How to install Jenkins in Centos 7 / Amazon Linux

==========

In this session, we are going to discuss how to install Jenkins in Centos 7.

Step 1: First we have to install java using below command

```
[root@ip-172-31-89-37 ~]# yum install  yum install java-11-openjdk -y
```

Output:

Step 1.1: Check whether java is installed or not using the below command.

```
[root@ip-172-31-89-37 ~]# sudo java -version
```

Output:

```
[root@ip-172-31-17-103 ~]# sudo java -version
openjdk version "11.0.13" 2021-10-19 LTS
OpenJDK Runtime Environment 18.9 (build 11.0.13+8-LTS)
OpenJDK 64-Bit Server VM 18.9 (build 11.0.13+8-LTS, mixed mode, sharing)
[root@ip-172-31-17-103 ~]#
```

Step 2: Install EPEL Repository

So first, We have to enable the epel repository in CentOS 7/Amazon linux . So using yum command you can install **epel-release** packages.

If you are using Centos 7 to install epel repository use below command

```
[root@ip-172-31-89-37 ~]# sudo yum install epel-release -y
```

Output:

```
Running transaction
  Installing : epel-release-7-11.noarch
  Verifying  : epel-release-7-11.noarch

Installed:
  epel-release.noarch 0:7-11

Complete!
```

If you are using Amazon Linux to install epel repository use below command

```
[root@ip-172-31-89-37 ~]# sudo amazon-linux-extras install epel -y
```

Output :

```
Running transaction
  Installing : epel-release-7-11.noarch
  Verifying  : epel-release-7-11.noarch

Installed:
  epel-release.noarch 0:7-11

Complete!
```

Step 3: The next step is to enable the Jenkins repository. To do that, import the GPG key using below curl command:

```
[root@ip-172-31-89-37 ~]# curl --silent --location http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo | sudo tee /etc/yum.repos.d/jenkins.repo
```

Output:

```
[root@ip-172-31-89-37 ~]# curl --silent --location http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo | sudo tee /etc/yum.repos.d/jenkins.repo
[jenkins]
name=Jenkins-stable
baseurl=http://pkg.jenkins.io/redhat-stable
gpgcheck=1
[root@ip-172-31-89-37 ~]#
[root@ip-172-31-89-37 ~]#
[root@ip-172-31-89-37 ~]#
```

Step 4: And add the jenkins repository to your system using below command:

```
[root@ip-172-31-89-37 ~]# sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

Step 5: Once the repository is enabled, install the latest stable version of Jenkins by below command:

```
[root@ip-172-31-89-37 ~]# sudo yum install jenkins -y
```

Output:

```
[root@ip-172-31-89-37 ~]# yum install jenkins -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                   | 3.7 kB  00:00:00
jenkins                                                                      | 2.9 kB  00:00:00
jenkins/primary_db                                                           |  42 kB  00:00:00
244 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package jenkins.noarch 0:2.346.3-1.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==============================================================================================================
 Package                   Arch                    Version                    Repository                Size
==============================================================================================================
Installing:
 jenkins                   noarch                  2.346.3-1.1                jenkins                   87 M

Transaction Summary
==============================================================================================================
Install  1 Package

Total download size: 87 M
Installed size: 87 M
Downloading packages:
jenkins-2.346.3-1.1.noarch.rpm                                               |  87 MB  00:00:10
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : jenkins-2.346.3-1.1.noarch                                                           1/1
  Verifying  : jenkins-2.346.3-1.1.noarch                                                           1/1

Installed:
  jenkins.noarch 0:2.346.3-1.1
```

Step 6: Once jenkins installation is done. Enable the Jenkins service to start on system boot use below command.

```
[root@ip-172-31-89-37 ~]# sudo systemctl enable jenkins
```

Output:

```
[root@ip-172-31-89-37 ~]# sudo systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
```

Step 7: Start the Jenkins service using below command:

```
[root@ip-172-31-89-37 ~]# sudo systemctl start  jenkins
                          (Or)
[root@ip-172-31-89-37 ~]# sudo service jenkins start
```

Step 8 : Check the status of the jenkins service using below command

```
[root@ip-172-31-89-37 ~]# sudo systemctl status  jenkins
                          (Or)
[root@ip-172-31-89-37 ~]# sudo service jenkins status
```

Output:

```
[root@ip-172-31-89-37 ~]# sudo systemctl status  jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2022-08-27 16:37:51 UTC; 14s ago
 Main PID: 17937 (java)
    Tasks: 42
   Memory: 383.6M
   CGroup: /system.slice/jenkins.service
           └─17937 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=%C/jenkins/war --httpPort=8080

Aug 27 16:37:26 ip-172-31-89-37.ec2.internal jenkins[17937]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Aug 27 16:37:26 ip-172-31-89-37.ec2.internal jenkins[17937]: *************************************************************
Aug 27 16:37:26 ip-172-31-89-37.ec2.internal jenkins[17937]: *************************************************************
Aug 27 16:37:26 ip-172-31-89-37.ec2.internal jenkins[17937]: *************************************************************
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal jenkins[17937]: 2022-08-27 16:37:51.166+0000 [id=28]        INFO        jenkins.InitReactorRunne...zation
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal jenkins[17937]: 2022-08-27 16:37:51.195+0000 [id=21]        INFO        hudson.lifecycle.Lifecyc...unning
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal systemd[1]: Started Jenkins Continuous Integration Server.
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal jenkins[17937]: 2022-08-27 16:37:51.300+0000 [id=42]        INFO        h.m.DownloadService$Down...taller
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal jenkins[17937]: 2022-08-27 16:37:51.301+0000 [id=42]        INFO        hudson.util.Retrier#star...mpt #1
Aug 27 16:37:51 ip-172-31-89-37.ec2.internal jenkins[17937]: 2022-08-27 16:37:51.304+0000 [id=42]        INFO        hudson.model.AsyncPeriod...840 ms
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-89-37 ~]#
```

Step 9: If you want to stop the Jenkins service use below command

```
[root@ip-172-31-89-37 ~]# sudo systemctl stop  jenkins
                          (Or)
[root@ip-172-31-89-37 ~]# sudo service jenkins stop
```

Step 10: If you want to restart the service use below command

```
[root@ip-172-31-89-37 ~]# sudo systemctl restart  jenkins
                          (Or)
[root@ip-172-31-89-37 ~]# sudo service jenkins restart
```

Step 11: Enable the port number using below commands

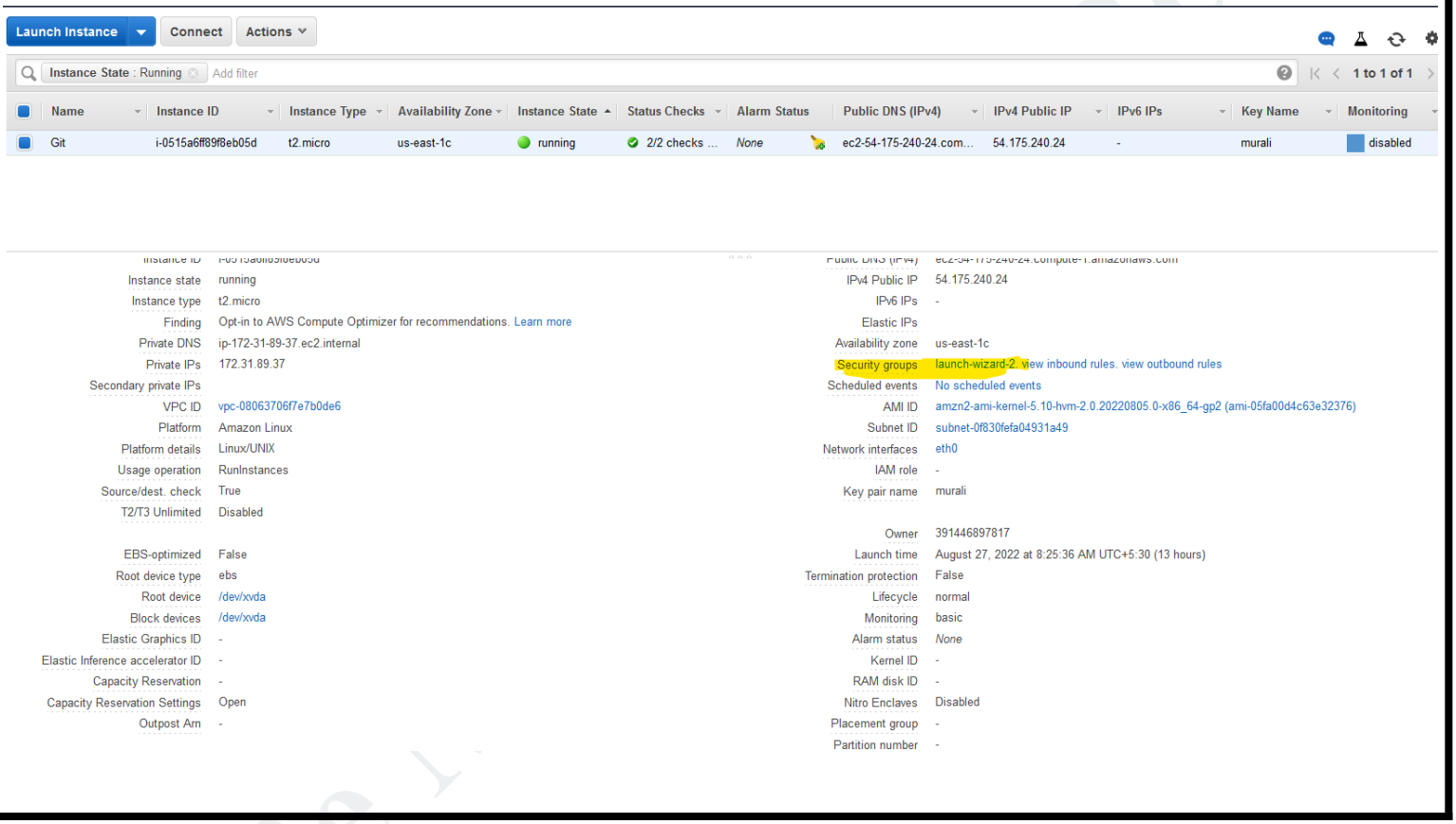
Note:

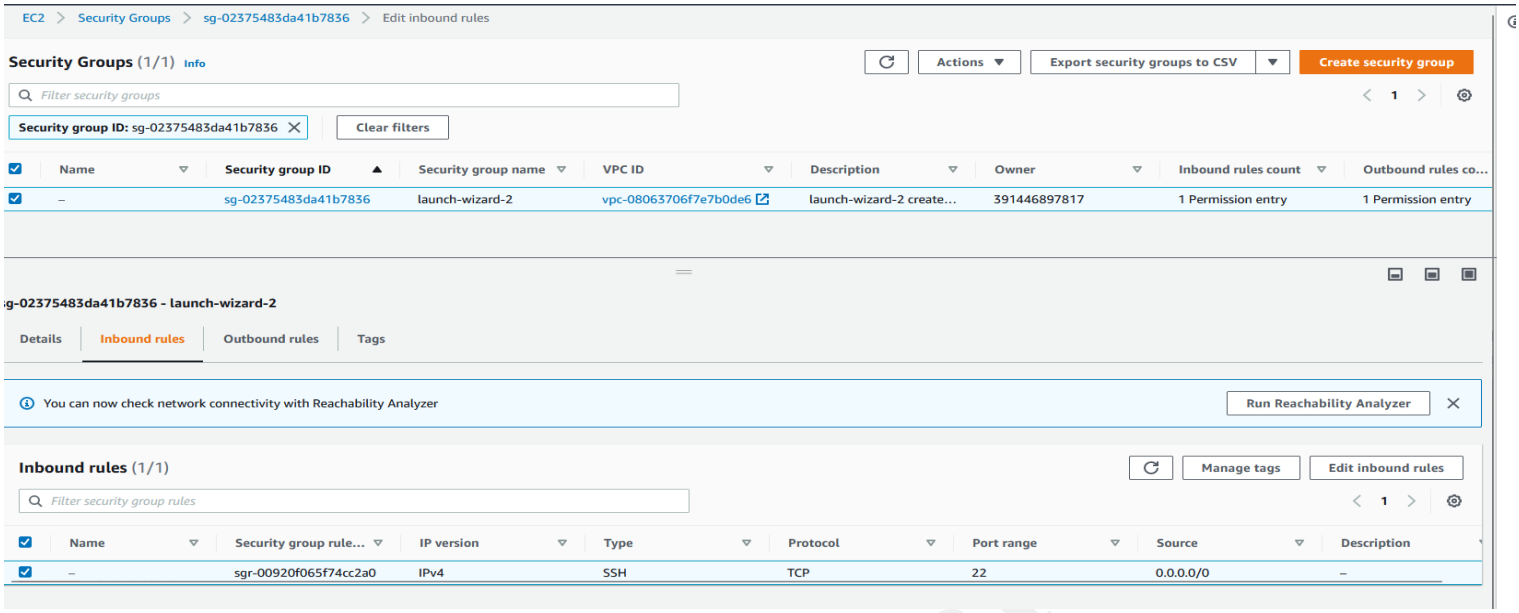
If your using on-premise server use below command to enabled the port number

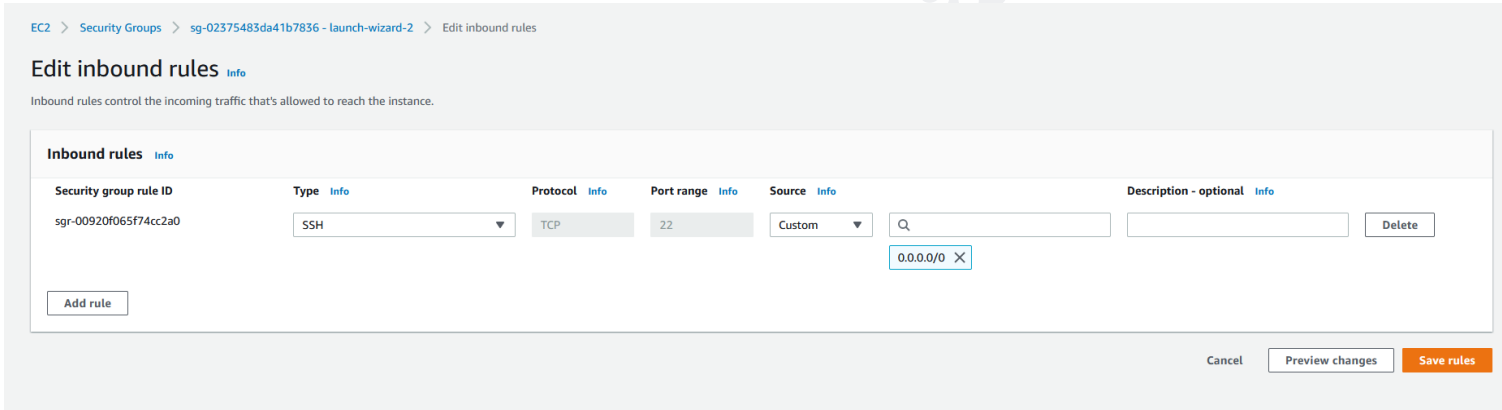
```
[root@ip-172-31-89-37 ~]# sudo firewall-cmd --permanent --zone=public --add-port=8080/tcp
[root@ip-172-31-89-37 ~]# sudo firewall-cmd --reload
```

If you are using an AWS EC2 instance, you have to open the Inbound rules 8080 port number in Security Groups for a particular server.

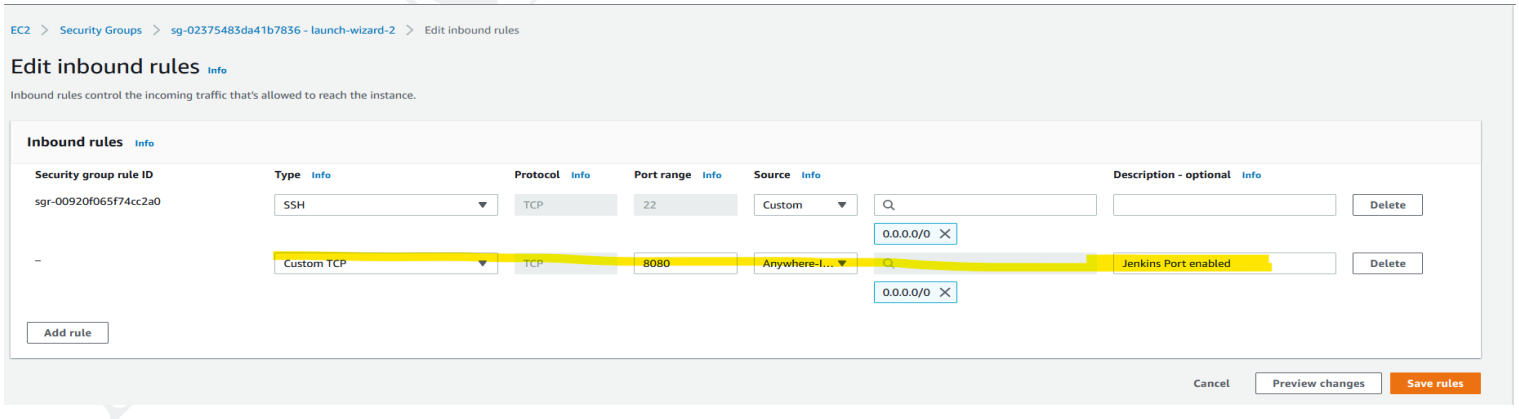
Example:







Add the 8080 port number in the Inbound rule.



Note: **If you are not enabled the 8080 port number , you can't access jenkins GUI.**

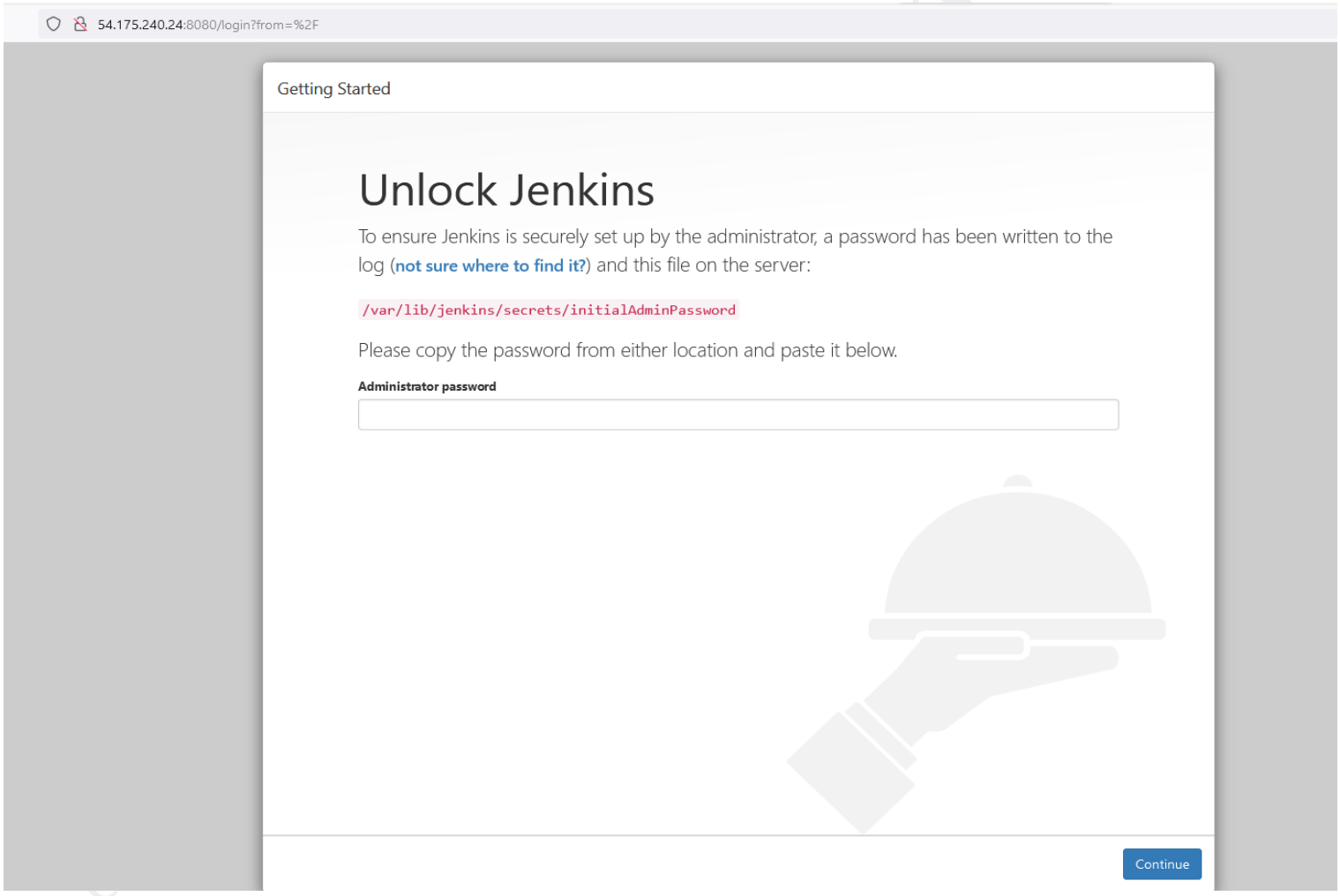
Step 12: To set up your new Jenkins installation, open any browser and type your domain or IP address followed by port 8080:

http://your_ip_or_domain:8080

Example:

http:54.175.240.24:8080

You should see something similar screen to the following will appear, prompting you to enter the Administrator password that is created during the installation:



Step 13: Use the following command to print the password on your terminal

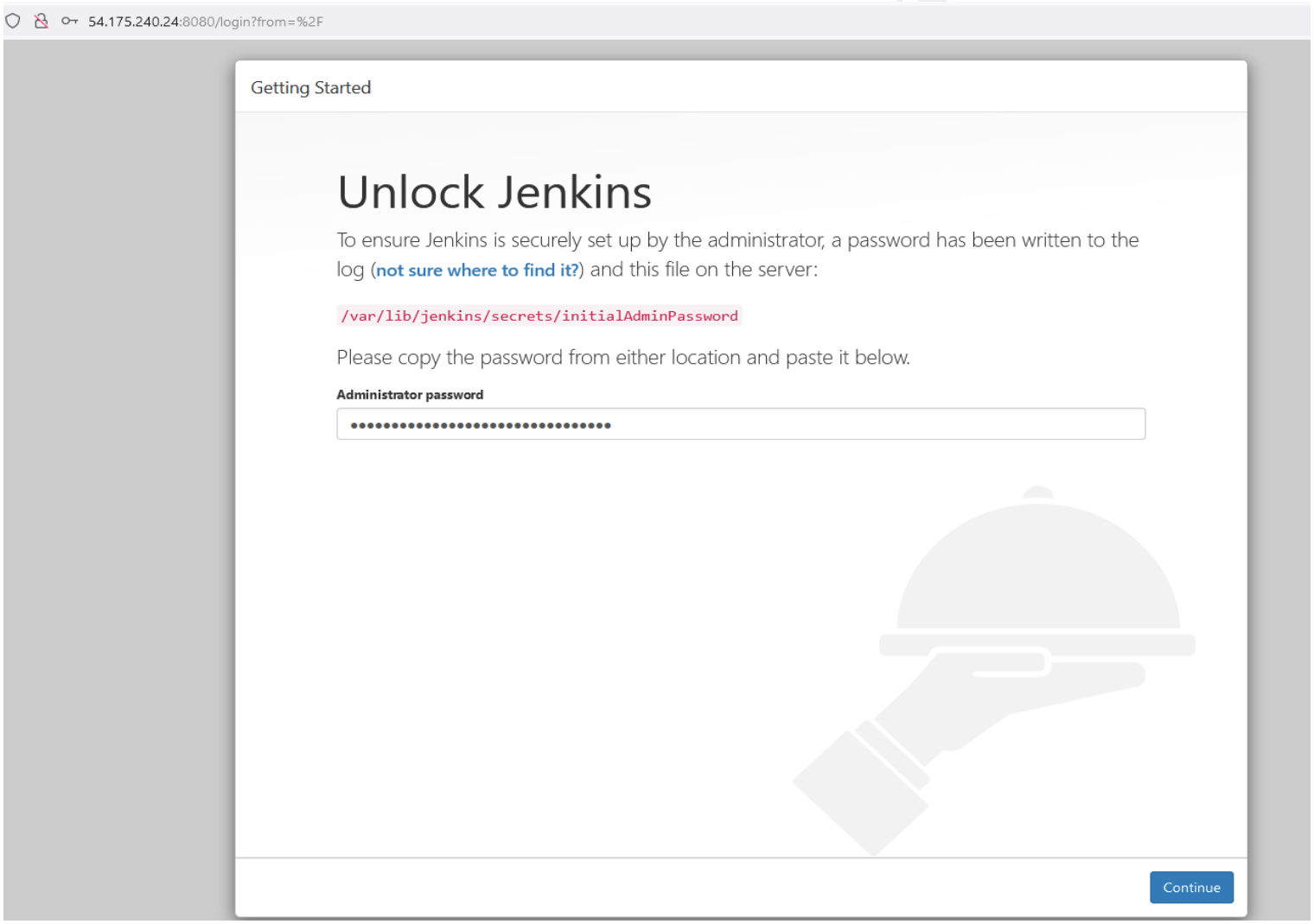
```
[root@ip-172-31-89-37 ~]# sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

You should see a 32-character long alphanumeric password as shown below

Output:

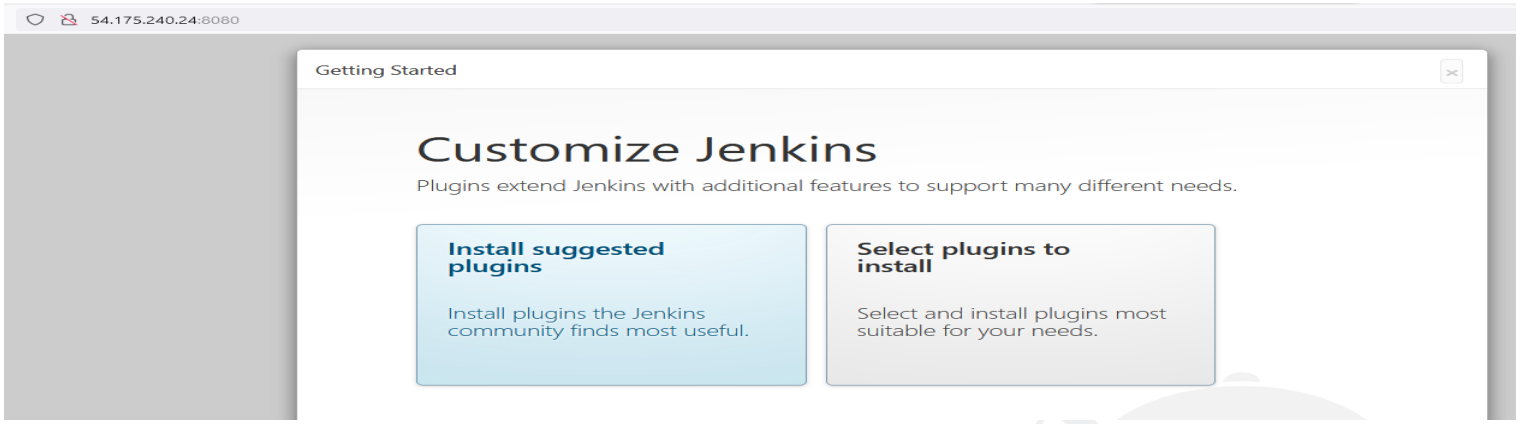
```
[root@ip-172-31-89-37 ~]# sudo cat /var/lib/jenkins/secrets/initialAdminPassword
1a69930df66848bf8bd4785258c0cedc
[root@ip-172-31-89-37 ~]#
```

Step 14: Copy the password from your terminal, paste it into the Administrator password field and click Continue.

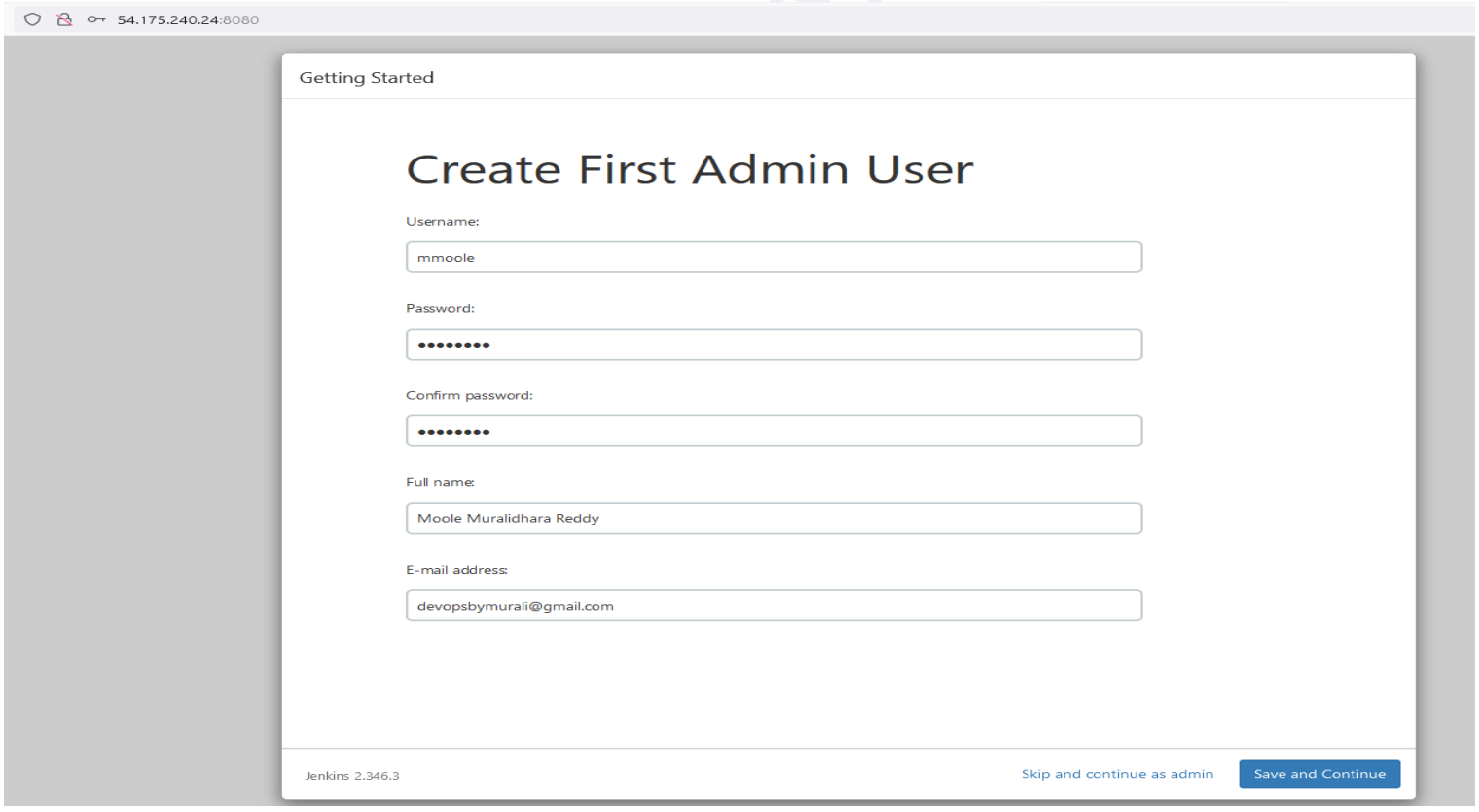


Step 15: In the second stage, you will be presented with 2 options: 'Install using suggested

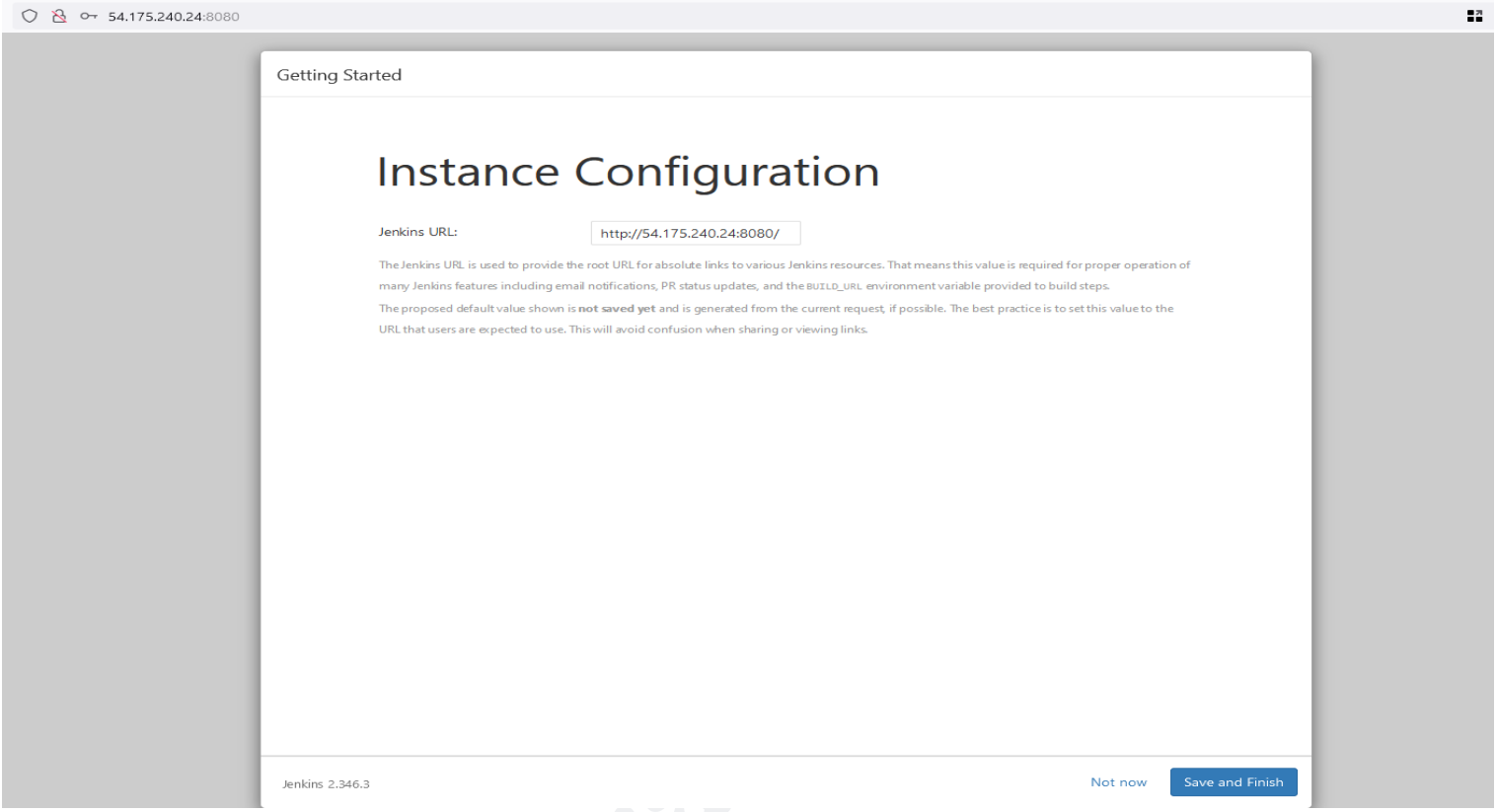
plugins‘ or ‘Select plugins to install‘
For now, click on ‘Install using suggested plugins‘ to install essential plugins for our setup.



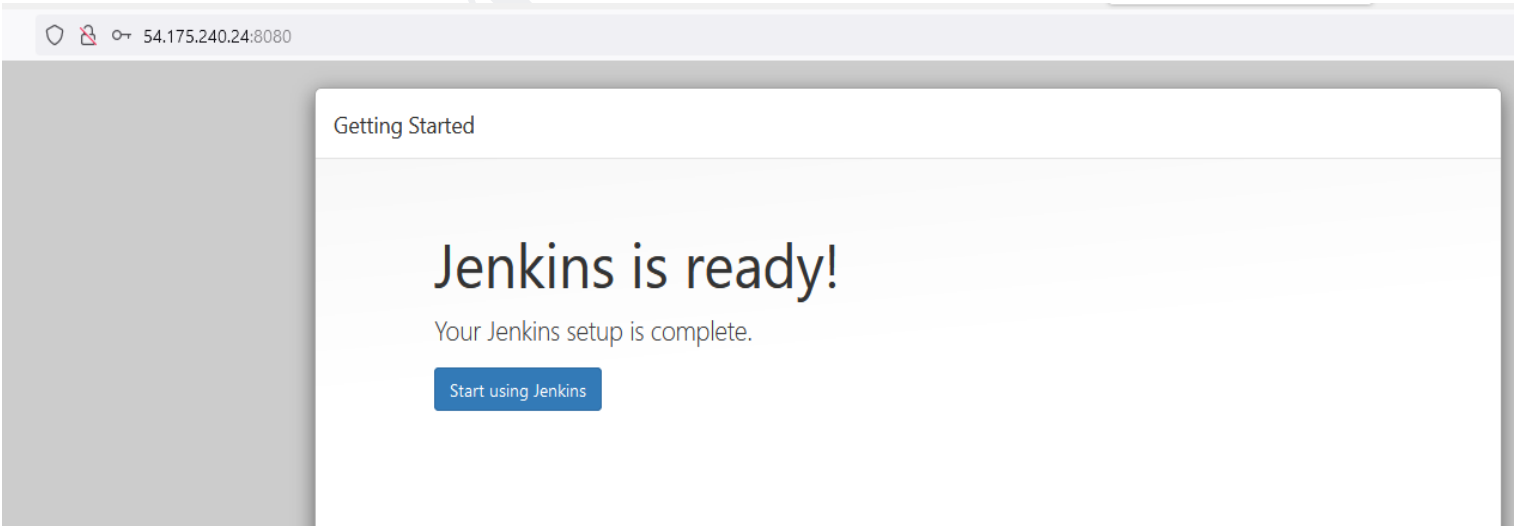
Shortly, the installation of the plugins will get underway.
Step 16: Once the installation is complete, you will be prompted to set up the first administrative user. Fill out all required information and click Save and Continue.



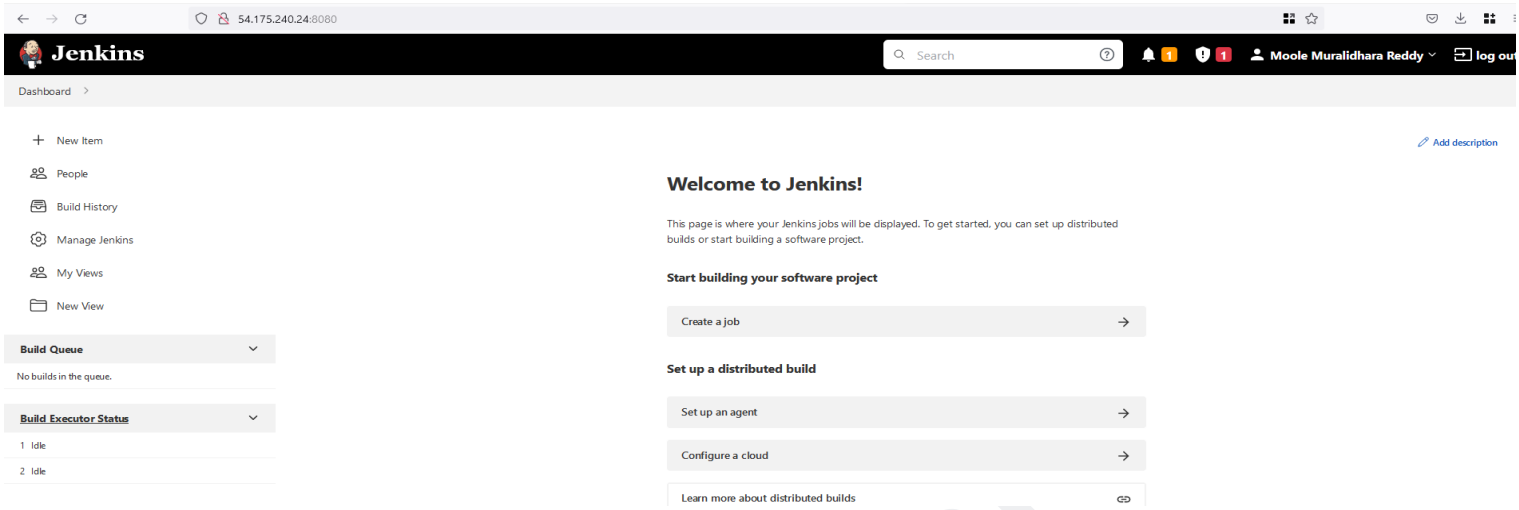
The ‘Instance Configuration‘ section will provide you with the default Jenkins URL. For simplicity, it’s recommended to leave it as it is and click ‘Save and Finish



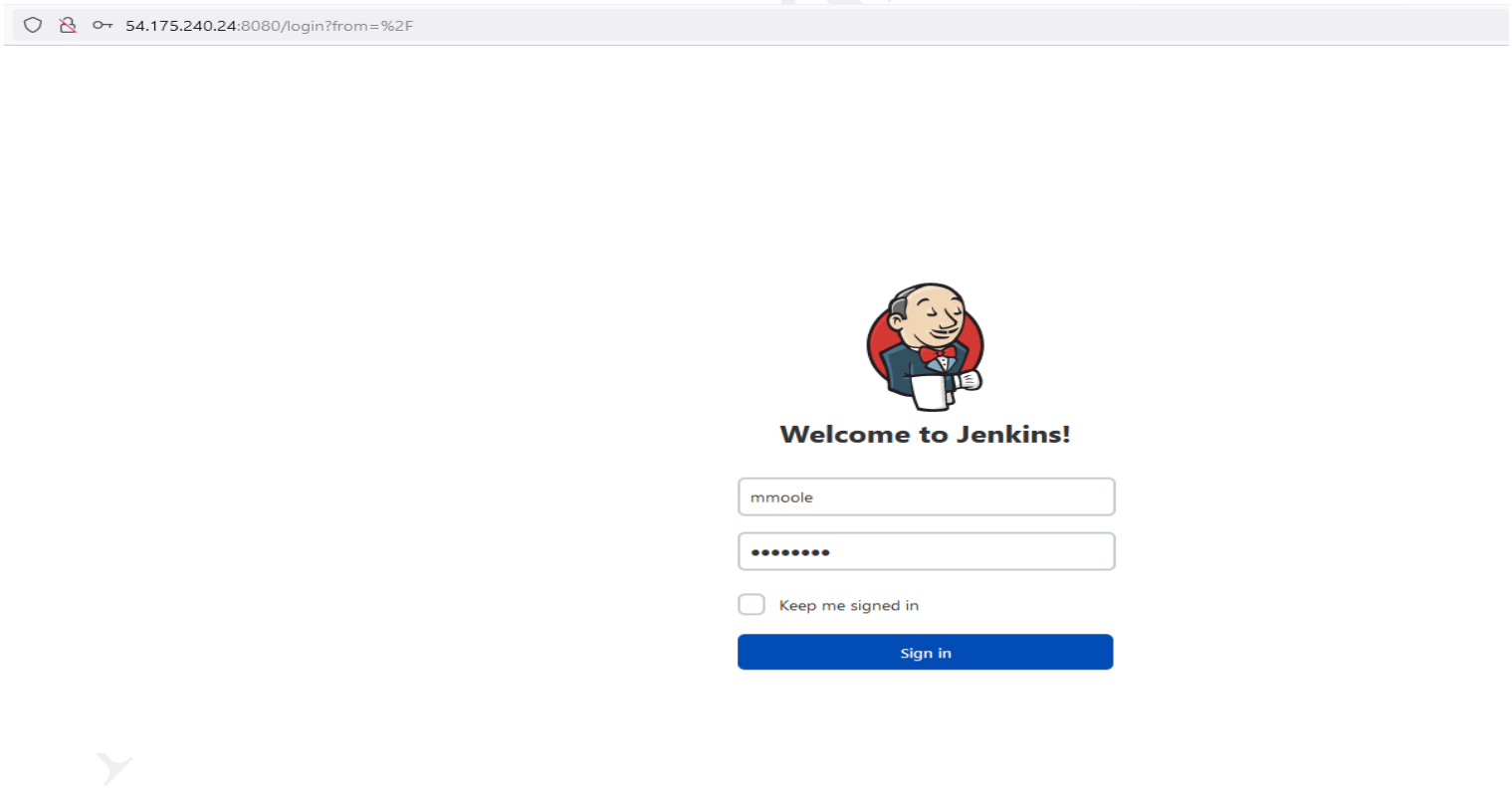
At this point, Jenkins setup is now complete. To access the Jenkins dashboard, simply click on ‘Start using Jenkins‘.



Jenkins's dashboard is displayed below



Next time you log into Jenkins, simply provide the username and password.



Conclusion:

In this tutorial, you have learnt how to install and complete the initial configuration of Jenkins on CentOS/RHEL based systems.

If you have any questions, please leave a comment below.

**** Thank you for watching this Video. We will see you in the next video. ****

About the Author:

MOOLE MURALIDHARA REDDY
Solution Architect / DevOps Consultant

- I am having rich experience in Devops and Cloud technologies and have done many projects on all varieties of tools which are hot cake in the market.
- I am passionate about learning new technology and teaching.
- My courses focus on providing students with an interactive and hands-on experience in learning new technology that makes learning really interesting.
- I have a wide range of experience in Telecom, Banking, Healthcare, Retail domains.
- I have been training people in newer technologies, like DevOps, AWS, Kubernetes, Terraform,Rancher, etc. and they have settled in MNC's and drawing respectable salaries.
- I have undergone many challenges and changed the entire phase of the projects.
- Certified in AWS, Kubernetes , Terraform, Linux and many to go.

Please check out my courses and join me with thousands of others who are learning the latest DevOps and Cloud tools!

Website Url: **https://www.techworldwithmurali.com/**
Linkedin Profile : **https://www.linkedin.com/in/moole-muralidhara-reddy/**
Youtube Channel : **https://www.youtube.com/c/TechWorldwithMurali**