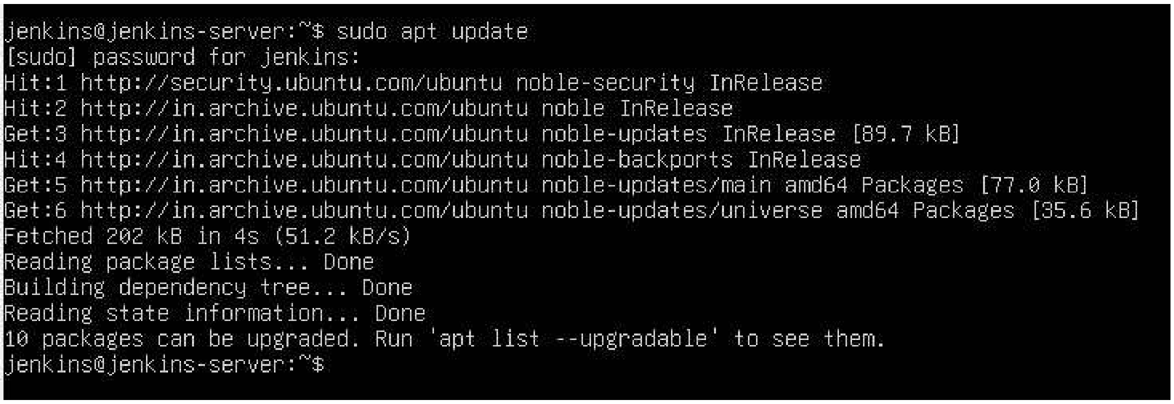
Restart Ubuntu server

sudo reboot

Installing update on Ubuntu server

Clear the console using the ”clear” command and update the system using “**sudo apt update**” command.



```
jenkins@jenkins-server:~$ sudo apt update
[sudo] password for jenkins:
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu noble InRelease
Get:3 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease [89.7 kB]
Hit:4 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:5 http://in.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [77.0 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [35.6 kB]
Fetched 202 kB in 4s (51.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
10 packages can be upgraded. Run 'apt list --upgradable' to see them.
jenkins@jenkins-server:~$
```

Installing Jenkins on Ubuntu server

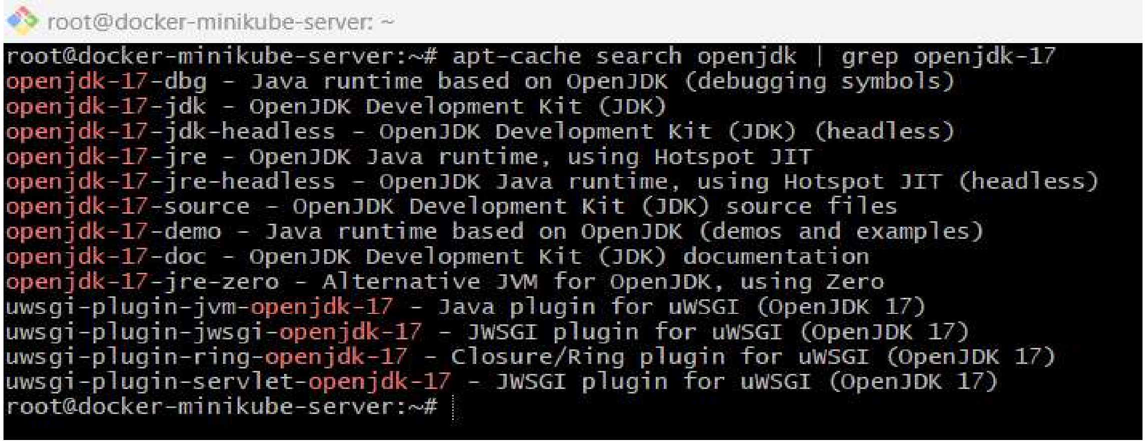
Jenkins is a open-source automation server that lets you build, test and deploy your code. Now, we will see how to install Jenkins on the newly created Ubuntu server.

Install Java as Jenkins pre-requisite

Jenkins is a Java based application. So, Java is a pre-requisite. Install Java with the following command in Ubuntu server –

sudo apt update

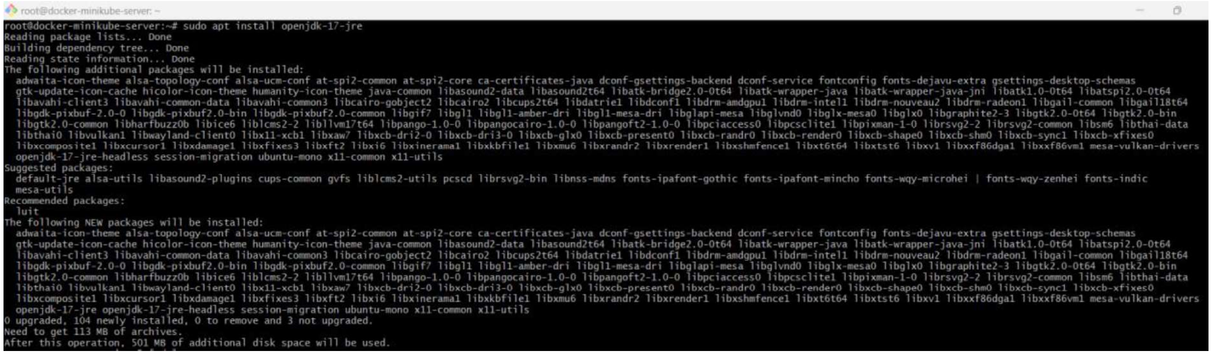
sudo apt upgrade



```
root@docker-minikube-server:~# apt-cache search openjdk | grep openjdk-17
openjdk-17-dbg - Java runtime based on OpenJDK (debugging symbols)
openjdk-17-jdk - OpenJDK Development Kit (JDK)
openjdk-17-jdk-headless - OpenJDK Development Kit (JDK) (headless)
openjdk-17-jre - OpenJDK Java runtime, using Hotspot JIT
openjdk-17-jre-headless - OpenJDK Java runtime, using Hotspot JIT (headless)
openjdk-17-source - OpenJDK Development Kit (JDK) source files
openjdk-17-demo - Java runtime based on OpenJDK (demos and examples)
openjdk-17-doc - OpenJDK Development Kit (JDK) documentation
openjdk-17-jre-zero - Alternative JVM for OpenJDK, using Zero
uwsgi-plugin-jvm-openjdk-17 - Java plugin for uWSGI (OpenJDK 17)
uwsgi-plugin-jwsgi-openjdk-17 - JWSGI plugin for uWSGI (OpenJDK 17)
uwsgi-plugin-ring-openjdk-17 - Closure/Ring plugin for uWSGI (OpenJDK 17)
uwsgi-plugin-servlet-openjdk-17 - JWSGI plugin for uWSGI (OpenJDK 17)
root@docker-minikube-server:~#
```

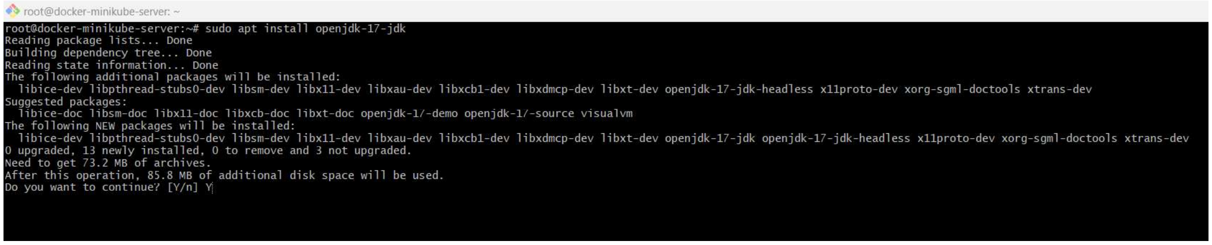
Depending on your version of Ubuntu, you may be able to install OpenJDK 17 JDE and JRE using the following command. This is only applicable if these packages are available in your distribution:

sudo apt install openjdk-17-jre



```
root@docker-minikube-server:~# sudo apt install openjdk-17-jre
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme alsa-topology-conf alsa-ucm-conf at-spi2-common at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service fontconfig fonts-dejavu-extra gsettings-desktop-schemas
  gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme java-common libasound2-data libasound2t64 libatk-bridge2.0-0t64 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0t64 libatspi2.0-0t64
  libavahi-client3 libavahi-common-data libavahi-common3 libcairo-gobject2 libcairo2 libcups2t64 libdatrie1 libdconf1 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libgail-common libgail18t64
  libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0t64 libgtk2.0-bin
  libgtk2.0-common libharfbuzz0b libice6 liblcms2-2 libllvm17t64 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpciaccess0 libpcsclite1 libpixman-1-0 librsvg2-2 librsvg2-common libsm6 libthai-data
  libthai0 libvulkan1 libwayland-client0 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0
  libxcomposite1 libxcursor1 libxdamage1 libxfixes3 libxft2 libxi6 libxinerama1 libxkbfile1 libxmu6 libxrandr2 libxrender1 libxshmfence1 libxt6t64 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers
  openjdk-17-jre-headless session-migration ubuntu-mono x11-common x11-utils
Suggested packages:
  default-jre alsa-utils libasound2-plugins cups-common gvfs liblcms2-utils pcscd librsvg2-bin libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic
  mesa-utils
Recommended packages:
  luit
The following NEW packages will be installed:
  adwaita-icon-theme alsa-topology-conf alsa-ucm-conf at-spi2-common at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service fontconfig fonts-dejavu-extra gsettings-desktop-schemas
  gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme java-common libasound2-data libasound2t64 libatk-bridge2.0-0t64 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0t64 libatspi2.0-0t64
  libavahi-client3 libavahi-common-data libavahi-common3 libcairo-gobject2 libcairo2 libcups2t64 libdatrie1 libdconf1 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libgail-common libgail18t64
  libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0t64 libgtk2.0-bin
  libgtk2.0-common libharfbuzz0b libice6 liblcms2-2 libllvm17t64 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpciaccess0 libpcsclite1 libpixman-1-0 librsvg2-2 librsvg2-common libsm6 libthai-data
  libthai0 libvulkan1 libwayland-client0 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0
  libxcomposite1 libxcursor1 libxdamage1 libxfixes3 libxft2 libxi6 libxinerama1 libxkbfile1 libxmu6 libxrandr2 libxrender1 libxshmfence1 libxt6t64 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 mesa-vulkan-drivers
  openjdk-17-jre openjdk-17-jre-headless session-migration ubuntu-mono x11-common x11-utils
0 upgraded, 104 newly installed, 0 to remove and 3 not upgraded.
Need to get 113 MB of archives.
After this operation, 501 MB of additional disk space will be used.
```

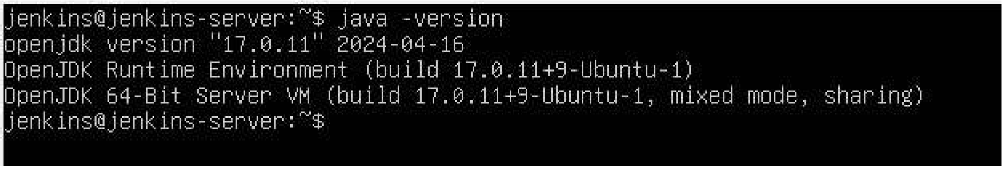
sudo apt install openjdk-17-jdk



```
root@docker-minikube-server:~# sudo apt install openjdk-17-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libice-dev libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk-headless x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-17-demo openjdk-17-source visualvm
The following NEW packages will be installed:
  libice-dev libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk openjdk-17-jdk-headless x11proto-dev xorg-sgml-doctools xtrans-dev
0 upgraded, 13 newly installed, 0 to remove and 3 not upgraded.
Need to get 73.2 MB of archives.
After this operation, 85.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Confirm the installation by running the following command.

java -version



```
jenkins@jenkins-server:~$ java -version
openjdk version "17.0.11" 2024-04-16
OpenJDK Runtime Environment (build 17.0.11+9-Ubuntu-1)
OpenJDK 64-Bit Server VM (build 17.0.11+9-Ubuntu-1, mixed mode, sharing)
jenkins@jenkins-server:~$
```

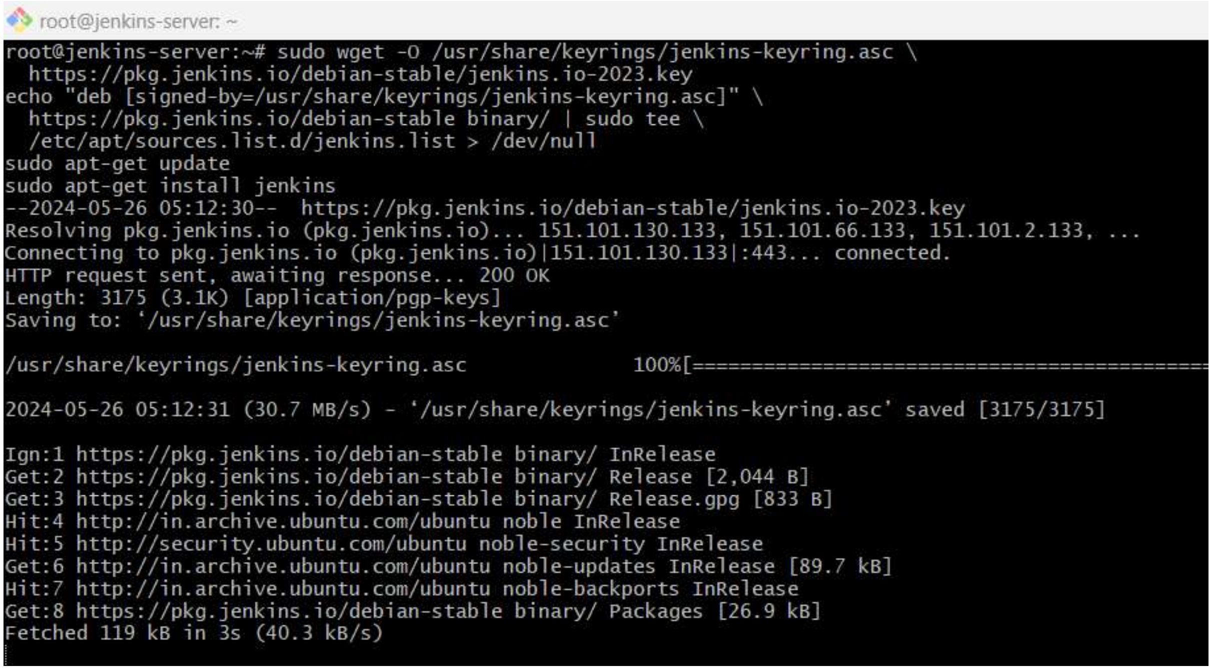
Install Jenkins

Run the below commands to install Jenkins on Ubuntu:

1st command:

sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \

  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key



```
root@jenkins-server:~# sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
--2024-05-26 05:12:30--  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.130.133, 151.101.66.133, 151.101.2.133, ...
Connecting to pkg.jenkins.io (pkg.jenkins.io)|151.101.130.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: ‘/usr/share/keyrings/jenkins-keyring.asc’

/usr/share/keyrings/jenkins-keyring.asc              100%[=============================================

2024-05-26 05:12:31 (30.7 MB/s) - ‘/usr/share/keyrings/jenkins-keyring.asc’ saved [3175/3175]

Ign:1 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:2 https://pkg.jenkins.io/debian-stable binary/ Release [2,044 B]
Get:3 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Hit:4 http://in.archive.ubuntu.com/ubuntu noble InRelease
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:6 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease [89.7 kB]
Hit:7 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:8 https://pkg.jenkins.io/debian-stable binary/ Packages [26.9 kB]
Fetched 119 kB in 3s (40.3 kB/s)
```

2nd command:

echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \

  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

  /etc/apt/sources.list.d/jenkins.list > /dev/null

3rd command:

sudo apt-get update

4th command:

sudo apt-get install jenkins

After installation, please run the below commands to configure Jenkins:

Enable Jenkins

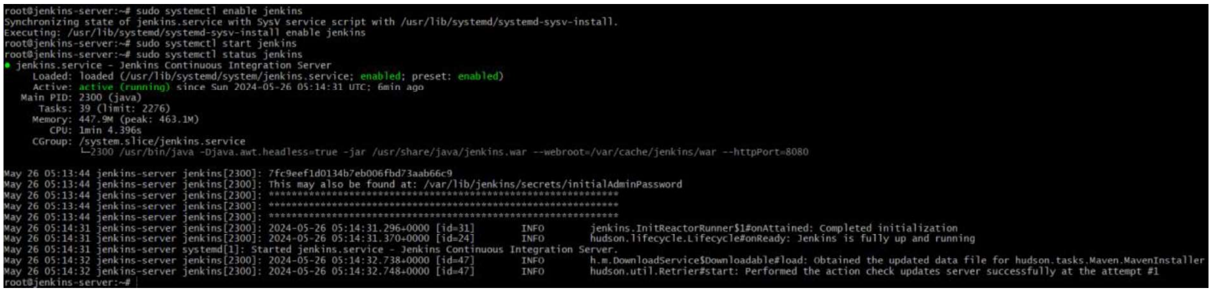
sudo systemctl enable jenkins

Start Jenkins

sudo systemctl start jenkins

Check status of Jenkins

sudo systemctl status jenkins



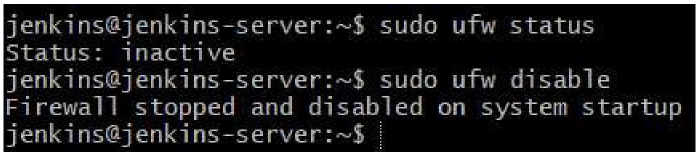
```
root@jenkins-server:~# sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
root@jenkins-server:~# sudo systemctl start jenkins
root@jenkins-server:~# sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Sun 2024-05-26 05:14:31 UTC; 6min ago
   Main PID: 2300 (java)
      Tasks: 39 (limit: 2276)
     Memory: 447.9M (peak: 463.1M)
        CPU: 1min 4.396s
     CGroup: /system.slice/jenkins.service
             └─2300 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

May 26 05:13:44 jenkins-server jenkins[2300]: 7fc9eef1d0134b7eb006fbd73aab66c9
May 26 05:13:44 jenkins-server jenkins[2300]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
May 26 05:13:44 jenkins-server jenkins[2300]: *************************************************************
May 26 05:13:44 jenkins-server jenkins[2300]: *************************************************************
May 26 05:13:44 jenkins-server jenkins[2300]: *************************************************************
May 26 05:14:31 jenkins-server jenkins[2300]: 2024-05-26 05:14:31.296+0000 [id=31]       INFO       jenkins.InitReactorRunner$1#onAttained: Completed initialization
May 26 05:14:31 jenkins-server jenkins[2300]: 2024-05-26 05:14:31.370+0000 [id=24]       INFO       hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
May 26 05:14:31 jenkins-server systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
May 26 05:14:32 jenkins-server jenkins[2300]: 2024-05-26 05:14:32.738+0000 [id=47]       INFO       h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.Maven.MavenInstaller
May 26 05:14:32 jenkins-server jenkins[2300]: 2024-05-26 05:14:32.748+0000 [id=47]       INFO       hudson.util.Retrier#start: Performed the action check updates server successfully at the attempt #1
root@jenkins-server:~#
```

[Optional] If needed, **disable firewall** on Ubuntu server:

sudo ufw status

sudo ufw disable

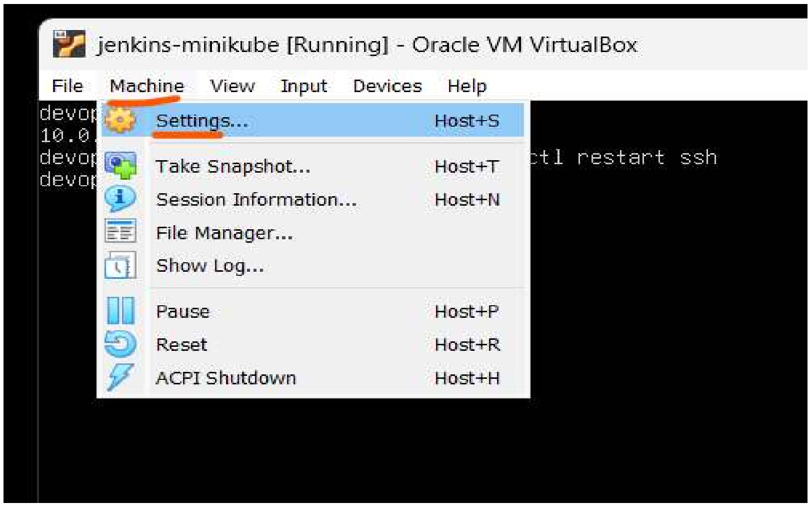


```
jenkins@jenkins-server:~$ sudo ufw status
Status: inactive
jenkins@jenkins-server:~$ sudo ufw disable
Firewall stopped and disabled on system startup
jenkins@jenkins-server:~$
```

Set port forwarding for Jenkins (guest / Ubuntu to host / Windows)

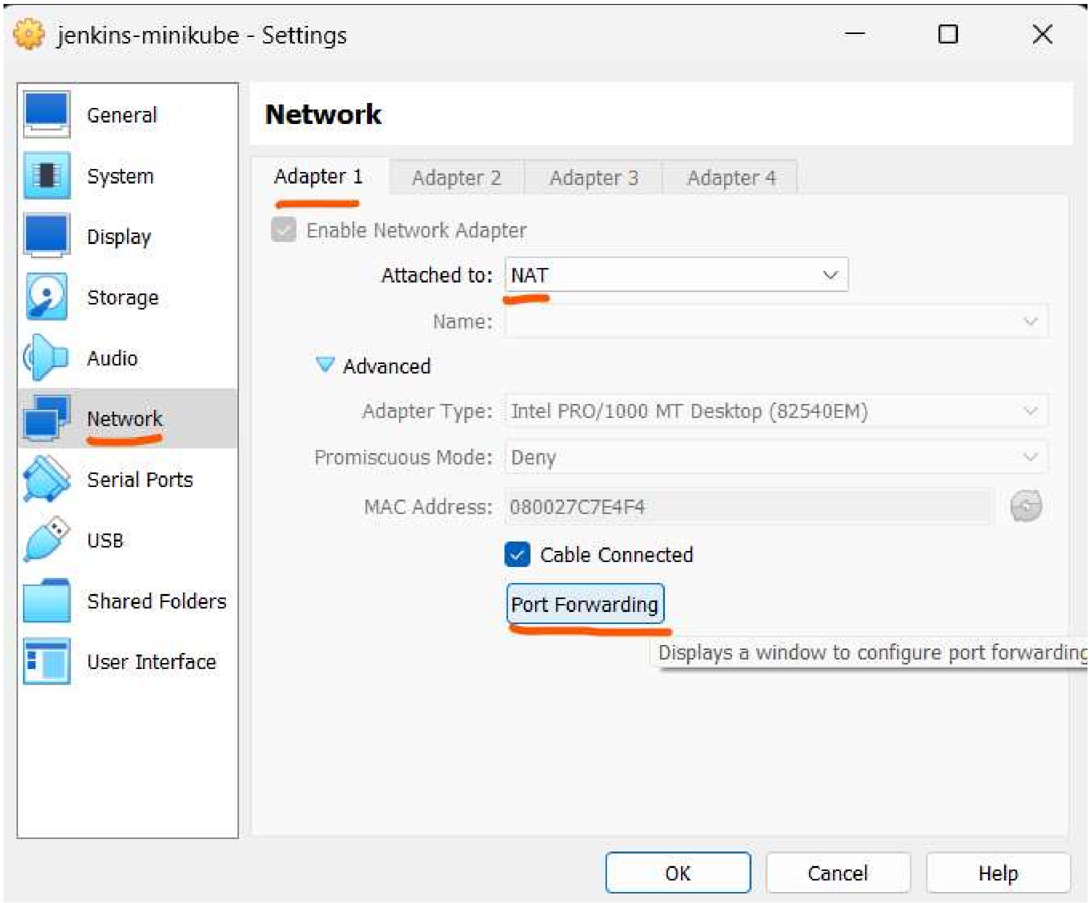
If you haven't enabled port forwarding yet, please follow the below step. Otherwise ignore.

Go to Ubuntu server VirtualBox console, click on Machine → Settings



```
devop
10.0.
devop                                              ctl restart ssh
devop
```

Click on network → Adapter 1 → NAT → click on Port Forwarding



☑ Enable Network Adapter

Attached to: NAT

Name:

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

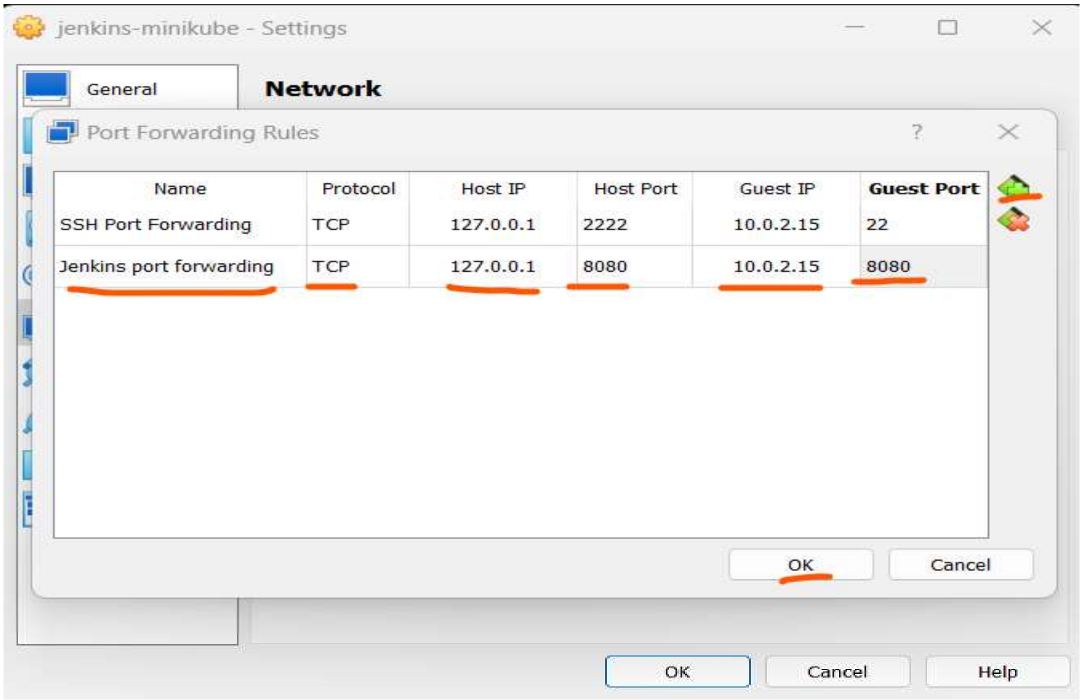
Promiscuous Mode: Deny

MAC Address: 080027C7E4F4

☑ Cable Connected

Add port forwarding for Jenkins like this:

Click on Add sign (green colour) and put Jenkins port forwarding as below:



Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
SSH Port Forwarding	TCP	127.0.0.1	2222	10.0.2.15	22
Jenkins port forwarding	TCP	127.0.0.1	8080	10.0.2.15	8080

Access Jenkins (on VirtualBox) from Windows machine

If you have already installed Jenkins and configured port-forwarding, please proceed to next step. Otherwise, please follow steps for these.

Now open browser and access Jenkins:

http://<loopback address>:8080/

http://127.0.0.1:8080/

Or

http://localhost:8080

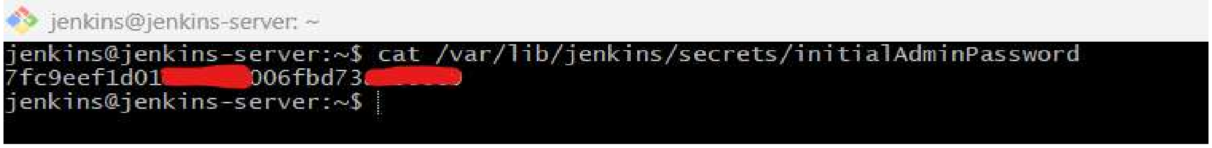
Unlock Jenkins / first time configuration

For the first time when you access Jenkins URL, you would be prompted to unlock Jenkins.

To ensure Jenkins is securely set up by the administrator, a password has been written to the log:

In the Ubuntu server, go to this location and read the file content →

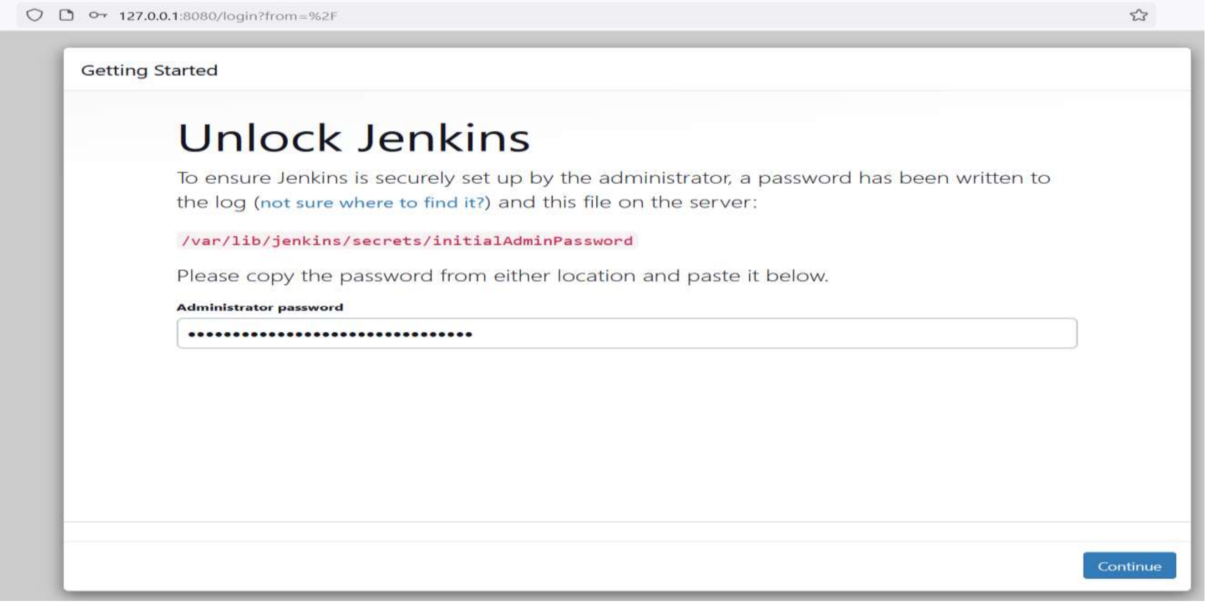
```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```



jenkins@jenkins-server: ~

```
jenkins@jenkins-server:~$ cat /var/lib/jenkins/secrets/initialAdminPassword
7fc9eef1d01[redacted]006fbd73[redacted]
jenkins@jenkins-server:~$ 
```

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



127.0.0.1:8080/login?from=%2F

Getting Started

Unlock Jenkins

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

••••••••••••••••••••••••••••••••

Click on Continue.

Customize Jenkins

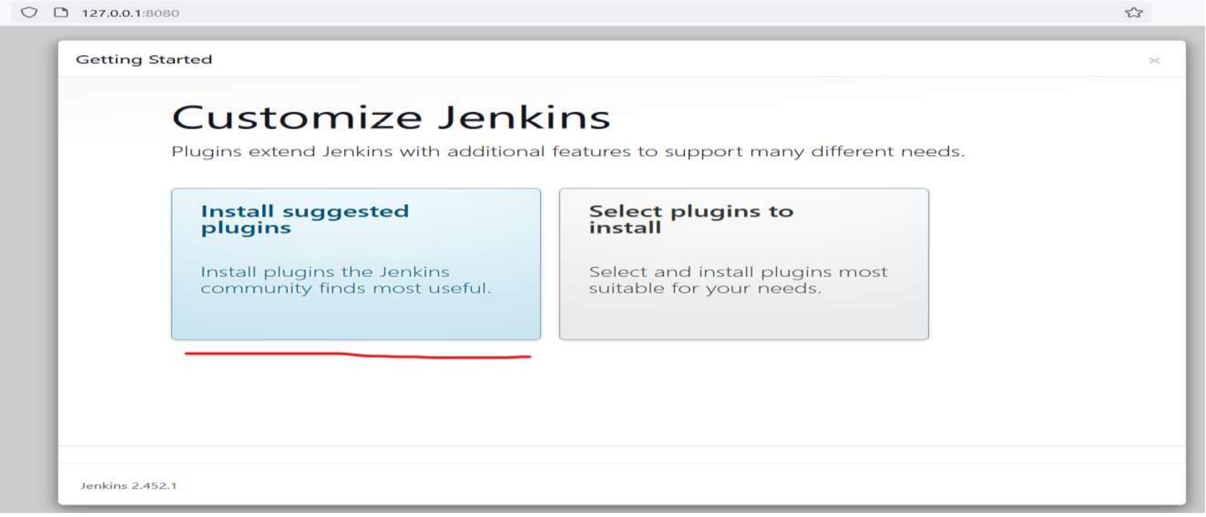
Initial Jenkins configuration

First time installation – default of set plugins.

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

For first time, you will get these options.

Please select – “Install suggested plugins”



Getting Started

Customize Jenkins

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

Install suggested plugins

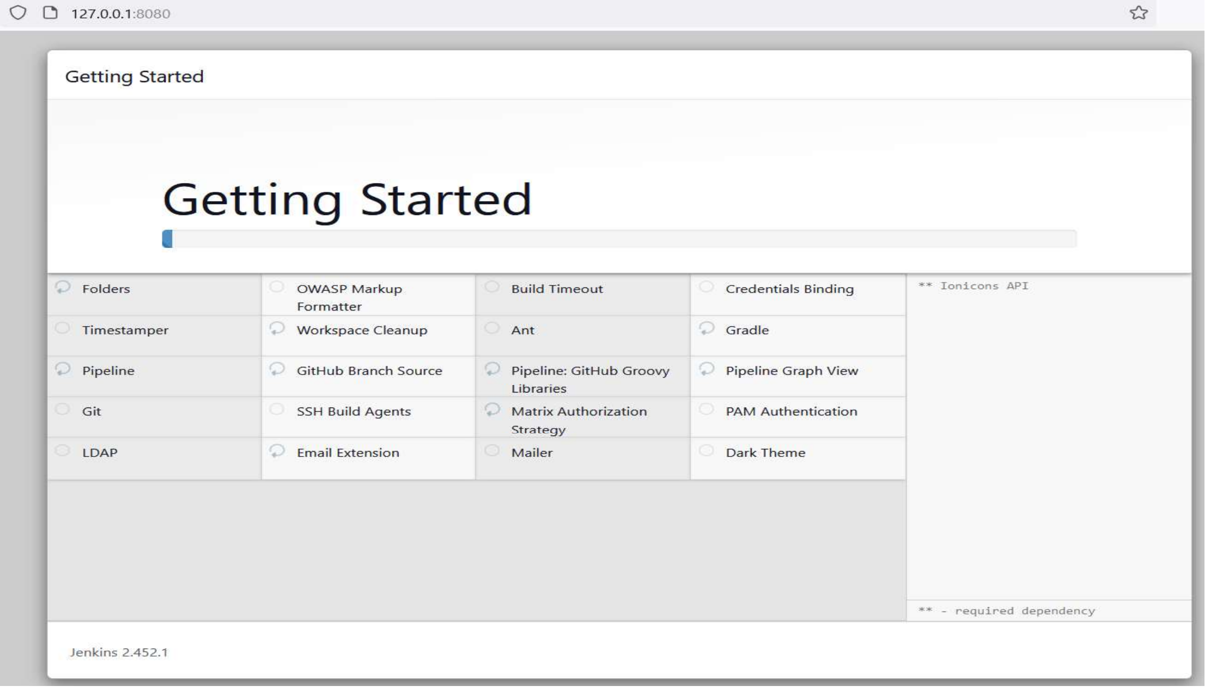
Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.452.1

You will able to see the progress of plugin installation.



Getting Started

Getting Started

Folders	OWASP Markup Formatter	Build Timeout	Credentials Binding
Timestamper	Workspace Cleanup	Ant	Gradle
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline Graph View
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication
LDAP	Email Extension	Mailer	Dark Theme

** Ionicons API

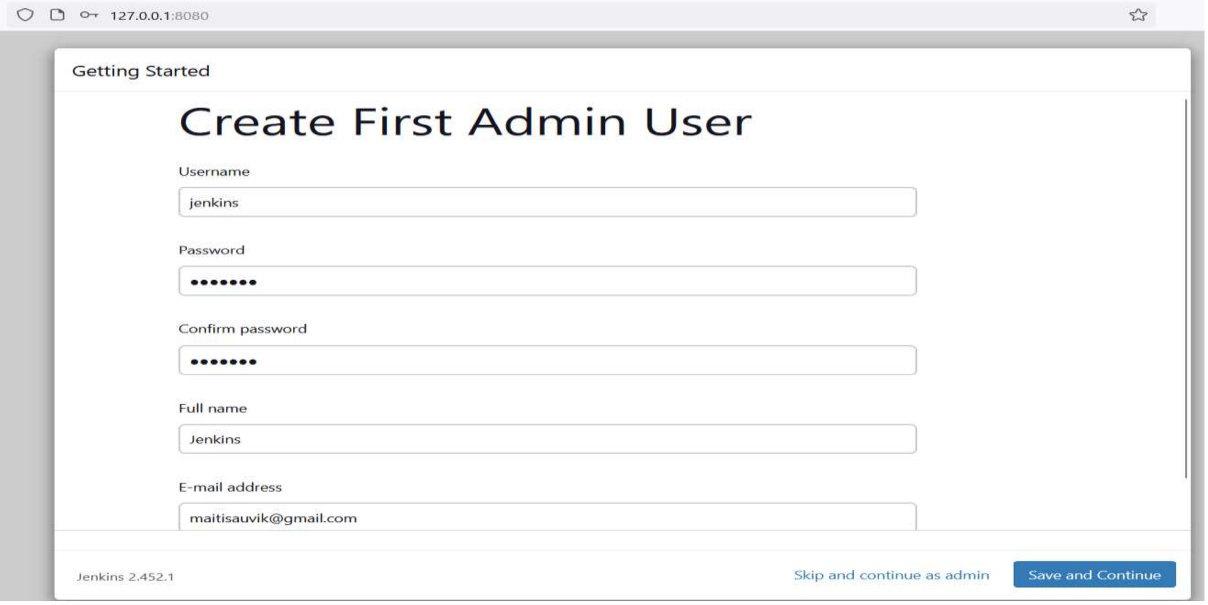
** - required dependency

Jenkins 2.452.1

Create First Admin User

Put your preferred username, password with confirming password, full name of user and e-mail id of user.

Click on Save and continue



Create First Admin User

Username: jenkins

Password: •••••••

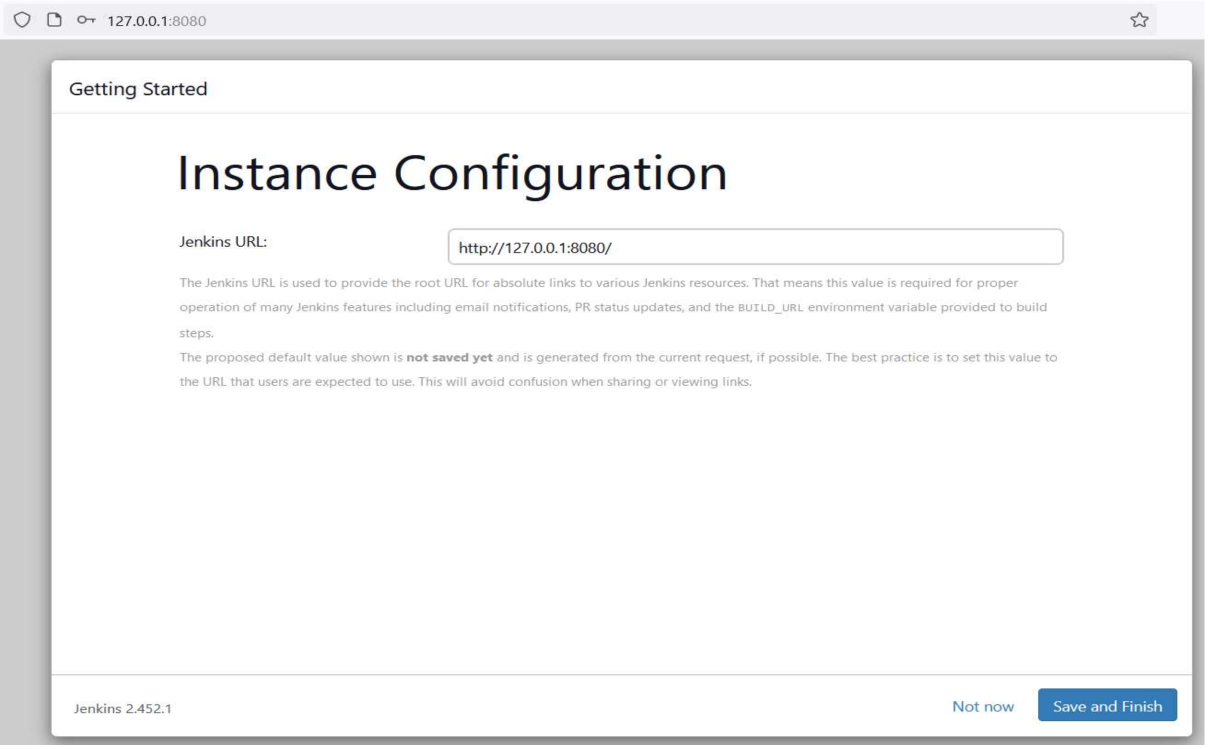
Confirm password: •••••••

Full name: Jenkins

E-mail address: maitisauvik@gmail.com

Instance Configuration

Leave the default Jankins as it is. Click on Save and Finish to continue.



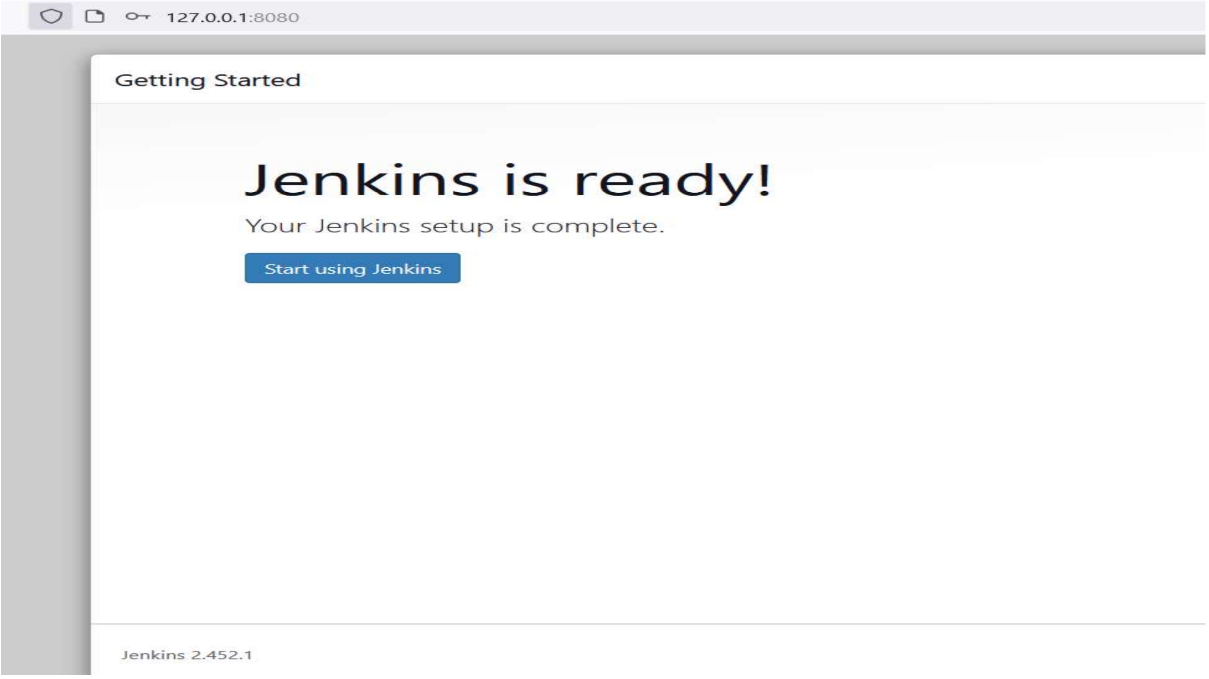
Instance Configuration

Jenkins URL: http://127.0.0.1:8080/

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

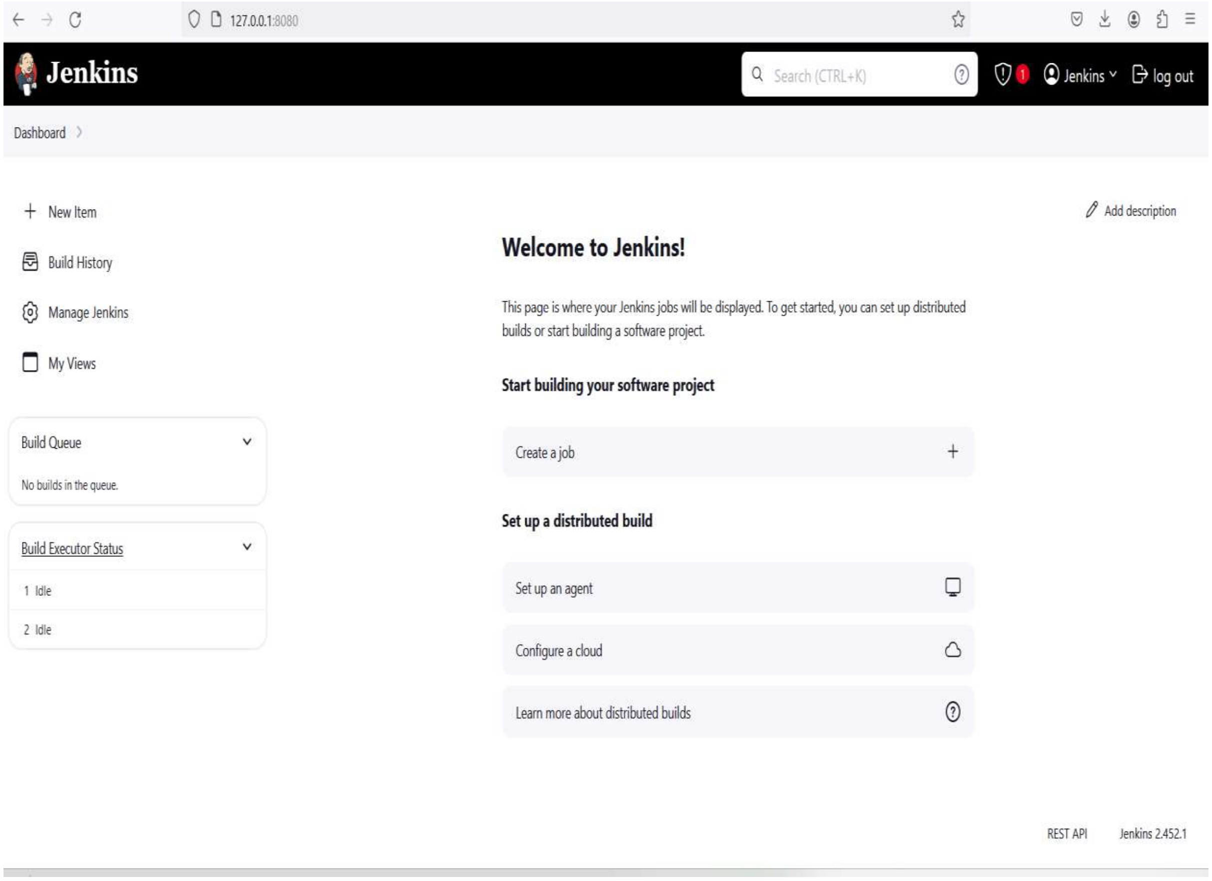
Now you should be able to see that Jenkins is ready for use.



Jenkins is ready!

Your Jenkins setup is complete.

You will get the Jenkins home page like below:



No builds in the queue.

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Set up a distributed build

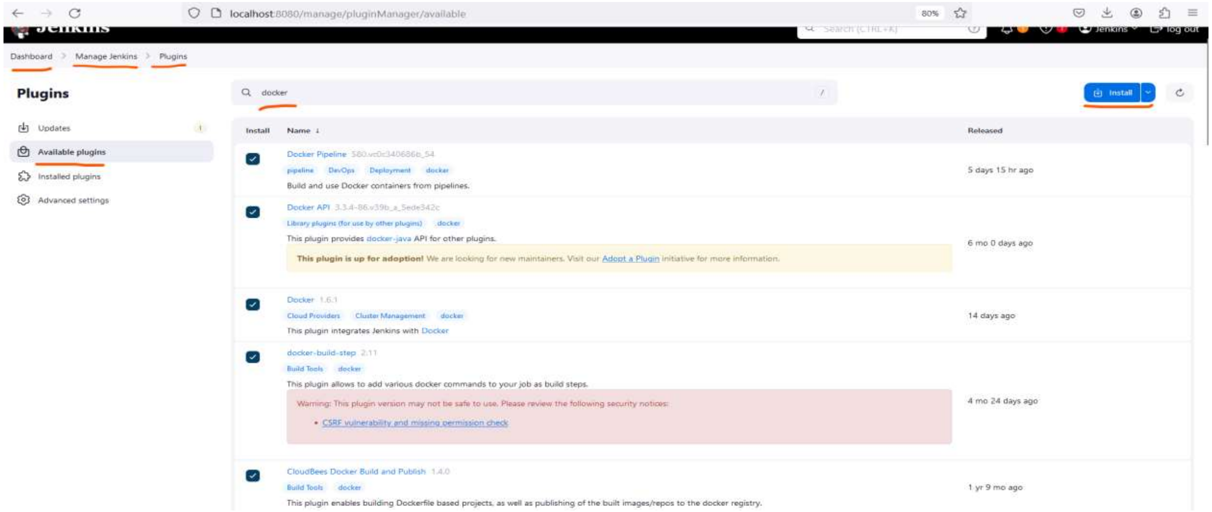
Jenkins plugins Installation

Ensure that the necessary Jenkins plugins are installed. It can be done through Jenkins Web interface: http://localhost:8080 from your windows machine

Docker plugin for Jenkins

Dashboard → Manage Jenkins → Plugins → Available plugins

Search for **docker** Select and install all docker related plugins as shown below.



localhost:8080/manage/pluginManager/available

Dashboard > Manage Jenkins > Plugins

Plugins

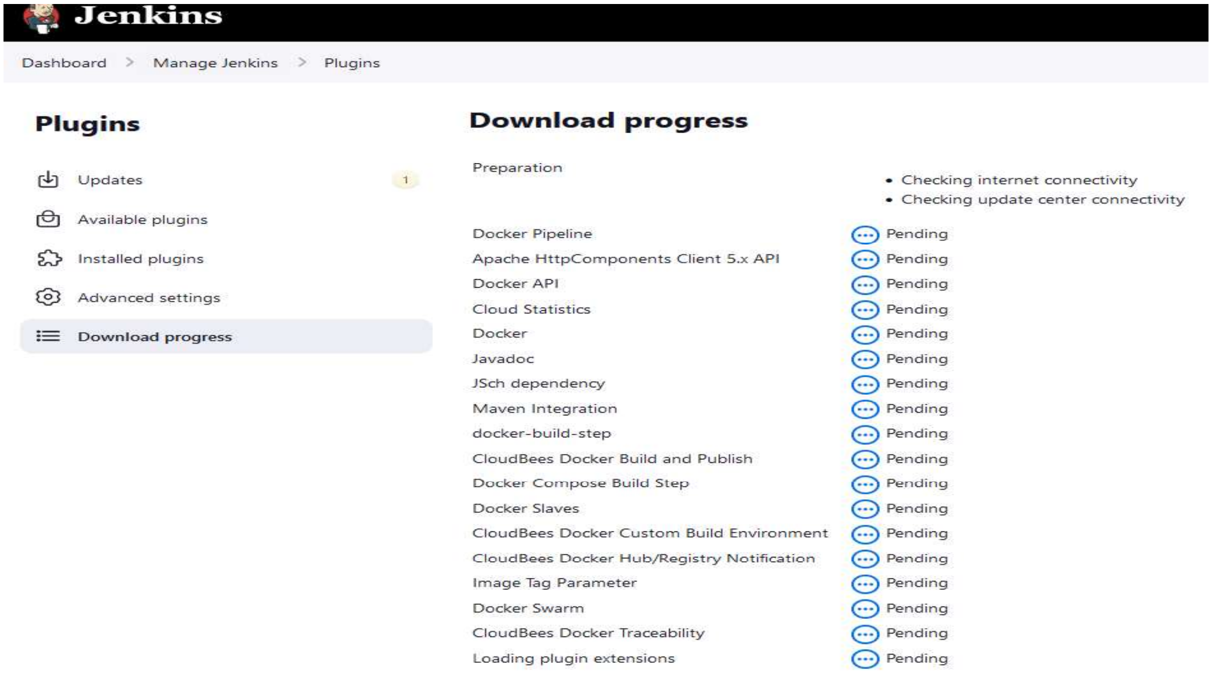
- Updates 1
- Available plugins
- Installed plugins
- Advanced settings

docker

Install

Install	Name ↓	Released
☑	Docker Pipeline 580.vc0c340686b_54 pipeline DevOps Deployment docker Build and use Docker containers from pipelines.	5 days 15 hr ago
☑	Docker API 3.3.4-86.v39b_a_5ede342c Library plugins (for use by other plugins) docker This plugin provides docker-java API for other plugins. **This plugin is up for adoption!** We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.	6 mo 0 days ago
☑	Docker 1.6.1 Cloud Providers Cluster Management docker This plugin integrates Jenkins with Docker	14 days ago
☑	docker-build-step 2.11 Build Tools docker This plugin allows to add various docker commands to your job as build steps. Warning: This plugin version may not be safe to use. Please review the following security notices: • CSRF vulnerability and missing permission check	4 mo 24 days ago
☑	CloudBees Docker Build and Publish 1.4.0 Build Tools docker This plugin enables building Dockerfile based projects, as well as publishing of the built images/repos to the docker registry.	1 yr 9 mo ago

Once you click "Install", you will see the progress.



Jenkins

Dashboard > Manage Jenkins > Plugins

Plugins

- Updates 1
- Available plugins
- Installed plugins
- Advanced settings
- Download progress

Download progress

Preparation

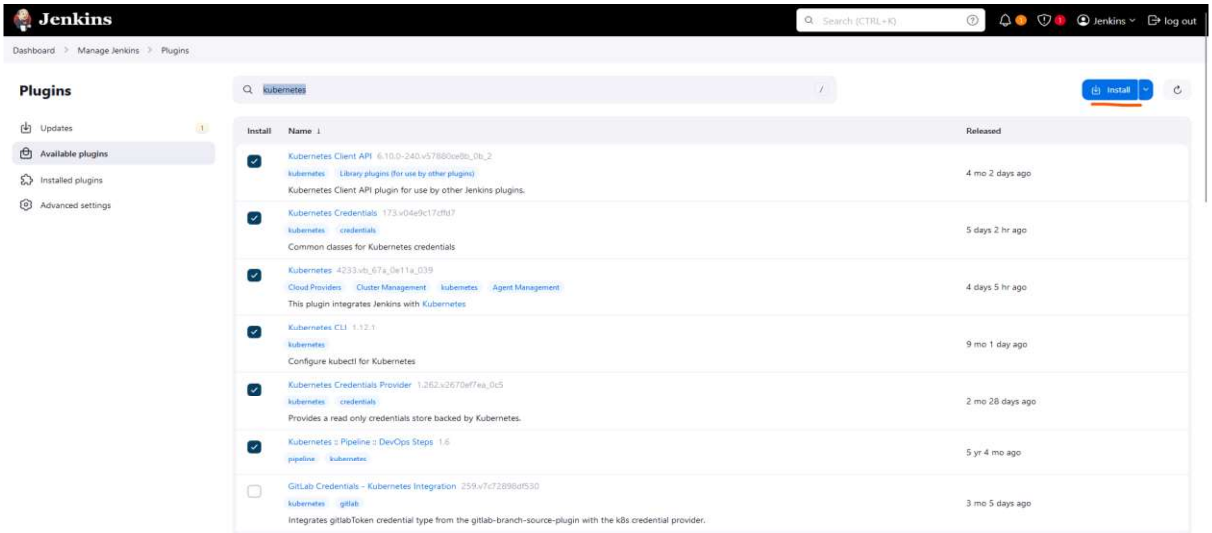
- Checking internet connectivity
- Checking update center connectivity

Docker Pipeline	Pending
Apache HttpComponents Client 5.x API	Pending
Docker API	Pending
Cloud Statistics	Pending
Docker	Pending
Javadoc	Pending
JSch dependency	Pending
Maven Integration	Pending
docker-build-step	Pending
CloudBees Docker Build and Publish	Pending
Docker Compose Build Step	Pending
Docker Slaves	Pending
CloudBees Docker Custom Build Environment	Pending
CloudBees Docker Hub/Registry Notification	Pending
Image Tag Parameter	Pending
Docker Swarm	Pending
CloudBees Docker Traceability	Pending
Loading plugin extensions	Pending

Kubernetes plugin for Jenkins

Jenkins URL → Dashboard → Manage Jenkins → Plugins → Available plugins

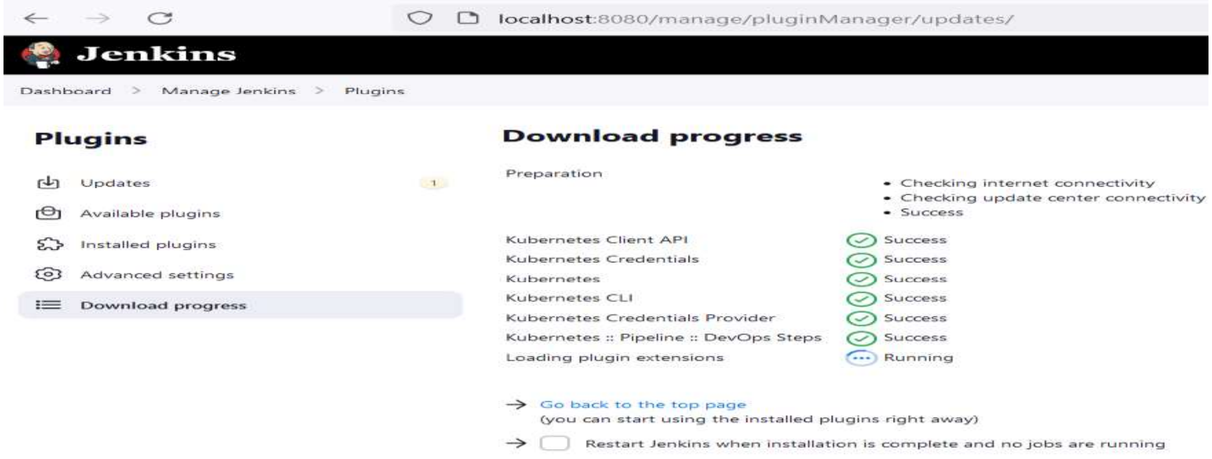
Search for **kubernetes**. Select and install all kubernetes related plugins as shown below.



Search: kubernetes

Install	Name ↓	Released
☑	Kubernetes Client API 6.10.0-240.v57880ce8b_0b_2 kubernetes Library plugins (for use by other plugins) Kubernetes Client API plugin for use by other Jenkins plugins.	4 mo 2 days ago
☑	Kubernetes Credentials 173.v04e9c17cffd7 kubernetes credentials Common classes for Kubernetes credentials	5 days 2 hr ago
☑	Kubernetes 4233.vb_67a_0e11a_039 Cloud Providers Cluster Management kubernetes Agent Management This plugin integrates Jenkins with Kubernetes	4 days 5 hr ago
☑	Kubernetes CLI 1.12.1 kubernetes Configure kubectl for Kubernetes	9 mo 1 day ago
☑	Kubernetes Credentials Provider 1.262.v2670ef7ea_0c5 kubernetes credentials Provides a read only credentials store backed by Kubernetes.	2 mo 28 days ago
☑	Kubernetes :: Pipeline :: DevOps Steps 1.6 pipeline kubernetes	5 yr 4 mo ago
☐	GitLab Credentials - Kubernetes Integration 259.v7c72898df530 kubernetes gitlab Integrates gitlabToken credential type from the gitlab-branch-source-plugin with the k8s credential provider.	3 mo 5 days ago

Once you click “Install”, you will see the progress.



Download progress

Preparation
- Checking internet connectivity
- Checking update center connectivity
- Success

Kubernetes Client API	Success
Kubernetes Credentials	Success
Kubernetes	Success
Kubernetes CLI	Success
Kubernetes Credentials Provider	Success
Kubernetes :: Pipeline :: DevOps Steps	Success
Loading plugin extensions	Running

→ Go back to the top page
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running

Once done, go to Ubuntu server where Jenkins has been installed and run the below command to restart Jenkins service:

```
sudo service jenkins restart
```

Open web browser and access Jenkins URL (http://localhost:8080/) once again.