Prerequisites:

1. Java

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

JDK : jdk-8u121

1. rpm –Uvh jdk-8u121.rpm
2. alternatives –install /usr/bin/java java /usr/java/latest/bin/java 200000
3. alternatives –install /usr/bin/javac javac /usr/java/latest/bin/javac 200000
4. alternatives –install /usr/bin/jar jar /usr/java/latest/bin/jar 200000
5. vi /etc/rc.local (specify java home)----export = “usr/java/latest”

**Jenkins**

**Installation:**

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 install -y jenkins-2.19.4-1.1 (stable version/ Certification version)
4. yum install yum-utils && yum-config-manager --disable jenkins (Avoid jenkins update)
5. Check port 8080 opened or not
   1. netstat -tulpn | grep 8080
6. start Jenkins
   1. systemctl start jenkins
   2. watch n=1 “netstat –tulpn | grep 8080”
7. systemctl enable jenkins
8. Now login in browser with initial admin password

**UserManagement and Security:**

1. Manage jenkins > Configure Global Security> Authoriazation (Matrix based security)
2. ---------------------🡪 Manage Users > Add user

**Adding Jenkin Slave:**

1. Adding slave server of jenkin’s to avoid load on master server by using plugins

**Prerequisites**

Spin up the node (cloud instance, or hardware).

Install and configure Java as described in the Install Prerequisite section.

Generate an RSA key pair for the Jenkins user on the Jenkins Master.

**JAVA SETUP ON SLAVE**

1. Go to: http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
2. Download the appropriate java jdk version from the Oracle website. For this course we use jdk-8u121.
3. Copy the package from your local environment to the target server.

Below is an example using scp:

1. scp jdk-8u121-linux-x64.rpm user@your-server:/home/user/
2. Install the jdk package.
3. rpm -Uvh jdk-8u121-linux-x64.rpm
4. Setup Alternatives for Java:

alternatives --install /usr/bin/java java /usr/java/latest/bin/java 200000

alternatives --install /usr/bin/javac javac /usr/java/latest/bin/javac 200000

alternatives --install /usr/bin/jar jar /usr/java/latest/bin/jar 200000

Note: Check out Terry’s nugget on this for more detail:

1. “Setting local and global java environment variables”

Set JAVA\_HOME environmental variable in rc.local.

vi /etc/rc.local

export JAVA\_HOME=”/usr/java/latest”

1. Jenkins Slave Setup

FROM THE TARGET SLAVE NODE’S CONSOLE

Switch to the “root” user.

sudo su

Add a jenkins user with the home “/var/lib/jenkins”.

useradd -d /var/lib/jenkins jenkins

1. FROM THE JENKINS MASTER

Copy the id\_rsa.pub key from the Jenkins user on the master.

cat /var/lib/jenkins/.ssh/id\_rsa.pub

1. FROM THE TARGET SLAVE NODE’S CONSOLE

Create an authorized\_keys file for the Jenkins user.

mkdir /var/lib/jenkins/.ssh

vi /var/lib/jenkins/.ssh/authorized\_keys

Paste the key from the Jenkins master into the file vim. Save with “:wq”.

1. FROM THE JENKINS DASHBOARD

Click “Manage Jenkins” from the left panel.

Click “Manage Nodes.”

Click “Add Node.”

Set a name for your node (e.g. “Slave 1”).

Select “Permanent Node.”

Set “Remote root directory” to ‘/var/lib/jenkins.’

Set “Usage” to “Use this node as much as possible.”

Set “Launch Method” to “Launch slave agents via SSH.”

Set “Host” to your nodes fqdn (e.g. brandon4232.mylabserver.com).

Select “Add” under “Credentials.”

Set “Kind” to “SSH Username with private key.”

Set “Username” to “jenkins.”

Set “Private key” to “From the Jenkins Master.”

Click “Add.”

Choose the new credential from the “Credentials” dropdown.

Click “Save.”

The agent should now be available for use.