JENKINS

What is Jenkins?

Jenkins is a self-contained, open source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software.

Jenkins can be installed through native system packages, Docker, or even run standalone by any machine with a Java Runtime Environment (JRE) installed.

Installing Jenkins

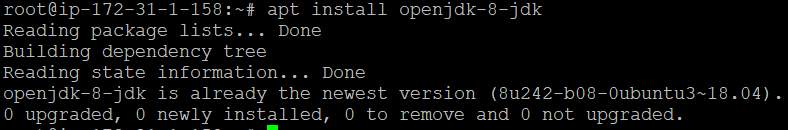
To install Jenkins on your Ubuntu system, follow these steps

Install Java.

Since Jenkins is a Java application, the first step is to install Java. Update the package index and install the Java 8 OpenJDK package with the following commands:

#sudo apt update

#sudo apt install openjdk-8-jdk



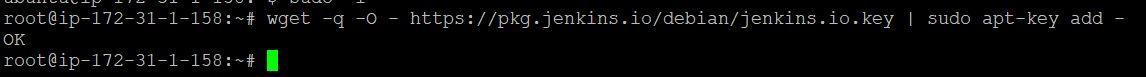
The current version of Jenkins does not support Java 10 (and Java 11) yet. If you have multiple versions of Java installed on your machine make sure Java 8 is the default Java version.

**java -version**

Add the Jenkins Debian repository.

Import the GPG keys of the Jenkins repository using the following wget command:

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



The command above should output OK which means that the key has been successfully imported and packages from this repository will be considered trusted.

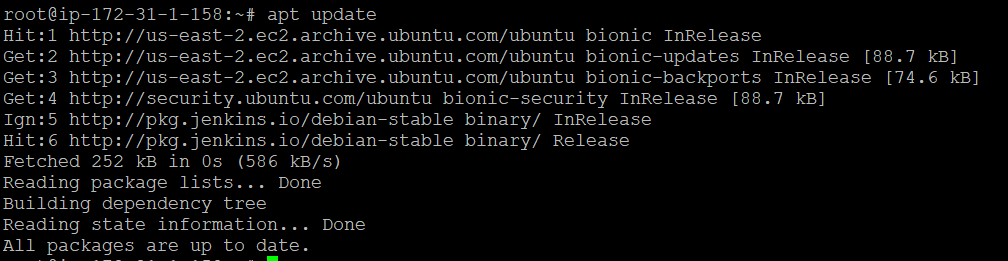
Next, add the Jenkins repository to the system with:

#sudo sh -c 'echo deb [http://pkg.jenkins.io/debian-stable binary/](http://pkg.jenkins.io/debian-stable%20binary/) > /etc/apt/sources.list.d/jenkins.list'

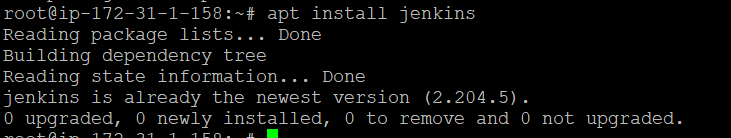
Install Jenkins.

Once the Jenkins repository is enabled, update the apt package list and install the latest version of Jenkins by typing:

#sudo apt update

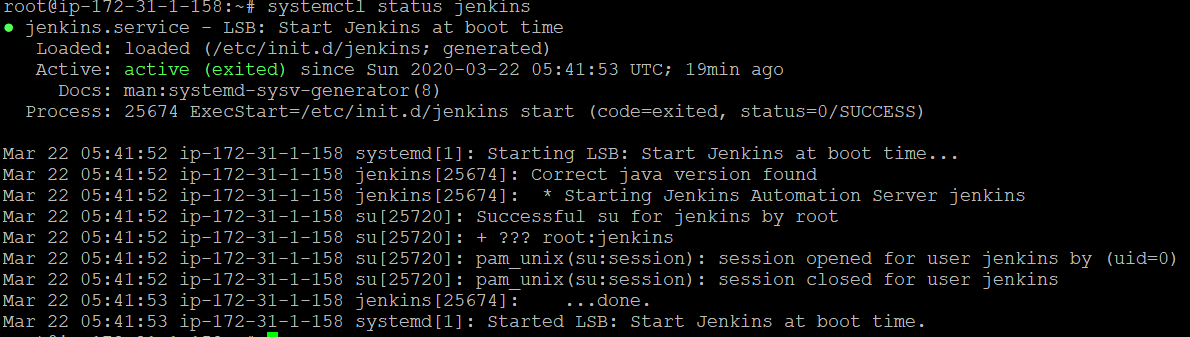


#sudo apt install jenkins



Jenkins service will automatically start after the installation process is complete. You can verify it by printing the service status:

#systemctl status jenkins



You should see something similar to this:

● jenkins.service - LSB: Start Jenkins at boot time

Loaded: loaded (/etc/init.d/jenkins; generated)

Active: active (exited) since Wed 2018-08-22 13:03:08 PDT; 2min 16s ago

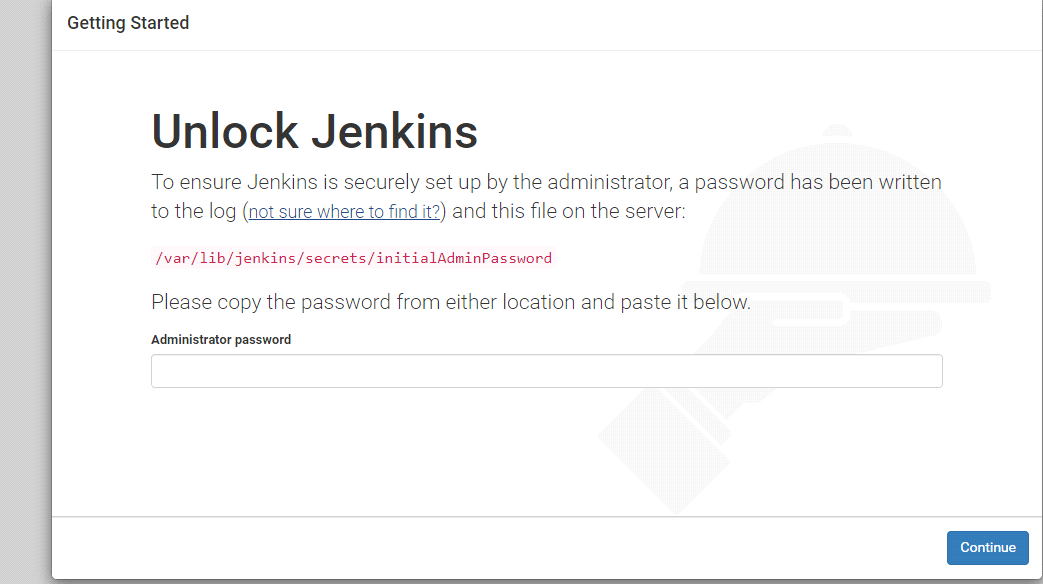
Docs: man:systemd-sysv-generator(8)

Tasks: 0 (limit: 2319)

CGroup: /system.slice/jenkins.service

Setting Up Jenkins

To set up your new Jenkins installation, open your browser, type your domain or IP address followed by port 8080, [http://your\_ip\_or\_domain:8080](http://your_ip_or_domain:8080/) and screen similar to the following will be displayed:

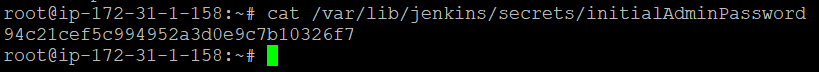


During the installation, the Jenkins installer creates an initial 32-character long alphanumeric password. Use the following command to print the password on your terminal:

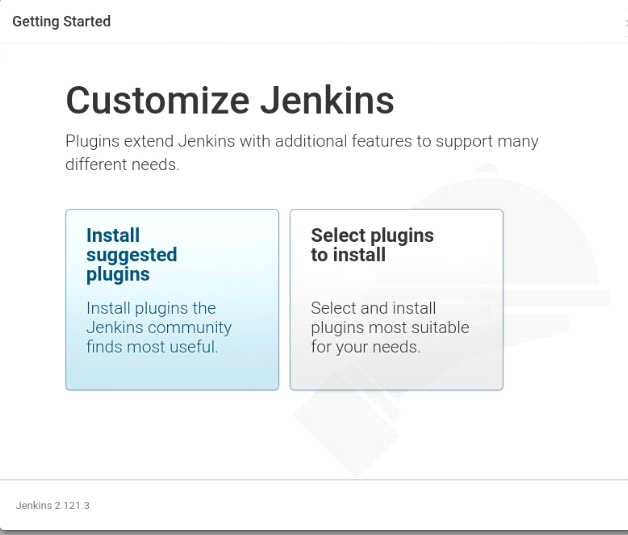
#sudo cat /var/lib/jenkins/secrets/initialAdminPassword

OUT PUT

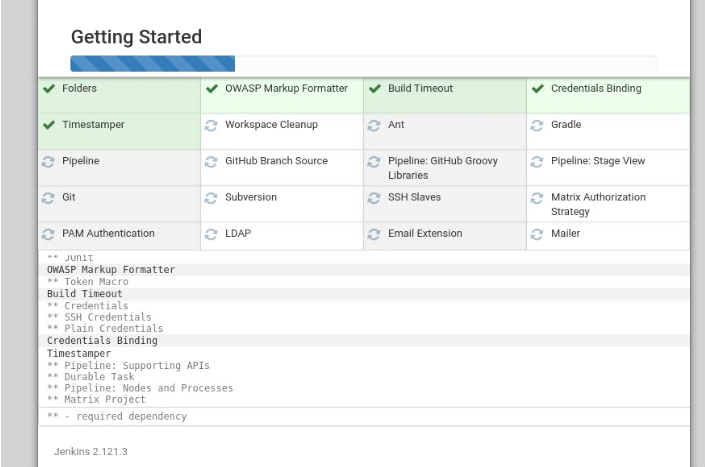
2115173b548f4e99a203ee99a8732a32



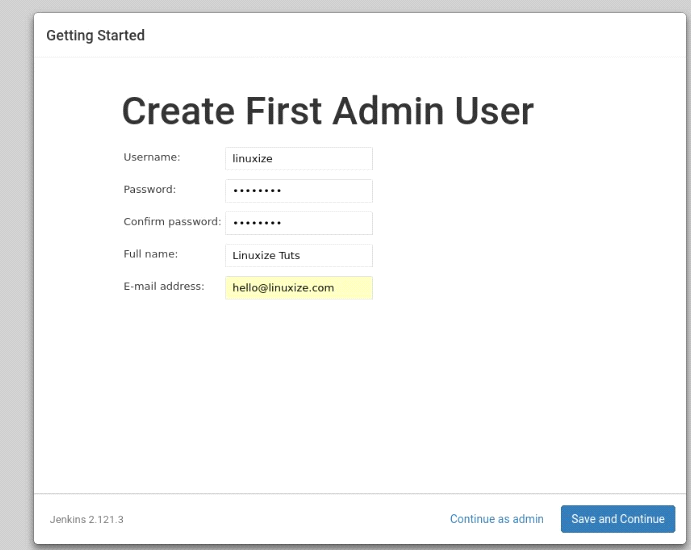
Copy the password from your terminal, paste it into the Administrator password field and click Continue.



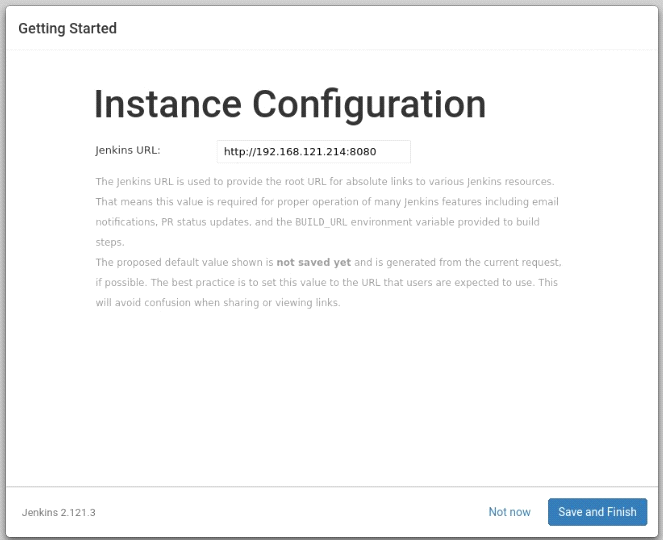
On the next screen, the setup wizard will ask you whether you want to install suggested plugins or you want to select specific plugins. Click on the Install suggested plugins box, and the installation process will start immediately.



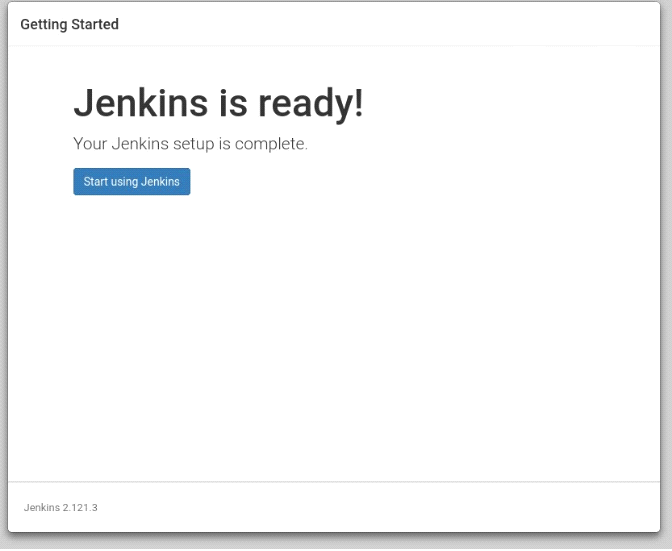
Once the plugins are installed, you will be prompted to set up the first admin user. Fill out all required information and click Save and Continue.



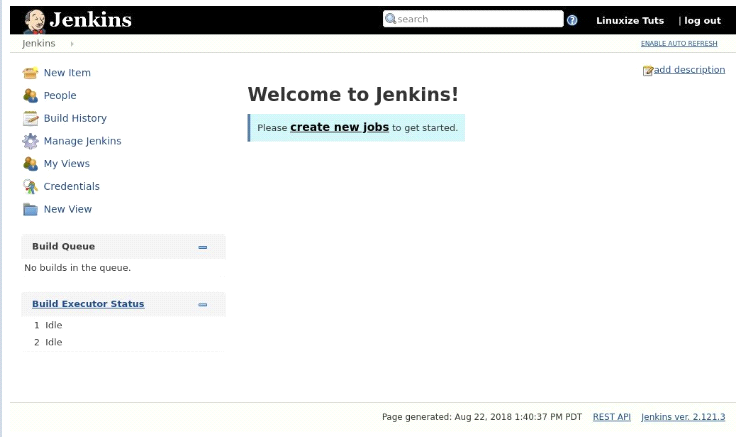
The next page will ask you to set the URL for your Jenkins instance. The field will be populated with an automatically generated URL.



Confirm the URL by clicking on the Save and Finish button and the setup process will be completed.



Click on the Start using Jenkins button and you will be redirected to the Jenkins dashboard logged in as the admin user you have created in one of the previous steps.



At this point, you’ve successfully installed Jenkins on your system.

----------------------------------------------------------------------------------

**Install maven software in local machine.**

sudo apt update

sudo apt install maven

mvn -version

----------------------------------------------------------------------------------

**Jenkins Global tool configuration settings**

java path - /usr/lib/jvm/java-8-openjdk-amd64/

maven path - /usr/share/maven

-------------------------------------------------------------------

**Install Tomcat server**

1) **install java**

sudo apt update

sudo apt install openjdk-8-jdk

java -version

2) **Download tomcat packages from** [**https://tomcat.apache.org/download-80.cgi**](https://tomcat.apache.org/download-80.cgi) **onto /opt on EC2 instance**

**Note: Make sure you change <version> with the tomcat version which you download.**

cd /

cd /opt

wget <https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.81/bin/apache-tomcat-8.5.81.tar.gz>

tar -zvxf /opt/apache-tomcat-8.5.81.tar.gz

3)**give executing permissions to startup.sh and shutdown.sh which are under bin**

ps -ef | grep tomcat : Running services

cd apache-tomcat-8.5.81

cd bin

ls -ltr

chmod +x ./startup.sh :give the permission for execute or start

chmod +x ./shutdown.sh

**Create soft link for easy start or shutdown**

echo $PATH

ln -s /opt/apache-tomcat-8.5.81/bin/startup.sh /usr/local/bin/tomcatup

ln -s /opt/apache-tomcat-8.5.81/bin/shutdown.sh /usr/local/bin/tomcatdown

tomcatup

access tomcat application from browser on port 8080

**http:// HYPERLINK "http://%3cpublic\_ip%3e:8080/"< HYPERLINK "http://%3cpublic\_ip%3e:8080/"Public\_IP HYPERLINK "http://%3cpublic\_ip%3e:8080/"> HYPERLINK "http://%3cpublic\_ip%3e:8080/":8080**

**Using unique ports for each application is a best practice in an environment. But tomcat and Jenkins runs on ports number 8080. Hence lets change tomcat port number to 8090. Change port number in conf/server.xml file under tomcat home**

cd /opt/apache-tomcat-8.5.81/conf

vi server.xml

# update port number in the "connecter port" field in server.xml

# restart tomcat after configuration update

tomcatdown

tomcatup

Note: Once change the port no from 8080 to 8090. please enable the 8090 port as inbound poer rule in cloud VM(Ubuntu server)

**4)Update the configuration to login into tomcat server**

Now application is accessible on port 8090. but tomcat application doesnt allow to login from browser. changing a default parameter in context.xml does address this issue

#search for context.xml

find / -name context.xml

above command gives 3 context.xml files. comment () Value ClassName field on files which are under webapp directory. After that restart tomcat services to effect these changes. At the time of writing this lecture below 2 files are updated.

vi /opt/tomcat/webapps/host-manager/META-INF/context.xml

vi /opt/tomcat/webapps/manager/META-INF/context.xml

# Restart tomcat services

tomcatdown

tomcatup

5)**Update users information in the tomcat-users.xml file goto tomcat home directory and Add below users to conf/tomcat-users.xml file**

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>

<user username="deployer" password="deployer" roles="manager-script"/>

<user username="tomcat" password="s3cret" roles="manager-gui"/>

------------------------------------------------------------------------------------------------

**Deploy jenkins build o/p to tomcat server**

-> Add "*Deploy to container"* plugin in Mange Jenkins plugin

->Add the credential of tomcat server for deployer user in Global credentials

->In post build steps - fill below infomation



Build now-

-> Login the tomcat server - find the location for the latest updates

/opt/apache-tomcat-8.5.81/webapps/ROOT

-----------------------------------------------------------------------------------------------

**Jenkins Build periodically:**

If you want to schedule your project build periodically then you should select the Build periodically option from the build triggers.

This field follows the syntax of cron (with minor differences). Specifically, each line consists of 5 fields separated by TAB or whitespace:

MINUTE HOUR DOM MONTH DOW

MINUTE Minutes within the hour (0–59)

HOUR The hour of the day (0–23)

DOM The day of the month (1–31)

MONTH The month (1–12)

DOW The day of the week (0–7) where 0 and 7 are Sunday.

\*/2 \* \* \* \* (Every 2 mins )

**Poll SCM:**

Poll SCM periodically polls the SCM to check whether changes were made (i.e. new commits) and builds the project if new commits were pushed since the last build.

\*/2 \* \* \* \* (Every 2 mins )

------------------------------------------------------------------------------------------------