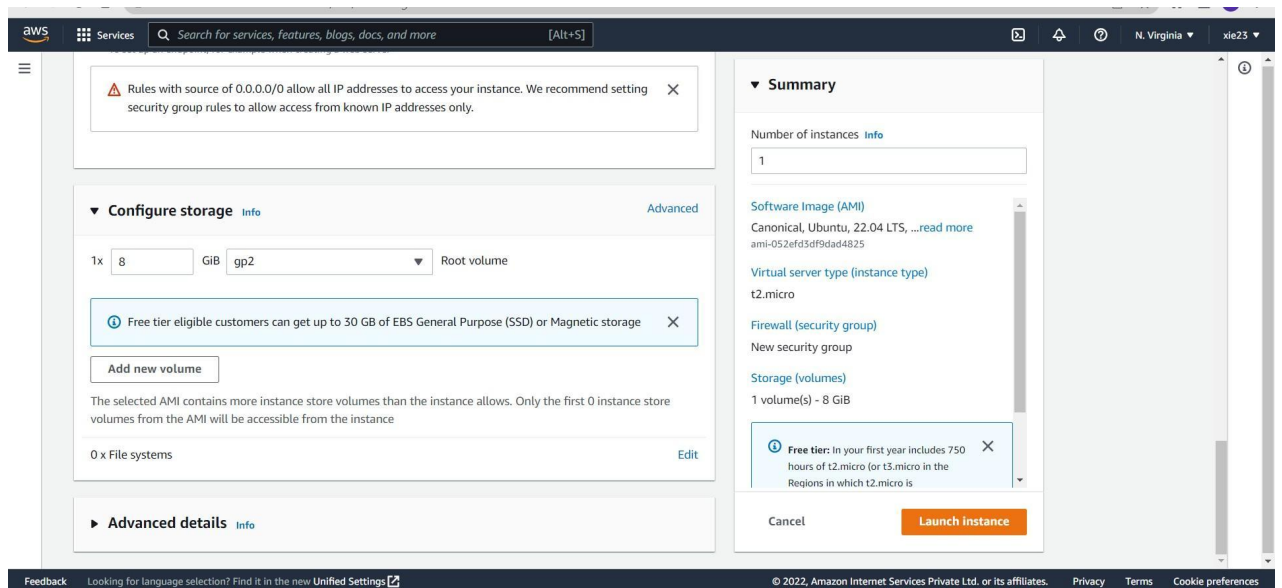
The screenshot shows the AWS Management Console interface for creating a new instance. The 'Instance type' is set to t2.micro. The 'Key pair (login)' section shows a dropdown for 'Key pair name - required' with a 'Create new key pair' button. The 'Network settings' section shows 'vpc-0326be2a8620a30ce' and 'No preference' for the subnet. The 'Summary' section shows 'Number of instances' as 1, 'Software Image (AMI)' as Canonical, Ubuntu, 22.04 LTS, 'Virtual server type (instance type)' as t2.micro, 'Firewall (security group)' as New security group, and 'Storage (volumes)' as 1 volume(s) - 8 GiB. A 'Free tier' notification is visible. The 'Launch instance' button is highlighted.

#### 5. Give the Key pair name and select .ppk file format and then click on Create key pair

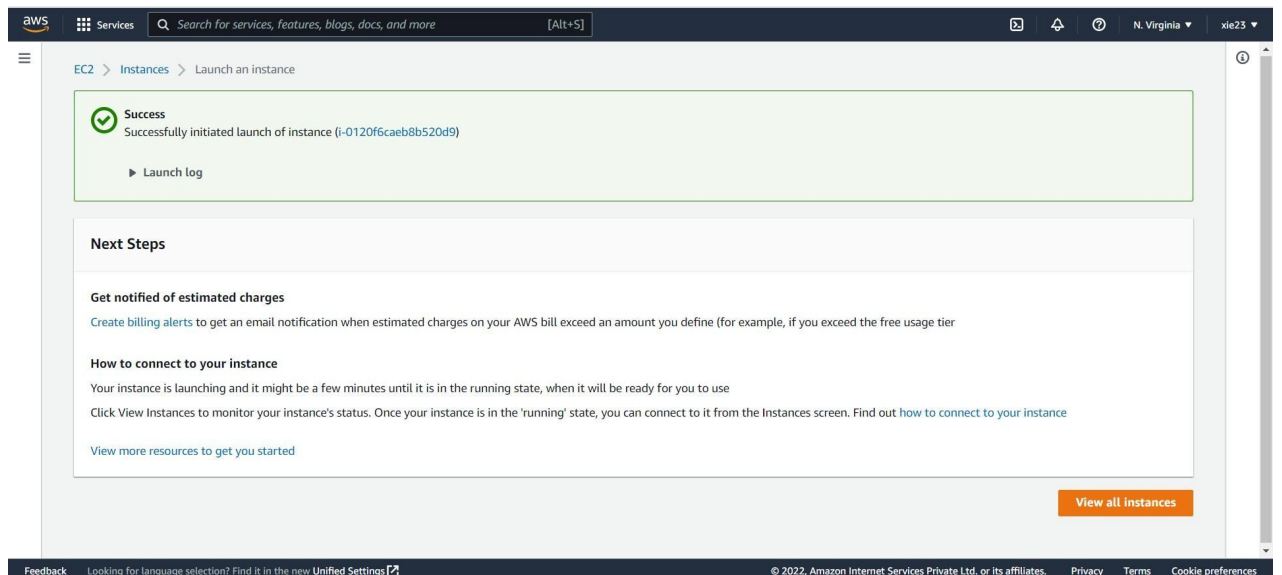
The 'Create key pair' dialog box is shown. It contains the following fields and options:

- Key pair name:** A text input field containing 'new 1'.
- Key pair type:** Radio buttons for 'RSA' (selected) and 'ED25519'.
- Private key file format:** Radio buttons for '.pem' and '.ppk' (selected).
- Buttons:** 'Cancel' and 'Create key pair'.

## 6. Click on Launch Instance

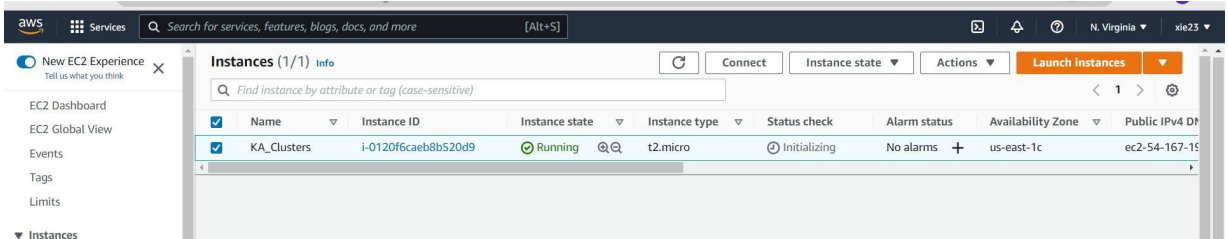


## 7. Successfully instance is launch



## Step 2: Connection

### 1. Goto Instances and select the newly created instance and then click on connect



## 2. Click on Connect

The screenshot shows the AWS Management Console interface. At the top, there's a dark blue header with the 'Services' menu and a search bar. Below the header, a breadcrumb trail reads 'EC2 > Instances > i-0120f6caeb8b520d9 > Connect to instance'. The main content area is titled 'Connect to instance' with an 'Info' link. It instructs the user to connect to instance i-0120f6caeb8b520d9 (KA\_Clusters) using any of the options: 'EC2 Instance Connect' (selected), 'Session Manager', 'SSH client', or 'EC2 serial console'. Below these tabs, the 'Instance ID' is shown as i-0120f6caeb8b520d9 (KA\_Clusters), the 'Public IP address' as 54.167.19.178, and the 'User name' as 'ubuntu' in a text input field. A note states: 'Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.' At the bottom right, there are 'Cancel' and 'Connect' buttons.

## 3. Establishing Connection

This screenshot shows the AWS Management Console during the connection establishment phase. The top header is identical to the previous screenshot. The main content area is a large black rectangle with a white circular progress indicator in the center and the text 'Establishing Connection ...' below it. Below this area, the instance details are listed: 'i-0120f6caeb8b520d9 (KA\_Clusters)' and 'Public IPs: 54.167.19.178 Private IPs: 172.31.93.67'. The footer contains a 'Feedback' link, a language selection prompt, the copyright notice '© 2022, Amazon Internet Services Private Ltd. or its affiliates.', and links for 'Privacy', 'Terms', and 'Cookie preferences'.

```
aws Services Q Search for services, features, blogs, docs, and more [Alt+S]

System information as of Sun Sep 11 12:46:08 UTC 2022

System load: 0.0830078125      Processes:          99
Usage of /: 19.1% of 7.58GB    Users logged in:   0
Memory usage: 21%             IPv4 address for eth0: 172.31.93.67
Swap usage: 0%

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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-172-31-93-67:~$

i-0120f6caeb8b520d9 (KA_Clusters)
PublicIPs: 54.167.19.178 PrivateIPs: 172.31.93.67
```

**Step 3:** Run the command “sudo su -” to goto root

```
ubuntu@ip-172-31-93-67:~$ sudo su -
root@ip-172-31-93-67:~#
```

**Step 4:** Install all the updates

```
root@ip-172-31-93-67:~# sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [544 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [129 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [8168 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [306 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [47.5 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [524 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [254 kB]
```



## Step 5: Install Docker and check its version

```
root@ip-172-31-93-67:~# apt install docker.io
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-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 78 not upgraded.
Need to get 65.6 MB of archives.
After this operation, 283 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-1ubuntu3 [34.4 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 runc amd64 1.1.0-0ubuntu1 [4087 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 containerd amd64 1.5.9-0ubuntu3 [27.0 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 dns-root-data all 2021011101 [5256 B]
```

```
root@ip-172-31-93-67:~# docker --version
Docker version 20.10.12, build 20.10.12-0ubuntu4
root@ip-172-31-93-67:~#
```

## Step 6: Enable Docker and then check docker status

```
root@ip-172-31-93-67:~# sudo systemctl enable docker
root@ip-172-31-93-67:~#
```

```
root@ip-172-31-93-67:~# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-09-11 12:53:49 UTC; 3min 37s ago
 TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 2549 (dockerd)
      Tasks: 7
     Memory: 38.0M
        CPU: 269ms
    CGroup: /system.slice/docker.service
            └─2549 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.356235070Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.356399659Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.356555442Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.426023858Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.705280711Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.799925454Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.880040777Z" level=
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.880496977Z" level=
Sep 11 12:53:49 ip-172-31-93-67 systemd[1]: Started Docker Application Container Engine.
Sep 11 12:53:49 ip-172-31-93-67 dockerd[2549]: time="2022-09-11T12:53:49.919670674Z" level=
lines 1-22/22 (END)
```

## Step 7 : Curl

```
root@ip-172-31-93-67:~# curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
root@ip-172-31-93-67:~#
```

## Step 8: Add repository

```
root@ip-172-31-93-67:~# sudo apt-add-repository "deb http://apt.kubernetes.io/ kubernetes-xenial main"
Repository: 'deb http://apt.kubernetes.io/ kubernetes-xenial main'
Description:
Archive for codename: kubernetes-xenial components: main
More info: http://apt.kubernetes.io
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Adding deb entry to /etc/apt/sources.list.d/archive_uri-http_apt_kubernetes_io-jammy.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/archive_uri-http_apt_kubernetes_io-jammy.list
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:4 https://packages.cloud.google.com/apt kubernetes-xenial InRelease [9383 B]
Get:6 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 Packages [58.4 kB]
Fetched 282 kB in 1s (519 kB/s)
Reading package lists... Done
W: http://apt.kubernetes.io/dists/kubernetes-xenial/InRelease: Key is stored in legacy trusted.gpg keyring
for details.
root@ip-172-31-93-67:~#
```

## Step 9: Install Kubeadm

```
root@ip-172-31-93-67:~# sudo apt-get install kubeadm kubelet kubect1
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  conntrack cri-tools ebtables kubernetes-cni socat
The following NEW packages will be installed:
  conntrack cri-tools ebtables kubeadm kubect1 kubelet kubernetes-cni socat
0 upgraded, 8 newly installed, 0 to remove and 78 not upgraded.
Need to get 75.9 MB of archives.
After this operation, 310 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 conntrack amd64 1:1.4.6-2build2 [33.5 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 ebtables amd64 2.0.11-4build2 [84.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 socat amd64 1.7.4.1-3ubuntu4 [349 kB]
Get:4 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 cri-tools amd64 1.24.2-00 [12.3 MB]
Get:5 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubernetes-cni amd64 0.8.7-00 [25.0 MB]
Get:6 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubelet amd64 1.25.0-00 [19.5 MB]
Get:7 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubect1 amd64 1.25.0-00 [9500 kB]
Get:8 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubeadm amd64 1.25.0-00 [9213 kB]
Fetched 75.9 MB in 2s (46.8 MB/s)
Selecting previously unselected package conntrack.
(Reading database ... 63966 files and directories currently installed.)
```

## Step 10: Set on hold

```
root@ip-172-31-93-67:~# sudo apt-mark hold kubeadm kubelet kubect1
kubeadm set on hold.
kubelet set on hold.
kubect1 set on hold.
root@ip-172-31-93-67:~#
```



### Step 11: Check kubeadm version

```
root@ip-172-31-93-67:~# kubeadm version
kubeadm version: &version.Info{Major:"1", Minor:"25", GitVersion:"v1.25.0", GitCommit:"
a866cbe2e5bbaa01cfd5e969aa3e033f3282a8a2", GitTreeState:"clean", BuildDate:"2022-08-23T
17:43:25Z", GoVersion:"go1.19", Compiler:"gc", Platform:"linux/amd64"}
root@ip-172-31-93-67:~#
```