



## Chapter 6 Lab: Setting Up Jenkins

Overview	1
Exercise 6.1. Set Up Jenkins	1
Cleanup	9

## Overview

In this lab, you will gain hands-on experience installing the Jenkins Server using Docker containers. The Jenkins container image is a stripped down version of Jenkins and does not contain all the plugins. We will understand how to set up Jenkins in a matter of a few minutes and create a simple job.

## Exercise 6.1. Set Up Jenkins

Jenkins provides hundreds of plugins to support building, deploying and automating any project. As an extensible automation server, Jenkins can be used as a simple CI server or turned into the continuous delivery hub for any project.

For simplicity's sake, we will use Docker containers to set up the Jenkins server.

For other installation methods refer to the [Jenkins documentation](#).

**Note: Ensure Docker is already installed on your machine.**

Set up Jenkins server on a Docker container and store Jenkins Home on Docker Volume.

1. Create the Jenkins container by executing the following command:

```
$ docker run --name myjenkins1 -dit -p 8080:8080 -p 50000:50000 -v jenkins_data:/var/jenkins_home jenkins/jenkins:lts
```

2. List the container running the following command:

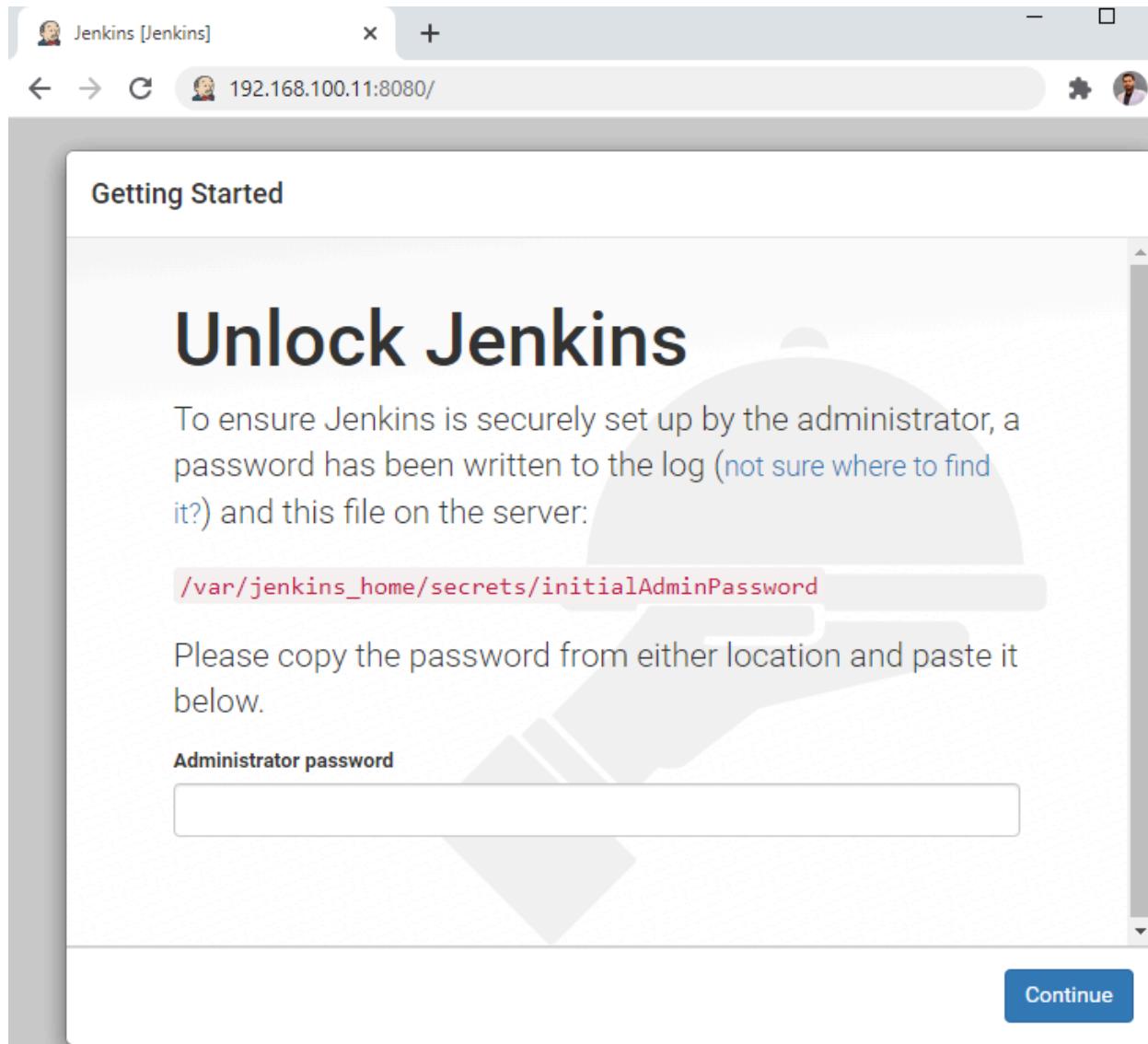
```
$ docker container ls
```

3. View the logs and capture the initial admin password for Jenkins by executing the following command:

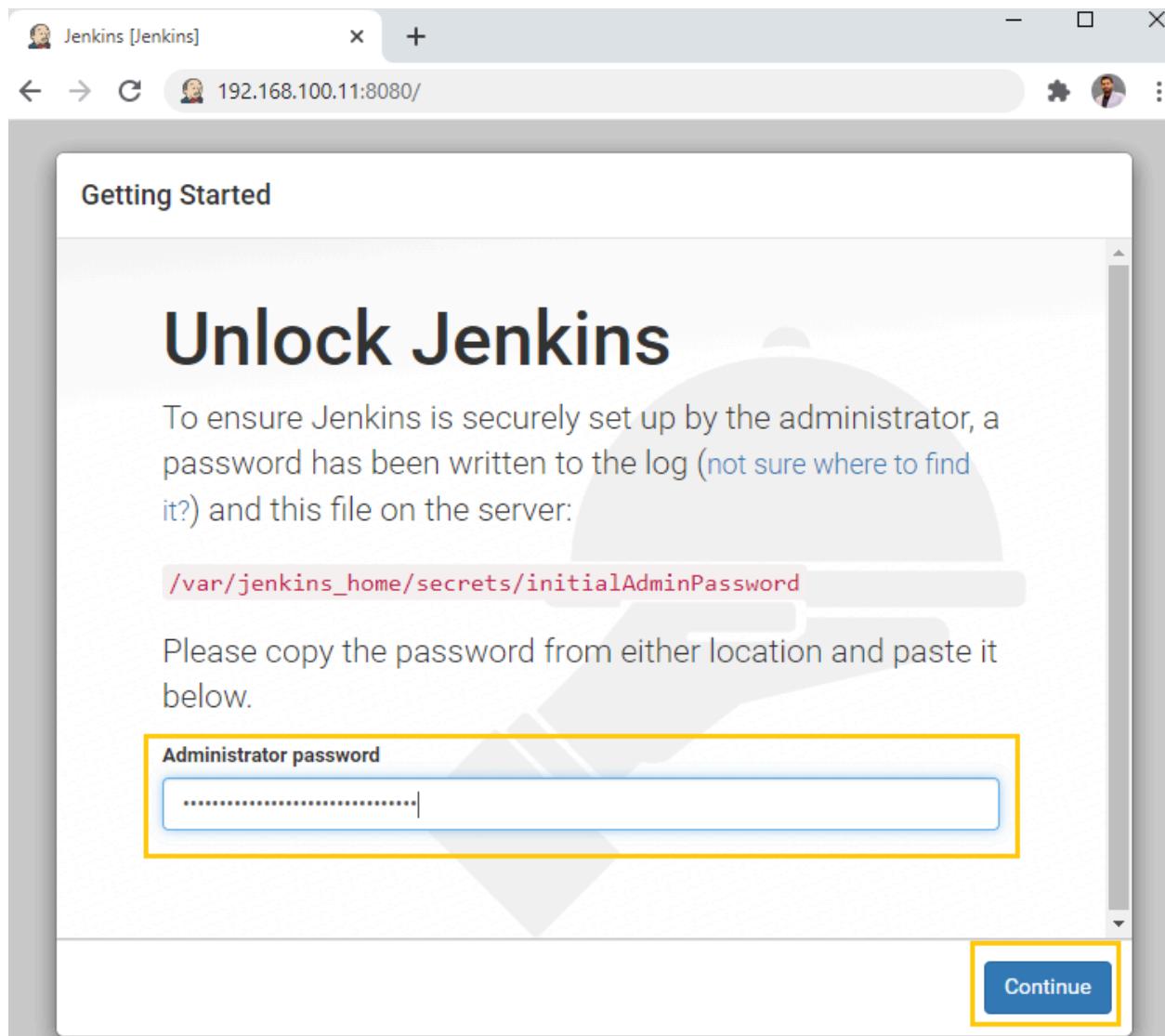
```
$ docker logs myjenkins1 | grep -B 5 initialAdminPassword
```

4. Access the Jenkins server and perform the initial configuration.

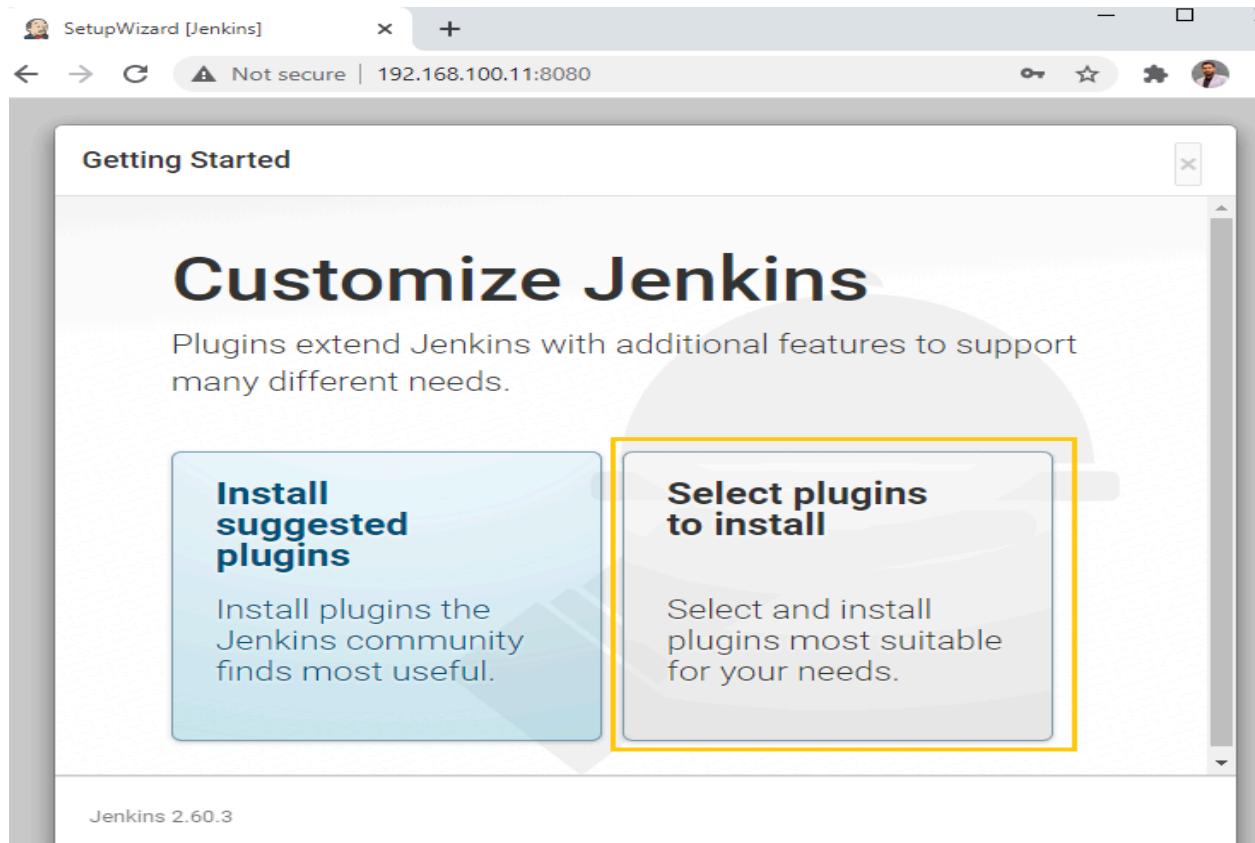
Open the browser and access the URL: `http://HostIP:8080` (`curl ifconfig.io` to get your public IP).



5. Enter the **initial admin password** and click on continue:



- Choose the “**Select Plugins to install**” option and click on “**none**”, then click on **Install**. The initial installation does not contain any plugins by default. We can install plugins of our choice. However, to keep it simple, we will choose “none” in this lab and move forward.



Getting Started

Organization and Administration

Build Features

**Build Tools**

Build Analysis and Reporting

Pipelines and Continuous Delivery

Source Code Management

Distributed Builds

User Management and Security

Notifications and Publishing

All | **None** | Suggested

Note that the full list of plugins is not shown here. Additional plugins can be installed in the [Plugin Manager](#) once the initial setup is complete. [See the Wiki](#) for more information.

## Organization and Administration (0/3)

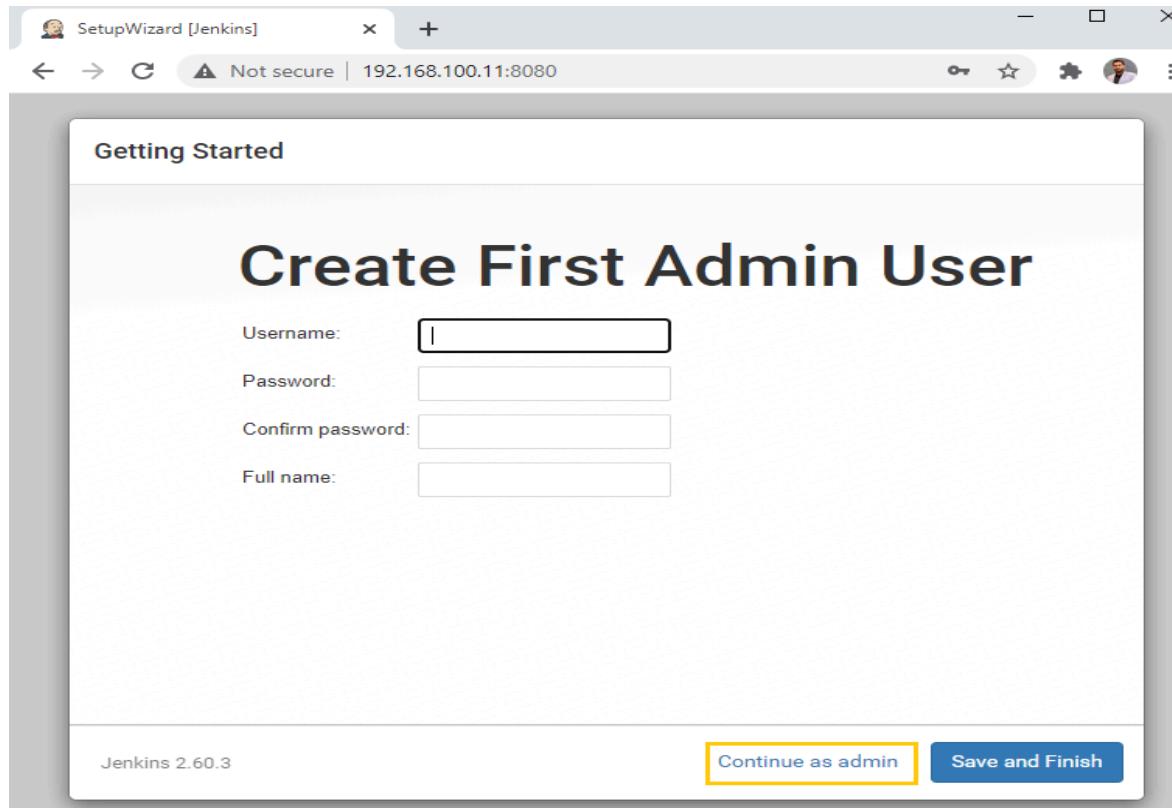
- Dashboard View** Customizable dashboard that can present various views of job information.
- Folders** This plugin allows users to create "folders" to organize jobs. Users can define custom taxonomies (like by project type, organization type etc). Folders are nestable and you can define views within folders. Maintained by CloudBees, Inc.
- OWASP Markup Formatter** Uses the [OWASP Java HTML Sanitizer](#) to allow safe-seeming HTML markup to be entered in project descriptions and the like.

## Build Features (0/10)

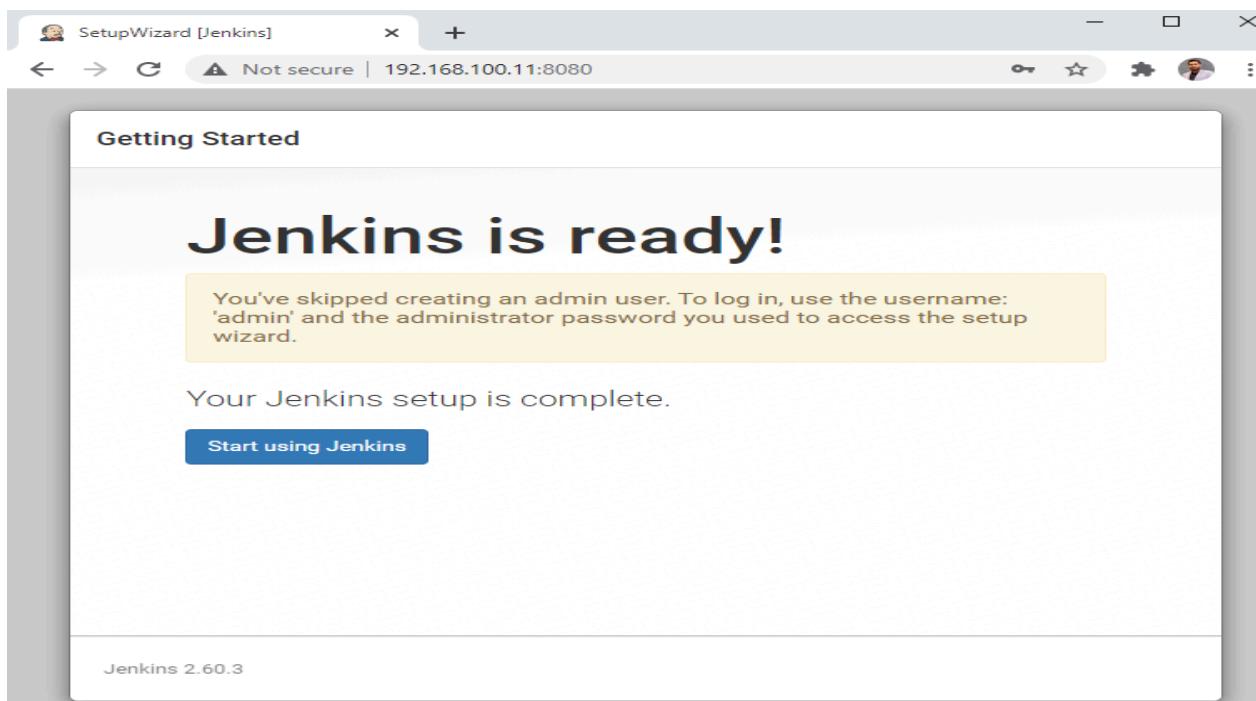
Jenkins 2.60.3

Back **Install**

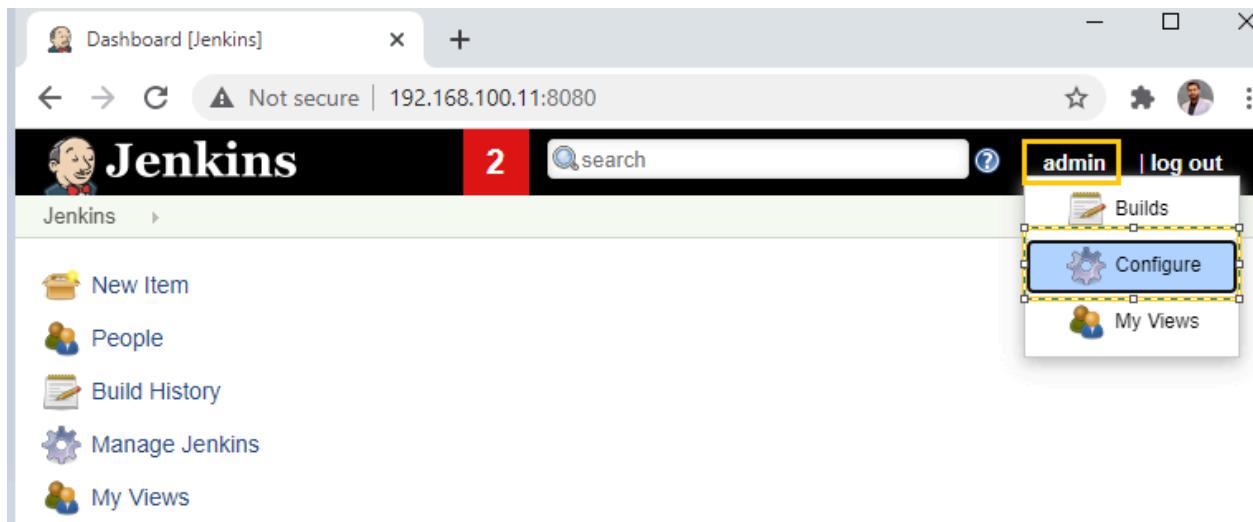
7. Admin user can be created during this step. We will skip the step and select “**Continue as admin**” and proceed to the next step.



8. Jenkins is set up and ready to use. Click on Start using Jenkins to proceed further.



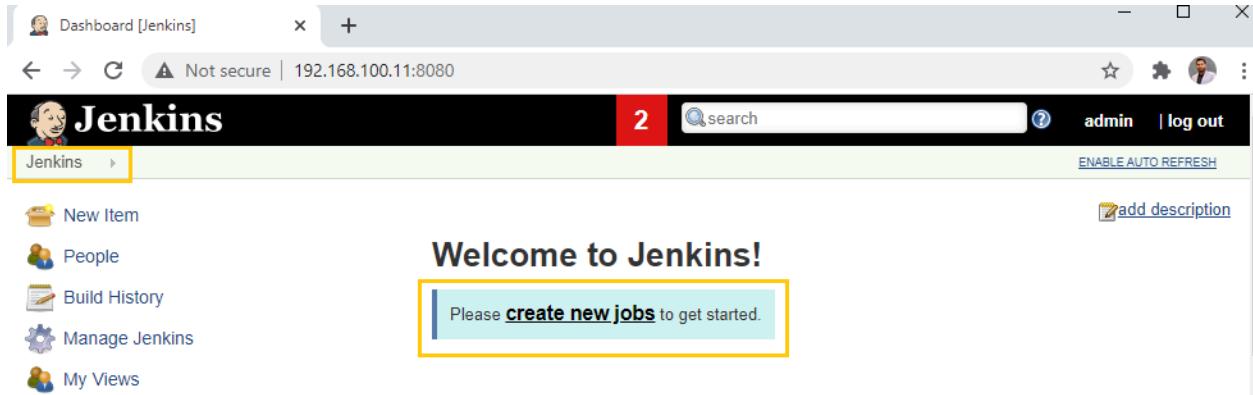
9. The initial admin password was generated during installation; reset the password of the admin user and change it to one of your choice.



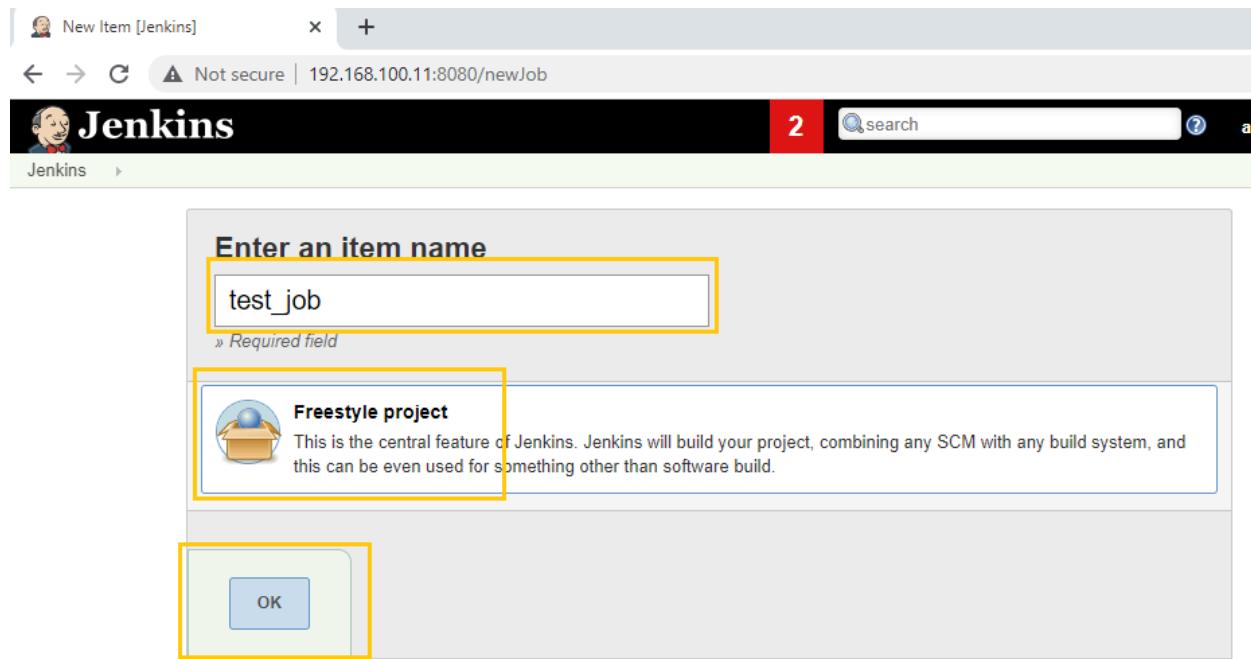
Update the Password and confirm.

A screenshot of a password update form. It has two input fields: "Password" containing "....." and "Confirm Password" also containing ".....".

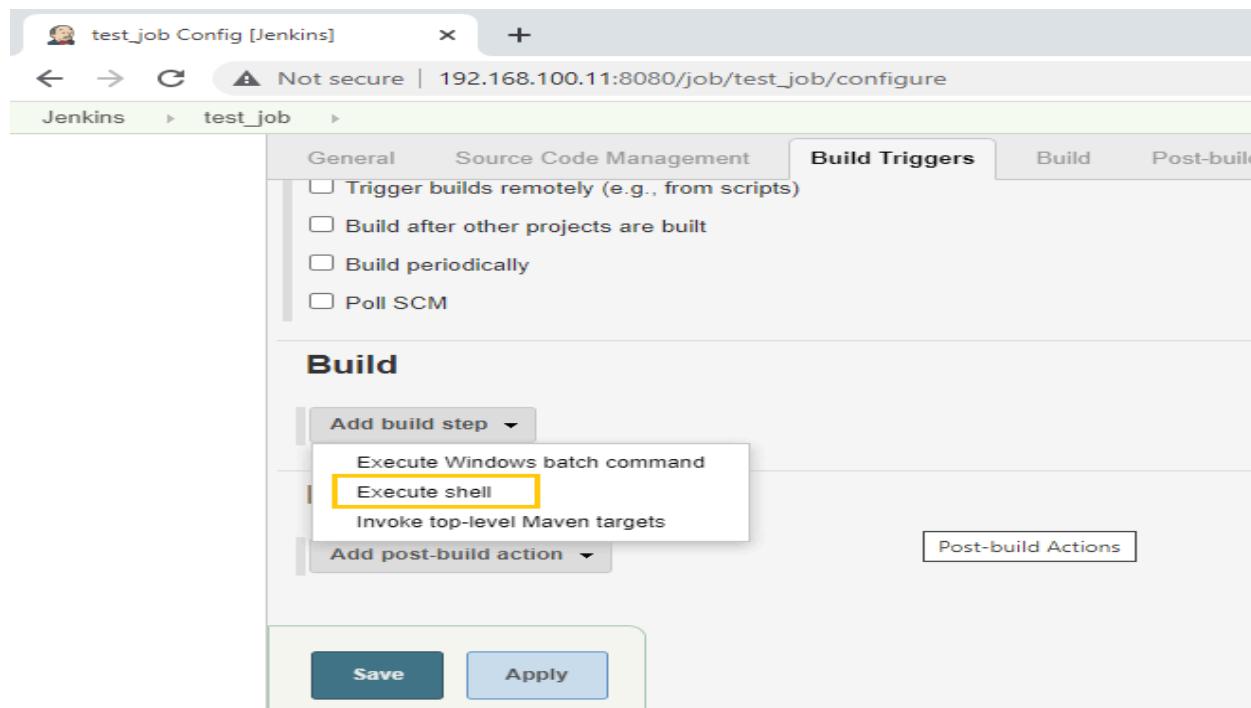
10. Now that Jenkins is ready, create a new job to test the working on Jenkins.



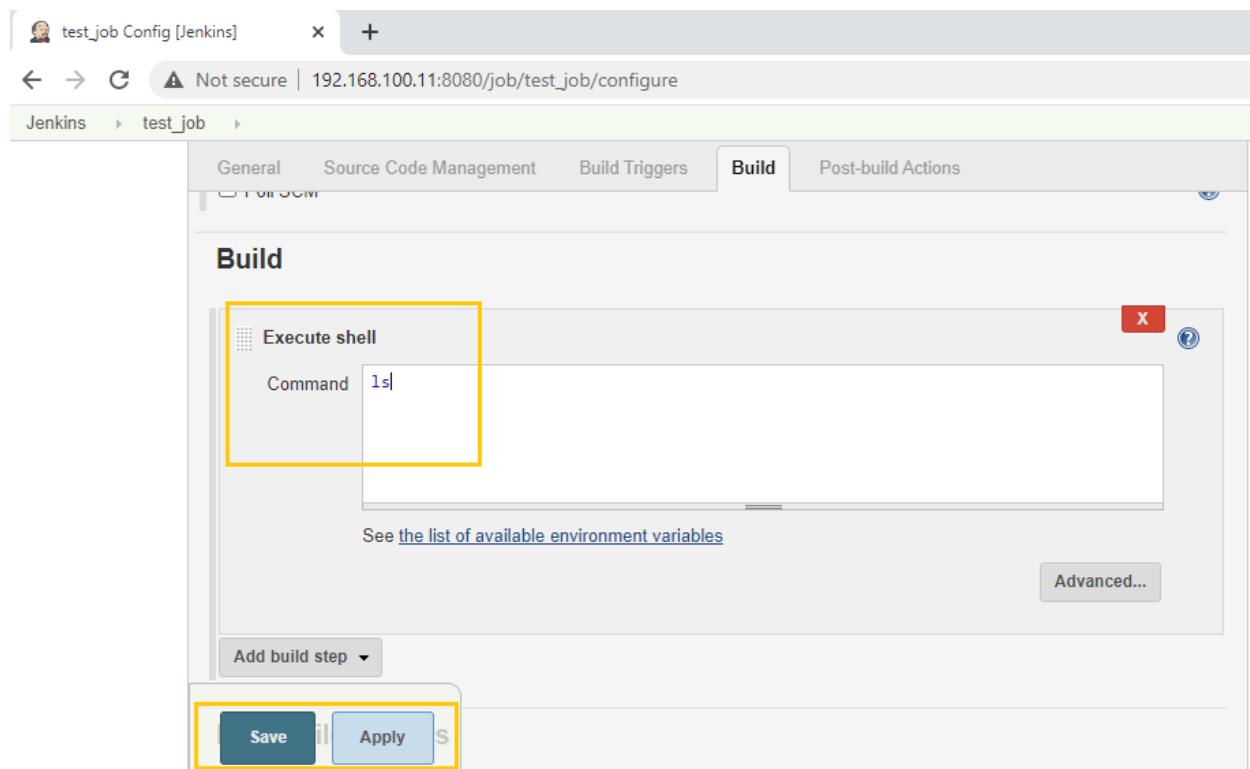
11. Notice that there are no projects available; this is because we skipped installing additional plugins during the setup. We can add plugins later, for now we will make use of the default Freestyle Project plugin available and set up a test job to execute a command.



12. Under the **Build** section, select **Execute shell** to execute a command.



13. Instead of creating a complex command, we are just executing a simple **ls** command. Type **ls** in the command box, save and exit.



14. On the Jenkins dashboard you will find the newly created Project **test\_job**.

The screenshot shows the Jenkins interface for the project 'test\_job'. The left sidebar contains links: Back to Dashboard, Status, Changes, Workspace, Build Now (which is highlighted), Delete Project, and Configure. The main content area displays the title 'Project test\_job' and two links: 'Workspace' and 'Recent Changes'. Below this is a 'Permalinks' section with a search bar and RSS links. A 'Build History' section shows a single build entry: '#1 Mar 7, 2024 7:58 AM'.

15. Click on **Build Now** on the left hand pane, to execute the command or to run the build.

The screenshot shows the Jenkins interface for the project 'test\_job'. The left sidebar shows the 'Build Now' link is now highlighted. The main content area displays the title 'Project test\_job' and two links: 'Workspace' and 'Recent Changes'. Below this is a 'Permalinks' section with a search bar and RSS links. A 'Build History' section shows two build entries: '#1 Mar 7, 2024 7:58 AM' and '#2 Mar 7, 2024 7:58 AM'.

16. The build job is completed in a few seconds; click on the build number in build history to view the console output, which displays the log output of the completed job.

The screenshot shows a browser window with the title "test\_job #1 Console [Jenkins]". The address bar indicates the URL is "Not secure 192.168.100.11:8080/job/test\_job/1/console". The Jenkins logo is at the top left, and the user "admin" is logged in at the top right. The main content area is titled "Console Output". On the left, there's a sidebar with links: "Back to Project", "Status", "Changes", "Console Output" (which is highlighted with a yellow box), "View as plain text", "Edit Build Information", and "Delete Build". The "Console Output" section contains the following log text:  
Started by user admin  
Building in workspace /var/jenkins\_home/workspace/test\_job  
[test\_job] \$ /bin/sh -xe /tmp/jenkins4594231735066938114.sh  
+ ls  
Finished: SUCCESS

17. Congratulations!!! You have successfully installed Jenkins, configured it, created and executed a simple job on Jenkins.

If you like, go ahead and explore the Jenkins Server by adding new plugins and creating pipelines.

18. Clean up by logging off the Jenkins Server, then stop and remove the **myjenkins** container, by executing the following commands.

```
$ docker container stop myjenkins1  
$ docker container rm myjenkins1
```

```
$ docker container rm `docker container ls -a -q` -f  
$ docker image rm `docker image ls -q` -f  
$ docker volume prune -f
```