How to Install Jenkins with Docker and Docker Compose on Arch Linux

atlantic.net/dedicated-server-hosting/how-to-install-jenkins-with-docker-and-docker-compose-on-arch-linux/

Hitesh Jethva April 27, 2023

Table of Contents

Jenkins is an accessible, open-source, self-hosted automation server that helps developers automate all parts of the software development process. It is based on Java and offers 1800+ plugins to support the automation of all kinds of tasks. Jenkins aims to continuously deliver your software by integrating with many testing and deployment technologies. It is cross-platform and can run on all major platforms including, macOS, Linux, and Windows.

In this post, we will show you how to install Jenkins on Arch Linux.

Step 1 – Configure the Repository

By default, the default repository is outdated in Arch Linux, so you will need to modify the default mirror list. You can do it by editing the mirror list configuration file:

```
nano  /etc/pacman.d/mirrorlist
```

Remove all lines and add the following lines:

```
## Score: 0.7, United States
Server = http://mirror.us.leaseweb.net/archlinux/$repo/os/$arch
## Score: 0.8, United States
Server = http://lug.mtu.edu/archlinux/$repo/os/$arch
Server = http://mirror.nl.leaseweb.net/archlinux/$repo/os/$arch
## Score: 0.9, United Kingdom
Server = http://mirror.bytemark.co.uk/archlinux/$repo/os/$arch
## Score: 1.5, United Kingdom
Server = http://mirrors.manchester.m247.com/arch-linux/$repo/os/$arch
Server = http://archlinux.dcc.fc.up.pt/$repo/os/$arch
## Score: 6.6, United States
Server = http://mirror.cs.pitt.edu/archlinux/$repo/os/$arch
## Score: 6.7, United States
Server = http://mirrors.acm.wpi.edu/archlinux/$repo/os/$arch
## Score: 6.8, United States
Server = http://ftp.osuosl.org/pub/archlinux/$repo/os/$arch
## Score: 7.1, India
Server = http://mirror.cse.iitk.ac.in/archlinux/$repo/os/$arch
## Score: 10.1, United States
Server = http://mirrors.xmission.com/archlinux/$repo/os/$arch
```

Save and close the file, then update all the package indexes with the following command:

```
pacman -Syu
```

Step 2 – Install Docker

In this post, we will use Docker to deploy Jenkins. You can install Docker and Docker Compose using the following command.

```
pacman -S docker docker-compose
```

Once the Docker is installed, enable the Docker service to start at system reboot.

```
systemctl enable docker
```

Next, restart your system to apply the system.

```
reboot
```

Step 3 – Create a Docker Compose File for Jenkins

First, create directories for Jenkins using the following command.

```
mkdir jenkins jenkins_home
```

Next, navigate to the Jenkins directory and create a docker-compose.yml file.

```
cd jenkins
nano docker-compose.yml
```

Add the following configurations.

```
version: '3.7'
services:
  jenkins:
    image: jenkins/jenkins:lts
    privileged: true
    user: root
    ports:
      - 8080:8080
      - 50000:50000
    container_name: jenkins-lts
    volumes:
      - ~/jenkins_home:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
      - /usr/local/bin/docker:/usr/local/bin/docker
```

Save and close the file when you are done.

Step 4 – Start Jenkins Container

At this point, the docker-compose.yml file is ready to launch the Jenkins container. Run the following command inside the Jenkins directory to create a container.

```
docker-compose up -d
```

You should see the following output.

```
[+] Running 14/14
 ✔ jenkins 13 layers [            ]      0B/0B      Pulled
24.0s
   ✔ 32fb02163b6b Pull complete
8.9s
   ✔ c09d5e9e1188 Pull complete
14.3s
   ✔ a56533012712 Pull complete
15.0s
   ✔ 7936e107ffe7 Pull complete
15.1s
   ✔ 3ca683058265 Pull complete
15.2s
   ✔ c2ecd304b4b8 Pull complete
17.4s
   ✔ be3512d810d6 Pull complete
17.5s
   ✔ 56b37d7c2a7a Pull complete
17.9s
   ✔ 99ed1e723e52 Pull complete
22.5s
   ✔ 256db5485b13 Pull complete
22.6s
   ✔ ee8c7eaf5e6b Pull complete
22.7s
   ✔ 509f66c2f317 Pull complete
22.8s
   ✔ 820296a845d6 Pull complete
22.9s
[+] Running 2/2
 ✔ Network jenkins_default  Created
0.1s
 ✔ Container jenkins-lts    Started
0.6s
```

You can verify the Jenkins container status using the following command.

```
docker-compose ps
```

You will get the following output.

```
NAME                IMAGE                  COMMAND                  SERVICE
CREATED              STATUS              PORTS
jenkins-lts         jenkins/jenkins:lts   "/usr/bin/tini -- /u…"   jenkins
About a minute ago   Up About a minute   0.0.0.0:8080->8080/tcp, :::8080->8080/tcp,
0.0.0.0:50000->50000/tcp, :::50000->50000/tcp
```

You will also need Jenkins's initial admin password to perform the Jenkins web-based installation. You can get the Jenkins admin password with the following command.

```
docker exec jenkins-lts cat /var/jenkins_home/secrets/initialAdminPassword
```

You should see the password in the following output.

```
99b844a4ad13404796e1ab8bcf05edd1
```

You can also check the Jenkins logs to get the password.

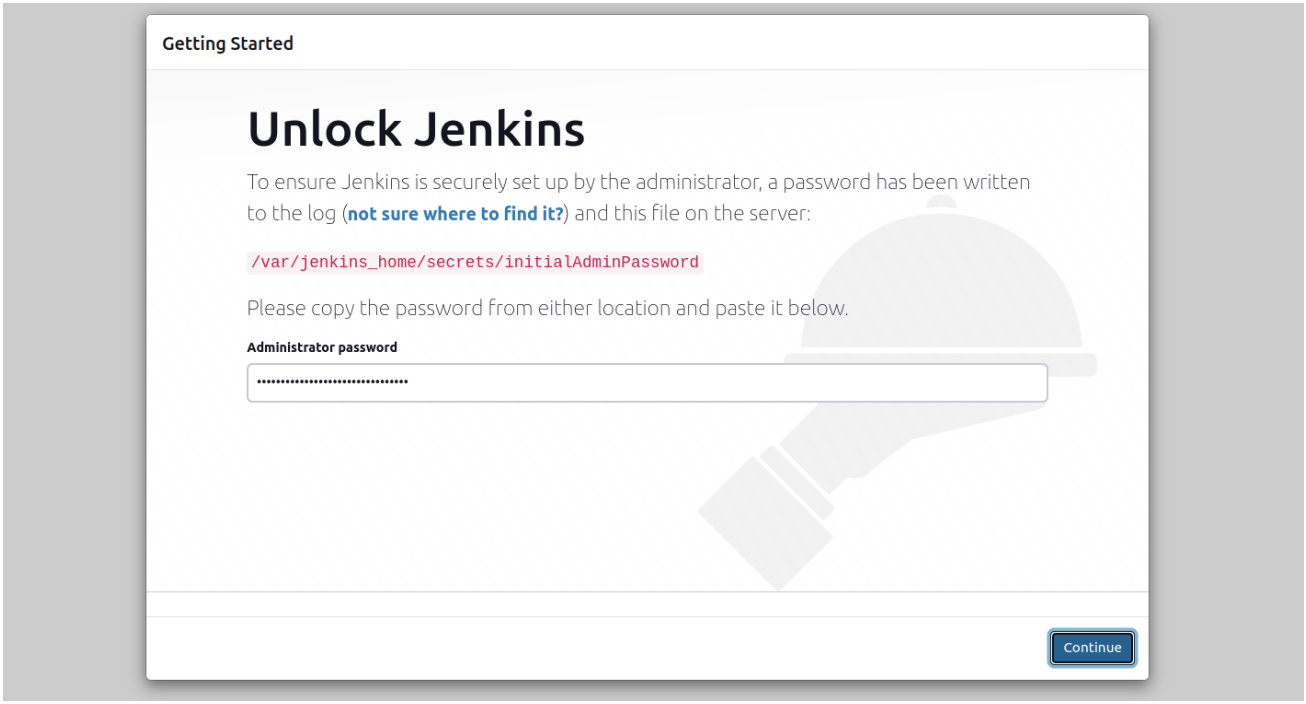
```
docker logs jenkins-lts | less
```

You should see the following output.

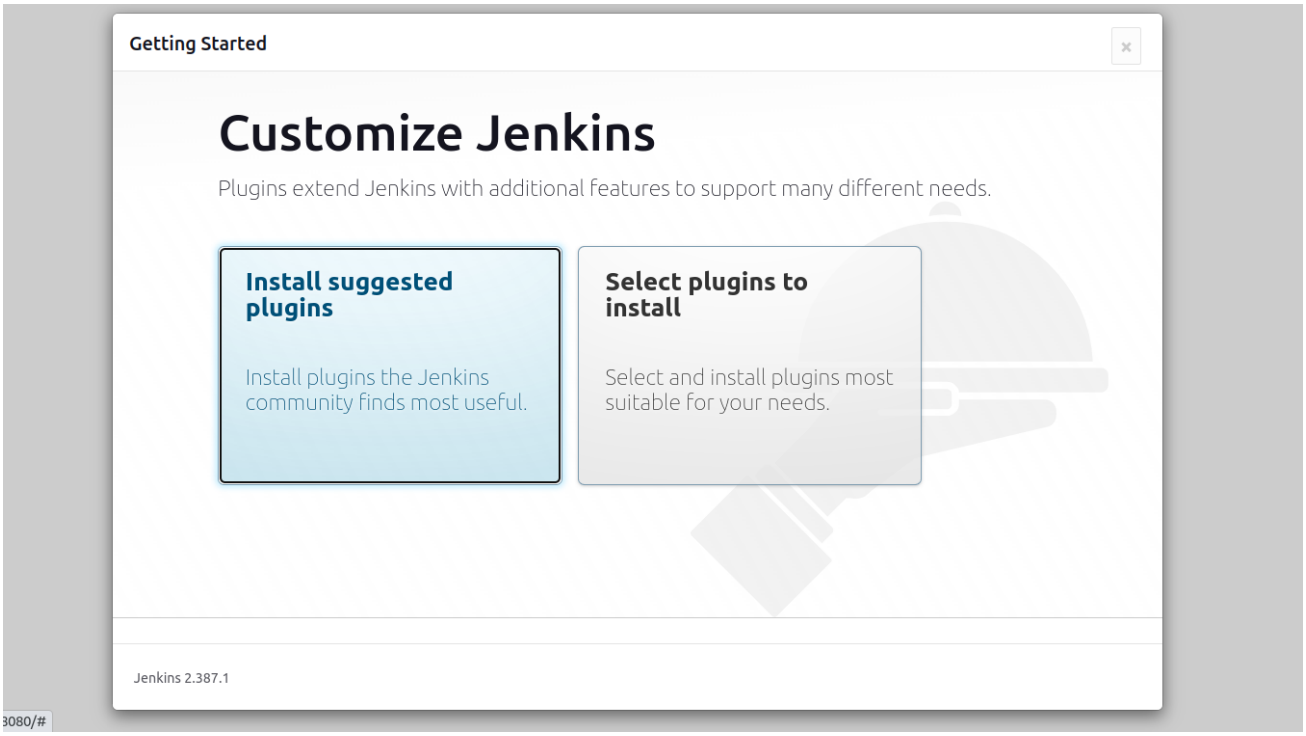
```
Jenkins initial setup is required. An admin user has been created and a password
generated.
Please use the following password to proceed to installation:

99b844a4ad13404796e1ab8bcf05edd1

Running from: /usr/share/jenkins/jenkins.war
webroot: /var/jenkins_home/war
This may also be found at: /var/jenkins_home/secrets/initialAdminPassword

*************************************************************
*************************************************************
*************************************************************

2023-03-26 05:25:19.934+0000 [id=28]    INFO
jenkins.InitReactorRunner$1#onAttained: Completed initialization
2023-03-26 05:25:19.956+0000 [id=22]    INFO    hudson.lifecycle.Lifecycle#onReady:
Jenkins is fully up and running
2023-03-26 05:25:20.151+0000 [id=44]    INFO
h.m.DownloadService$Downloadable#load: Obtained the updated data file for
hudson.tasks.Maven.MavenInstaller
2023-03-26 05:25:20.153+0000 [id=44]    INFO    hudson.util.Retrier#start: Performed
the action check updates server successfully at the attempt #1
```

Step 5 – Access Jenkins Web UI

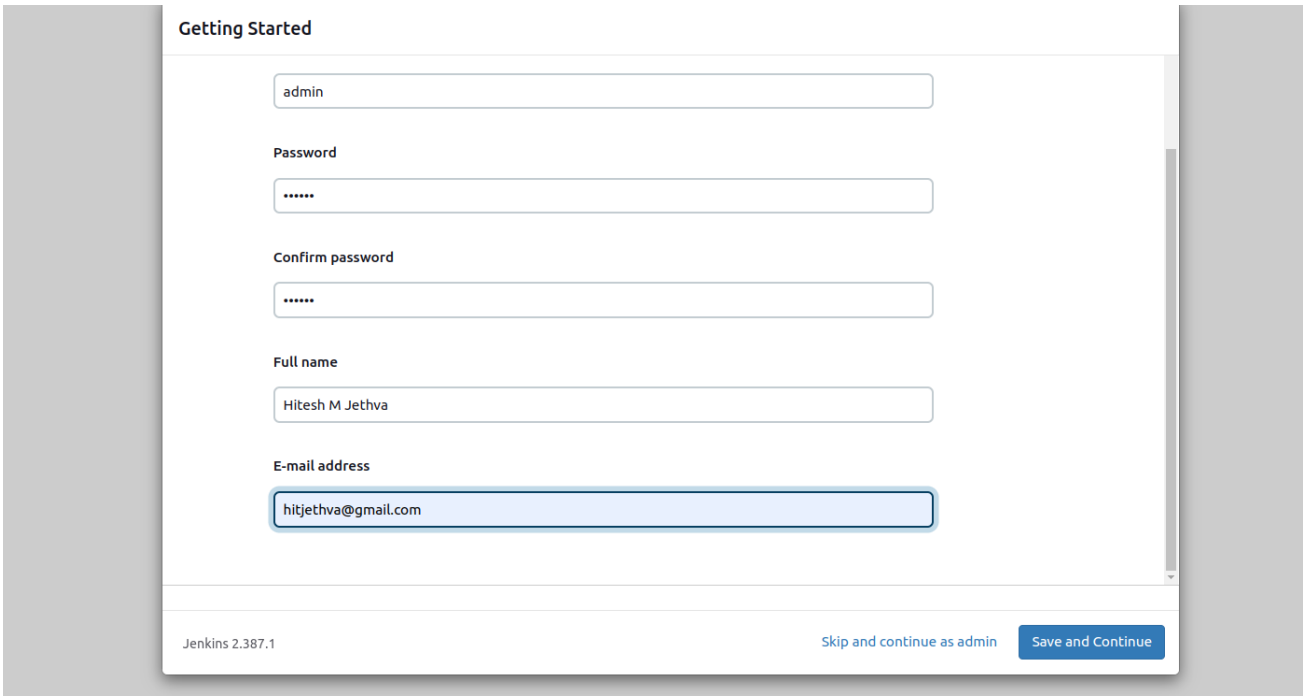
At this point, Jenkins is installed and listens on port 8080. You can now access it using the URL **http://your-server-ip:8080.** You should see the Jenkins initial setup password screen.



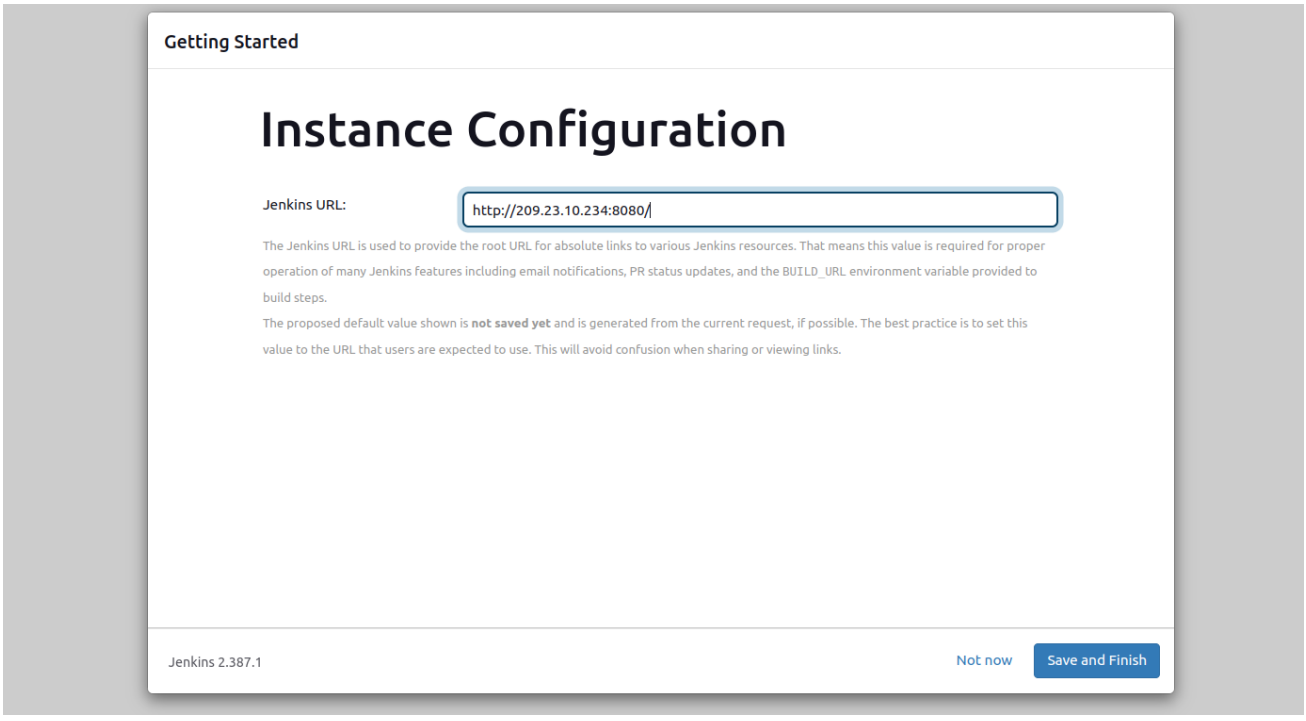
Provide your password and click on the **Continue** button. You should see the customized Jenkins screen.



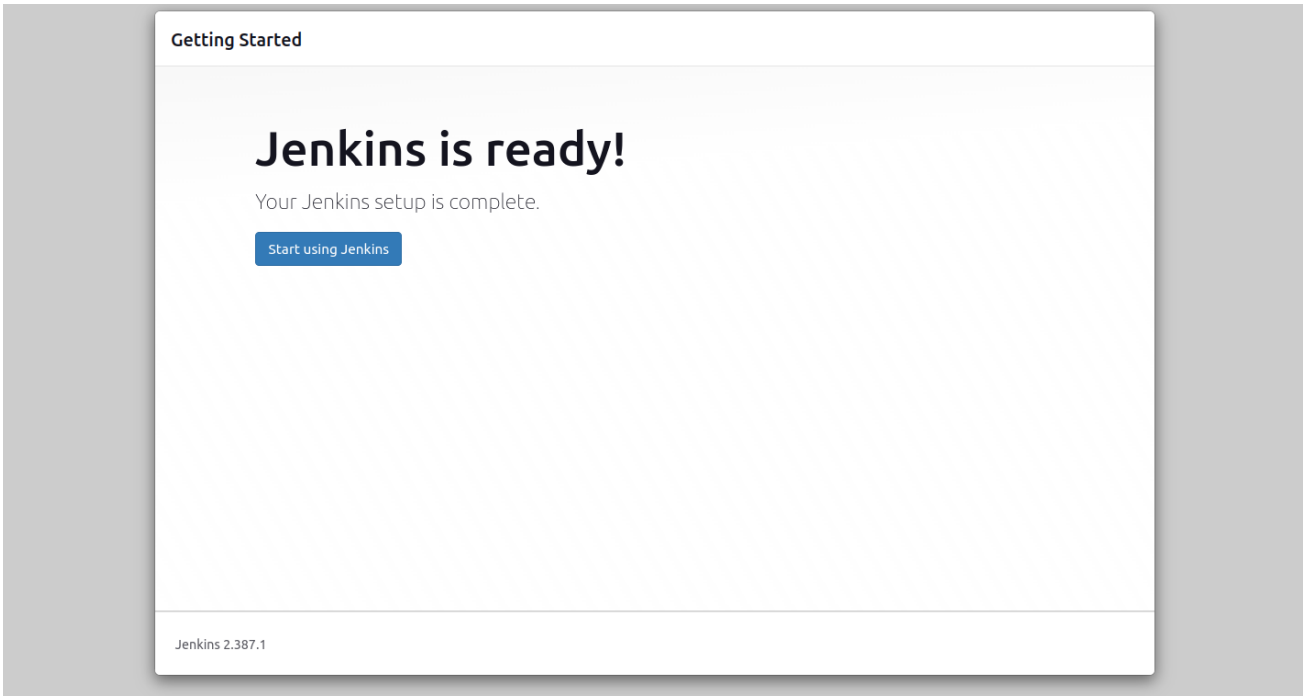
Click on Install **suggested plugins.** You should see the Getting Started screen.



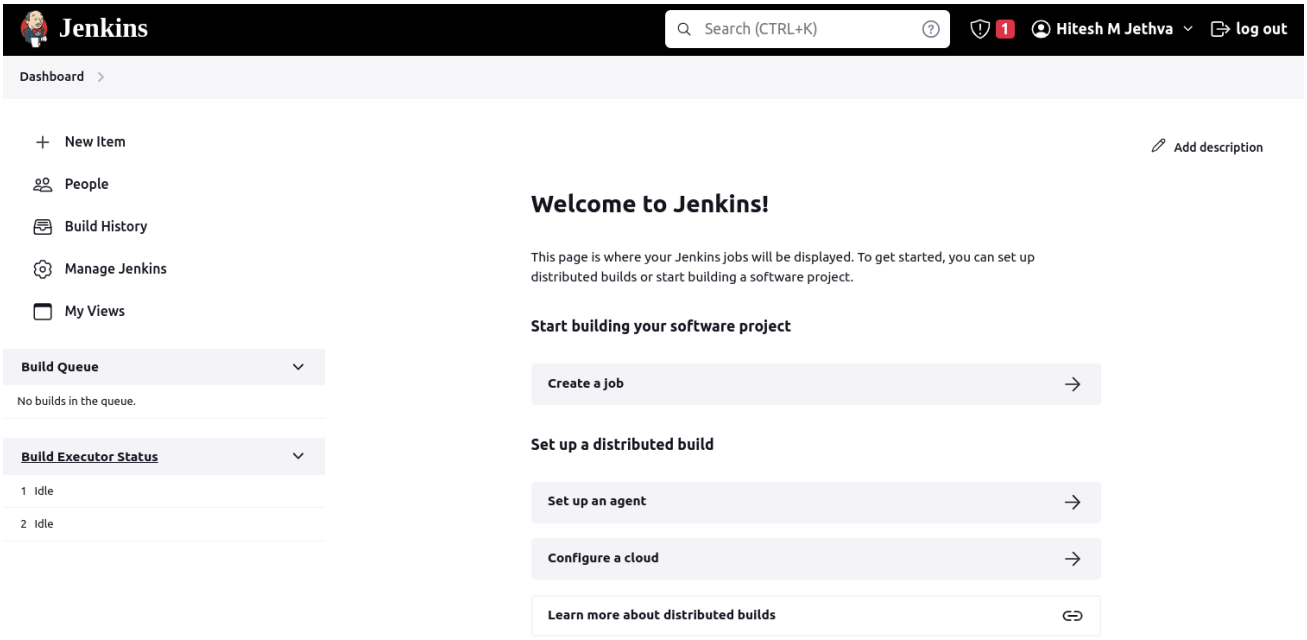
Create your new admin user and click on the **Save and Continue** buttons. You should see the instance configuration screen.



Click on the **Save and Finish** button. You should see the following screen.



Click on the **Start using Jenkins.** You should see the Jenkins dashboard.



Conclusion

In this post, we explained how to install Jenkins with Docker on Arch Linux. You can now explore the Jenkins features and implement them in your organization to automate all kinds of tasks. You can also try to deploy Jenkins on dedicated server hosting from Atlantic.Net!