

Step 1. Create a EC2 instance and configured Jenkins and install docker

- Create EC2 using ubuntu

The first screenshot shows the 'Launch an instance' page. The 'Name' field is set to 'cicd-project-ec2'. The 'Application and OS Images (Amazon Machine Image)' section shows the 'Ubuntu' AMI selected. The 'Summary' panel on the right shows the configuration: 1 instance, Amazon Linux 2023 AMI 2023.5.2, t2.micro instance type, new security group, and 1 volume of 8 GiB. The 'Launch instance' button is visible.

The second screenshot shows the 'Amazon Machine Image (AMI)' selection screen. The 'Ubuntu' AMI is selected. The 'Summary' panel on the right shows the configuration: 1 instance, Canonical, Ubuntu, 24.04 LTS, ami-04a81a99f5ec58529, t2.micro instance type, new security group, and 1 volume of 8 GiB. The 'Launch instance' button is visible.

The third screenshot shows the 'Instances (1)' page. The instance 'cicd-project-ec2' is listed with the status 'Running'. The 'Instance state' column shows a green checkmark and the word 'Running'. The 'Instance ID' is 'i-00d43417509aa02d2'. The 'Instance type' is 't2.micro'. The 'Status check' shows '2/2 checks passed'. The 'Availability' is 'us-east-1c'.

- Update the package and Install Java 17 using below commands:

sudo apt update

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162: /home/ubuntu# sudo apt update
```

sudo apt install fontconfig openjdk-17-jre

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# sudo apt install fontconfig openjdk-17-jre
```

Now you can check the Java Version using command

java -version

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu# java -version
openjdk version "17.0.11" 2024-04-16
OpenJDK Runtime Environment (build 17.0.11+9-Ubuntu-1)
OpenJDK 64-Bit Server VM (build 17.0.11+9-Ubuntu-1, mixed mode, sharing)
root@ip-172-31-19-162:/home/ubuntu#
```

- Installing Jenkins using this link and below commands:

<https://www.jenkins.io/doc/book/installing/linux/#debianubuntu>

sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
<https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key>

echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
<https://pkg.jenkins.io/debian-stable> binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu# sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
--2024-07-07 12:46:13-- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.38.133, 2a04:4e42:78::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|146.75.38.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings 100%[=====>] 3.10K --.-KB/s in 0s

2024-07-07 12:46:13 (46.4 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]

root@ip-172-31-19-162:/home/ubuntu# echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu#
```

sudo apt-get update

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu# sudo apt-get update
```

sudo apt-get install Jenkins -y

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# sudo apt-get install jenkins -y
```

sudo systemctl enable jenkins

BY: - CLOUD TECH BURNER(CONNECT TO YOUTUBE CHANNEL FOR MORE)

sudo systemctl start jenkins

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# sudo systemctl start jenkins
root@ip-172-31-19-162:/home/ubuntu# sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/sy
stemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
root@ip-172-31-19-162:/home/ubuntu#
```

- Install Docker using below commands:-

Add Docker's official GPG key:

1. sudo apt-get update
2. sudo apt-get install ca-certificates curl
3. sudo install -m 0755 -d /etc/apt/keyrings
4. sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
5. sudo chmod a+r /etc/apt/keyrings/docker.asc

Add the repository to Apt sources:

1. echo \

```
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
```

```
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
```

```
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

2. sudo apt-get update
3. sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

sudo usermod -aG docker \${USER}

- Add Jenkins to the docker group

sudo usermod -aG docker jenkins

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# sudo apt install apt-transport-https ca-cert
ificates curl software-properties-common -y
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# curl -fsSL https://download.docker.com/linux
/ubuntu/gpg | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (s
ee apt-key(8)).
OK
root@ip-172-31-19-162:/home/ubuntu#
```

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu#
root@ip-172-31-19-162:/home/ubuntu# sudo add-apt-repository "deb [arch=amd64] ht
tps://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
root@ip-172-31-19-162:/home/ubuntu# sudo usermod -aG docker ${USER}
root@ip-172-31-19-162:/home/ubuntu# sudo usermod -aG docker Jenkins
usermod: user 'Jenkins' does not exist
root@ip-172-31-19-162:/home/ubuntu# sudo usermod -aG docker jenkins
root@ip-172-31-19-162:/home/ubuntu#
```

Step 2. Setup AWS Credentials on an EC2 instance and Create EKS Cluster Multinode 2 Worker

- To setup AWS Credentials use below commands:

```
# curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

```
# sudo apt install unzip
```

```
# unzip awscliv2.zip
```

```
# sudo ./aws/install
```

```
# aws configure
```

```
root@ip-172-31-19-162: /home/ubuntu
root@ip-172-31-19-162:/home/ubuntu# curl "https://awscli.amazonaws.com/awscli-ex
e-linux-x86_64.zip" -o "awscliv2.zip"
root@ip-172-31-19-162:/home/ubuntu# sudo apt install unzip
root@ip-172-31-19-162:/home/ubuntu# unzip awscliv2.zip
root@ip-172-31-19-162:/home/ubuntu# sudo ./aws/install
```

- To Generate Access Key and Secret Access Key:-

Click on Security and Credentials:-

