

## CI CD

### CI

In continuous integration after a code commit the software is build abd tested immediately ,In a large projects many developers commits are made many times during a day,with each commit code is build and tested,if the test is passed build is tested for deployment,if deployment is a success the code is pushed to production,this commit,build,test and deply is a continupous process and hence the name is continuous integration.

1)Launch ubuntu 20 machine and allow the port no.8080 in security groups I.e., click on add rule andallow port no.like below:

Custom tcp rule 8080 anywhere

The screenshot shows the AWS Management Console interface for launching an EC2 instance. At the top, there's a navigation bar with the AWS logo, 'Services' link, a search bar, and user information. Below this is a banner for the new launch instance wizard. The main content area shows the 'Step 1: Choose an Amazon Machine Image (AMI)' screen. It includes a progress bar with steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The AMI selection area lists two options: 'Ubuntu Server 20.04 LTS (HVM), SSD Volume Type' and 'Microsoft Windows Server 2019 Base'. Each option shows its icon, name, AMI ID, architecture, and a 'Select' button. The Ubuntu option also includes a 'Free tier eligible' badge and a link to Canonical's website. The Windows option also includes a 'Free tier eligible' badge. On the right side, there are radio buttons to select the architecture: '64-bit (x86)' (selected) and '64-bit (Arm)'. A 'Cancel and Exit' link is visible in the top right corner of the AMI selection area.

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

EC2

You've been invited to try an early, beta iteration of the new launch instance wizard. We will continue to improve the experience over the next few months. We're asking customers for their feedback on this early release. To exit the new launch instance wizard at any time, choose the **Cancel** button. [Try it now!](#)

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

Cancel and Exit

Apache 2.2, MySQL 5.5, PHP 5.3, and Ruby 1.8.7 available.  
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

**Ubuntu Server 20.04 LTS (HVM), SSD Volume Type** - ami-0fb653ca2d3203ac1 (64-bit x86) / ami-02af65b2d1ebdfafc (64-bit Arm) [Select](#)

**Free tier eligible** Ubuntu Server 20.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).  
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

☒ 64-bit (x86)  
☐ 64-bit (Arm)

**Microsoft Windows Server 2019 Base** - ami-0c4a11a8d0e503812 [Select](#)

**Windows** Microsoft Windows 2019 Datacenter edition. [English]  
**Free tier eligible** Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

64-bit (x86)

# CI CD

aws

Services

Search for services, features, blogs, docs, and more

[Alt+S]

Ohio

sudheer

EC2

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

**Assign a security group:** ☒ Create a **new** security group  
☐ Select an **existing** security group

**Security group name:**

**Description:**

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop
Custom TCP	TCP	8080	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop

**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

## CI CD

Click on connect and type the below

commands:

- 1) sudo -i
- 2) apt-get update
- 3) apt-get install default-jdk-y
- ) java -version

```
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-172-31-35-108:~$ sudo -i
root@ip-172-31-35-108:~# apt-get update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [144 kB]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse Translation-en [104 kB]
Get:10 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 c-n-f Metadata [9136 B]
Get:11 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1680 kB]
Get:12 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [317 kB]
```

i-00deded8b1b26d2bb (Jenkins1)

CI CD

## CI CD

```
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ca-certificates (20210119~20.04.2) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up openjdk-11-jre-headless:amd64 (11.0.14.1+1-0ubuntu1~20.04) ...
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/java to provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jjs to provide /usr/bin/jjs (jjs) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/keytool to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/rmid to provide /usr/bin/rmid (rmid) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/rmiregistry to provide /usr/bin/rmiregistry (rmiregistry) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/pack200 to provide /usr/bin/pack200 (pack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/unpack200 to provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/lib/jexec to provide /usr/bin/jexec (jexec) in auto mode
root@ip-172-31-35-108:~# java -version
openjdk version "11.0.14.1" 2022-02-08
OpenJDK Runtime Environment (build 11.0.14.1+1-Ubuntu-0ubuntu1.20.04)
OpenJDK 64-Bit Server VM (build 11.0.14.1+1-Ubuntu-0ubuntu1.20.04, mixed mode, sharing)
root@ip-172-31-35-108:~#
```

4)Google search install jenkins on ubuntu 20 and click on first link I.e digital ocean and

follow the steps.wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo

apt-key add -

sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ >  
/etc/apt/sources.list.d/jenkins.list

'sudo apt update

sudo apt install jenkins

Sudo systemctl start

jenkins

sudo systemctl status jenkins

The screenshot shows a web browser window with a Bing search result. The search query is "install+jenkins+on+ubuntu20&cvid=375473ab0d6b436cb8becafe141ee1e&qs=edge.0.69159j691...". The top result is "How To Install Jenkins on Ubuntu 20.04 | DigitalOcean" with a URL "https://www.digitalocean.com/.../how-to-install-jenkins-on-ubuntu-20-04". It is published on 22/06/2020 by Lyn Muldr and has an estimated reading time of 6 minutes. Below the title, it says "How To Install Jenkins on Ubuntu 20.04 Step 1 — Installing Jenkins. The version of Jenkins included with the default Ubuntu packages is often behind the latest... Step 2 — Starting Jenkins...". There is a section "EXPLORE FURTHER" with links to "How to Install Jenkins on Ubuntu 20.04 - Linuxize", "How to Install and Configure Jenkins on Ubuntu 20.04 LTS", "Official Jenkins image to use from Docker Hub", "How To Install Jenkins on Ubuntu 20.04/18.04 - Tecmint", and "How to Install Jenkins on Ubuntu 20.04 - Liquid Web". To the right of these links are the domain names: linuxize.com, www.linuxtechi.com, www.jenkins.io, www.tecmint.com, and www.liquidweb.com. Below this is another result "How to Install Jenkins on Ubuntu 20.04 - Linuxize" with a URL "https://linuxize.com/post/how-to-install-jenkins-on-ubuntu-20-04" and an estimated reading time of 4 minutes. It lists steps: "Installing Java", "Jenkins is a Java application and requires Java 8 or later to be installed on the system. We'll...", "Installing Jenkins", and "Installation: Jenkins on Ubuntu is relatively straightforward. We'll enable the Jenkins APT".

## CI CD

```
Setting up libsisu-inject-java (0.3.3-1) ...
Setting up libsisu-plexus-java (0.3.3-3) ...
Setting up libmaven3-core-java (3.6.3-1) ...
Setting up maven (3.6.3-1) ...
update-alternatives: using /usr/share/maven/bin/mvn to provide /usr/bin/mvn (mvn) in auto mode
root@ip-172-31-35-108:~# mvn -version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.14.1, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "5.11.0-1022-aws", arch: "amd64", family: "unix"
root@ip-172-31-35-108:~# git --version
git version 2.25.1
root@ip-172-31-35-108:~# wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
OK
root@ip-172-31-35-108:~#
```

i-00deded8b1b26d2bb (Jenkins1)

Public IPs: 18.191.193.136 Private IPs: 172.31.35.108

```
The following NEW packages will be installed:
ca-certificates-java fontconfig-config fonts-dejavu-core java-common libavahi-client3 libavahi-common-data
libavahi-common3 libcups2 libfontconfig1 libgraphite2-3 libharfbuzz0b libjpeg-turbo8 libjpeg8 liblcms2-2 libpcsclite1
openjdk-11-jre-headless
0 upgraded, 16 newly installed, 0 to remove and 91 not upgraded.
The following additional packages will be installed:
net-tools
The following NEW packages will be installed:
jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 91 not upgraded.
Need to get 91.4 MB of archives.
After this operation, 95.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20180626.aebd88e-1ubuntu1 [1
96 kB]
Get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.332.1 [91.2 MB]
Fetched 91.4 MB in 6s (14.5 MB/s)
Selecting previously unselected package net-tools.
(Reading database ... 65359 files and directories currently installed.)
Preparing to unpack .../net-tools 1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../jenkins_2.332.1_all.deb ...
Unpacking jenkins (2.332.1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up jenkins (2.332.1) ...
```

i-00deded8b1b26d2bb (Jenkins1)

Public IPs: 18.191.193.136 Private IPs: 172.31.35.108

## CI CD

```
Setting up jenkins (2.332.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /lib/systemd/system/jenkins.service.
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for systemd (245.4-4ubuntu3.13) ...
root@ip-172-31-35-108:~# systemctl start jenkins
root@ip-172-31-35-108:~# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2022-03-30 04:51:00 UTC; 2min 6s ago
     Main PID: 18940 (java)
       Tasks: 35 (limit: 1147)
      Memory: 304.6M
      CGroup: /system.slice/jenkins.service
              └─18940 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins

Mar 30 04:50:31 ip-172-31-35-108 jenkins[18940]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Mar 30 04:50:31 ip-172-31-35-108 jenkins[18940]: *****
Mar 30 04:50:31 ip-172-31-35-108 jenkins[18940]: *****
Mar 30 04:50:31 ip-172-31-35-108 jenkins[18940]: *****
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.783+0000 [id=29] INFO jenkins.InitReac
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.817+0000 [id=22] INFO hudson.lifecycle
Mar 30 04:51:00 ip-172-31-35-108 systemd[1]: Started Jenkins Continuous Integration Server.
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.945+0000 [id=44] INFO h.m.DownloadServ
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.946+0000 [id=44] INFO hudson.util.Retri
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.953+0000 [id=44] INFO hudson.model.Asy
lines 1-19/19 (END)
```

i-00deded8b1b26d2bb (Jenkins1)

Public IPs: 18.191.193.136 Private IPs: 172.31.35.108

```
Mar 30 04:51:00 ip-172-31-35-108 systemd[1]: Started Jenkins Continuous Integration Server.
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.945+0000 [id=44] INFO h.m.DownloadServ
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.946+0000 [id=44] INFO hudson.util.Retri
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.953+0000 [id=44] INFO hudson.model.Asy
lines 1-19/19 (END)
```

```
root@ip-172-31-35-108:~# ufw allow 8080
rules updated
rules updated (v6)
root@ip-172-31-35-108:~# ufw allow OpenSSH
rules updated
rules updated (v6)
root@ip-172-31-35-108:~# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
firewall is active and enabled on system startup
root@ip-172-31-35-108:~# ufw status
status: active
```

to	Action	From
-	-----	----
8080	ALLOW	Anywhere
OpenSSH	ALLOW	Anywhere
8080 (v6)	ALLOW	Anywhere (v6)
OpenSSH (v6)	ALLOW	Anywhere (v6)

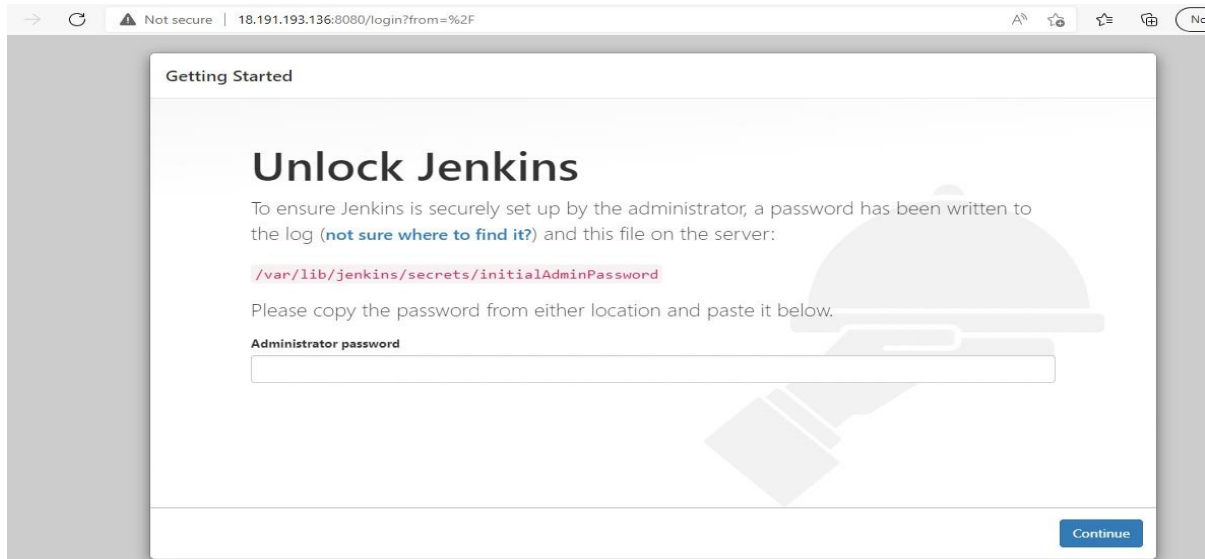
```
root@ip-172-31-35-108:~#
```

## CI CD

5) Now open AWS console copy public ip of jenkins server instance.



## CI CD



```
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.946+0000 [id=44] INFO hudson.util.Retri
Mar 30 04:51:00 ip-172-31-35-108 jenkins[18940]: 2022-03-30 04:51:00.953+0000 [id=44] INFO hudson.model.Asy
lines 1-19/19 (END)

root@ip-172-31-35-108:~# ufw allow 8080
Rules updated
Rules updated (v6)
root@ip-172-31-35-108:~# ufw allow OpenSSH
Rules updated
Rules updated (v6)
root@ip-172-31-35-108:~# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@ip-172-31-35-108:~# ufw status
Status: active

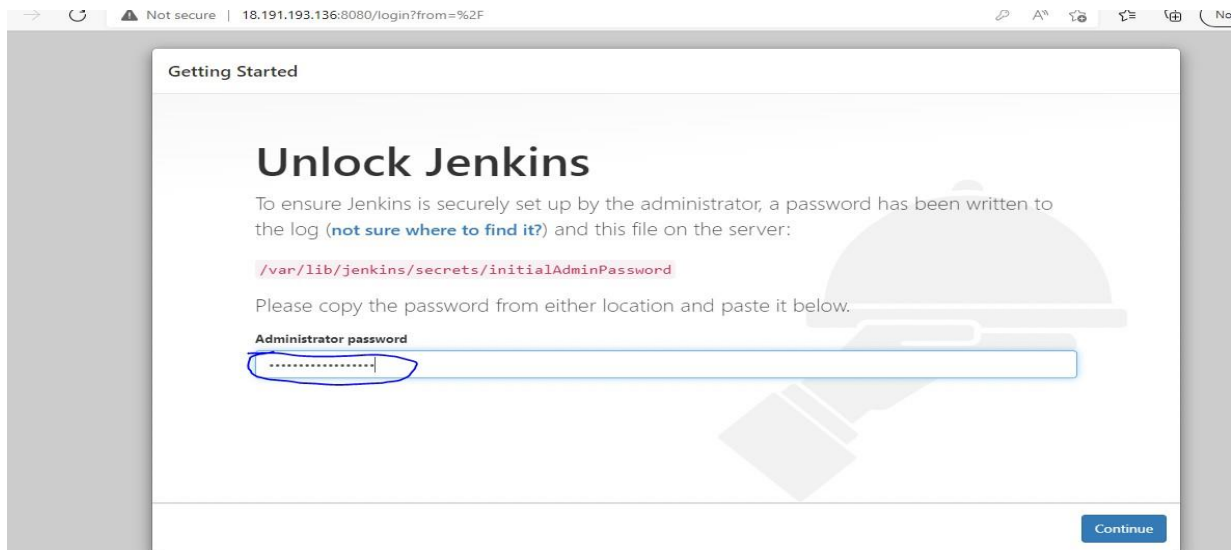
To Action From
--
8080 ALLOW Anywhere
OpenSSH ALLOW Anywhere
8080 (v6) ALLOW Anywhere (v6)
OpenSSH (v6) ALLOW Anywhere (v6)

root@ip-172-31-35-108:~# cat /var/lib/jenkins/secrets/initialAdminPassword
b7a4154120954a9baecca9c38b1a5597
root@ip-172-31-35-108:~#
```

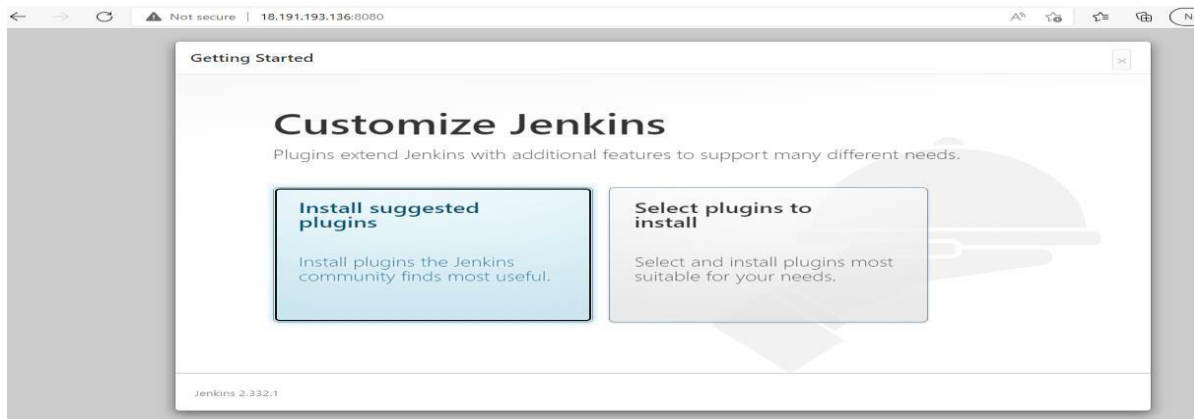
i-00deded8b1b26d2bb (Jenkins1)

6) To access jenkins <http://<public ip>:8080>

## CI CD

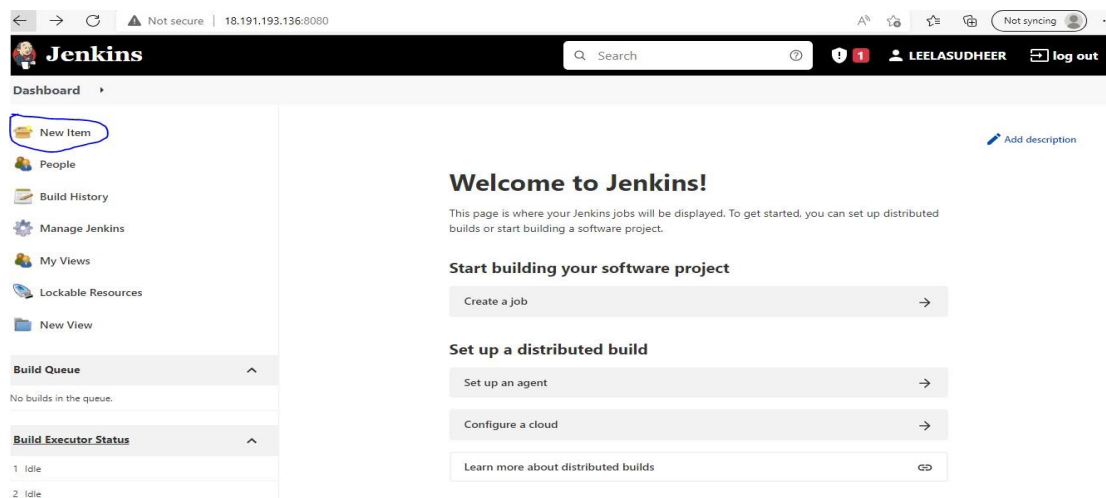
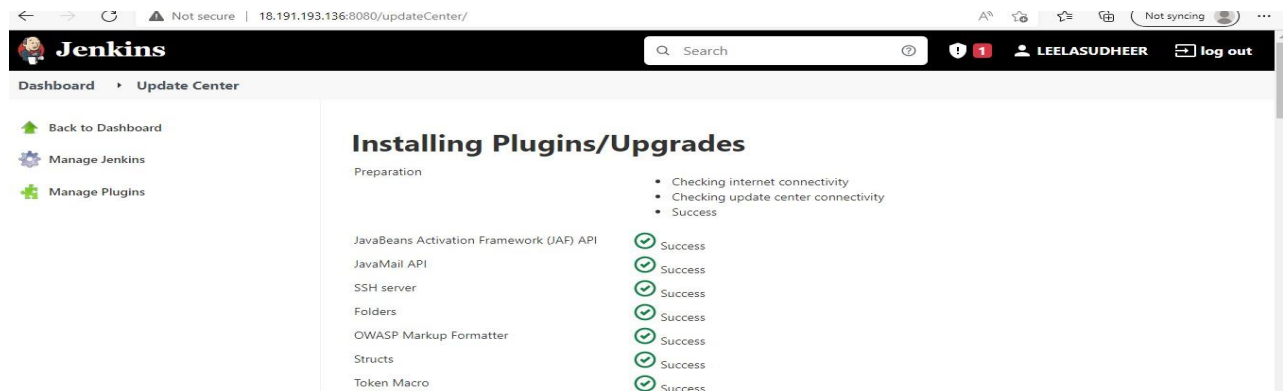
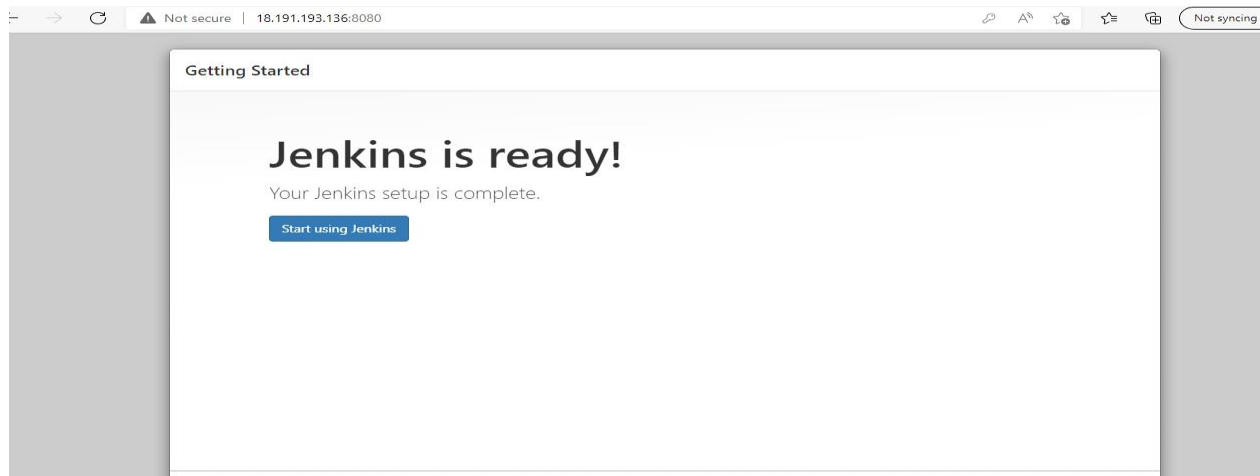


7) Click on the "install suggested plugins" button



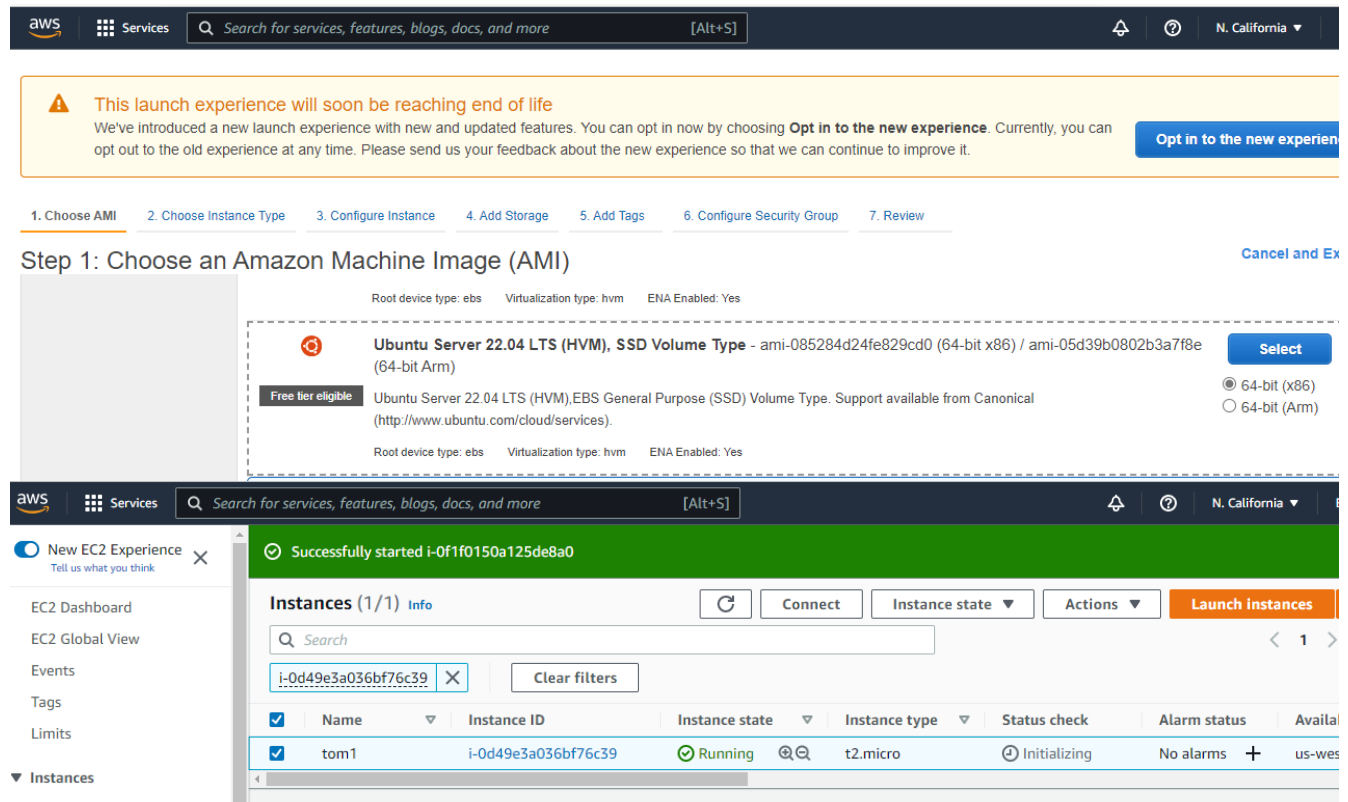


# CI CD



Install tomcat:

1) Launch ubuntu 20 machine and allow the port no.8080 in security groups I.e., click on add rule and allow port no.like below:



The screenshot shows the AWS Management Console interface for launching an EC2 instance. The top navigation bar includes the AWS logo, 'Services' link, a search bar, and the region 'N. California'. A warning banner at the top states: 'This launch experience will soon be reaching end of life. We've introduced a new launch experience with new and updated features. You can opt in now by choosing **Opt in to the new experience**. Currently, you can opt out to the old experience at any time. Please send us your feedback about the new experience so that we can continue to improve it.' Below the banner, the 'Launch wizard' progress bar shows steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, 7. Review. The 'Step 1: Choose an Amazon Machine Image (AMI)' is selected. It displays the 'Ubuntu Server 22.04 LTS (HVM), SSD Volume Type' (ami-085284d24fe829cd0) as the chosen AMI. Below the AMI selection, a table shows the instance 'tom1' with ID 'i-0d49e3a036bf76c39' in a 'Running' state. The instance type is 't2.micro' and the status check is 'Initializing'.

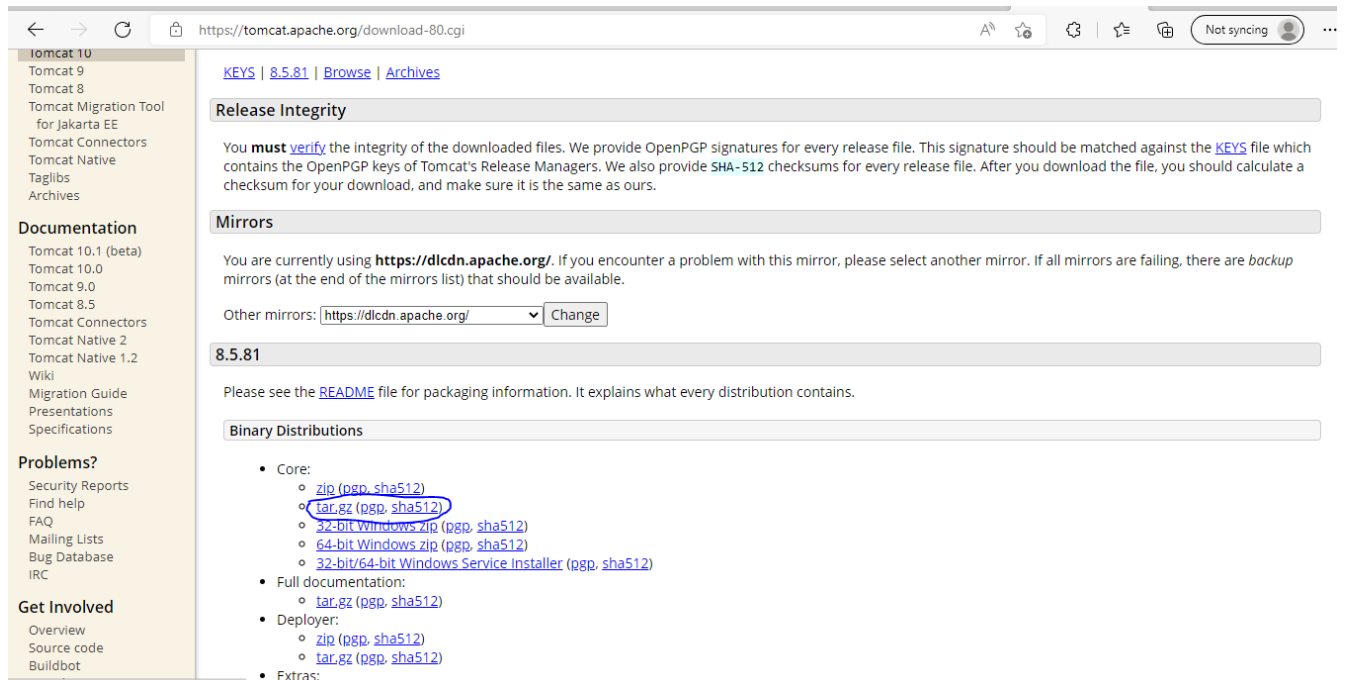
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
tom1	i-0d49e3a036bf76c39	Running	t2.micro	Initializing	No alarms	us-west-2

2) Download tomcat packages from <https://tomcat.apache.org/download-80.cgi> onto /opt on EC2 instance

Note: Make sure you change <version> with the tomcat version which you download.

Copy the tomcat8 tar.gz file

## CI CD



The screenshot shows the Apache Tomcat download page at <https://tomcat.apache.org/download-80.cgi>. The left sidebar contains navigation links for Tomcat versions (10, 9, 8), tools, connectors, native libraries, and documentation. The main content area includes sections for Release Integrity, Mirrors, 8.5.81 version information, and Binary Distributions. The Binary Distributions section lists download links for Core, Full documentation, Deployer, and Extras, with [tar.gz \(pgp, sha512\)](#) highlighted.

```
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of Wed Jul 13 09:26:00 UTC 2022

System load: 0.0          Processes:              98
Usage of /:  19.2% of 7.58GB Users logged in:             0
Memory usage: 21%        IPv4 address for eth0: 172.31.31.239
Swap usage:  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

Last login: Wed Jul 13 05:44:33 2022 from 14.143.3.38
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-31-239:~$ wget https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.81/bin/apache-tomcat-8.5.81.tar.gz
```

```
*** System restart required ***
Last login: Wed Jul 13 06:22:30 2022 from 14.143.3.38
ubuntu@ip-172-31-6-92:~$ ls
apache-tomcat-8.5.81  apache-tomcat-8.5.81.tar.gz
ubuntu@ip-172-31-6-92:~$ |
```

## CI CD

```
*** System restart required ***
Last login: Wed Jul 13 06:22:30 2022 from 14.143.3.38
ubuntu@ip-172-31-6-92:~$ ls
apache-tomcat-8.5.81  apache-tomcat-8.5.81.tar.gz
ubuntu@ip-172-31-6-92:~$ cd apache-tomcat-8.5.81
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81$ ls
BUILDING.txt  LICENSE  README.md  RUNNING.txt  conf  logs  webapps
CONTRIBUTING.md  NOTICE  RELEASE-NOTES  bin  lib  temp  work
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81$ cd bin
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ ls
bootstrap.jar  ciphers.sh  commons-daemon-native.tar.gz  daemon.sh  shutdown.bat  tomcat-native.tar.gz
catalina-tasks.xml  commons-daemon.jar  digest.bat  shutdown.sh  tool-wrapper.bat
catalina.bat  configtest.bat  digest.sh  startup.bat  tool-wrapper.sh
catalina.sh  configtest.sh  setclasspath.bat  startup.sh  version.bat
ciphers.bat  configtest.sh  setclasspath.sh  tomcat-juli.jar  version.sh
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ ./startup.sh
Using CATALINA_BASE:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_HOME:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_TMPDIR: /home/ubuntu/apache-tomcat-8.5.81/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/ubuntu/apache-tomcat-8.5.81/bin/bootstrap.jar:/home/ubuntu/apache-tomcat-8.5.81/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ |
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ cd ..
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81$ cd conf
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/conf$ ls
catalina  catalina.properties  jaspic-providers.xml  logging.properties  tomcat-users.xml  web.xml
catalina.policy  context.xml  jaspic-providers.xsd  server.xml  tomcat-users.xsd
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/conf$ sudo vi tomcat-user.xml
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/conf$ sudo vi tomcat-users.xml
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/conf$ |

The users below are wrapped in a comment and are therefore ignored. If you
wish to configure one or more of these users for use with the manager web
application, do not forget to remove the <!-- ... --> that surrounds them. You
will also need to set the passwords to something appropriate.
-->
<!--
<user username="admin" password="<must-be-changed>" roles="manager-gui"/>
<user username="robot" password="<must-be-changed>" roles="manager-script"/>
-->
<!--
The sample user and role entries below are intended for use with the
examples web application. They are wrapped in a comment and thus are ignored
when reading this file. If you wish to configure these users for use with the
examples web application, do not forget to remove the <!-- ... --> that surrounds
them. You will also need to set the passwords to something appropriate.
-->
<!--
<role rolename="tomcat"/>
<role rolename="role1"/>
<user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
<user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
<user username="role1" password="<must-be-changed>" roles="role1"/>
-->
<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>
<user username="deployer" password="deployer" roles="manager-script"/>
<user username="admin" password="admin" roles="manager-gui"/>
</tomcat-users>
```

3) copy the public ip and paste it browser tomcat started.

## CI CD

The screenshot shows the Apache Tomcat 8.5.81 installation page. At the top, there's a navigation bar with links: Home, Documentation, Configuration, Examples, Wiki, Mailing Lists, and a Find Help button. Below this, a green banner says "If you're seeing this, you've successfully installed Tomcat. Congratulations!". To the left of the banner is the Tomcat logo (a cat). To the right, under "Recommended Reading:", there are links for "Security Considerations How-To", "Manager Application How-To", and "Clustering/Session Replication How-To". Further right are buttons for "Server Status", "Manager App", and "Host Manager". Below the banner is a "Developer Quick Start" section with links for "Tomcat Setup", "First Web Application", "Realms & AAA", "JDBC DataSources", "Examples", "Servlet Specifications", and "Tomcat Versions". At the bottom, there are three yellow boxes: "Managing Tomcat" (with links for "Release Notes" and "Changelog"), "Documentation" (with links for "Tomcat 8.5 Documentation", "Tomcat 8.5 Configuration", "Tomcat Wiki", and "Tomcat 8.5 Bug Database"), and "Getting Help" (with links for "FAQ and Mailing Lists", "tomcat-announce", "tomcat-users", "taglibs-user", and "tomcat-dev").

4) Then open Jenkins console click on manage Jenkins, manage credentials add tomcat credentials.

The screenshot shows the Jenkins Credentials page for a credential named 'tomcatcred'. The breadcrumb trail is: Dashboard > Credentials > System > Global credentials (unrestricted) > admin/\*\*\*\*\* (tomcatcred). On the left, there are buttons for 'Update', 'Delete', and 'Move'. The main form has the following fields: 'Scope' (a dropdown menu set to 'Global (Jenkins, nodes, items, all child items, etc)'), 'Username' (a text field containing 'admin'), 'Treat username as secret' (an unchecked checkbox), 'Password' (a text field with a lock icon and the word 'Concealed', and a 'Change Password' button), 'ID' (a text field containing 'tomcatcred'), and 'Description' (a text field containing 'tomcatcred'). At the bottom, there is a 'Save' button.

5) Click on manage Jenkins manage plugins and click on available tab to select deploy to war container plugin move Jenkins dash board click on new item and select ok and sourcecodemanagement give git hub repository url and build section select invoke top level maven select give maven version under goals clean install package and post-build actions give WAR/EAR FILE give \*\*/\*.war and add tomcat credentials and tomcat url click on apply and save.



# CI CD

← → ↻ ⚠ Not secure | 13.57.245.241:8080/view/all/newJob

**Jenkins** 🔍 Search ? 🔔 1 🛡 1 👤 Leelasudheer ⌵ 🗂

Dashboard > All >

## Enter an item name

Cicd

» Required field



### Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



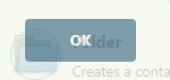
### Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



### Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a

3.57.245.241:8080

← → ↻ ⚠ Not secure | 13.57.245.241:8080/job/Cicd/configure

Dashboard > Cicd >

General **Source Code Management** Build Triggers Build Environment Build Post-build Actions

☐ None

☒ Git ?

Repositories ?

Repository URL ?

https://github.com/Leelasudheer/Hello-World.git

Credentials ?

- none -

+ Add

Advanced...

Add Repository

Save Apply



## CI CD

← → ↻ ⚠ Not secure | 13.57.245.241:8080/job/Cicd/configure

Dashboard > Cicd >

General Source Code Management Build Triggers Build Environment Build **Post-build Actions**

+ Add

Tomcat URL ?

http://54.67.85.114:8090/

Advanced...

Add Container ▾

☐ Deploy on failure

Add post-build action ▾

Save Apply

6) Click on build now build is success.

← → ↻ ⚠ Not secure | 13.57.245.241:8080/job/Cicd/

**Jenkins** 🔍 Search ? 🔔 1 🛡 1 👤 Leelasudheer ▾ 🚪 log out

Dashboard > Cicd >

↑ Back to Dashboard

📄 Status

</> Changes

📁 Workspace

▶ Build Now

⚙ Configure

🗑 Delete Project

✎ Rename

📁 **Build History** trend ▾

🔍 Filter builds...

🟢 #13  
Jul 13, 2022 12:36 PM

### Project Cicd

📁 Workspace

</> Recent Changes

### Permalinks

- Last build (#13), 21 min ago
- Last stable build (#13), 21 min ago
- Last successful build (#13), 21 min ago
- Last failed build (#12), 25 min ago
- Last unsuccessful build (#12), 25 min ago
- Last completed build (#13), 21 min ago

✎ Add description

🚫 Disable Project

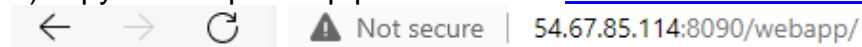
## CI CD

7) Check your tomcat cmd cd webapps prompt war file is moved are not.

```
ubuntu@ip-172-31-6-92: ~/apache-tomcat-8.5.81/bin
catalina.bat      digest.bat        tomcat-native.tar.gz
catalina.sh       digest.sh         tool-wrapper.bat
ciphers.bat       setclasspath.bat tool-wrapper.sh
ciphers.sh        setclasspath.sh  version.bat
commons-daemon-native.tar.gz shutdown.bat      version.sh
commons-daemon.jar shutdown.sh
configtest.bat    startup.bat
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ ./shutdown.sh
Using CATALINA_BASE:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_HOME:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_TMPDIR: /home/ubuntu/apache-tomcat-8.5.81/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/ubuntu/apache-tomcat-8.5.81/bin/bootstrap.jar:/home
/ubuntu/apache-tomcat-8.5.81/bin/tomcat-juli.jar
Using CATALINA_OPTS:
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ ./startup.sh
Using CATALINA_BASE:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_HOME:   /home/ubuntu/apache-tomcat-8.5.81
Using CATALINA_TMPDIR: /home/ubuntu/apache-tomcat-8.5.81/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /home/ubuntu/apache-tomcat-8.5.81/bin/bootstrap.jar:/home
/ubuntu/apache-tomcat-8.5.81/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/bin$ cd ..
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81$ ls
BUILDING.txt  LICENSE  README.md  RUNNING.txt  conf  logs  webapps
CONTRIBUTING.md  NOTICE  RELEASE-NOTES  bin      lib  temp  work
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81$ cd webapps
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/webapps$ ls
ROOT  docs  examples  host-manager  manager
ubuntu@ip-172-31-6-92:~/apache-tomcat-8.5.81/webapps$ ls
ROOT  docs  examples  host-manager  manager  webapp  webapp.war
```

## CI CD

8) Copy tomcat public ip paste browser [54.67.85.114:8090/webapp/](http://54.67.85.114:8090/webapp/)



# Hello All, !!!

## this is my master branch new

Tomcat installation using these commands.

```
1 sudo apt-get update -y
2 sudo apt-get install openjdk-8*
3 https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.81/bin/apache-tomcat-8.5.81.tar.gz
4 sudo wget https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.81/bin/apache-tomcat-8.5.81.tar.gz
5 wget https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.81/bin/apache-tomcat-8.5.81.tar.gz
6 ls
7 tar-xvzf apache-tomcat-8.5.81.tar.gz
8 tar -xvzf apache-tomcat-8.5.81.tar.gz
9 ls
10 cd apache-tomcat-8.5.81/
11 ls
12 cd bin/
13 ls
14 cd ./shutdown.sh
15 ./shutdown.sh
16 ./startup.sh
17 sudo find / -name context.xml
18 sudo vi /home/ubuntu/apache-tomcat-8.5.81/webapps/host-manager/META-INF/context.xml
19 sudo vi /home/ubuntu/apache-tomcat-8.5.81/webapps/manager/META-INF/context.xml
20 sudo vi /home/ubuntu/apache-tomcat-8.5.81/webapps/examples/META-INF/context.xml
21 sudo vi /home/ubuntu/apache-tomcat-8.5.81/conf/context.xml
22 ./shutdown.sh
23 ./startup.sh
```

## CI CD

```
24 cd ..
25 cd conf/
26 ls
27 sudo vi tomcat-users.xml
28 cd ..
29 ls
30 cd bin
31 ./shutdown.sh
32 ./startup.sh
```