Configuring Jenkins server

Provision a Ec2-instance

```
1)sudo yum update -y

2)sudo yum install java-1.8.0-openjdk -y

***set permanent path for java
JAVA_HOME= JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-0.el7_7.x86_64

PATH=$PATH:$JAVA_HOME:$HOME/bin

export PATH

4)_sudo yum install git -y
5)yum install wget -y
```

Installing Jenkins

```
 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
4) sudo systemctl start jenkins.service
5) sudo systemctl enable jenkins.service
```

Integrating Jenkins with docker as a slave node to create Docker container and Push it onto ECR

Provision AWS ec2 instance

Configuring docker on (centos7)

```
1)sudo yum update -y
2)sudo yum install java-1.8.0-openjdk -y
```

To check java homepath

```
sudo alternatives --config java
```

```
***set permanent path for java
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-1.el7_6.x86_64
```

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

export PATH

source ~/.bash_profile

sudo yum install -y yum-utils device-mapper-persistent-data lvm2

sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

sudo yum install docker-ce -y
```

Add your user to the docker group with the following command.

```
$ sudo usermod -aG docker $(whoami)
```

Set Docker to start automatically at boot time:

```
$ sudo systemctl enable docker.service
```

Finally, start the Docker service:

```
$ sudo systemctl start docker.service
```

From root execute below commands

```
useradd dockeradmin
passwd dockeradmin
usermod -aG docker dockeradmin
```

now go to Jenkins configure Jenkins→ manage nodes and cloud –> (on the left)new-node
now, it will open a window where
newnode: give any name based on the requirement(in mycase docker)
click perminant agent
click next

now

Name: give any name
Description: write some description for the node
# of executers: give a number(I choose 2)
Remote root directory : (I have created a user for docker which is dockeradmin so go to Su – **dockeradmin** if ask for password I have created
. password as well which is **dockeradmin**)

```
Cd ~
Mkdir jenkins
Mkdir docker
Cd docker
```

Pwd(/home/dockeradmin/jenkins/docker). → this is Remote root directory

Now,

Labels: docker

Usage: use this node as much as possible
Launch method: launch agents via ssh
Now,
Host: give the private ip of the docker instance
Credentials: we need to configure the user-id and ssh keys follow the bellow steps
1)login into docker host
2)sudo su –
3)su – dockeradmin
4)cd ~
5) **ssh-keygen -t rsa -b 4096 -C** xxx@gmail.com
6)cd .ssh
7)cat id_rsa.pub > authorized_keys
8)chmod 700 authorized_keys
So now we have configure the keys in docker host(slave)

Now we have to configure the jenkins instance to tell jenkins that the slave node is available in this particular IP address(Private) goahead and grab the key so that the masternode can communicate with it

Configuration on jenkins master
1)cd /var/lib/jenkins/.ssh(if no .ssh then go to the location and do ssh-keygen command)
2)ssh-keyscan -H slavenode privateIP or Hostname >>/var/lib/jenkins/.ssh/known_hosts
3)chown jenkins:jenkins known_hosts
4)chmod 700 known_hosts

Now again go to jenkins console
Manage jenkins →configure credentials →on the left side click credentials -> click global credentials → add credentials

Kind: ssh username with private key
Scope: global

ID: give any name

Description: give some description

Username: dockeradmin(since I have created user and generated ssh keys in .ssh of docker admin)

Private Key: for this go to docker admin/.ssh folder and cat id_rsa and copy the content and paste it here

Now **save**

Now, lets get back to node configuration

in Credentials: select the credentials we just saved in global credentials
Host Key Verification Strategy: known hosts file Verification Strategy.
Availability: keep this agent online as much as possible.

Now click save.
Done.

The configuration is done go to status of node and check weather it is connected or not in the status section.