Installation of Jenkins with Docker on Ubuntu

Requirement:-

- A server running on Ubuntu 18.0 with 2 GB of RAM.
- A root password is set up on the server.

Getting Started:-

Steps to follow to install jenkins with docker

Step 1: Update server's repository

For update the server's repository with the latest version

```
sudo apt-get update -y
```

Once the repository has been updated, restart server to apply all these changes.

Step 2: Installation of Docker

a) Download and add Docker CE GPG key:

```
Sudo Wget https://download.docker.com/linux/ubuntu/gpg
Sudo apt-key add gpg
```

b) Add the Docker CE repository to APT:

```
sudo nano /etc/apt/sources.list.d/docker.list
```

And add this line:

```
deb [arch=amd64] https://download.docker.com/linux/ubuntu
xenial stable
```

Save and close the file . then update the repository

```
sudo apt-get update -y
```

c) Install Docker CE using following command:

```
sudo apt-get install docker-ce -y
```

d) Verify the Docker installation using following command :

```
sudo systemctl status docker
```

Step 3: Creation of Docker Volume for Data and Log

To create a data and log volumes to backup Jenkins data and configurations including, logs, plugins, plugin configuration and job config use the following command:

```
sudo docker volume create jenkins-data
sudo docker volume create jenkins-log
```

For list the volume use the following command:

```
sudo docker volume ls
```

Step4: Install Jenkins with Docker

a) For create a docker file to pull and build Jenkins images with required settings.

First Make one directory named “docker” and create a docker file in it using following command:

```
mkdir docker
nano docker/dockerfile
```

And add following lines:

```
FROM jenkins/jenkins
LABEL
maintainer="your_email@example.com"
USER root
RUN mkdir /var/log/jenkins
RUN mkdir /var/cache/jenkins
RUN chown -R jenkins:jenkins
/var/log/jenkins
RUN chown -R jenkins:jenkins
/var/cache/jenkins
USER jenkins

ENV JAVA_OPTS="-Xmx8192m"
ENV JENKINS_OPTS="--handlerCountMax=300
--logfile=/var/log/jenkins/jenkins.log
--webroot=/var/cache/jenkins/war
```

Save and close the file. Then, build the Jenkins image with the following command:

```
cd docker
sudo docker build -t <image_name> .
```

Step 5: Run Jenkins Container with Data and Log Volume

For running Jenkins container by specifying data and log volumes as a mount point use following command:

```
sudo docker run -p 8080:8080 -p 50000:50000
--name=jenkins-master --mount source=jenkins-log,
target=/var/log/jenkins --mount
source=jenkins-data,target=/var/jenkins_home -d
<Image_name>
```

Verify the running container with the following command:

```
sudo docker ps
```

Check the jenkins log file whether everything is working fine or not with following command:

```
sudo docker exec jenkins-master tail -f
/var/log/jenkins/jenkins.log
```

It will show the following output:

```
b437ba21655a44cda66a75b8fbddf5b8

This may also be found at:
/var/jenkins_home/secrets/initialAdminPassword

*************************************************************
*************************************************************
*************************************************************

2021-03-16 06:42:50.393+0000 [id=38]  INFO
jenkins.InitReactorRunner$1#onAttained: Completed
initialization
2021-03-16 06:42:50.400+0000 [id=22]  INFO
hudson.WebAppMain$3#run: Jenkins is fully up and running
2021-03-16 06:42:50.577+0000 [id=56]  INFO
h.m.DownloadService$Downloadable#load: Obtained the updated
data file for hudson.tasks.Maven.MavenInstaller
2021-03-16 06:42:50.577+0000 [id=56]  INFO
hudson.util.Retrier#start: Performed the action check
updates server successfully at the attempt #1
2021-03-16 06:42:50.578+0000 [id=56]  INFO
hudson.model.AsyncPeriodicWork#lambda$doRun$0: Finished
Download metadata. 7,187 ms
```

Note:- if output is look like following:

```
*************************************************************
*************************************************************
*************************************************************

2021-03-16 06:42:50.393+0000 [id=38]  INFO
jenkins.InitReactorRunner$1#onAttained: Completed initialization
2021-03-16 06:42:50.400+0000 [id=22]  INFO  hudson.WebAppMain$3#run:
Jenkins is fully up and running
2021-03-16 06:42:50.577+0000 [id=56]  INFO
h.m.DownloadService$Downloadable#load: Obtained the updated data file
for hudson.tasks.Maven.MavenInstaller
2021-03-16 06:42:50.577+0000 [id=56]  INFO  hudson.util.Retrier#start:
Performed the action check updates server successfully at the attempt
#1
2021-03-16 06:42:50.578+0000 [id=56]  INFO
hudson.model.AsyncPeriodicWork#lambda$doRun$0: Finished Download
metadata. 7,187 ms
```

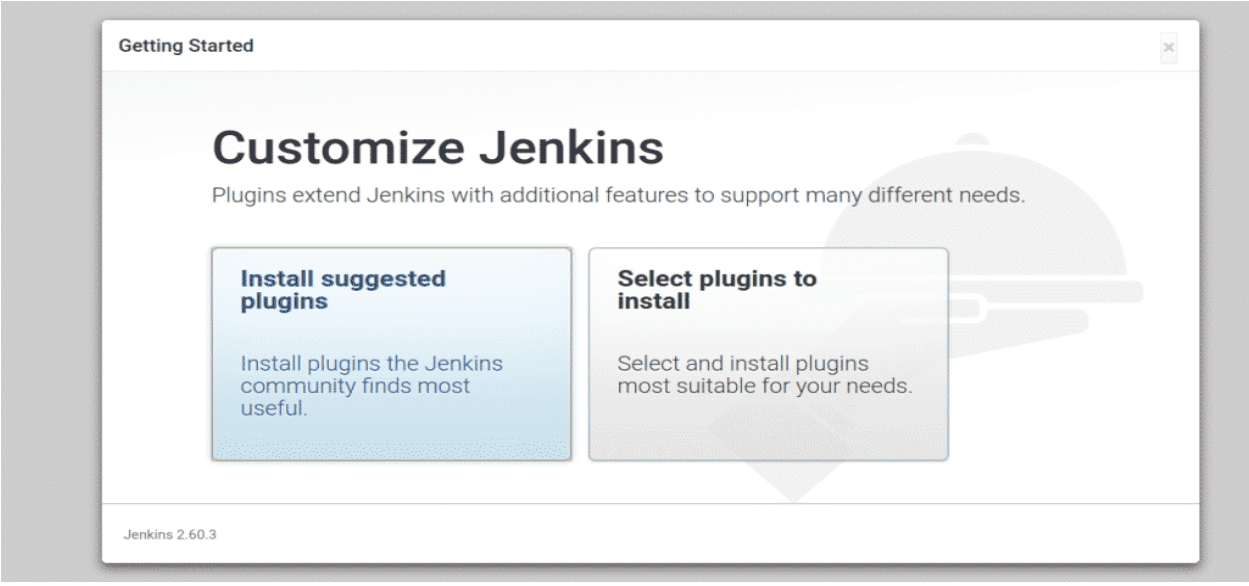
Then initial password can be find at the following location:

```
sudo docker exec -it <Container_id>
bash
cd var/log/jenkins
cat jenkins.log
```

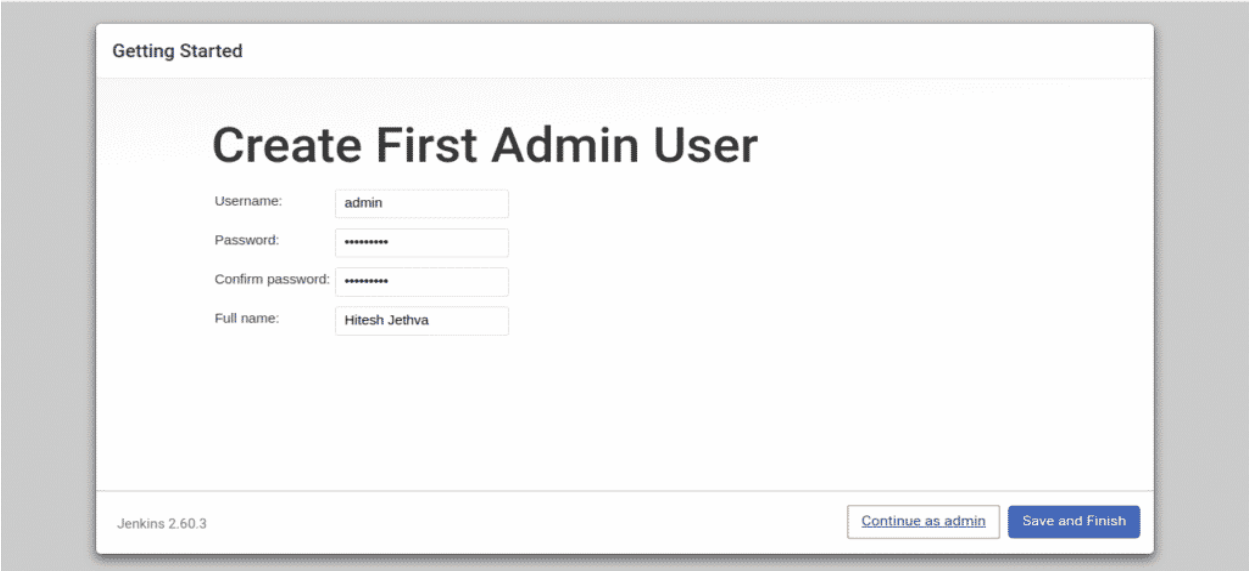
Step 6: Access Jenkins Web Interface

Open web browser and type the URL **http://your-server-ip:8080**.
It will redirected to the Jenkins setup screen as shown below:

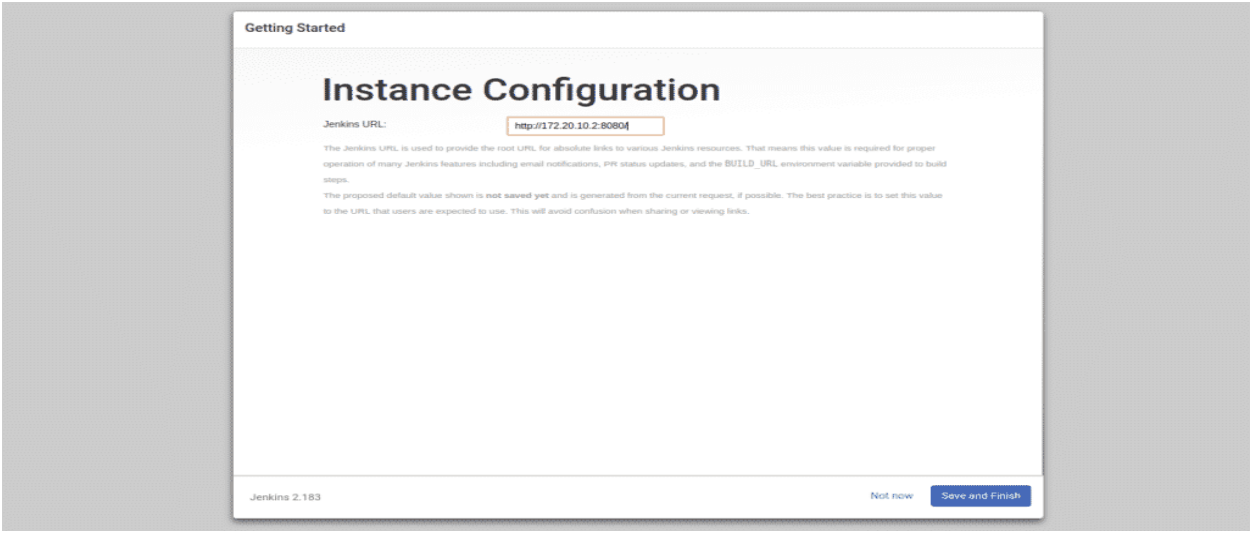
Provide administrator password and click on the **Continue** button. it should show the following page:



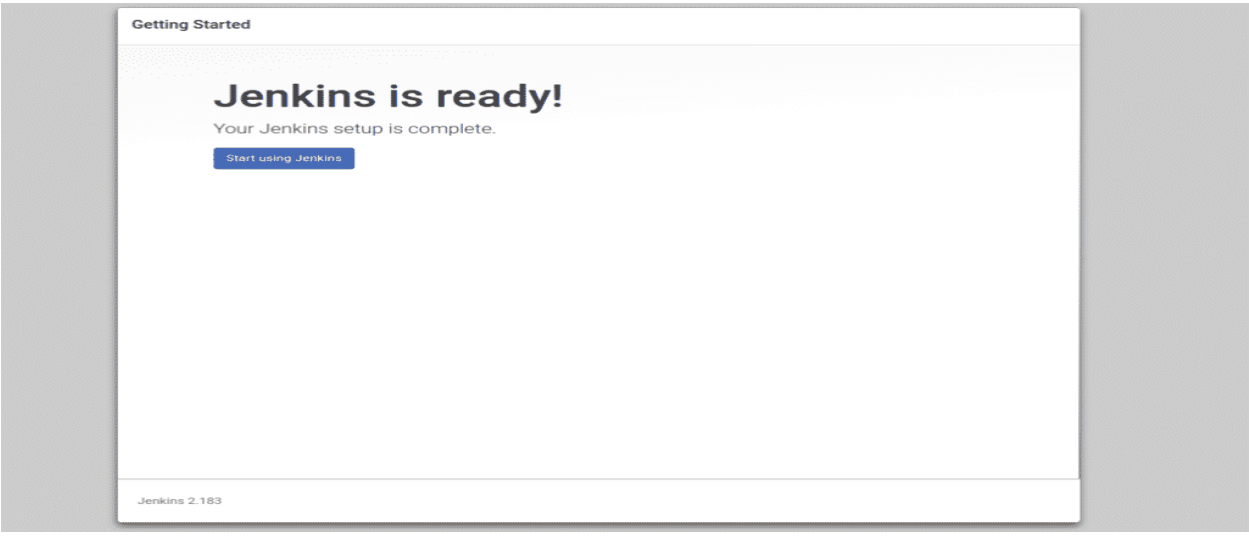
Now, click on the "**Install suggested plugins**" to install the required plugins. Once the installation has been finished. following page will appear:



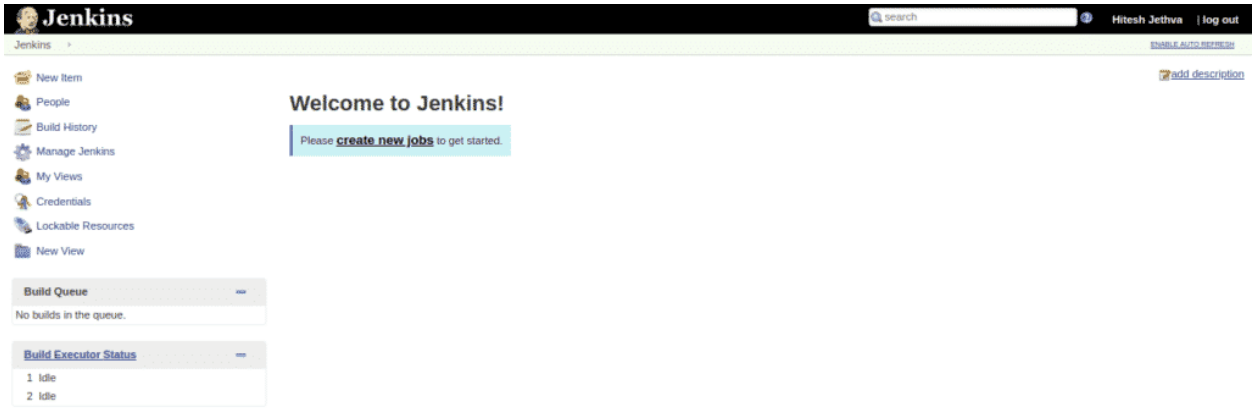
Now, provide admin username, password and full name , password then click on the **Save and Finish** button. You should see the following page:



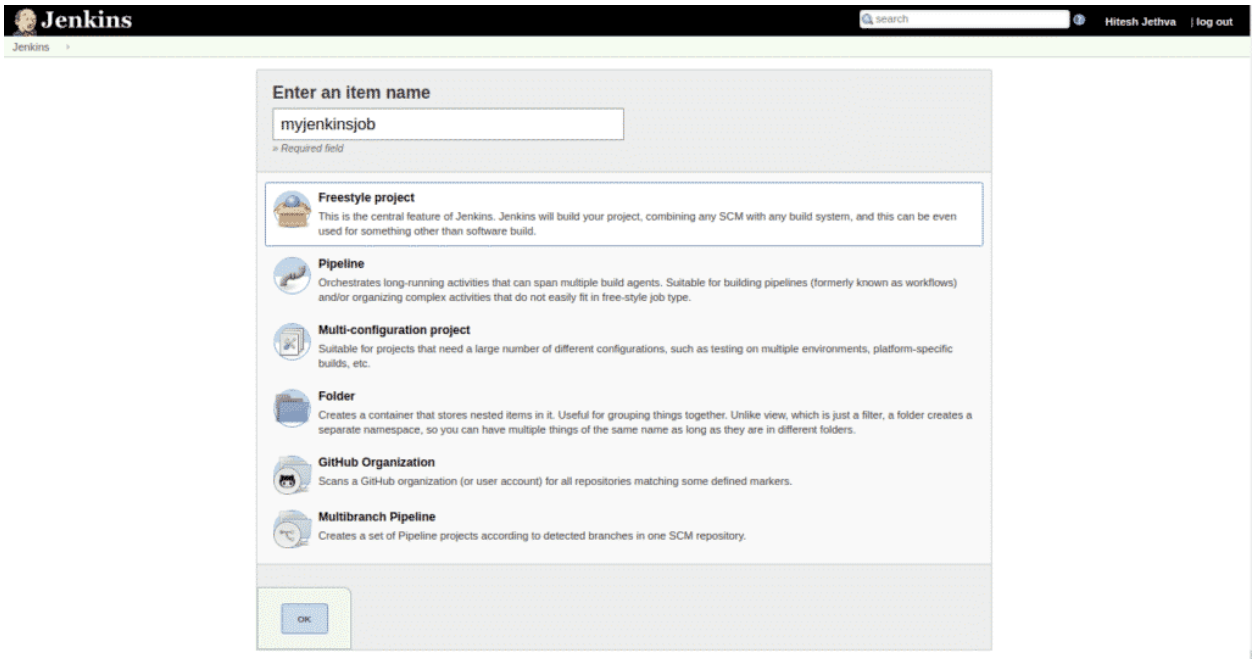
Now, just click on the **Save and Finish** button. Once the setup completed successfully, you should see the following page:



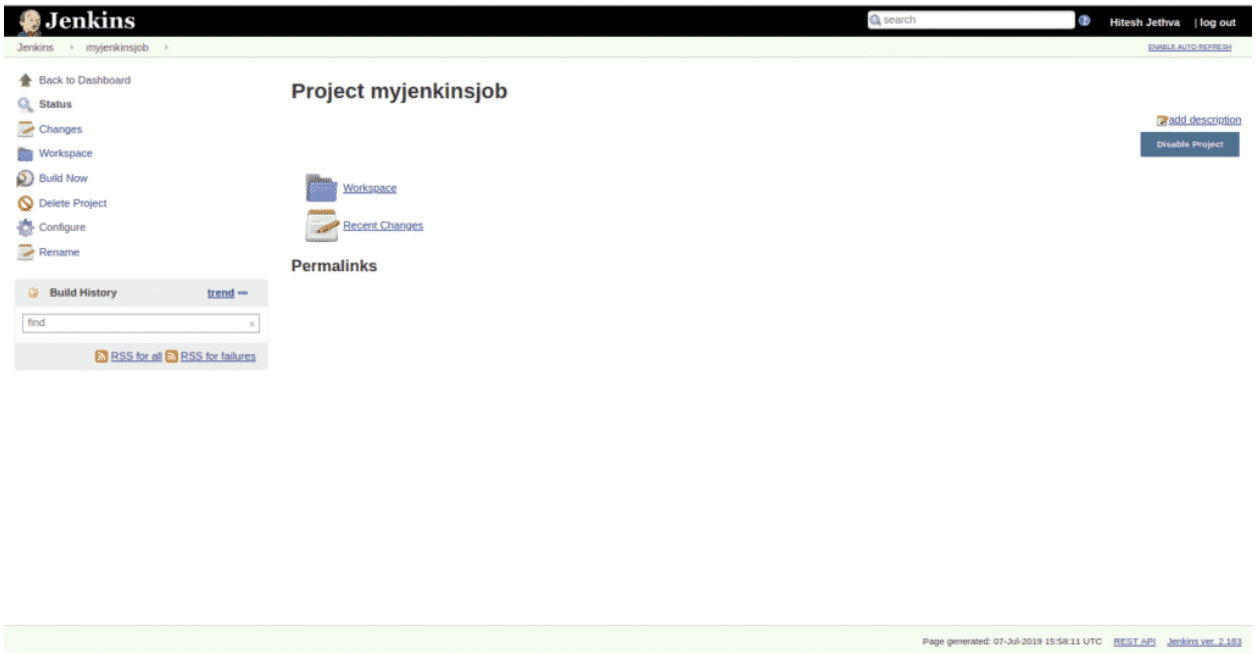
Now, click on the "**Start using Jenkins**". it will redirected to the Jenkins dashboard as shown in the following page:



Now, click on the “**create new jobs**” button,it should show the following page::



Now, provide your job name and click on the **OK** button. You should see the following page:



Step 7: Test Jenkins Persistent Data

For testing whether Jenkins data and log are still persisting after removing the Jenkins container.

First stop and delete the Jenkins container with the following command:

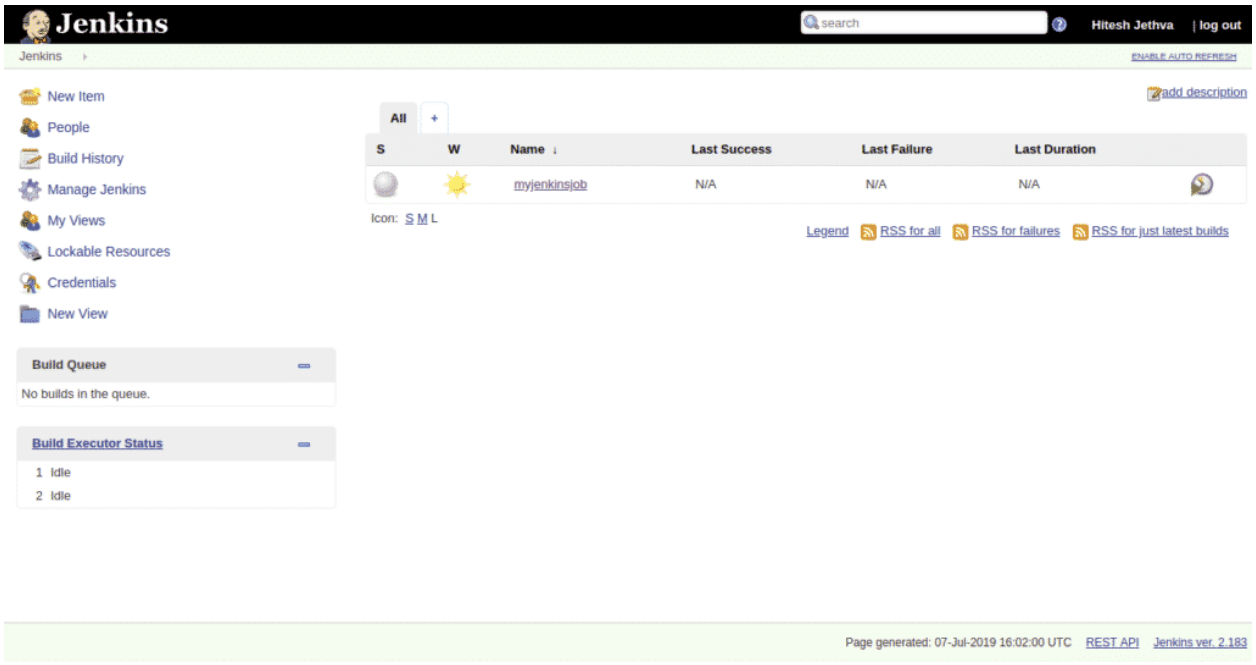
```
sudo docker stop jenkins-master
sudo docker rm jenkins-master
```

Now, start the Jenkins container again with the following command:

```
sudo docker run -p 8080:8080 -p 50000:50000
--name=jenkins-master --mount source=jenkins-log,
target=/var/log/jenkins
--mount source=jenkins-data,target=/var/jenkins_home -d
<image_name>
```

Once the Jenkins container has been started, open your web browser and type the URL **http://your-server-ip:8080**. You will be redirected to the following page:

Now, provide admin user name and password then click on the **Sign in** button. You should see the Jenkins dashboard in the following page:



That means it preserved all the data, logs, setup configuration and plugin installs.Already created job is still there.