

Install Git-jenkins-maven-tomcat

Please follow below steps.

1.) Open Amazon console Create EC2 instance and give Security group 8080 & 22

2.) Open Ec2 terminal Install Java

```
yum install java-1.8*
```

```
java -version
```

```
find / -name jre
```

```
/etc/alternatives/jre
```

```
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2.x86_64/jre
```

```
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2.x86_64-debug/jre
```

```
/usr/lib/jvm/jre
```

```
/usr/lib/jvm-exports/jre
```

```
/usr/share/javadoc/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2-debug/jre
```

```
/usr/share/javadoc/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2/jre
```

In vi /etc/profile

```
fi
```

```
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2.x86_64
```

```
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL JAVA_HOME
```

```
sudo su -
```

```
Last login: Sun Dec 30 12:51:16 UTC 2018 on pts/1
```

```
java -version
```

```
openjdk version "1.8.0_191"
```

```
OpenJDK Runtime Environment (build 1.8.0_191-b12)
```

```
OpenJDK 64-Bit Server VM (build 25.191-b12, mixed mode)
```

3.) Install Jenkins go to terminal please follow below steps

```
yum -y install wget
```

```
wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

```
rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

```
yum -y install jenkins
```

Start Jenkins in Terminal please follow below steps

```
systemctl status jenkins
```

```
systemctl start jenkins
```

Setup jenkins at boot level in terminal give below command

```
chkconfig jenkins on
```

Accessing Jenkins please open new tab in browser give your ec2 publicip & port accessing 8080

<http://YOUR-SERVER-PUBLIC-IP:8080>

After above step we want unlock the jenkins copy below path

`/var/lib/jenkins/secrets/initialAdminPassword`-----paste this path in terminal

`cat /var/lib/jenkins/secrets/initialAdminPassword`

0232fb27dbf94ac0a2ba6dab6b9aa329----like this code will come paste it on jenkins page to unlock the page

Jenkins page is opened go to corner of right side select admin option go to configure change the password the logout and give username as admin & your password

Go to Jenkins console configure jenkins path like click on manage jenkins----select global tool configuration-----set jdk path

Go to terminal type `find / -name jre`-----keep the path in jenkins console

4.) Configure git plugging on jenkins

Open our Ec2 terminal type below command

`yum install git -y`

Go to jenkins console----go to manage jenkins----go to manage plugins-----go to search---type github-----install without restart----Plugins will download

Go to manage jenkins-----go to global tool configuration-----set the path-----username Default-----path git

5.) Install & configure Maven build tool on Jenkins

Open terminal go to `cd /opt` under this we create maven directory follow below command

`mkdir maven`

go to `cd /opt/maven`

download the maven version-----`wget`

<http://mirrors.estointernet.in/apache/maven/maven-3/3.6.0/binaries/apache-maven-3.6.0-bin.tar.gz>

tar file will be download please extract it like-----`tar -zxvf apache-maven-3.6.0-bin.tar.gz`

file will be xtracted-----go `cd apache-maven-3.6.0-bin.tar.gz`

type command like-----`mv * /opt/maven/`

type----`cd ..`-----`rm -rf apache-maven-3.6.0-bin.tar.gz` `apache-maven-3.6.0`

Setup M2_HOME and M2 paths in `.bash_profile` of user and add these to path variable

Go to `vi ~/.bash_profile`-----specity the path as below

`M2_HOME=/opt/maven/`

`M2=$M2_HOME/bin`

`PATH=$PATH:$HOME/bin:$M2_HOME:$M2_HOME/bin`

`sudo su -` (this will be refresh & show the version on maven path)

mvn -version (shows like below lines)

```
Apache Maven 3.6.0 (97c98ec64a1fdfee7767ce5ffb20918da4f719f3; 2018-10-24T18:41:47Z)
Maven home: /opt/maven
Java version: 1.8.0_191, vendor: Oracle Corporation, runtime:
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2.x86_64/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "4.14.77-81.59.amzn2.x86_64", arch: "amd64", family: "unix"
```

Go to jenkins console----go to manage jenkins-----go to manage plugins-----in search type maven invoker-----download plugin without restart the plugin---Plugin will download
Go to manage jenkins----go to global tool configuration-----go to maven option-----specify name as maven-----path /opt/maven
Go to create job---type name----select maven project-----select source code (git)-----keep the github application hello-world url-----in build triggers poll scm give 5stars * * * * *-----in build we have pom.xml type in goals (clean install package)

6.) Tomcat installation on EC2 instance

Go to AWS console launch ec2 new instance in security groups keep 8080 & 8090 ports launched the instance.

Go to ec2 terminal

```
Install java
yum install java-1.8*
```

specify path on vi /etc/profile/

```
fi
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-0.amzn2.x86_64
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL JAVA_HOME
```

Go to cd /opt create tomcat directory

```
mkdir tomcat
cd /opt/tomcat
wget http://mirrors.wuchna.com/apachemirror/tomcat/tomcat-8/v8.5.37/bin/apache-tomcat-8.5.37.tar.gz
tar -xvzf /opt/apache-tomcat-8.5.35.tar.gz
cd apache-tomcat-8.5.35
mv * /opt/tomcat
rm -rf apache-tomcat-8.5.37 apache-tomcat-8.5.37.tar.gz
```

Go to cd /opt/tomcat/bin

Give executing permissions to startup.sh and shutdown.sh which are under bin

chmod +x /opt/tomcat/bin/startup.sh shutdown.sh-----type this command in cd /opt/tomcat/bin

create softlink files for tomcat startup.sh and shutdown.sh-----in cd /opt/tomcat/bin

```
ln -s /opt/tomcat/bin/startup.sh /usr/local/bin/tomcatup
```

```
ln -s /opt/tomcat/bin/startup.sh /usr/local/bin/tomcatdown
```

start tomcat server-----tomcatup-----stop tomcat server-----tomcatdown

Access tomcat application from browser on port 8080

http://<Public_IP>:8080

Change the port number go to cd /opt/tomcat/conf under this

vi server.xml-----to search type 8080-----change the port number to 8090 in ec2 security group access port to 8090

```
<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
-->
<Connector port="8090" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
    port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />
-->
```

Go to cd /opt/tomcat/bin shutdown & startup the server (or) type tomcatdown & tomcatup to restart the server after changing the port to 8090

Now application is accessible on port 8090. but tomcat application doesnt allow to login from browser. changing a default parameter in context.xml does address this issue

Search for context.xml

```
find / -name context.xml
```

```
/opt/tomcat/conf/context.xml
/opt/tomcat/webapps/host-manager/META-INF/context.xml
/opt/tomcat/webapps/manager/META-INF/context.xml
```

change 2 files that is-----vi /opt/tomcat/webapps/host-manager/META-INF/context.xml-----changes as below

```
<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
:wq!
```

change 2nd file that is-----vi /opt/tomcat/webapps/manager/META-INF/context.xml-----changes as below

```
<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
```

:wq!

Restart the tomcat server tomcatdown & tomcatup

Check the tomcat server it is up are not type commands-----ps -ef|grep tomcat

Update users information in the tomcat-users.xml file go to tomcat home directory and Add below users to conf/tomcat-user.xml file

Go to cd /opt/tomcat/conf/-----open-----vi tomcat-user.xml----add below mentioned lines

```
<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="tomcat" password="s3cret" roles="manager-gui"/>
```

Restart the tomcat server tomcatdown & tomcatup

Go to tomcat console tomcatec2publici:8090

Give username:tomcate

Password:s3cret

Go to jenkins console----go to manage jenkins----go to manage plugins-----go to search---Deploye to container-----install without restart----Plugins will download

Go to credentials-----go to global----select adding some credentials----username deployer-----password deployer-----ID Tomcat_Credentials----Description Tomcat_Credentials

Finally login to tomcat console-----Create New Job in Jenkins-----select free style-----under source code give github URL hello world-----select poll SCM (* * * * *)----under build pom.xml,goals (clean install package)-----Post-build Actions Select (Deploy war/ear)----**/*.war---give deployer----under container add container select tomcat 8X give deployer credentials----give tomcat access URL http://tomcatpublicIP:8090 Apply & save-----Build now

