**Install Jenkins on AWS EC2**

Jenkins is a self-contained Java-based program, ready to run out-of-the-box, with packages for Windows, Mac OS X and other Unix-like operating systems. As an extensible automation server, Jenkins can be used as a simple CI server or turned into the continuous delivery hub for any project.

**Prerequisites:**

EC2 RHEL 7.x Instance

With Internet Access

Security Group with Port 8080 open for internet

Java v1.8.x

Install Java

We will be using open java for our demo, Get latest version from http://openjdk.java.net/install/

Create 2 Linux instances, one for Jenkins Master and the other for Jenkins Slave and follow below steps.

Then switch as root user: **sudo -i**

**Installing java 8 on master node:**

By default, Amazon Linux comes with java 1.7. So, first uninstall it and then install java8.

**Uninstalling java7:** **yum remove java**

**Installing java8:****~~yum install java-1.8\*~~**

**OR** # **yum install java-1.8.0-openjdk-devel**

**Confirm Java Version:** **java -version**

Let’s set java home

**~~Find java path:~~****~~find /usr/lib/jvm/java-1.8\* | head -n 3~~**

**Find the path like this: /usr/lib/jvm/java-1.8.0-openjdk**

**Then copy the last line and then open .bash\_profile**: **vi .bash\_profile**

JAVA\_HOME=copied path

PATH=$PATH:$JAVA\_HOME:$HOME/bin

export PATH

**# To set it permanently update your .bash\_profile: source ~/.bash\_profile**

**Verify java home: echo $JAVA\_HOME**

**Then verify java version [root@~]#** **java -version**

The output should be something like this,

**openjdk version "1.8.0\_151"**

**OpenJDK Runtime Environment (build 1.8.0\_151-b12)**

**OpenJDK 64-Bit Server VM (build 25.151-b12, mixed mode)**

**Install Jenkins on Master instance:**

You can install Jenkins using the rpm or by setting up the repo. We will setup the repo so that we can update it easily in future. Get latest version of jenkins from https://pkg.jenkins.io/redhat-stable/

**Installing Jenkins:** **yum -y install wget**

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

**2) rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key**

**3) yum -y install jenkins**

**Starting Jenkins:** **service jenkins start** OR

**systemctl start jenkins**

# Setup Jenkins to start at boot,

**systemctl enable jenkins**

**Installing java 8 on all Slave instance:**

By default, Amazon Linux comes with java 1.7. So, first uninstall it and then install java8.

**Uninstalling java7:** **yum remove java**

**Installing java8:** **~~yum install java-1.8\*~~**

# **yum install java-1.8.0-openjdk-devel**

**Confirm Java Version:** **java -version**

Let’s set java home

**~~Find java path:~~****~~find /usr/lib/jvm/java-1.8\* | head -n 3~~**

**Find the path like this: /usr/lib/jvm/java-1.8.0-openjdk**

**Then copy the last line and then open .bash\_profile**: **vi .bash\_profile**

JAVA\_HOME=copied path

PATH=$PATH:$JAVA\_HOME

export JAVA\_HOME

**# To set it permanently update your .bash\_profile: source ~/.bash\_profile**

The output should be something like this,

**[root@~]#** **java -version**

**openjdk version "1.8.0\_151"**

**OpenJDK Runtime Environment (build 1.8.0\_151-b12)**

**OpenJDK 64-Bit Server VM (build 25.151-b12, mixed mode)**

**Setup Jenkins Slave (Repeat steps for all slaves):**

# Create user and add the user to wheel group(make sure username is same on all slave nodes)

**useradd jenkins-slave**

Then set password for the user.

**passwd jenkins-slave**

Then give sudo privileges to the user. Run **visudo** command to edit the **/etc/sudoers** file.

Find the lines in the file that grant **sudo** access to users in the group **wheel** when enabled.

## Allows people in group wheel to run all commands

# %wheel ALL=(ALL) ALL

Remove the comment character (#) at the start of the second line. This enables the configuration option.

Then save your changes and exit the editor.

Then Add the user you created to the wheel group using the usermod command.

**# usermod -aG wheel jenkins-slave**

Then switch as Jenkins-slave user.

**sudo su - jenkins-slave**

Then Create SSH Keys

**ssh-keygen -t rsa -N "" -f /home/jenkins-slave/.ssh/id\_rsa**

# The private and public keys will be created at these locations `/home/jenkins-slave/.ssh/id\_rsa` and `/home/jenkins-slave/.ssh/id\_rsa.pub`

Then get into **.ssh** directory: **cd .ssh**

Then load the content of id\_rsa.pub file to authorized\_keys file: **cat id\_rsa.pub > authorized\_keys**

Then give 700 permissions to authorized\_keys file: **chmod 700 authorized\_keys**

**Configuration on Master**

Copy the slave node's public key[id\_rsa.pub] to Master Node's known\_hosts file

**mkdir -p /var/lib/jenkins/.ssh**

**cd /var/lib/jenkins/.ssh**

**Repeat below steps for all slaves:**

**By changing the private ip address of slaves, run the below command:**

**ssh-keyscan -H SLAVE-NODE-PRIVATE\_IP-OR-HOSTNAME >> /var/lib/jenkins/.ssh/known\_hosts**

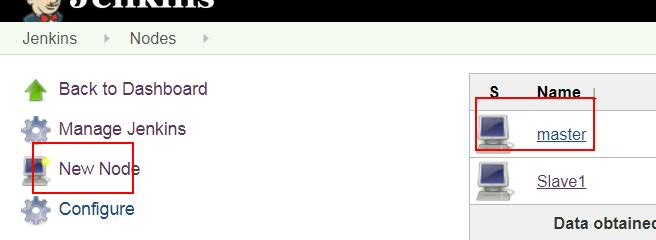
**Example: # ssh-keyscan -H 172.31.38.42 >>/var/lib/jenkins/.ssh/known\_hosts**

**chown jenkins:jenkins known\_hosts**

**chmod 700 known\_hosts**

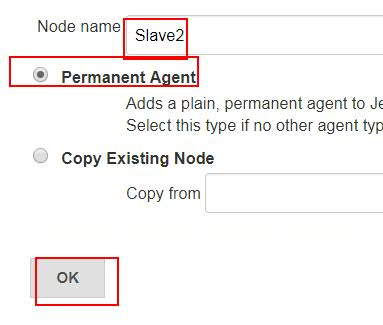
**Creating Nodes on Master (Repeat steps for 2 slaves):**

Then login to Jenkins admin console on master. Go to **Manage Jenkins**, then select **Manage Nodes**.

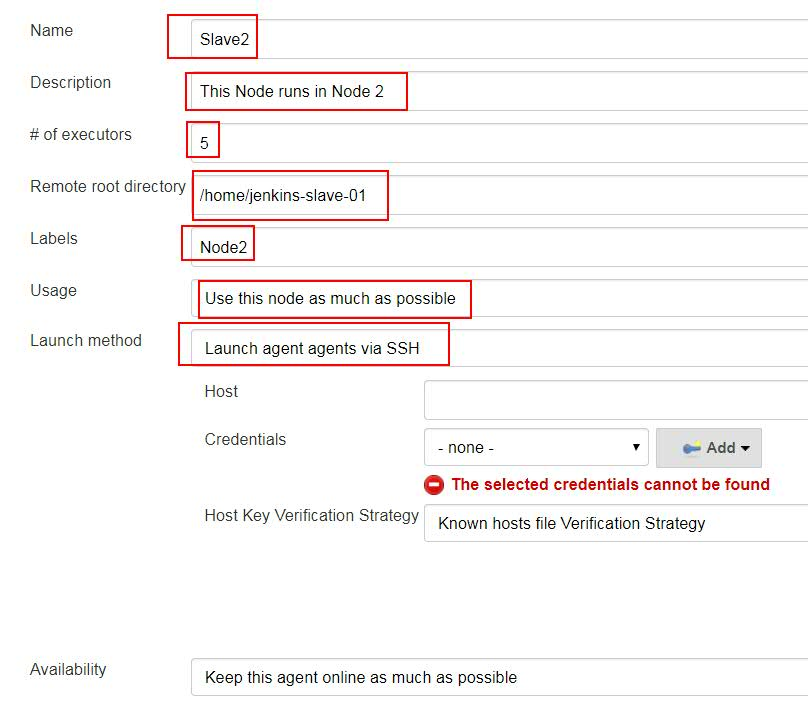


Then select **New Node**.

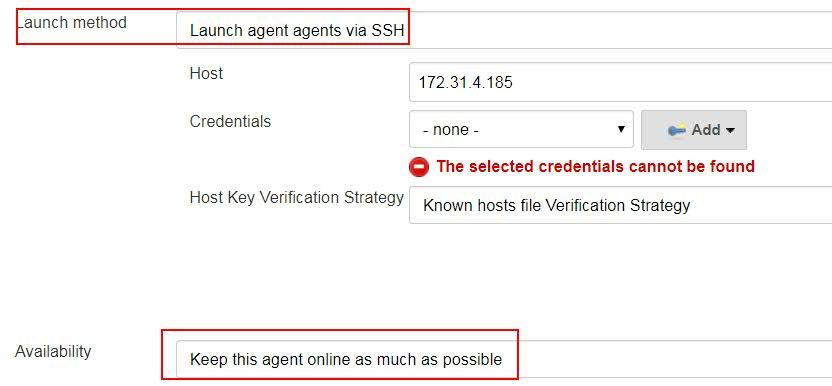
Then give the Node name and Select Permanent Agent. Ex: **Name**: **Slave2**



Then click on OK. Then give the details, like below.



Give the node details. Select the **Launch agent via SSH** agent option to connect, then give the Private IP address if the salve instance , Ex**: HOST: 172.31.4.185**, then select the add user option.

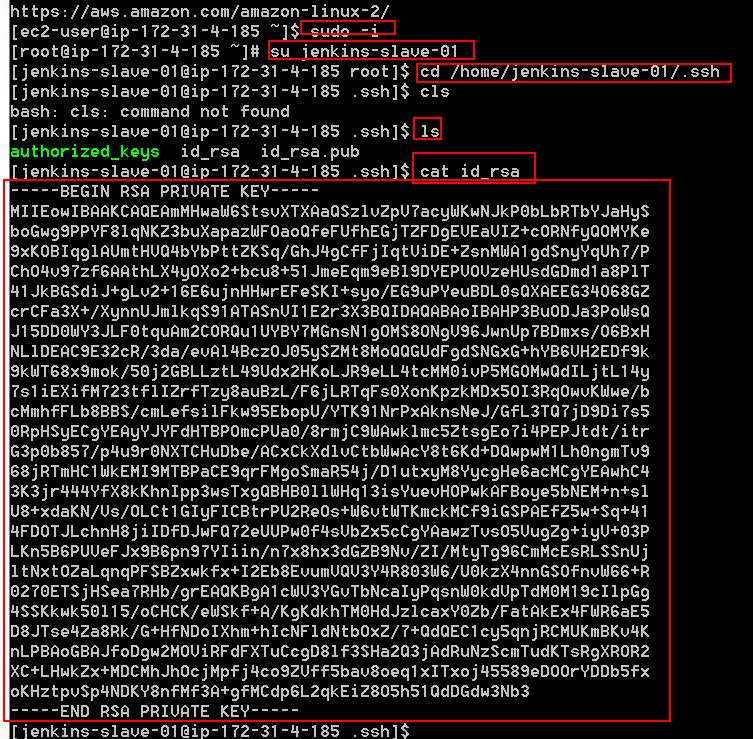


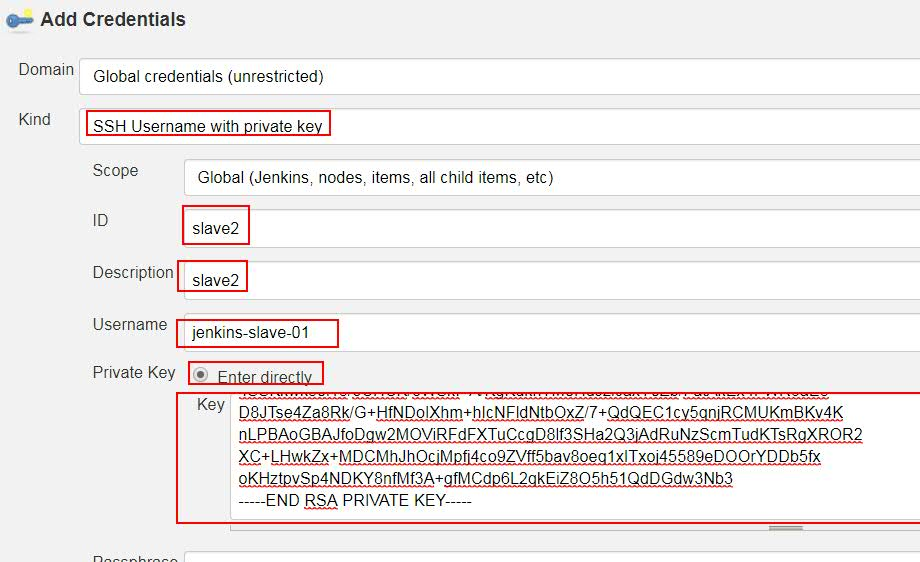
Then select the option as **username with private key**. Give the username and private key and click on Add.

**How to get private Key?** Connect to Save machine. Switch as Jenkins-slave user, then go to the .shh directory(**cd /home/Jenkins-slave/.ssh**)

Then print the id\_rsa key **Ex:** **cat id\_rsa**.

Then copy it and paste it.



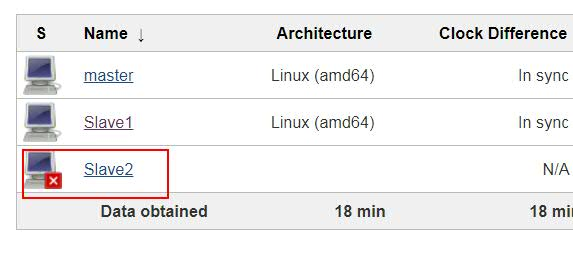


Then click on add.

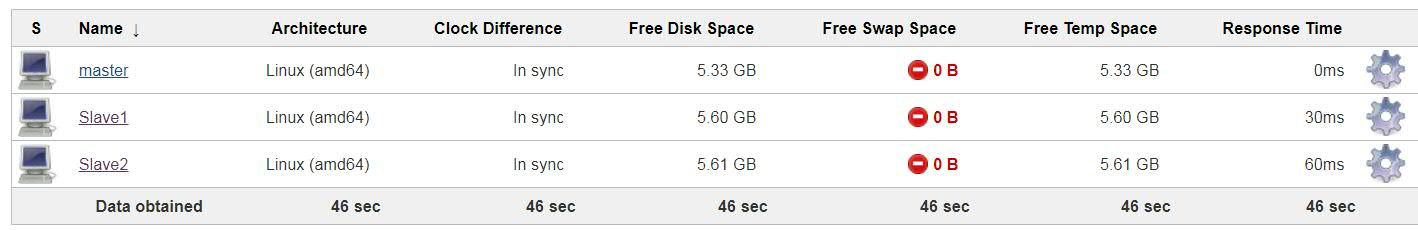
Then select the added credentials.



Then create the node by clicking on **Save**, Then click on the node and then click on Launch agent.



Then you can see the 2 nodes are connected.



If you get any issue with below error message, then it means, the user don’t have permissions to Jenkins home directory where the agent.jar file has to be copied.  
Caused by: com.trilead.ssh2.SFTPException: Permission denied (SSH\_FX\_PERMISSION\_DENIED: The user does not have sufficient permissions to perform the operation

**For that set permissions using below command:**

**sudo chmod -R 777 /home/Jenkins**

**Steps to install Tomcat on Linux RedHat:**

**# yum install tomcat tomcat-webapps tomcat-admin-webapps**

**# systemctl start tomcat**

**# systemctl enable tomcat**

**# vi /usr/share/tomcat/conf/tomcat-users.xml**

**Add below text:**

**<user username="admin" password="passw0rd" roles="admin-gui,manager-gui,manager-script"/>**

**<user username="admin1" password="passw0rd" roles="admin-gui"/>**

**<user username="admin2" password="passw0rd" roles="admin-gui,manager-gui"/>**

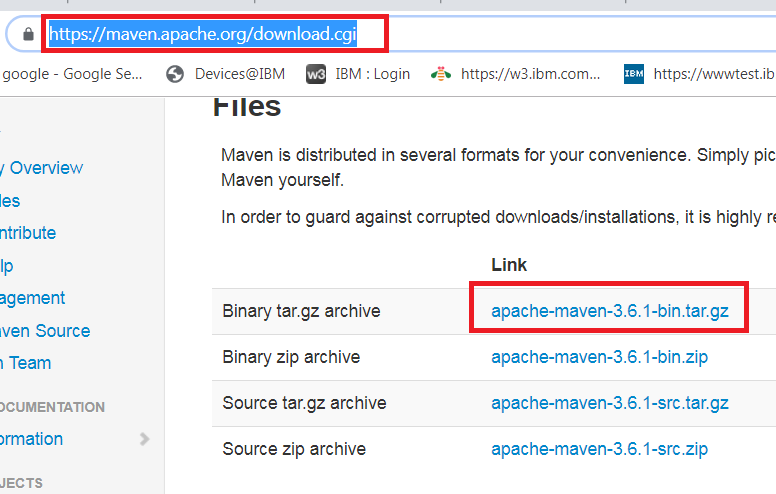
**Then change the port number in server.xm if required.**

**vi /usr/share/tomcat/conf/server.xml**

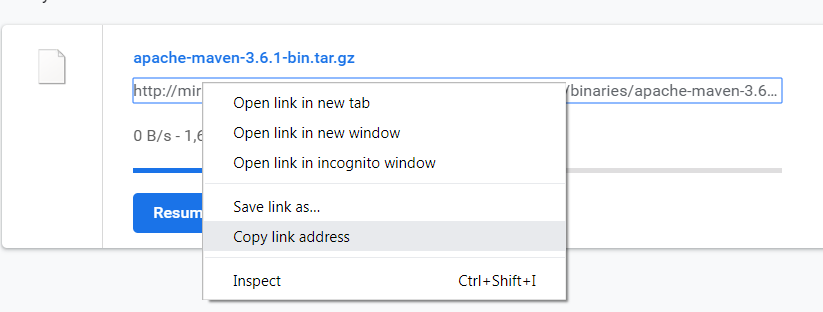
**systemctl restart tomcat**

**Steps to install Maven on Linux RedHat:**

**Go to Maven site(**<https://maven.apache.org/download.cgi>**) and click on tar/zip file to download.**

****

**Then it starts downloading, then copy the download url.**

****

**Then connect to your linux machine and go to /opt directory where we normally keeps out downloads.  
# cd /opt**

**Then execute below command to download on Linux:**

**Then use wget command and the copied url to download maven on Linux. Example is like below.**

**# wget** [**http://mirrors.estointernet.in/apache/maven/maven-3/3.6.1/binaries/apache-maven-3.6.1-bin.tar.gz**](http://mirrors.estointernet.in/apache/maven/maven-3/3.6.1/binaries/apache-maven-3.6.1-bin.tar.gz)

**This will download a tar file and we have to extract it.**

**Ex: # tar -xvf apache-maven-3.6.1-bin.tar.gz**

**Then get in to maven bin directory:**

**# cd apache-maven-3.6.1/bin**

**Then set the maven home. For that first create M2\_HOME environmental variable:**

**# export M2\_HOME=/opt/apache-maven-3.6.1/bin**

**Then set M2\_HOME variable to PATH.**

**# export PATH=$PATH:$M2\_HOME**

**Then check Maven version**

**# mvn -version**

**Steps to install Git on Linux RedHat:**

**# yum install git -y**

**Then check version**

**# git --version**