

# Aim:

## Installation and Configuration of Jenkins

## Procedure and Screenshots.

Jenkins is an open source automation server that has become a crucial component in the likes of Kubernetes and GitOps. Jenkins enables the continuous integration and delivery of software. Jenkins includes a number of plugins to help the automation of building and deploying your applications.

### Steps for Installation:

1. First of all, jenkins **needs** jdk, so on ubuntu vm of ours go ahead and issue:

```
sudo apt update && sudo apt install default-jdk
```

```
#if on ubuntu server, default-jdk-headless
```

```
#Once the installation finishes check the java version by,
```

```
java -version.
```



```
uwsgi-plugin-jvm-openjdk-7/trusty-updates,trustly-security 1.9.17.1-Subuntu0.1 amd64
Java plugin for uWSGI (OpenJDK 7)

uwsgi-plugin-jwsgi-openjdk-6/trusty-updates,trustly-security 1.9.17.1-Subuntu0.1 amd64
JWSGI plugin for uWSGI (OpenJDK 6)

uwsgi-plugin-jwsgi-openjdk-7/trusty-updates,trustly-security 1.9.17.1-Subuntu0.1 amd64
JWSGI plugin for uWSGI (OpenJDK 7)

vagrant@vagrant-ubuntu-trusty-64:~$ sudo apt install default-jdk
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  ca-certificates-java default-jre default-jre-headless fonts-dejavu-extra
  java-common libasynsnc0 libatk-wrapper-java libatk-wrapper-java-jni libflac8
  libgltf4 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libice-dev libnspr4
  libnss3 libnss3-nssdb libpccsclite libpthread-stubs0-dev libpulse0 libscptpl
  libsm-dev libsndfile1 libvorbisenc2 libx11-dev libx11-doc libxau-dev
  libxbl-dev libxkb-common libxkb-dev libxkb-tools openjdk-7-jdk openjdk-7-jre
  openjdk-7-jre-headless tzdata-java x11proto-core-dev x11proto-input-dev
  x11proto-kb-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  equiva librsync2-common gvfs libice-doc pccsd pulseaudio libsm-doc libxkb-doc
  libxkb-tools openjdk-7-demo openjdk-7-source visualvm icedtea-7-plugin
  icedtea-7-jre-jawm libnss-nss sun-java6-fonts fonts-ipafont-gothic
  fonts-ipafont-mincho fonts-woy-microhei fonts-woy-zehnei ttf-indic-fonts
The following NEW packages will be installed:
  ca-certificates-java default-jdk default-jre default-jre-headless
  fonts-dejavu-extra java-common libasynsnc0 libatk-wrapper-java
  libatk-wrapper-java-jni libflac8 libgltf4 libgtk2.0-0 libgtk2.0-bin
  libgtk2.0-common libice-dev libnspr4 libnss3 libnss3-nssdb
  libpccsclite libpthread-stubs0-dev libpulse0 libscptpl libsm-dev
  libsndfile1 libvorbisenc2 libx11-dev libx11-doc libxau-dev
  libxbl-dev libxkb-common libxkb-dev libxkb-tools libxkb-dev
  libxkb-tools openjdk-7-jdk openjdk-7-jre openjdk-7-jre-headless
  tzdata-java x11proto-core-dev x11proto-input-dev x11proto-kb-dev
  xorg-sgml-doctools xtrans-dev
0 upgraded, 41 newly installed, 0 to remove and 1 not upgraded.
Need to get 65.4 MB of archives.
After this operation, 120 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu/ trusty/main libasynsnc0 amd64 0.8-4ubuntu2 [11.9 kB]
Get:2 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libgtk2.0-common all 2.24.23-0ubuntu1.4 [121 kB]
Get:3 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libgtk2.0-0 amd64 2.24.23-0ubuntu1.4 [1,739 kB]
Get:4 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libnspr4 amd64 2:4.13.1-0ubuntu0.14.04.1 [110 kB]
Get:5 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libnss3-nssdb all 2:3.28.4-0ubuntu0.14.04.5 [10.6 kB]
Get:6 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libnss3 amd64 2:3.28.4-0ubuntu0.14.04.5 [1,124 kB]
Get:7 http://archive.ubuntu.com/ubuntu/ trusty/main ca-certificates-java all 20130815ubuntu1 [13.4 kB]
Get:8 http://archive.ubuntu.com/ubuntu/ trusty-updates/main tzdata-java all 2013a-0ubuntu0.14.04 [70.0 kB]
Get:9 http://archive.ubuntu.com/ubuntu/ trusty/main java-common all 0.51 [130 kB]
Get:10 http://archive.ubuntu.com/ubuntu/ trusty-updates/main libpccsclite amd64 1.8.10-1ubuntu1.1 [21.1 kB]
5% [Working]
401 kB/s 2min 34s
```

2. Download and install the necessary GPG key with the command

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

### 3. Add the necessary repository with the command

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

### 4. Add the universe repository with the command

```
sudo add-apt-repository universe
```

### 5. Update apt with the command `sudo apt-get update`

### 6. Install Jenkins with the command

```
sudo apt-get install jenkins -y
```

Allow the installation to complete.

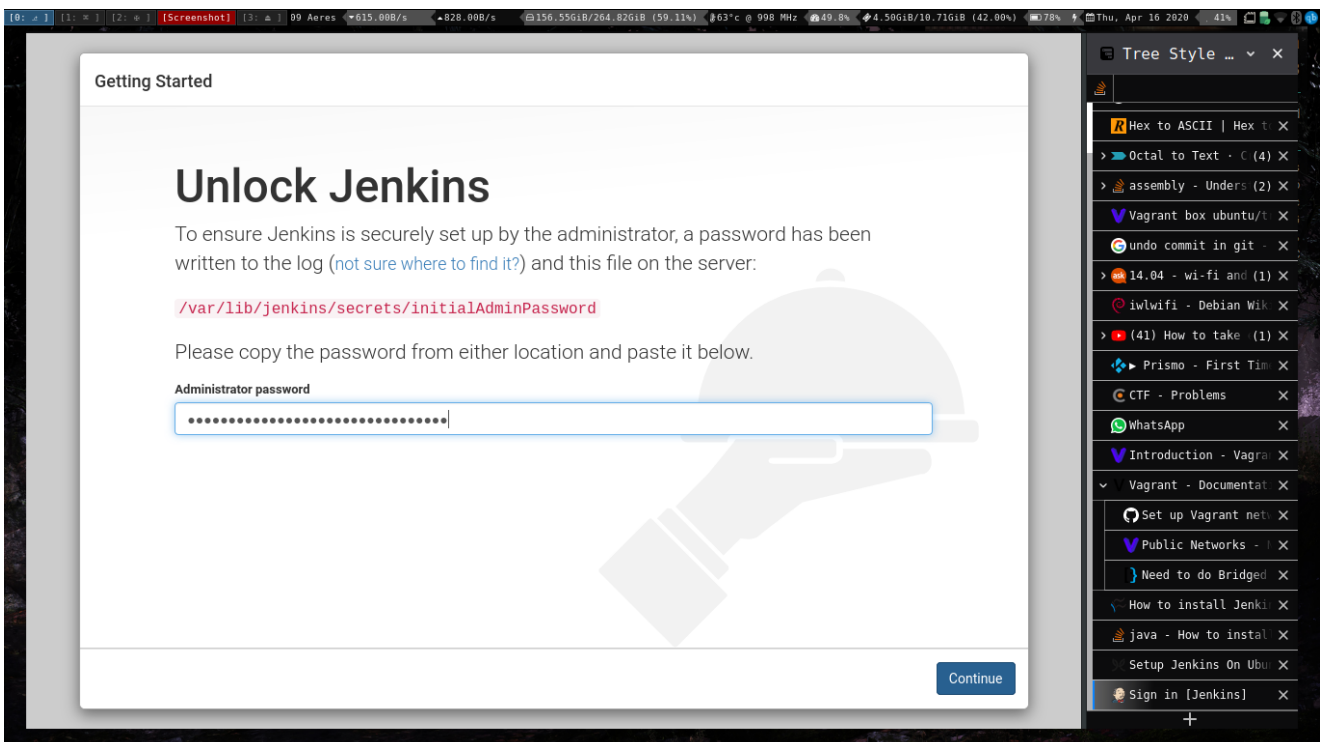
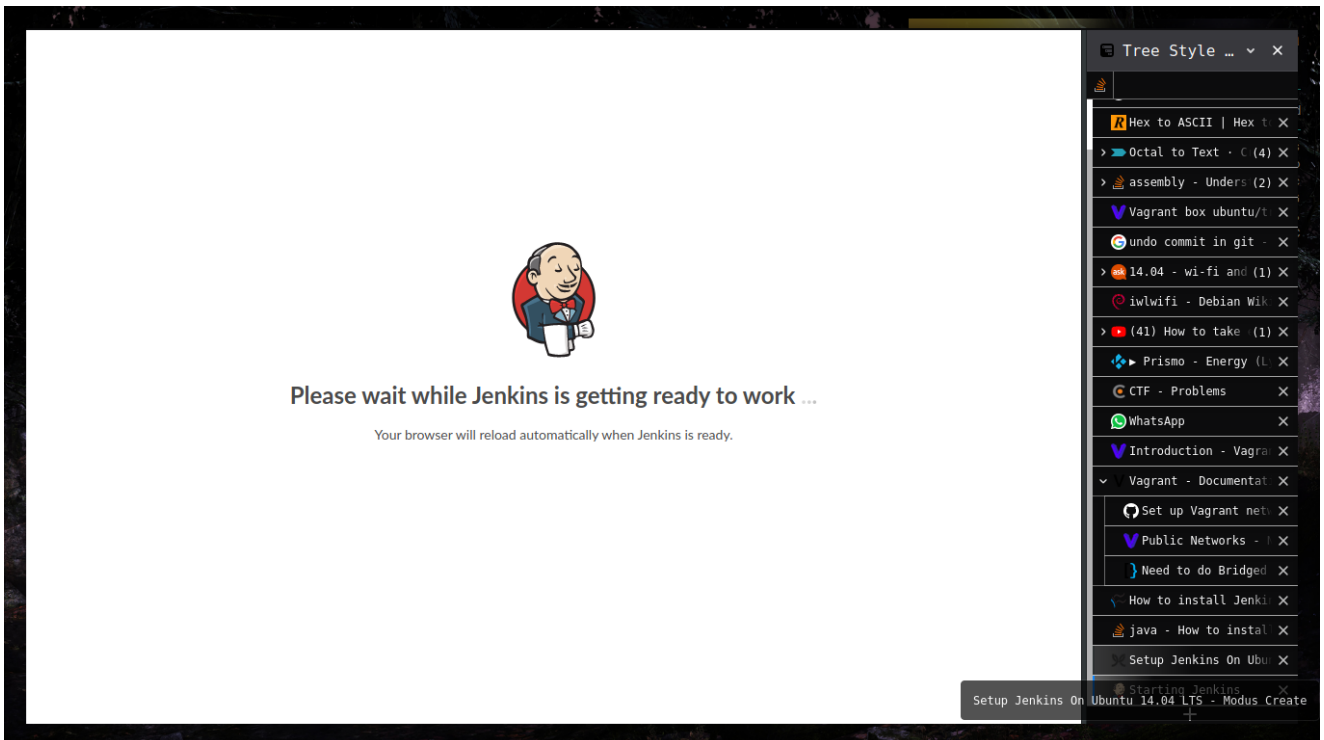


```
.agentlib-<libname>[=<options>]
    load native agent library <libname>, e.g. -agentlib:hprof
    see also, -agentlib:jdwp=help and -agentlib:hprof=help
-agentpath-<pathname>[=<options>]
    load native agent library by full pathname
-javaagent:<jarpath>[=<options>]
    load Java programming language agent, see java.lang.Instrument
-splash:<imagepath>
    show splash screen with specified image
See http://www.oracle.com/technetwork/java/javase/documentation/index.html for more details.
vagrant@vagrant-ubuntu-trusty-64:~$ java -version
java version "1.7.0_201"
OpenJDK Runtime Environment (IcedTea 2.6.17) (7u211-2.6.17-0ubuntu0.1)
OpenJDK 64-Bit Server VM (build 24.201-b00, mixed mode)
vagrant@vagrant-ubuntu-trusty-64:~$ wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add
OK
vagrant@vagrant-ubuntu-trusty-64:~$ sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/s
ources.list.d/jenkins.list'
vagrant@vagrant-ubuntu-trusty-64:~$ sudo add-apt-repository universe
'universe' distribution component is already enabled for all sources.
vagrant@vagrant-ubuntu-trusty-64:~$ sudo apt update && sudo apt install jenkins
Ign http://archive.ubuntu.com trusty InRelease
Ign http://pkg.jenkins.io binary/ InRelease
Hit http://archive.ubuntu.com trusty-updates InRelease
Hit http://security.ubuntu.com trusty-security InRelease
Hit http://archive.ubuntu.com trusty-backports InRelease
Get:1 http://pkg.jenkins.io binary/ Release.gpg [195 B]
Hit http://archive.ubuntu.com trusty Release.gpg
Hit http://archive.ubuntu.com trusty-updates/main amd64 Packages
Get:2 http://pkg.jenkins.io binary/ Packages [17.2 kB]
Hit http://archive.ubuntu.com trusty-updates/restricted Sources
Hit http://security.ubuntu.com trusty-security/main amd64 Packages
Hit http://archive.ubuntu.com trusty-updates/universe Sources
Get:3 http://pkg.jenkins.io binary/ Packages [17.2 kB]
Hit http://security.ubuntu.com trusty-security/universe Sources
Hit http://archive.ubuntu.com trusty-updates/multiverse Sources
Hit http://security.ubuntu.com trusty-security/main amd64 Packages
Hit http://archive.ubuntu.com trusty-updates/main amd64 Packages
Hit http://archive.ubuntu.com trusty-updates/restricted amd64 Packages
Hit http://security.ubuntu.com trusty-security/universe amd64 Packages
Hit http://archive.ubuntu.com trusty-updates/universe amd64 Packages
Hit http://security.ubuntu.com trusty-security/multiverse amd64 Packages
Hit http://archive.ubuntu.com trusty-updates/restricted Translation-en
Hit http://security.ubuntu.com trusty-security/main Translation-en
Hit http://archive.ubuntu.com trusty-updates/main Translation-en
Hit http://security.ubuntu.com trusty-security/universe Translation-en
Hit http://archive.ubuntu.com trusty-updates/multiverse Translation-en
Hit http://security.ubuntu.com trusty-security/main Translation-en
Hit http://archive.ubuntu.com trusty-updates/universe Translation-en
Hit https://esm.ubuntu.com trusty-infra-security InRelease
Hit http://archive.ubuntu.com trusty Release
100% [Packages 5,856 KB] [Release gpgv 58.5 KB] [Waiting for headers] [Waiting ]
```

## How to access Jenkins?

1. Open a web browser and point it to [http://SERVER\\_IP:8080](http://SERVER_IP:8080) (where SERVER\_IP is the IP address of the hosting server).
2. You will then be prompted to copy and paste a password that was created during the Jenkins installation. To retrieve that password, go back to the terminal window and issue the command:

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

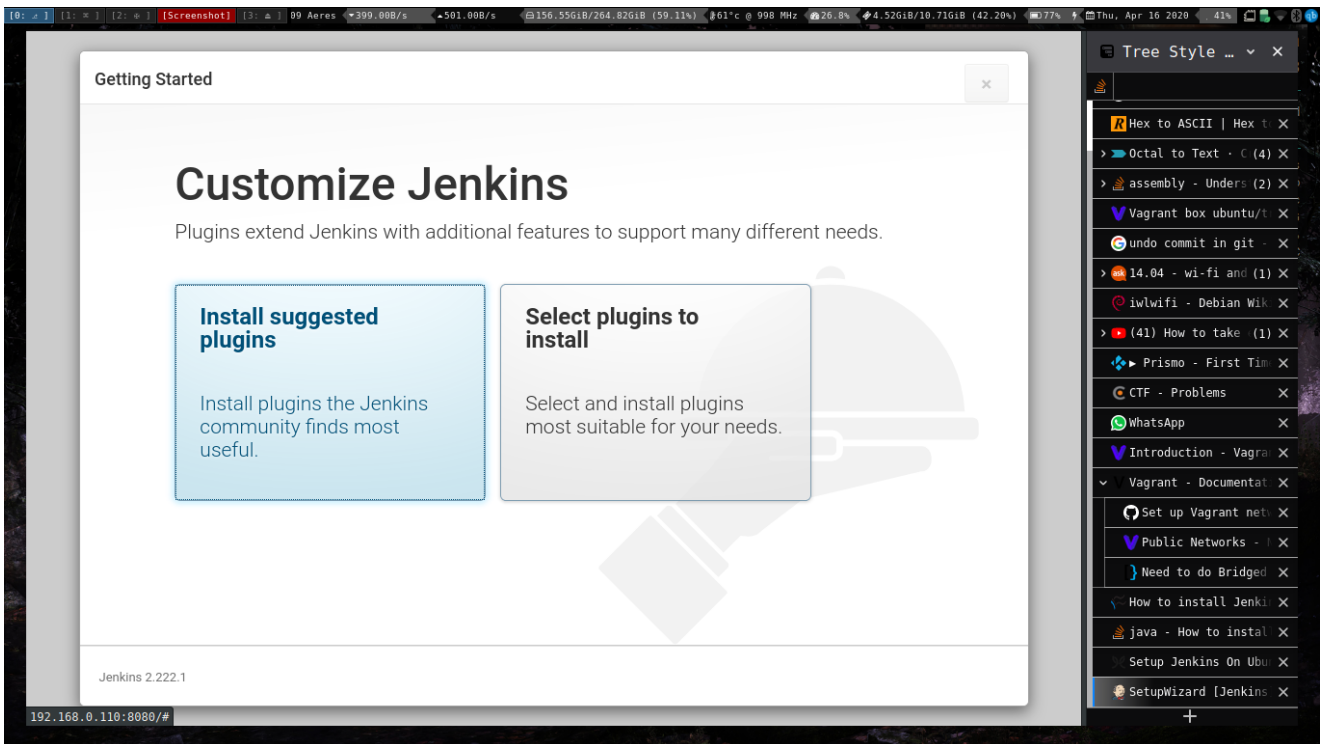


```
vagrant@vagrant-ubuntu-trusty-64:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
5ce54d5f6f14479d9170f7504db855aa
vagrant@vagrant-ubuntu-trusty-64:~$
```

```
1  [|||||] 38.1% Tasks: 153, 652 thr; 2 running
2  [|||||] 37.1% Load average: 1.87 2.90 2.58
3  [|||||] 38.9% Uptime: 02:36:02
4  [|||||] 33.1%
Mem [|||||] 4.18G/10.7G
Swap [|||||] 0K/0K
```

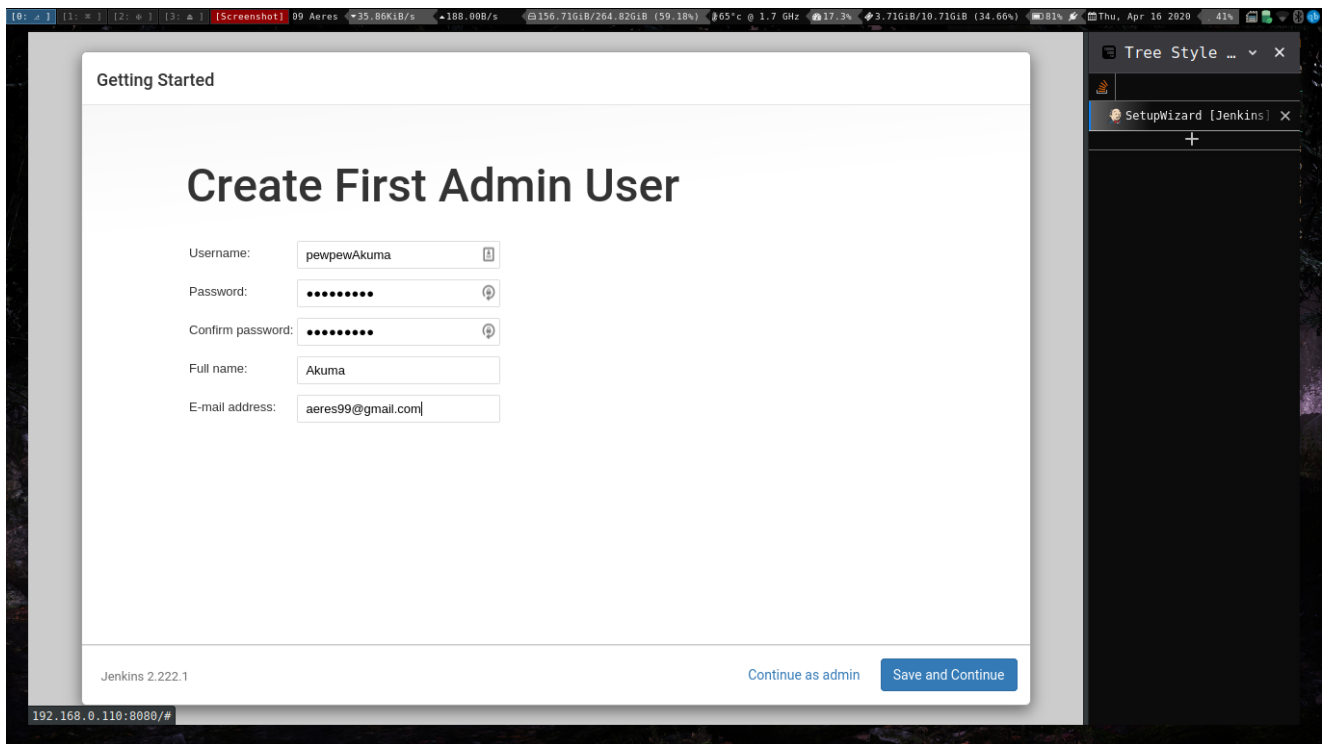
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
2052	root	20	0	954M	102M	73556	S	39.5	0.9	37:10.26	/usr/lib/xorg/Xorg -core :0
4634	akuma	20	0	815M	56824	41252	S	13.2	0.5	4:14.78	compton --config /home/akuma
4716	akuma	20	0	928M	13376	10188	S	11.9	0.1	14:00.59	conky -bc /home/akuma/.confi
14706	akuma	20	0	1563M	37652	23652	S	11.3	0.3	0:41.50	mpd -v
4756	akuma	9	-11	1778M	14996	9944	S	8.2	0.1	2:10.51	/usr/bin/pulseaudio --start
14817	akuma	20	0	690M	10648	9084	S	8.2	0.1	0:31.70	vis
2786	root	20	0	954M	102M	73556	S	7.5	0.9	8:43.93	/usr/lib/xorg/Xorg -core :0
2607	akuma	20	0	562M	33280	26396	S	6.9	0.3	0:34.04	termite
4152	netdata	20	0	22744	5648	2504	S	6.3	0.1	7:17.60	/usr/libexec/netdata/plugins
3881	akuma	20	0	59480	4964	3832	R	5.6	0.0	1:49.42	htop
4921	akuma	-6	0	1778M	14996	9944	S	4.4	0.1	0:10.14	/usr/bin/pulseaudio --start
14785	akuma	20	0	1563M	37652	23652	S	4.4	0.3	0:17.94	mpd -v
20058	akuma	20	0	4178M	2631	127M	S	4.4	2.4	10:11.51	/usr/bin/mindforger
4640	akuma	20	0	188M	31776	11580	T	3.0	0.3	2:03.01	/usr/bin/python3 /home/akuma

```
F1 Help F2 Setup F3 Search F4 Filter F5 Tree F6 Sort By F7 Nice F8 Nice # F9 Kill F10 Quit
```



Choose suggested plugins and then

Create the first user



Set the default path and you will be set to roll with Jenkins.

