

Penetration Testing Lab Setup: Jenkins

February 2, 2019 By Raj Chandel

Hey! You all know that we have performed so many CTF challenges and we got to know about Jenkins there. So let's know about Jenkins better. For this, we are here with the new challenges which you will face while performing CTF challenges. To do it in an easier way we are here with a new article. So let's do it.

Table of Content

Introduction of Jenkins

Lab setup

- Install Java
- Import the GPG keys
- Add the Jenkins repository
- Install Jenkins
- Setup Jenkins

Jenkins penetration testing

Exploiting Groovy Script

Introduction of Jenkins

Jenkins is an open source automation server written in Java that offers a simple way to set up a continuous CI / CD pipeline. It supports version control tools, including AccuRev, CVS, Subversion, Git, Mercurial, Perforce, TD/OVS, ClearCase, and RTC, and can execute Apache Ant, Apache Maven, and sbt based projects as well as arbitrary shell scripts and Windows batch commands. The creator of Jenkins is Kohsuke Kawaguchi. Jenkins achieves Continuous Integration with the help of plugins. Plugins allow the integration of Various DevOps stages. If you want to integrate a particular tool, you need to install the plugins for that tool. For example Git, Maven 2 project, Amazon EC2, HTML publisher etc.

Lab setup

Install Java

Now we need to install Jenkins and for this, it is mandatory that you are logged in from sudo user or root. Because Jenkins is a Java application, installing Java is the first step. Update the package index and install the OpenJDK Java 8 package using the following commands:

```
sudo apt update
sudo apt install openjdk-8-jdk
```

```
root@ubuntu:~# sudo apt install openjdk-8-jdk
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libgif7 libice-dev libpthread-stubs0-dev libsm-dev
  libx11-6 libx11-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
  libxt-dev openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
```

Import the GPG keys

```
wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add -
```

```
root@ubuntu:~# wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo
apt-key add -
OK
root@ubuntu:~#
```

Install Jenkins

When the key is added, the system returns all right. Next, add the Debian package repository to the source list of the server:

```
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/source
sudo apt update
```

The Jenkins version with the default Ubuntu packages is often behind the project's latest version. You can use project-maintained packages to install Jenkins to take advantage of the latest fixes and features. Now open the kali terminal and install Jenkins from the given link below-

```
sudo apt install jenkins
sudo ufw allow 8080
```

```

root@ubuntu:~# sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/
> /etc/apt/sources.list.d/jenkins.list'
root@ubuntu:~# sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu bionic-security InRelease
Ign:2 http://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Get:4 http://pkg.jenkins.io/debian-stable binary/ Release [2,042 B]
Hit:5 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Get:6 http://pkg.jenkins.io/debian-stable binary/ Release.gpg [181 B]
Get:7 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:8 http://pkg.jenkins.io/debian-stable binary/ Packages [14.2 kB]
Fetched 91.0 kB in 2s (41.6 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
383 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ubuntu:~# sudo apt install jenkins
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  daemon net-tools
The following NEW packages will be installed:
  daemon jenkins net-tools
0 upgraded, 3 newly installed, 0 to remove and 383 not upgraded.
Need to get 75.7 MB of archives.
After this operation, 77.1 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

You can use its status command to check that Jenkins has successfully started.

```
systemctl status jenkins
```

```

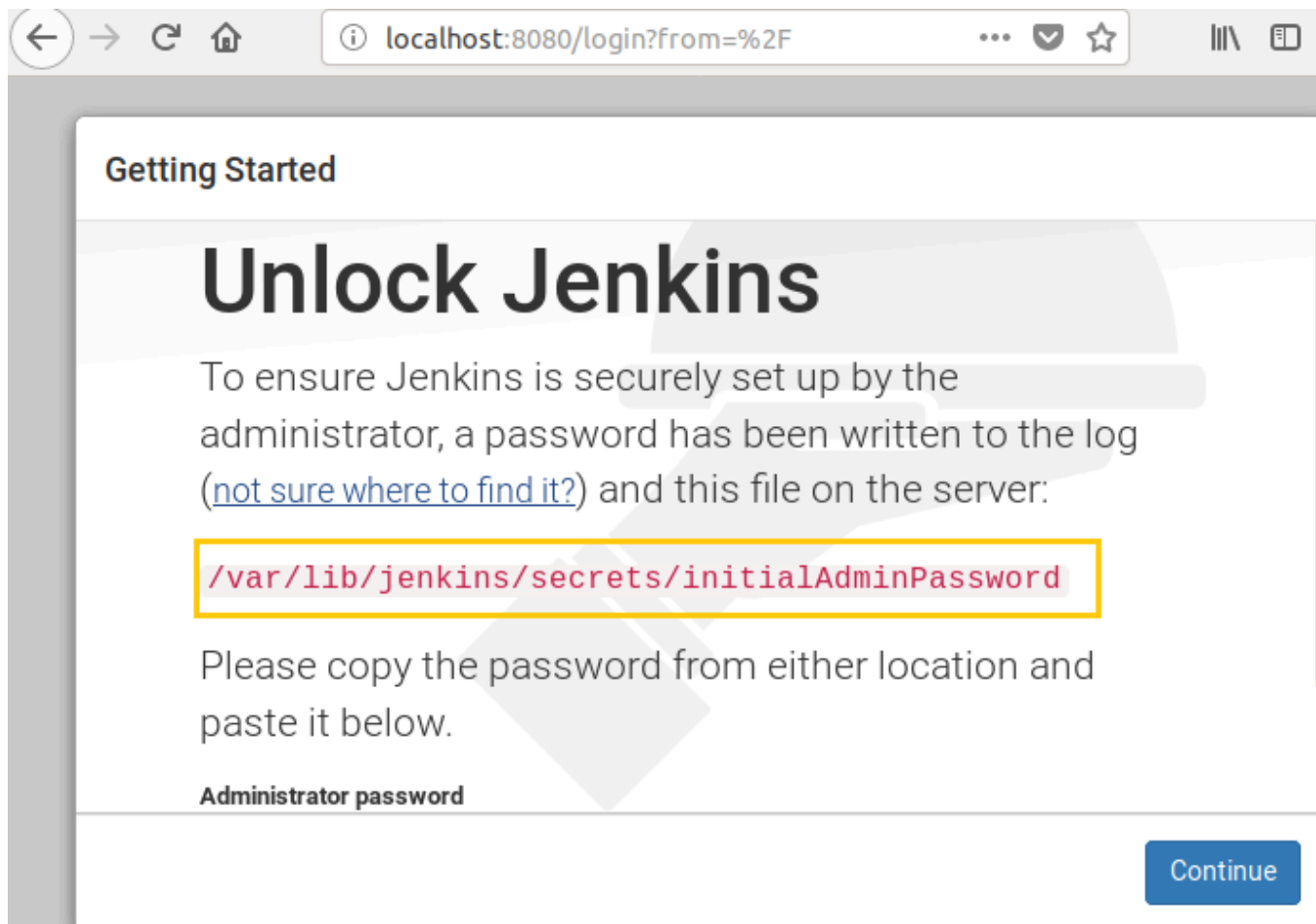
root@ubuntu:~# systemctl status jenkins
● jenkins.service - LSB: Start Jenkins at boot time
   Loaded: loaded (/etc/init.d/jenkins; generated)
   Active: active (exited) since Thu 2019-01-31 07:35:43 PST; 58s ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 0 (limit: 1085)
   CGroup: /system.slice/jenkins.service

Jan 31 07:35:37 ubuntu systemd[1]: Starting LSB: Start Jenkins at boot time..
Jan 31 07:35:38 ubuntu jenkins[9193]: Correct java version found
Jan 31 07:35:38 ubuntu jenkins[9193]: * Starting Jenkins Automation Server j
Jan 31 07:35:39 ubuntu su[9224]: Successful su for jenkins by root
Jan 31 07:35:39 ubuntu su[9224]: + ??? root:jenkins
Jan 31 07:35:39 ubuntu su[9224]: pam_unix(su:session): session opened for use
Jan 31 07:35:43 ubuntu jenkins[9193]: ...done.
Jan 31 07:35:43 ubuntu systemd[1]: Started LSB: Start Jenkins at boot time.

```

Visit Jenkins on its default port 8080 to set up your installation using your server domain name or IP address: *http://your server IP or domain:8080*

You should see the Unlock Jenkins screen displaying the location of the initial password:



In the terminal window, you need to use the cat command to display the password:

Copy the password from your terminal

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

A terminal window screenshot showing a root user at an Ubuntu machine. The command 'cat /var/lib/jenkins/secrets/initialAdminPassword' has been executed. The output is a long alphanumeric string: '7d5d9824bd9c41bbb421b379107ae90b'. This output is highlighted with a yellow rectangular box. A green cursor is visible at the end of the command line.

Copy the password from your terminal and paste it into the Administrator password field and click Continue.

Getting Started

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password



Continue

On the next page, you will be asked if you want to install suggested plugins or if you want to select specific plugins. Click the **Install suggested plugins** box and start the process of installation plugin instantly.

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.



Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

In my case, it took so much time to get all plugin installed successfully.

Getting Started

✖ Folders	🔗 OWASP Markup Formatter	🔗 Build Timeout	🔗 Credentials Binding	Folders ** JDK Tool ** Script Security
🔗 Timestampers	🔗 Workspace Cleanup	🔗 Ant	🔗 Gradle	
🔗 Pipeline	🔗 GitHub Branch Source	🔗 Pipeline: GitHub Groovy Libraries	🔗 Pipeline: Stage View	
🔗 Git	🔗 Subversion	🔗 SSH Slaves	🔗 Matrix Authorization Strategy	
🔗 PAM Authentication	🔗 LDAP	🔗 Email Extension	🔗 Mailer	

Once the installation is completed, you will get another page to create First Admin user account, fill the all essential details and click on “Save and Continue”.

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:



Jenkins 2.150.2

[Continue as admin](#)

[Save and Continue](#)

You will see a confirmation page that “Jenkins is ready”. To visit Jenkins main dashboard, click Start using Jenkins. Click Save and Finish after confirming the corresponding information.

Getting Started


Jenkins is ready!


Your Jenkins setup is complete.


[Start using Jenkins](#)




That’s wonderful! You have successfully installed Jenkins on your system.


 **Jenkins**


 **raj** | [log out](#)


Jenkins 


[ENABLE AUTO REFRESH](#)


 [New Item](#)


 [People](#)


 [Build History](#)

 [Manage Jenkins](#)

 [My Views](#)

 [Lockable Resources](#)

 [Credentials](#)

 [New View](#)


Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

 [add description](#)

Welcome to Jenkins!

Please **create new jobs** to get started.