How to Install Jenkins on CentOS 7

Jenkins is a popular open source CI (Continuous Integration) tool which is widely used for project development, deployment, and automation.

This article will guide you through the process of installing Jenkins on a CentOS 7 server instance. In order to facilitate visitors' access, Nginx will also be installed as the reverse proxy for Jenkins.

Prerequisites

Before proceeding, you must have:

- Deployed a CentOS 7 server instance from scratch.
- Logged into your machine as a non-root user with sudo privileges.

Step 1: Update your CentOS 7 system

One of the Linux system administrator's best practices is keeping a system up to date. Install the latest stable packages, then reboot.

```
sudo yum install epel-release

sudo yum update

sudo reboot
```

When the reboot finishes, login with the same sudo user.

Step 2: Install Java

Before you can install Jenkins, you need to setup a Java virtual machine on your system. Here, let's install the latest OpenJDK Runtime Environment 1.8.0 using YUM:

```
sudo yum install java-1.8.0-openjdk.x86_64 -y
```

After the installation, you can confirm it by running the following command:

```
java -version
```

This command will tell you about the Java runtime environment that you have installed:

```
openjdk version "1.8.0_91"

OpenJDK Runtime Environment (build 1.8.0_91-b14)

OpenJDK 64-Bit Server VM (build 25.91-b14, mixed mode)
```

In order to help Java-based applications locate the Java virtual machine properly, you need to set two environment variables: "JAVA_HOME" and "JRE_HOME".

```
sudo cp /etc/profile /etc/profile_backup

echo 'export JAVA_HOME=/usr/lib/jvm/jre-1.8.0-openjdk' | sudo tee -a /etc/profile

echo 'export JRE_HOME=/usr/lib/jvm/jre' | sudo tee -a /etc/profile source /etc/profil
e
```

Finally, you can print them for review:

```
echo $JAVA_HOME

echo $JRE_HOME
```

Step 3: Install Jenkins

Use the official YUM repo to install the latest stable version of Jenkins, which is `1.651.2` at the time of writing:

```
cd ~

sudo yum wget  -- Install the software wget

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

sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

sudo yum install jenkins
```

Start the Jenkins service and set it to run at boot time:

```
sudo systemctl start jenkins.service

sudo systemctl enable jenkins.service
```

In order to allow visitors access to Jenkins, you need to allow inbound traffic on port 8080:

```
sudo firewall-cmd --zone=public --permanent --add-port=8080/tcp

sudo firewall-cmd –reload

Note: -- If you are running Jenkins on AWS EC2 instance, then the above firewall configuration is not required. But make sure the firewall port is configured on Security Groups.
```

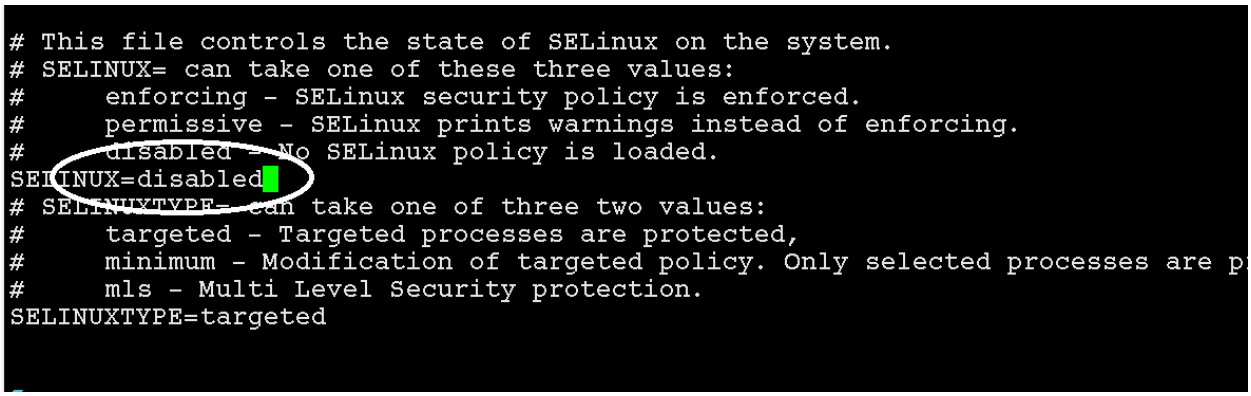
Now, test Jenkins by visiting the following address from your web browser:

```
http://<your-server-IP>:8080
```

If the page does not come up, try to disable the SELINUX.

Edit the selinux config file

vi /etc/selinux/config



save the file -> press 'ecs' and type ":wq" and enter.

REBOOT the linux machine and

Then try to access the jenkins web page.

```
http://<your-server-IP>:8080
```

Step 4: Install Nginx (optional)

In order to facilitate visitors' access to Jenkins, you can setup an Nginx reverse proxy for Jenkins, so visitors will no longer need to key in the port number 8080 when accessing your Jenkins application.

Install Nginx using YUM:

```
sudo yum install nginx
```

Modify the configuration of Nginx:

```
sudo vi /etc/nginx/nginx.conf
```

Find the two lines below:

```
location / {

}
```

Insert the six lines below into the { } segment:

```
proxy_pass http://127.0.0.1:8080;

proxy_redirect off;

proxy_set_header Host $host;

proxy_set_header X-Real-IP $remote_addr;

proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

proxy_set_header X-Forwarded-Proto $scheme;
```

The final result should be:

```
location / {

proxy_pass http://127.0.0.1:8080;

proxy_redirect off;

proxy_set_header Host $host;

proxy_set_header X-Real-IP $remote_addr;

proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

proxy_set_header X-Forwarded-Proto $scheme;

}
```

Save and quit:

```
:wq
```

Start and enable the Nginx service:

```
sudosystemctl start nginx.service

sudosystemctl enable nginx.service
```

Allow traffic on port 80:

```
sudo firewall-cmd --zone=public --permanent --add-service=http

sudo firewall-cmd --reload
```

Finally, visit the following address from your web browser to confirm your installation:

```
http://<your- server-IP>
```