**Install Jenkins on Ubuntu -** [**https://phoenixnap.com/kb/install-jenkins-ubuntu**](https://phoenixnap.com/kb/install-jenkins-ubuntu)

**Step 1: Install Java**

If you already have Java installed on your system, skip ahead to the next section.

It is recommended that you install OpenJDK from the default repositories. Open a terminal window and enter the following:

sudo apt update

sudo apt install openjdk-8-jdk

You’ll be asked to confirm the download and installation. Press **Y** and hit **Enter**, then allow the process to finish.

**Step 2: Add the Jenkins Repository**

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

Then add a Jenkins apt repository entry:

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

Update your local package index, then finally install Jenkins:

sudo apt-get update

sudo apt-get install jenkins

**Step 3: Set up Jenkins**

1. To launch and set up Jenkins, open a web browser, and navigate to the IP address of your server:

http://ip\_address\_or\_domain:8080

Use the actual IP address or domain name for the server you’re using Jenkins on.

2. You should see a page that prompts you to **Unlock Jenkins**. You’ll need the default password. You can get the default password by switching to a command line and entering the following:

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

3. The system returns an alphanumeric code. Enter that code, then click **Continue**.

4. Next, you are prompted to either **Install suggested plugins** or **Select plugins to install**. It’s fine to simply install the suggested plugins. You can always install more plugins later. The system continues the initial Jenkins setup.

5. Then you will be prompted to **Create First Admin User**. Enter the credentials you want to use for your Jenkins administrator, then **Save and Continue**.

6. After this, you should set up the **Instance Configuration**. This is the preferred network address for this Jenkins installation. Confirm the address you want to use for your server. This is most likely the same address you used to get to this configuration page. When you’re satisfied, click **Save and Finish**.

7. You should see a page that says **Jenkins is ready!** You can click **Start using Jenkins** to open the Jenkins dashboard.

**Step 4: Install docker**

1. Apt update
2. Apt install docker.io
3. Add Jenkins user to docker group sudo usermod -a -G docker Jenkins
4. Restart the servers ; if you getting permission error when running docker command run the following **chmod 777 /var/run/docker.sock**

**Step 5: Define webhook**

1. sudo wget -O /usr/local/bin/relay https://storage.googleapis.com/webhookrelay/downloads/relay-linux-amd64
2. sudo chmod +wx /usr/local/bin/relay
3. Login to : <https://my.webhookrelay.com/register>
4. Create new token   
     
   Graphical user interface, text, application

   Description automatically generated
5. Run the agent : relay login -k token-key-here -s token-secret-here
6. Take the URL output and define it as the webhook link in github   
     
   Graphical user interface, text

   Description automatically generated

**Install Jfrog doker repo on Ubuntu -** <https://www.jfrog.com/confluence/display/JFROG/JFrog+Container+Registry>

1. Disable IPv6 - <https://www.thegeekdiary.com/how-to-disable-ipv6-on-ubuntu-18-04-bionic-beaver-linux/>

1. Edit **/etc/default/grub** and append **ipv6.disable=1** to **GRUB\_CMDLINE\_LINUX** and **GRUB\_CMDLINE\_LINUX\_DEFAULT** like the following sample:  
FROM:

GRUB\_CMDLINE\_LINUX\_DEFAULT=""

GRUB\_CMDLINE\_LINUX=""

TO:

GRUB\_CMDLINE\_LINUX\_DEFAULT="ipv6.disable=1"

GRUB\_CMDLINE\_LINUX="ipv6.disable=1"

2. Run the **update-grub** command to regenerate the grub.cfg file:

# update-grub

3. Reboot the system to disable IPv6 support.

## Disabling IPv6 via sysctl settings

Alternatively, this can be done via sysctl settings.

1. To disable Ipv6 temporarily on the system apply the below sysctl settings:

$ sysctl -w net.ipv6.conf.all.disable\_ipv6=1  
$ sysctl -w net.ipv6.conf.default.disable\_ipv6=1  
$ sysctl -w net.ipv6.conf.lo.disable\_ipv6=1

2. In order to make the above sysctl changes permanent, we need to add them to **/etc/sysctl.conf** confirguation file.

net.ipv6.conf.all.disable\_ipv6=1

net.ipv6.conf.default.disable\_ipv6=1

net.ipv6.conf.lo.disable\_ipv6 = 1

3. Use “sysctl -p” to apply the changes from /etc/sysctl.conf file.

# sysctl -p

4. An output of 1 from the below command verifies that the IPv6 has been successfully disabled.

# cat /proc/sys/net/ipv6/conf/all/disable\_ipv6

1. Install docker : apt-get install docker.io
2. Download deb file
3. Install : dpkg -i <file-name>

Start Artifactory with:

> systemctl start artifactory.service

Check Artifactory status with:

> systemctl status artifactory.service

Installation directory was set to /opt/jfrog/artifactory

You can find more information in the log directory /opt/jfrog/artifactory/var/log

System configuration templates can be found under /opt/jfrog/artifactory/var/etc

Copy any configuration you want to modify from the template to /opt/jfrog/artifactory/var/etc/system.yaml

Triggering migration script, this will migrate if needed ...

Processing triggers for ureadahead (0.100.0-21) ...

Processing triggers for systemd (237-3ubuntu10.39) ...

1. service artifactory start
2. browse to <http://,server-ip.:8081>
   1. admin
   2. password
3. View log : tail -f $ARTIFACTORY\_HOME/logs/artifactory.log
4. Create local repository
5. Create virtual repository and assign the local repo \*
6. Create user
7. Click on Jfrom Container registry
   1. General 🡪 HTTP Setting
   2. Docker access: Repository path
   3. Server provided : embedded tomcat
   4. Public server name : this what you will use on the next step

\*The default installation is based on HTTP – therefore we need to create virtual repo and update docker client to receive HTTPS https://www.jfrog.com/confluence/display/JCR6X/Getting+Started+with+JFrog+Container+Registry+as+a+Docker+Registry

1. From the docker client (From the server you want to run docker commands)
   1. Create “daemon.json” file under /etc/docker
   2. Update the file   
      {

"insecure-registries" : ["<jfrog-server-ip>:8082"]

}

* 1. In order to login run   
     docker login -u <username -step10> -p <username pass> ["<jfrog-server-ip>::8082

1. In order to upload image run the following
   1. Download image
   2. Tag the image with the remote registry – for example   
      local image name : ngnix  
      local image tag : latest   
      local image id : ad4c705f24d3  
      remote local registry key : local  
       remote local registry ip: 10.201.55.76  
        
      docker image tag ad4c705f24d3 **10.201.55.76:8082/local/nginx:latest**
   3. To upload run : docker image push **10.201.55.76:8082/local/nginx:latest**
2. Adding Jenkins plugin https://www.jfrog.com/confluence/display/JFROG/Configuring+Jenkins+Artifactory+Plug-in