

# Steps for Continuous Integration

Step:1 Create an ec2 instance of type " t2 large "

The screenshot displays the AWS Management Console interface. At the top, a green banner indicates a successful request to manage IP addresses. Below this, the 'Instances' section shows a table with one instance, 'cicd-instance', which is in a 'Running' state. The instance details panel is open, showing various attributes such as Instance ID, Public IPv4 address, Instance state, Private IP DNS name, and Instance type.

**Instances (1/1)** Info

Last updated less than a minute ago

Find Instance by attribute or tag (case-sensitive)

Instance state = running

Clear filters

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 address
<input checked="" type="checkbox"/>	cicd-instance	i-0bd0a9dc5b3028909	Running	t2.large	...	...	us-east-1a	ec2-3-219-219-199.co...	3.219.219.199

**i-0bd0a9dc5b3028909 (cicd-instance)**

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

**Instance summary** Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0bd0a9dc5b3028909 (cicd-instance)	3.219.219.199   <a href="#">open address</a>	172.31.6.102
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-3-219-219-199.compute-1.amazonaws.com   <a href="#">open address</a>
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-6-102.ec2.internal	ip-172-31-6-102.ec2.internal	-
Answer private resource DNS name	Instance type	
-	t2.large	

# Step:2 Edit the inbound rules

aws

Services

Search

[Alt+S]

N. Virginia

Pushpa

EC2

>

Security Groups

>

sg-03989ab4b4baafbe5 - launch-wizard-5

>

Edit inbound rules

Edit inbound rules

Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules

Info

Security group rule ID	Type	Info	Protocol	Info	Port range	Info	Source	Info	Description - optional	Info	
sgr-09f9aaf0d4e3cb449	SSH		TCP		22		Custom				Delete
									0.0.0.0/0		
-	All traffic		All		All		Anywher...				Delete
									0.0.0.0/0		

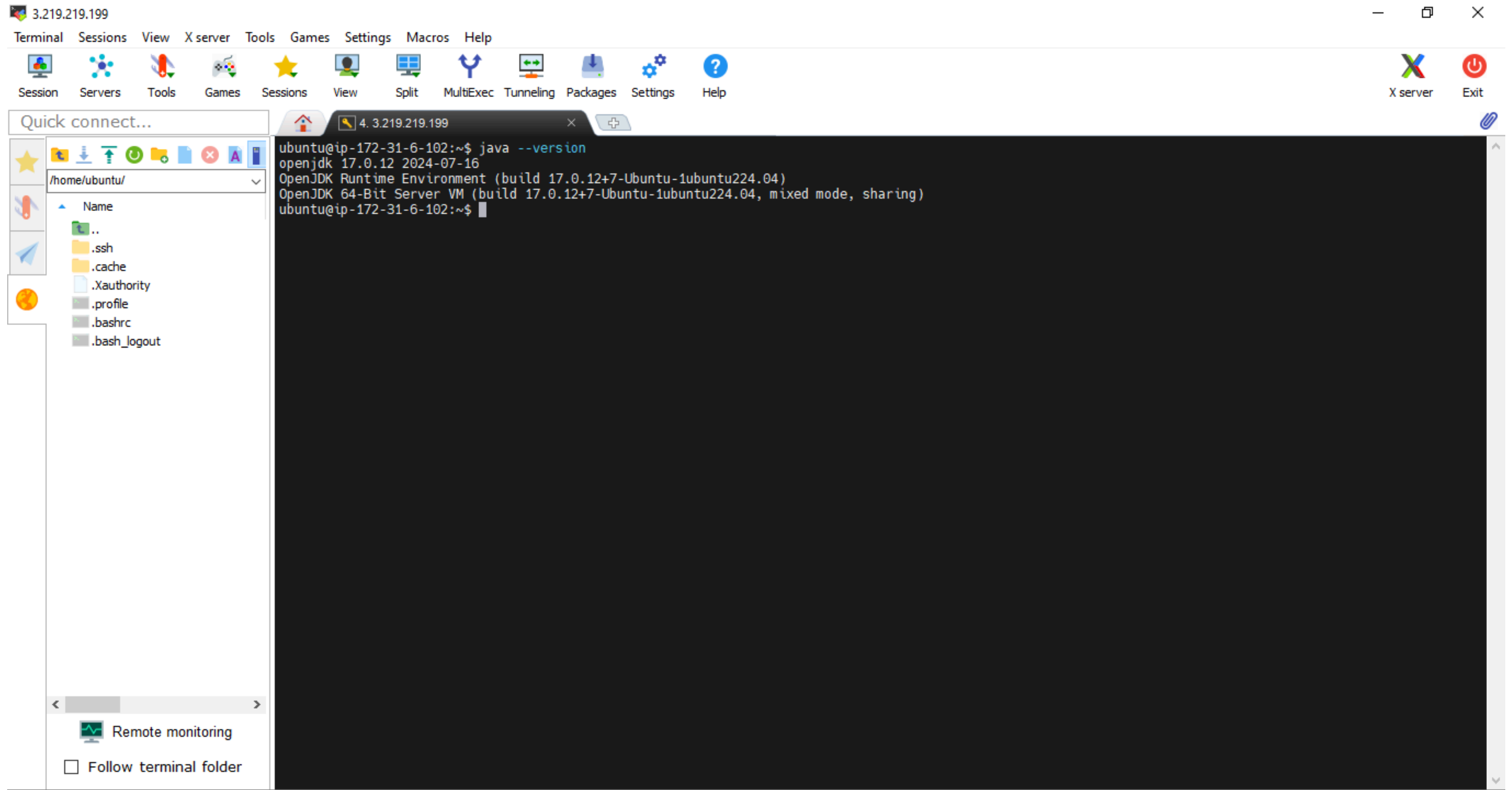
Add rule

Cancel

Preview changes

Save rules

## Step:3 connect to the ec2 instance and Install java



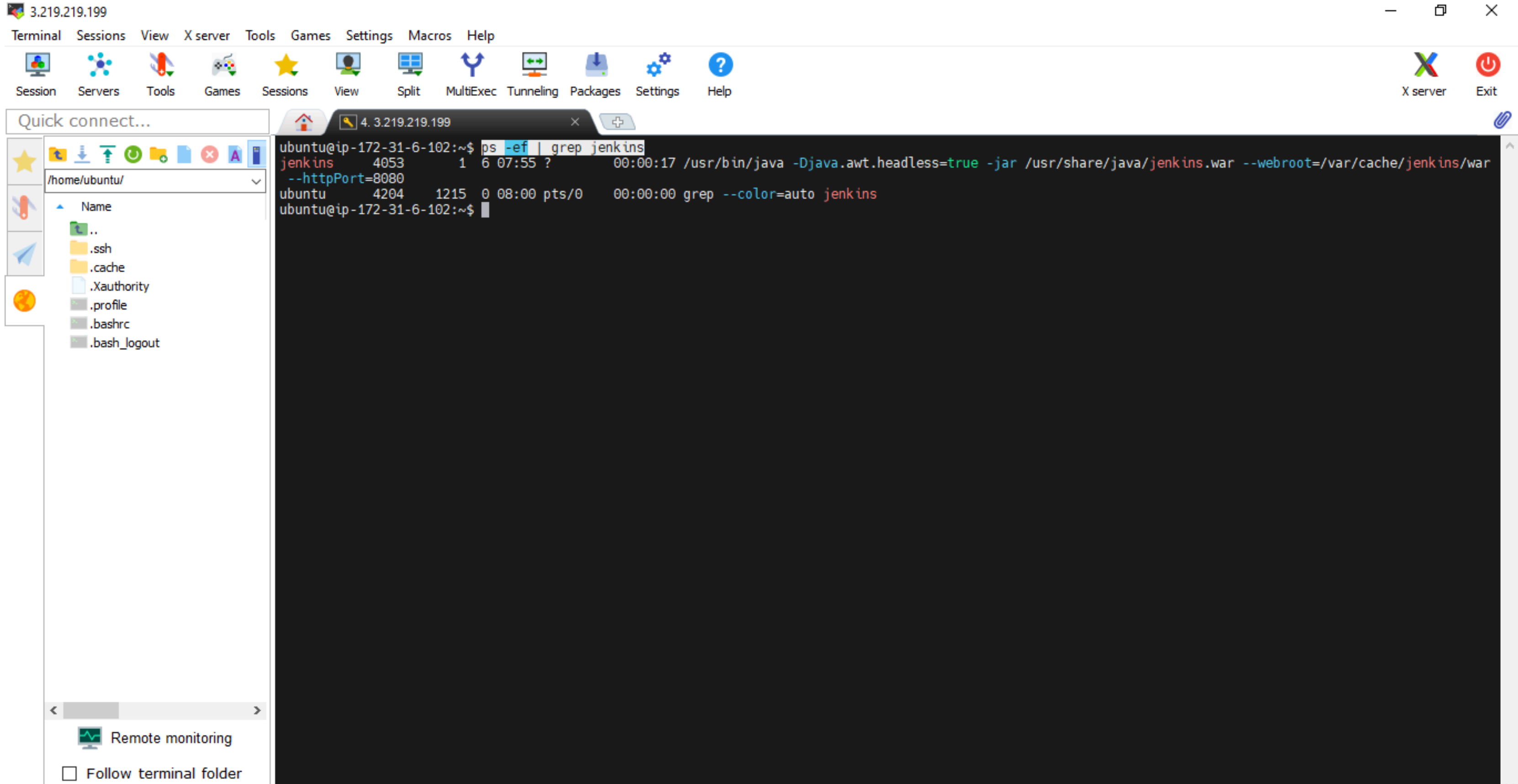
## Step:4 Install jenkins

The screenshot shows a terminal window titled "3.219.219.199" with a menu bar (Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help) and a toolbar. On the left is a "Quick connect..." sidebar with a file explorer showing the directory `/home/ubuntu/` containing files like `.ssh`, `.cache`, `.Xauthority`, `.profile`, `.bashrc`, and `.bash_logout`. The terminal displays the following commands and output:

```
ubuntu@ip-172-31-6-102:~$ curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee \
/usr/share/keyrings/jenkins-keyring.asc > /dev/null
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Ign:4 https://pkg.jenkins.io/debian binary/ InRelease
Get:5 https://pkg.jenkins.io/debian binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:8 https://pkg.jenkins.io/debian binary/ Packages [64.9 kB]
Fetched 67.8 kB in 0s (137 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... 50%
```

At the bottom of the sidebar, there are checkboxes for "Remote monitoring" (checked) and "Follow terminal folder" (unchecked).

# Verify if jenkins is installed or not



The screenshot shows a terminal window with the following content:

```
ubuntu@ip-172-31-6-102:~$ ps -ef | grep jenkins
jenkins 4053 1 6 07:55 ? 00:00:17 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war
--httpPort=8080
ubuntu 4204 1215 0 08:00 pts/0 00:00:00 grep --color=auto jenkins
ubuntu@ip-172-31-6-102:~$
```

The terminal window is titled "4. 3.219.219.199" and has a menu bar with options: Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help. The left sidebar shows a file explorer for the directory "/home/ubuntu/" with files: .., .ssh, .cache, .Xauthority, .profile, .bashrc, and .bash\_logout. The bottom of the window has a "Remote monitoring" checkbox and a "Follow terminal folder" checkbox.

step:5 Browse the public IP of the ec2 instance with port 8080 to access jenkins

← → ↻ ⚠ Not secure 3.219.219.199:8080/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:

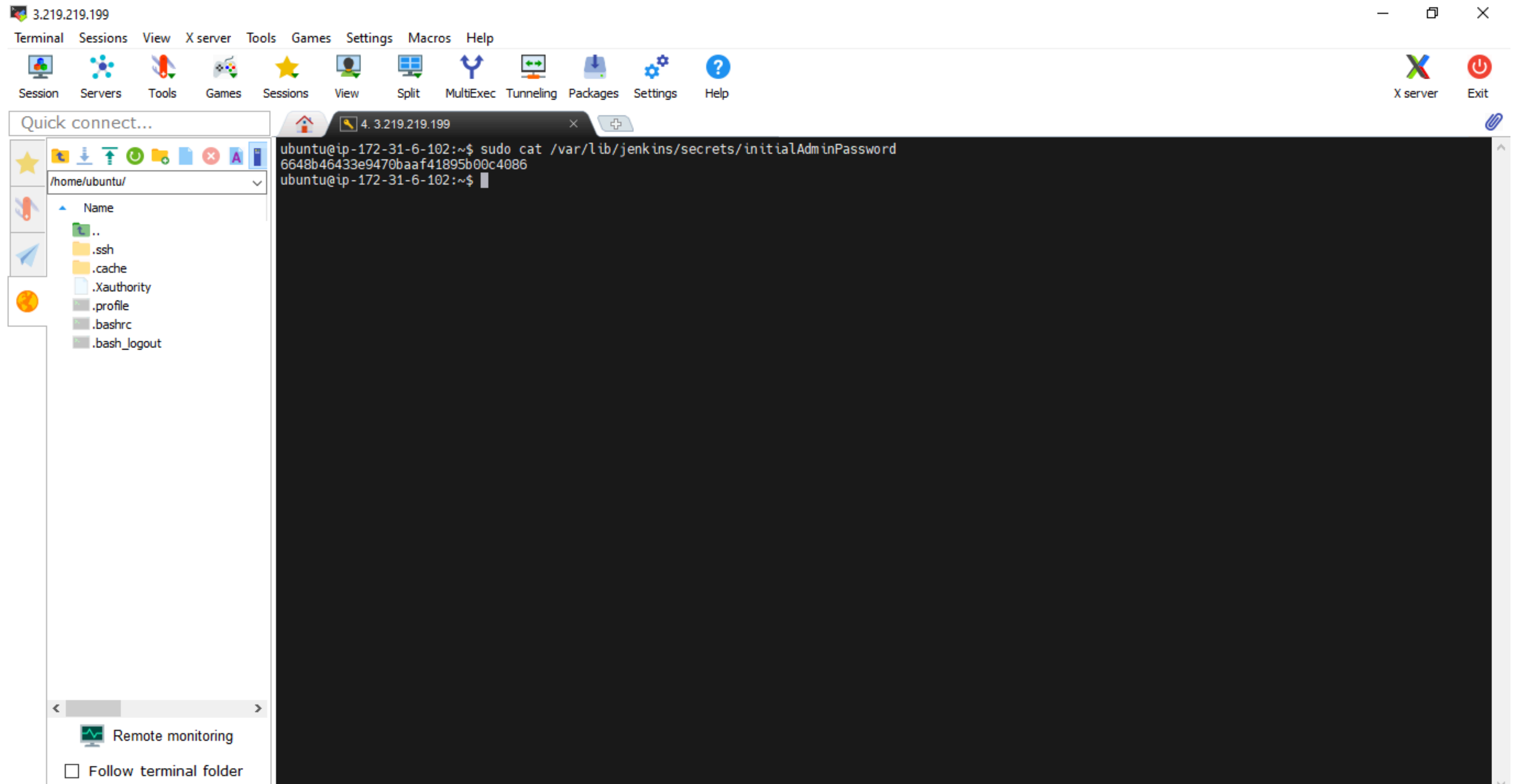
```
/var/lib/jenkins/secrets/initialAdminPassword
```

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

**Administrator password**

Continue

step:6 Copy the url and say "sudo cat url" to get the password for jenkins









step:8 Now select new item , give a name , and select pipeline and ok

← → ↻

Not secure 3.219.219.199:8080/view/all/newJob

☆


Dashboard > All > New Item

## New Item


Enter an item name

pushpa shree


Select an item type




**Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



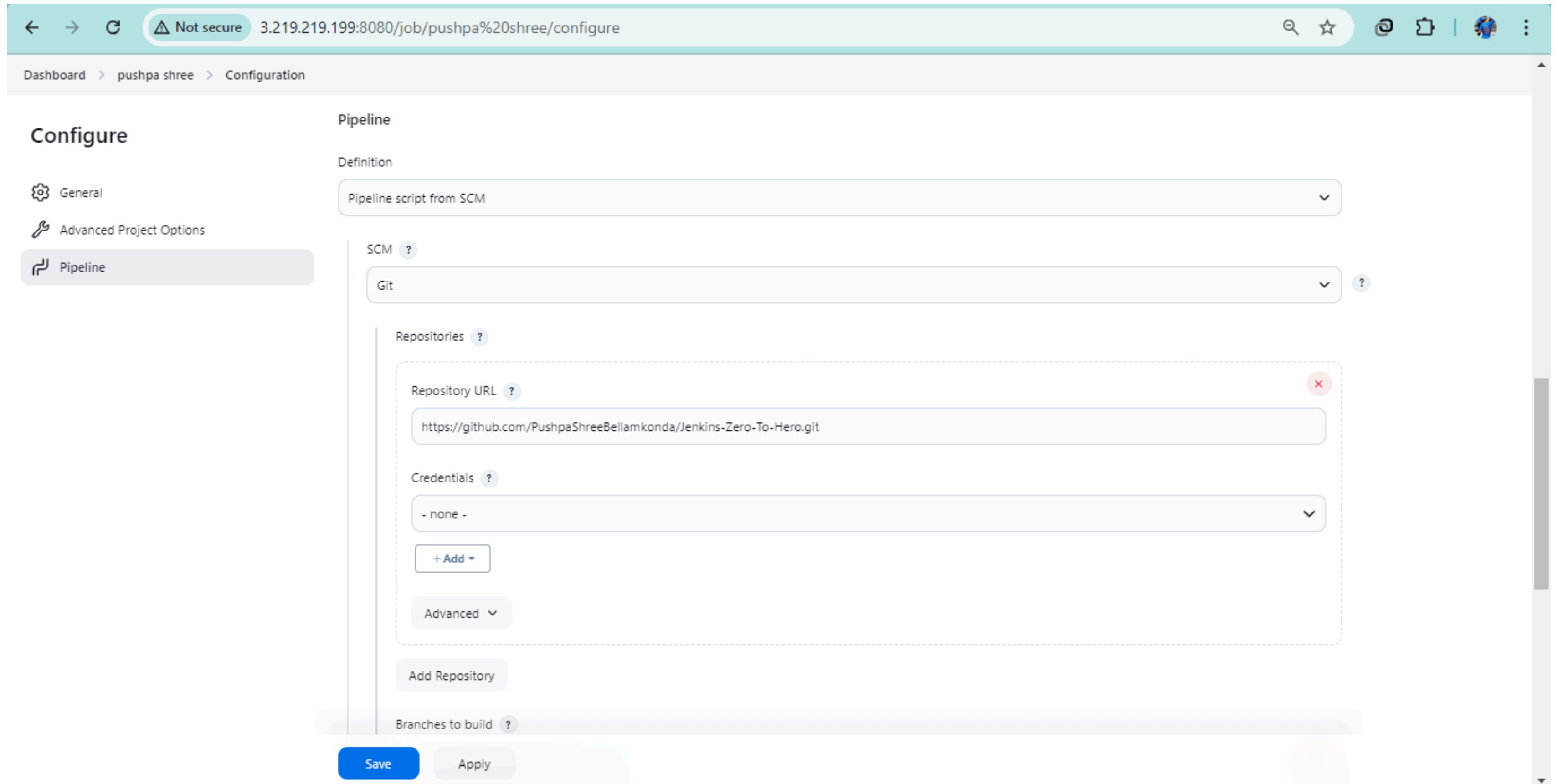
**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



**Folder**

OK

step:9 Now in the pipeline select "pipeline script from SCM" and give "github repo url" the "branch" and "Script path"



The screenshot shows the Jenkins Pipeline Configuration page. The browser address bar indicates the URL is 3.219.219.199:8080/job/pushpa%20shree/configure. The page has a breadcrumb trail: Dashboard > pushpa shree > Configuration. On the left, the 'Configure' section is active, with sub-tabs for General, Advanced Project Options, and Pipeline. The main area is titled 'Pipeline' and shows the 'Definition' as 'Pipeline script from SCM'. Below this, the 'SCM' is set to 'Git'. The 'Repositories' section contains one repository with the URL 'https://github.com/PushpaShreeBellamkonda/Jenkins-Zero-To-Hero.git' and credentials set to '- none -'. There is an '+ Add' button and an 'Advanced' dropdown. At the bottom, there is an 'Add Repository' button and a 'Branches to build' section. The page ends with 'Save' and 'Apply' buttons.

Dashboard > pushpa shree > Configuration

### Configure

- General
- Advanced Project Options
- Pipeline

#### Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/PushpaShreeBellamkonda/Jenkins-Zero-To-Hero.git

Credentials ?

- none -

+ Add

Advanced

Add Repository

Branches to build ?

Save Apply

## Configure

- ⚙ General
- 🔧 Advanced Project Options
- 🔗 Pipeline**

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▼

Script Path ?

java-maven-sonar-argocd-helm-k8s/spring-boot-app/JenkinsFile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Save

Apply

step:10 Install required plugins , " Docker Pipeline " and " Sonarqube Scanner "

←

→

↻

⚠ Not secure

3.219.219.199:8080/manage/pluginManager/available

🔍


☆

🔗

📁

👤

⋮

 **Jenkins**

🔍 Search (CTRL+K) ⓘ

🛡️ ⓘ 1

👤 Bellamkonda Pushpa Shree ▾

🚪 log out

Dashboard > Manage Jenkins > Plugins

Plugins

📄 Updates

**📁 Available plugins**

⚙️ Installed plugins

⚙️ Advanced settings

☰ Download progress

🔍 docker pipe ⓘ

📄 Install

Name ↓

Released

✓

Docker Pipeline

580.vc0c340686b\_54

pipeline DevOps Deployment docker

Build and use Docker containers from pipelines.

4 mo 16 days ago

REST API

Jenkins 2.479

## Plugins

sonar

Install



Updates



Available plugins



Installed plugins



Advanced settings



Download progress

Install

Name

Released



SonarQube Scanner 2.17.2

External Site/Tool Integrations

Build Reports

This plugin allows an easy integration of [SonarQube](#), the open source platform for Continuous Inspection of code quality.

7 mo 18 days ago



Sonar Quality Gates 315.v1f12b\_e81a\_3a\_4

Library plugins (for use by other plugins)

analysis

Other Post-Build Actions

Fails the build whenever the Quality Gates criteria in the Sonar 5.6+ analysis aren't met (the project Quality Gates status is different than "Passed")

1 mo 9 days ago



Quality Gates 2.5

Fails the build whenever the Quality Gates criteria in the Sonar analysis aren't met (the project Quality Gates status is different than "Passed")

Warning: This plugin version may not be safe to use. Please review the following security notices:

- [Credentials transmitted in plain text](#)

8 yr 4 mo ago



Sonargraph Integration 5.0.2

External Site/Tool Integrations

Build Reports

Other Post-Build Actions

This plugin integrates [Sonargraph](#) functionality into Jenkins, for Sonargraph versions 9 and 10

1 yr 3 mo ago



CodeSonar 3.5.0

DevOps

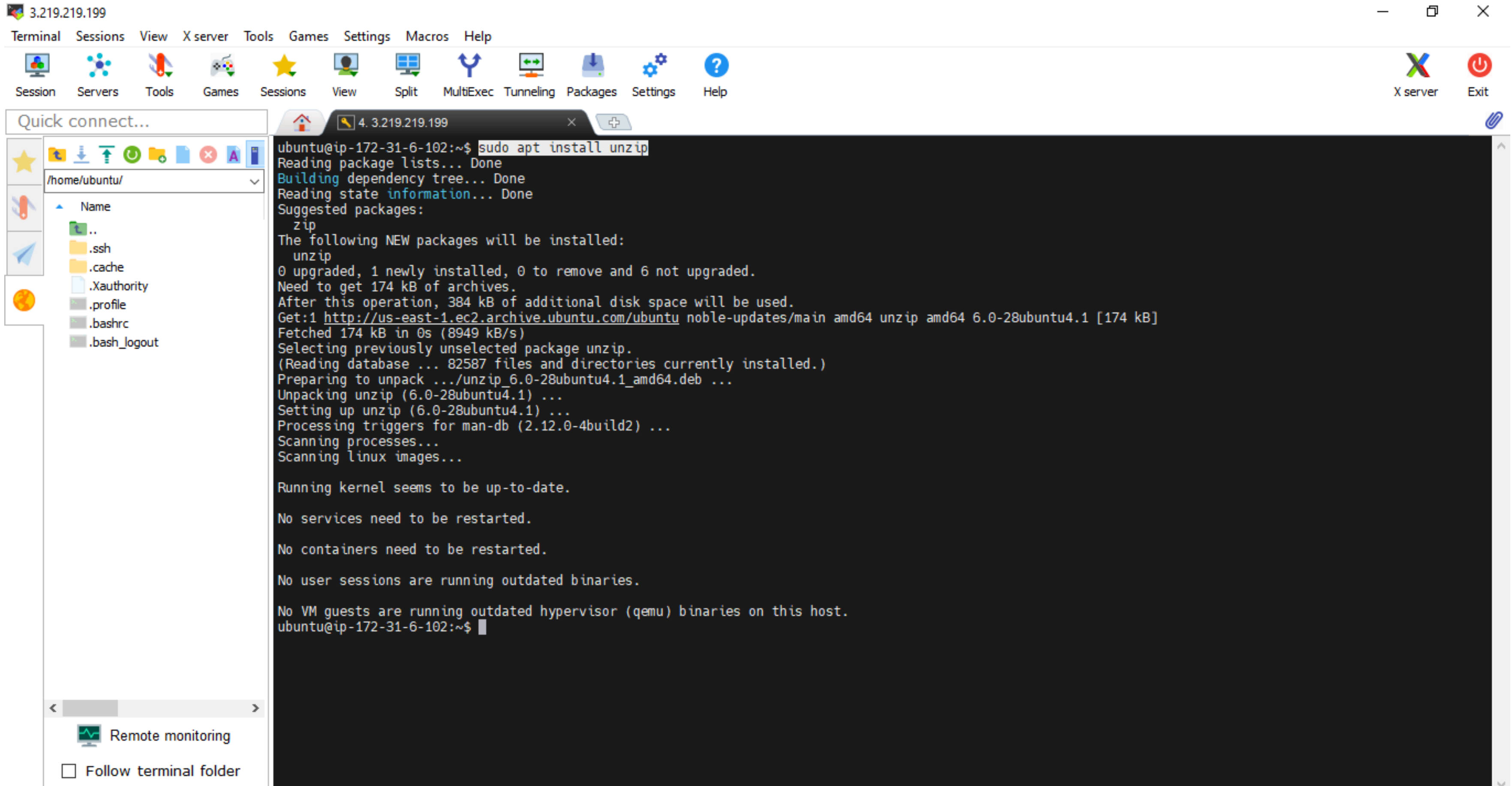
Build Notifiers

Build Reports

Other Post-Build Actions

11 mo ago

# step:11 Install sonarqube in the ec2 instance



The screenshot shows a terminal window titled "3.219.219.199" with a menu bar (Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help) and a toolbar. On the left, a sidebar shows a "Quick connect..." search bar and a file explorer for "/home/ubuntu/" with a list of files and folders: .., .ssh, .cache, .Xauthority, .profile, .bashrc, and .bash\_logout. The terminal output shows the command `sudo apt install unzip` being executed. The output includes package list reading, dependency tree building, state information reading, and suggested packages (zip, unzip). It lists the new packages to be installed (unzip), the disk space requirements (174 kB), and the source of the package (http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 unzip amd64 6.0-28ubuntu4.1 [174 kB]). The installation process is shown as successful, with the package being unpacked and triggers processed. The terminal ends with the prompt `ubuntu@ip-172-31-6-102:~$`.

```
ubuntu@ip-172-31-6-102:~$ sudo apt install unzip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 6 not upgraded.
Need to get 174 kB of archives.
After this operation, 384 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 unzip amd64 6.0-28ubuntu4.1 [174 kB]
Fetched 174 kB in 0s (8949 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 82587 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4.1_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4.1) ...
Setting up unzip (6.0-28ubuntu4.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-6-102:~$
```



3.219.219.199

Terminal Sessions View X server Tools Games Settings Macros Help

Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help

X server Exit

Quick connect...

/home/ubuntu/  
Name  
..  
.ssh  
.cache  
.Xauthority  
.profile  
.bashrc  
.bash\_logout

```
ubuntu@ip-172-31-6-102:~$ sudo adduser sonarqube
info: Adding user `sonarqube' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `sonarqube' (1001) ...
info: Adding new user `sonarqube' (1001) with group `sonarqube (1001)' ...
info: Creating home directory `/home/sonarqube' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for sonarqube
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
info: Adding new user `sonarqube' to supplemental / extra groups `users' ...
info: Adding user `sonarqube' to group `users' ...
ubuntu@ip-172-31-6-102:~$ sudo su - sonarqube
sonarqube@ip-172-31-6-102:~$ wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.4.0.54424.zip
--2024-10-05 08:38:34-- https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.4.0.54424.zip
Resolving binaries.sonarsource.com (binaries.sonarsource.com)... 99.84.191.23, 99.84.191.71, 99.84.191.75, ...
Connecting to binaries.sonarsource.com (binaries.sonarsource.com)|99.84.191.23|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 287666040 (274M) [binary/octet-stream]
Saving to: 'sonarqube-9.4.0.54424.zip'

sonarqube-9.4.0.54424.zip      100%[=====>] 274.34M  91.3MB/s   in 3.0s

2024-10-05 08:38:37 (91.3 MB/s) - 'sonarqube-9.4.0.54424.zip' saved [287666040/287666040]

sonarqube@ip-172-31-6-102:~$ unzip *
Archive: sonarqube-9.4.0.54424.zip
  creating: sonarqube-9.4.0.54424/
  inflating: sonarqube-9.4.0.54424/dependency-license.json
  creating: sonarqube-9.4.0.54424/bin/
  creating: sonarqube-9.4.0.54424/bin/jsw-license/
  inflating: sonarqube-9.4.0.54424/bin/jsw-license/LICENSE.txt
  creating: sonarqube-9.4.0.54424/bin/macosx-universal-64/
  creating: sonarqube-9.4.0.54424/bin/macosx-universal-64/lib/
```

Remote monitoring

☐ Follow terminal folder



3.219.219.199

Terminal Sessions View X server Tools Games Settings Macros Help



Quick connect...

4. 3.219.219.199

/home/ubuntu/

Name

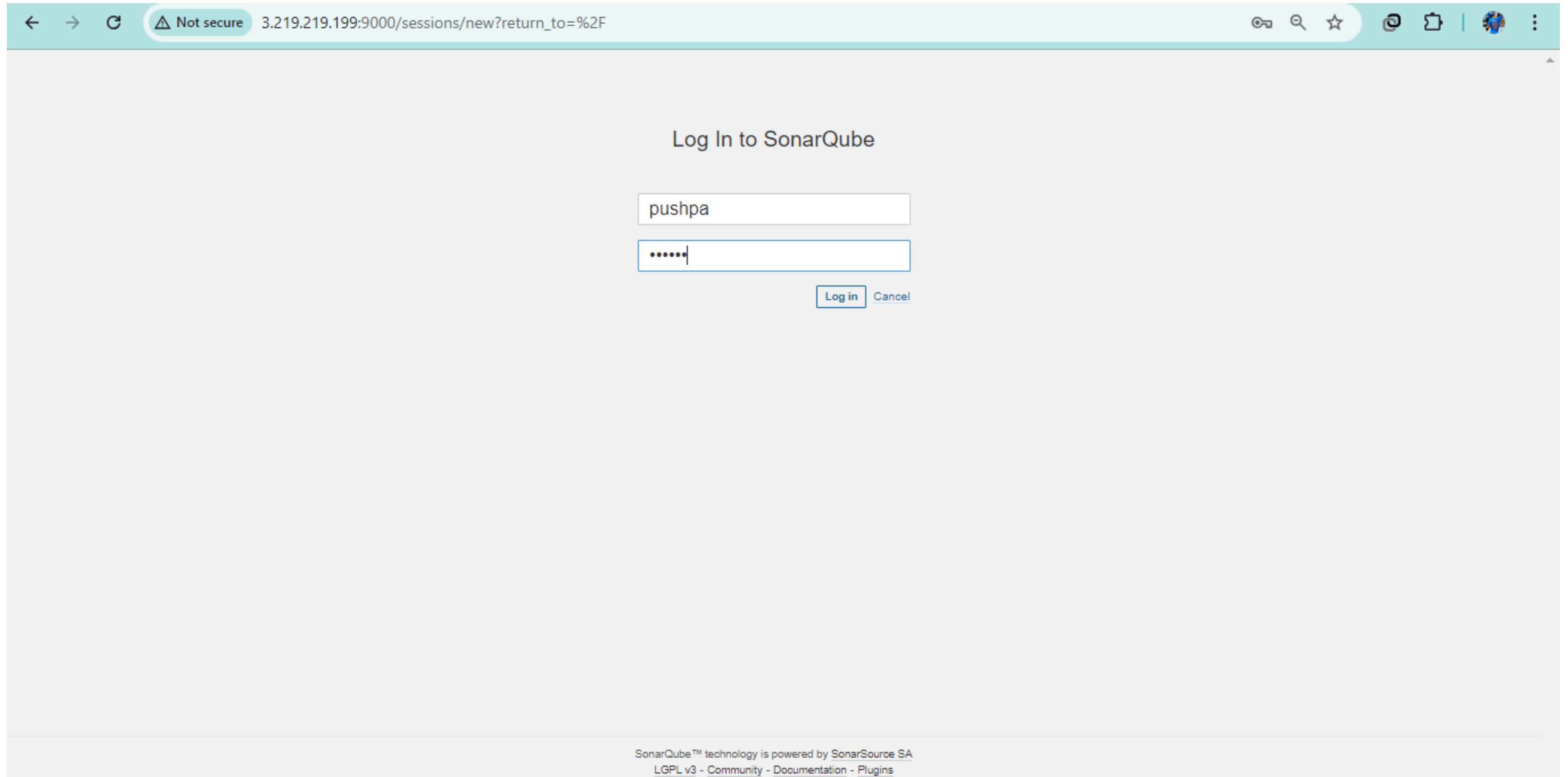
..  
..ssh  
..cache  
..Xauthority  
..profile  
..bashrc  
..bash\_logout

```
inflating: sonarqube-9.4.0.54424/web/images/embed-doc/images/encrypt-value.png
inflating: sonarqube-9.4.0.54424/web/images/embed-doc/images/short-lived-branch-concept.png
inflating: sonarqube-9.4.0.54424/web/images/embed-doc/sq-icon.svg
inflating: sonarqube-9.4.0.54424/web/images/embed-doc/twitter-icon.svg
inflating: sonarqube-9.4.0.54424/web/index.html
inflating: sonarqube-9.4.0.54424/web/.htaccess
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-76x76.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-144x144.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-120x120.png
inflating: sonarqube-9.4.0.54424/web/mstile-512x512.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-114x114.png
inflating: sonarqube-9.4.0.54424/web/WEB-INF/web.xml
inflating: sonarqube-9.4.0.54424/web/robots.txt
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-72x72.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-180x180.png
creating: sonarqube-9.4.0.54424/web/js/
inflating: sonarqube-9.4.0.54424/web/js/out3HYJVQV0.css.map
inflating: sonarqube-9.4.0.54424/web/js/out3HYJVQV0.css
inflating: sonarqube-9.4.0.54424/web/js/outEG4XZBCS.js.map
inflating: sonarqube-9.4.0.54424/web/js/outEG4XZBCS.js
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-152x152.png
inflating: sonarqube-9.4.0.54424/web/favicon.ico
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-60x60.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-57x57.png
inflating: sonarqube-9.4.0.54424/web/apple-touch-icon-precomposed.png
creating: sonarqube-9.4.0.54424/lib/jdbc/
creating: sonarqube-9.4.0.54424/lib/jdbc/mssql/
inflating: sonarqube-9.4.0.54424/lib/jdbc/mssql/mssql-jdbc-9.4.1.jre11.jar
creating: sonarqube-9.4.0.54424/lib/jdbc/postgresql/
inflating: sonarqube-9.4.0.54424/lib/jdbc/postgresql/postgresql-42.3.3.jar
creating: sonarqube-9.4.0.54424/lib/jdbc/h2/
inflating: sonarqube-9.4.0.54424/lib/jdbc/h2/h2-2.1.210.jar
inflating: sonarqube-9.4.0.54424/lib/sonar-shutdowner-9.4.0.54424.jar
creating: sonarqube-9.4.0.54424/elasticsearch/plugins/
sonarqube@ip-172-31-6-102:~$ chmod -R 755 /home/sonarqube/sonarqube-9.4.0.54424
sonarqube@ip-172-31-6-102:~$ chown -R sonarqube:sonarqube /home/sonarqube/sonarqube-9.4.0.54424
sonarqube@ip-172-31-6-102:~$ cd sonarqube-9.4.0.54424/bin/linux-x86-64
sonarqube@ip-172-31-6-102:~/sonarqube-9.4.0.54424/bin/linux-x86-64$ ./sonar.sh start
Starting SonarQube...
Started SonarQube.
sonarqube@ip-172-31-6-102:~/sonarqube-9.4.0.54424/bin/linux-x86-64$
```

Remote monitoring

☐ Follow terminal folder

step:12 Now browse the public IP of instance with port 9000 to access sonarqube and login to it



The screenshot shows a web browser window with the address bar displaying "3.219.219.199:9000/sessions/new?return\_to=%2F". The page title is "Log In to SonarQube". The login form consists of two input fields: the first contains the username "pushpa" and the second contains a masked password ".....". Below the password field are two buttons: "Log in" and "Cancel". At the bottom of the page, there is a footer that reads: "SonarQube™ technology is powered by SonarSource SA" followed by links for "LGPL v3", "Community", "Documentation", and "Plugins".

Log In to SonarQube

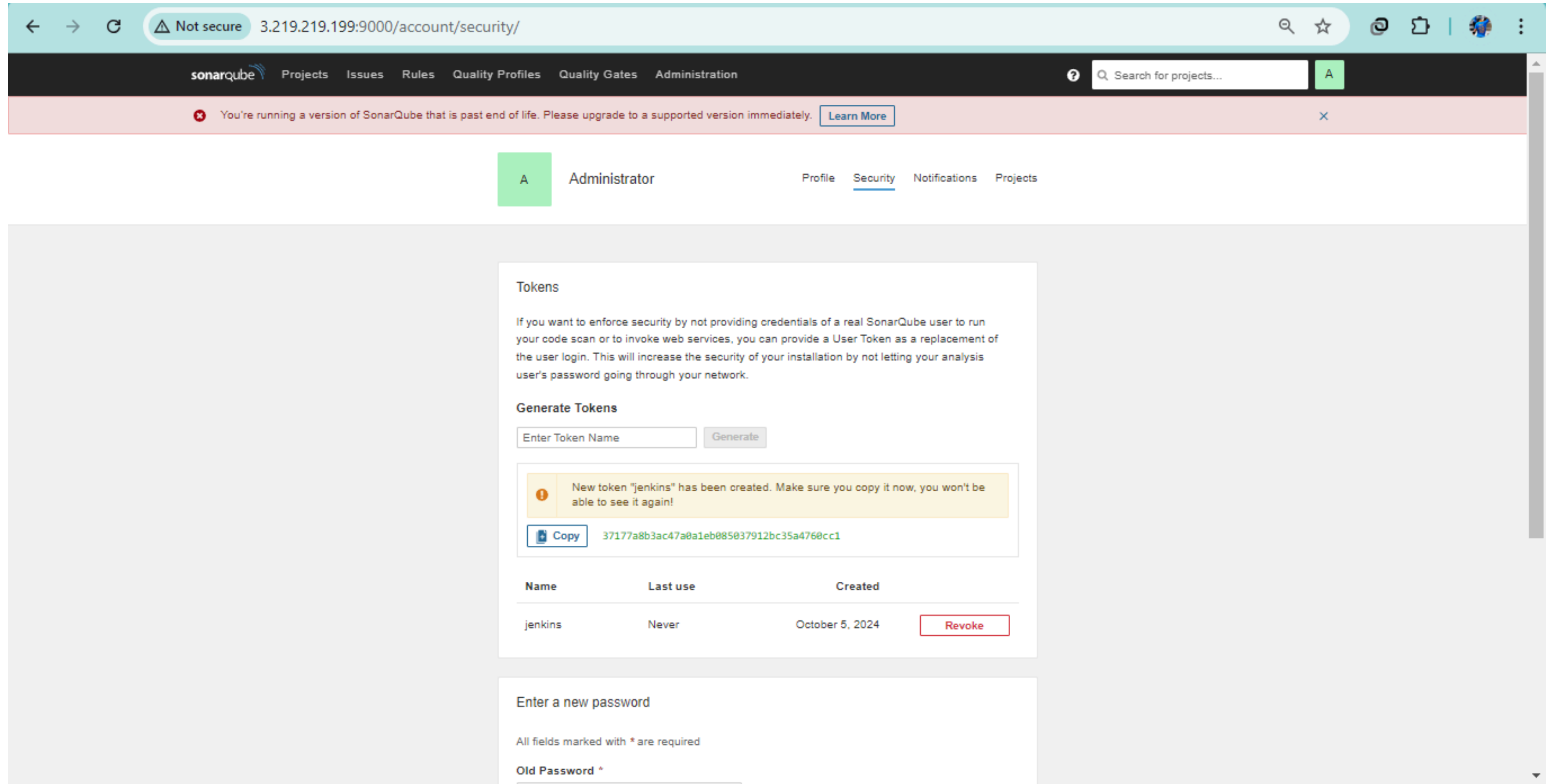
pushpa

.....

Log in Cancel

SonarQube™ technology is powered by SonarSource SA  
LGPL v3 - [Community](#) - [Documentation](#) - [Plugins](#)

step:13 Now create a token and by using this token we can make jenkins to interact with sonarqube



The screenshot shows the SonarQube web interface. The browser address bar indicates the URL is `3.219.219.199:9000/account/security/`. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. A search bar is present on the right. A red warning banner at the top states: "You're running a version of SonarQube that is past end of life. Please upgrade to a supported version immediately." Below this, the user profile "Administrator" is shown with tabs for Profile, Security (selected), Notifications, and Projects.

The main content area is titled "Tokens" and contains the following text:

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

**Generate Tokens**

Enter Token Name

A yellow notification box states: "New token 'jenkins' has been created. Make sure you copy it now, you won't be able to see it again!" Below this, a "Copy" button is next to the token value: `37177a8b3ac47a0a1eb085037912bc35a4760cc1`.

Name	Last use	Created	
jenkins	Never	October 5, 2024	<input type="button" value="Revoke"/>

Below the table, there is a section titled "Enter a new password" with the instruction "All fields marked with \* are required". The "Old Password \*" field is partially visible.

step:14 Now in jenkins create sonarqube credentials in order to make both jenkins and sonarqube interact with eachother

Not secure3.219.219.199:8080/manage/credentials/store/system/domain/\_/newCredentials

Jenkins

Search (CTRL+K)

Bellamkonda Pushpa Shree

log out

Dashboard

Manage Jenkins

Credentials

System

Global credentials (unrestricted)

New credentials

Kind

Secret text

Scope

Global (Jenkins, nodes, items, all child items, etc)

Secret

ID

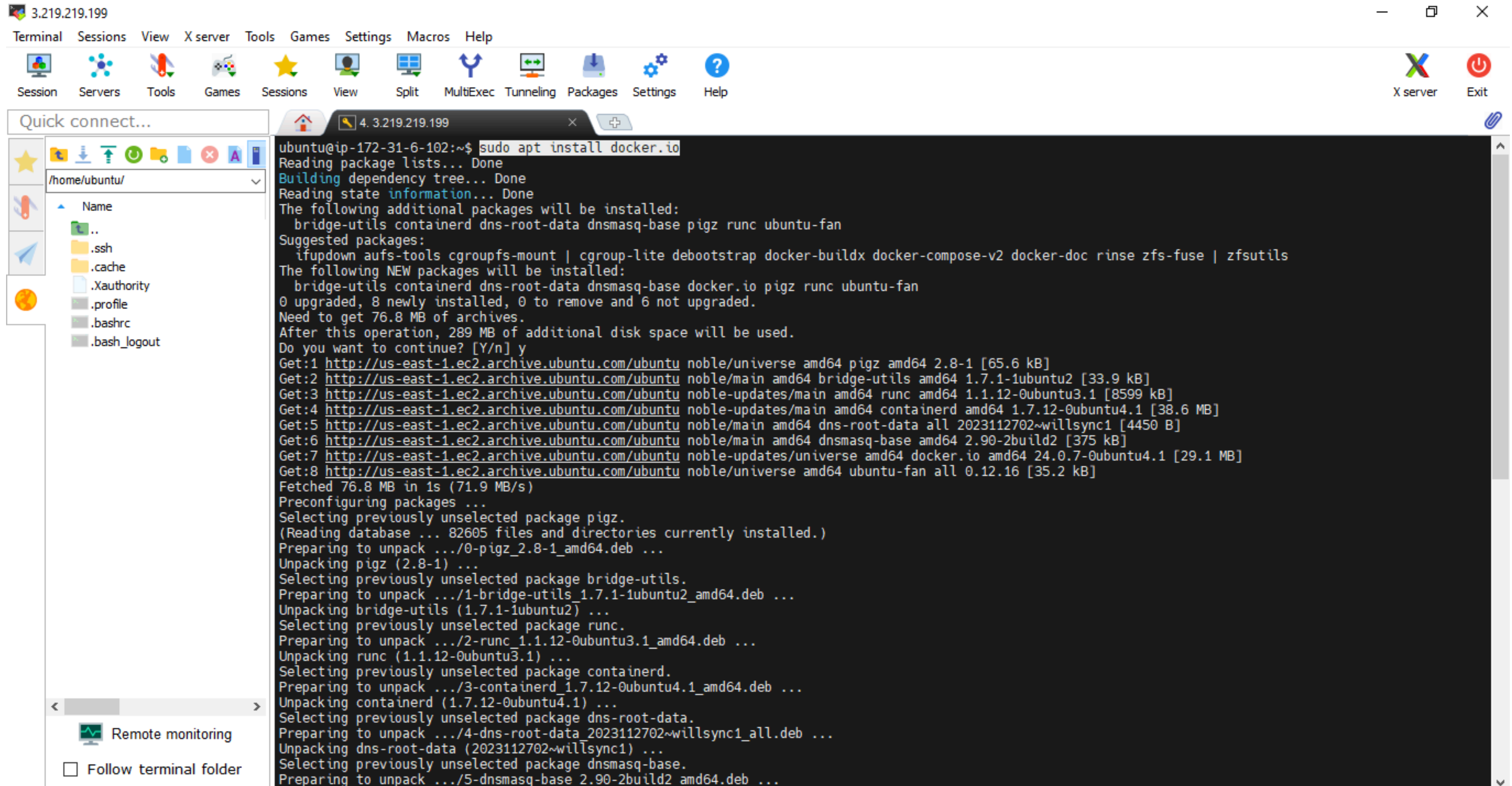
sonarqube

Description

Create

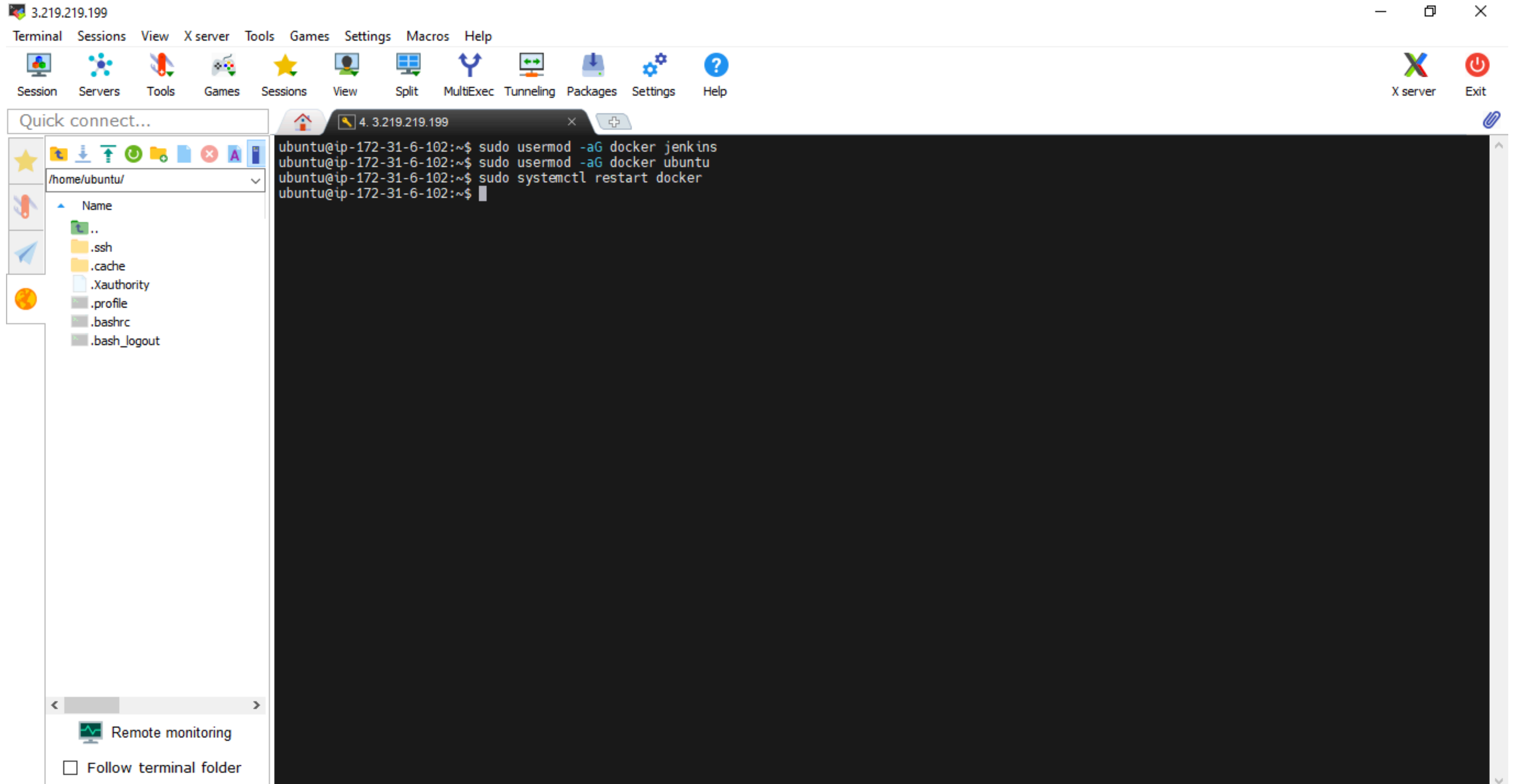


# step:15 Now install Docker inside ec2 instance



```
ubuntu@ip-172-31-6-102:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-buildx docker-compose-v2 docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 6 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-1ubuntu2 [33.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dns-root-data all 2023112702~willsync1 [4450 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dnsmasq-base amd64 2.90-2build2 [375 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 docker.io amd64 24.0.7-0ubuntu4.1 [29.1 MB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 76.8 MB in 1s (71.9 MB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 82605 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.8-1_amd64.deb ...
Unpacking pigz (2.8-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.7.1-1ubuntu2_amd64.deb ...
Unpacking bridge-utils (1.7.1-1ubuntu2) ...
Selecting previously unselected package runc.
Preparing to unpack .../2-runc_1.1.12-0ubuntu3.1_amd64.deb ...
Unpacking runc (1.1.12-0ubuntu3.1) ...
Selecting previously unselected package containerd.
Preparing to unpack .../3-containerd_1.7.12-0ubuntu4.1_amd64.deb ...
Unpacking containerd (1.7.12-0ubuntu4.1) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2023112702~willsync1_all.deb ...
Unpacking dns-root-data (2023112702~willsync1) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../5-dnsmasq-base_2.90-2build2_amd64.deb ...
```

# Now run these commands to give privileges and restart docker



# Now restart jenkins

←

→

↻

3.219.219.199:8080/restart

3.219.219.199:8080/restart

3.219.219.199:8080/restart - Google Search

Jenkins

Dashboard

Shree

log out

+ New Item

Build History

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

0/2

All

+

S	W	Name ↓	Last Success	Last Failure	Last Duration
...	☀	pushpa shree	N/A	N/A	N/A

Icon:

S

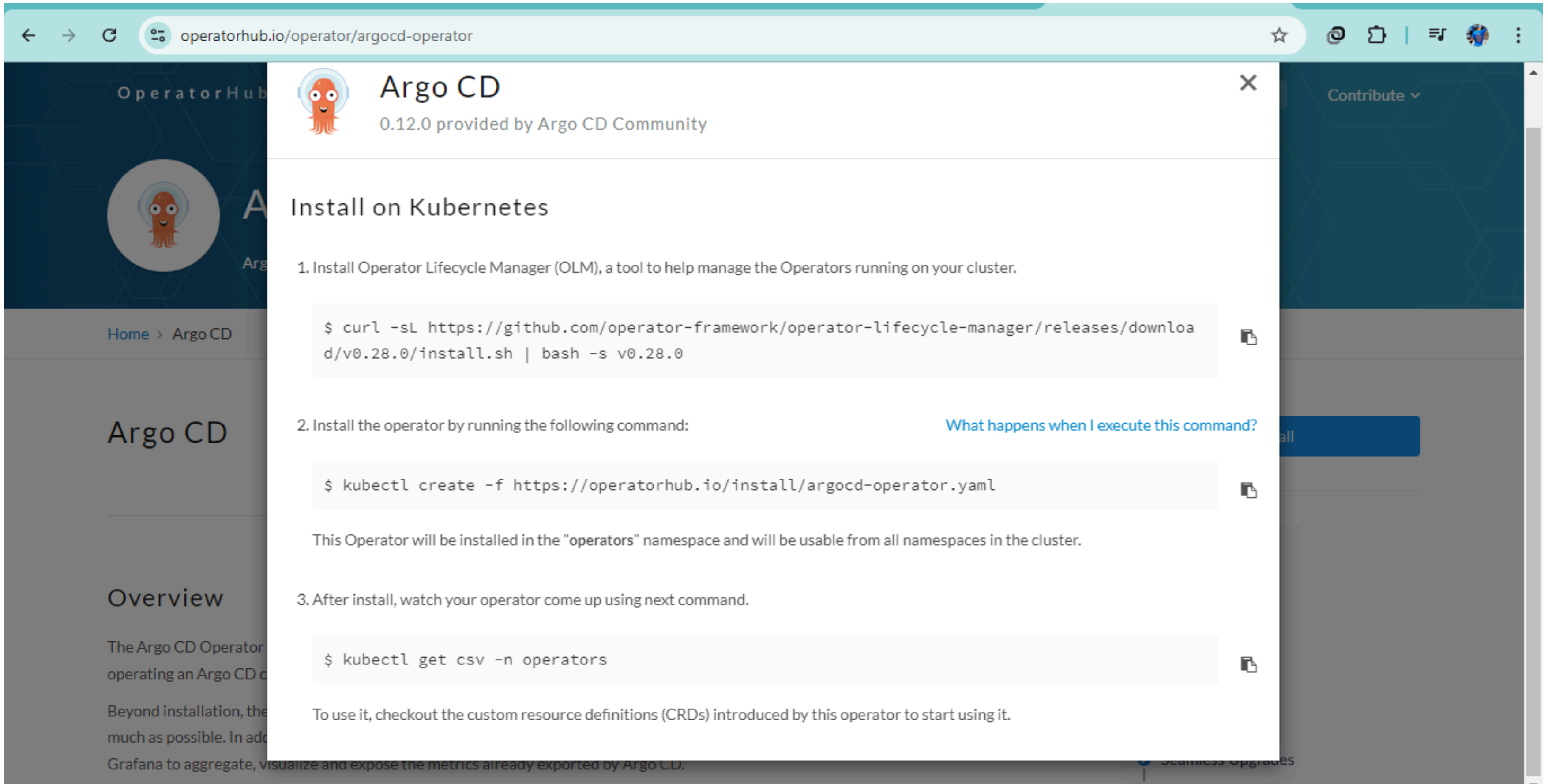
M

L

Add description



Now browse "operatorhub argocd install" to install argocd operator



The screenshot shows a web browser window with the URL `operatorhub.io/operator/argocd-operator`. The page features a dark blue sidebar on the left with the OperatorHub logo and navigation links like "Home" and "Argo CD". The main content area is titled "Argo CD" and "0.12.0 provided by Argo CD Community". It includes a section "Install on Kubernetes" with three numbered steps: 1. Install Operator Lifecycle Manager (OLM), 2. Install the operator by running the following command, and 3. After install, watch your operator come up using next command. Each step includes a terminal command snippet and a copy icon. A link "What happens when I execute this command?" is present next to the second step. The bottom of the page shows an "Overview" section with text about the Argo CD Operator.

OperatorHub

# Argo CD

0.12.0 provided by Argo CD Community

## Install on Kubernetes

1. Install Operator Lifecycle Manager (OLM), a tool to help manage the Operators running on your cluster.  

```
$ curl -sL https://github.com/operator-framework/operator-lifecycle-manager/releases/download/v0.28.0/install.sh | bash -s v0.28.0
```
2. Install the operator by running the following command:  

```
$ kubectl create -f https://operatorhub.io/install/argocd-operator.yaml
```

[What happens when I execute this command?](#)

This Operator will be installed in the "operators" namespace and will be usable from all namespaces in the cluster.
3. After install, watch your operator come up using next command.  

```
$ kubectl get csv -n operators
```

To use it, checkout the custom resource definitions (CRDs) introduced by this operator to start using it.

## Overview

The Argo CD Operator is a Kubernetes Operator that manages the lifecycle of Argo CD clusters. It can be used to install, upgrade, and manage Argo CD clusters in a Kubernetes cluster.

Beyond installation, the operator can be used to manage the lifecycle of Argo CD clusters, such as upgrading the version of Argo CD. In addition, the operator can be used to manage the lifecycle of Argo CD clusters, such as upgrading the version of Argo CD. In addition, the operator can be used to manage the lifecycle of Argo CD clusters, such as upgrading the version of Argo CD.

Grafana to aggregate, visualize and expose the metrics already exported by Argo CD.

# Now run those commands one by one

MINGW64:/c/Users/rgukt

```
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

rgukt@pushpaPC MINGW64 ~

```
$ curl -sL https://github.com/operator-framework/operator-lifecycle-manager/releases/download/v0.28.0/install.sh | bash -s v0.28.0
```

```
customresourcedefinition.apiextensions.k8s.io/catalogsources.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/clusterserviceversions.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/installplans.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/olmconfigs.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/operatorconditions.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/operatorgroups.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/operators.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/subscriptions.operators.coreos.com created
customresourcedefinition.apiextensions.k8s.io/catalogsources.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/clusterserviceversions.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/installplans.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/olmconfigs.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/operatorconditions.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/operatorgroups.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/operators.operators.coreos.com condition met
customresourcedefinition.apiextensions.k8s.io/subscriptions.operators.coreos.com condition met
```

```
namespace/olm created
```

```
namespace/operators created
```

```
serviceaccount/olm-operator-serviceaccount created
```

```
clusterrole.rbac.authorization.k8s.io/system:controller:operator-lifecycle-manager created
```

```
clusterrolebinding.rbac.authorization.k8s.io/olm-operator-binding-olm created
```

```
olmconfig.operators.coreos.com/cluster created
```

```
deployment.apps/olm-operator created
```

```
deployment.apps/catalog-operator created
```

```
clusterrole.rbac.authorization.k8s.io/aggregate-olm-edit created
```

```
clusterrole.rbac.authorization.k8s.io/aggregate-olm-view created
```

```
operatorgroup.operators.coreos.com/global-operators created
```

```
operatorgroup.operators.coreos.com/olm-operators created
```

```
clusterserviceversion.operators.coreos.com/packageserver created
```

```
catalogsource.operators.coreos.com/operatorhubio-catalog created
```

```
Waiting for deployment "olm-operator" rollout to finish: 0 of 1 updated replicas are available...
```

```
deployment "olm-operator" successfully rolled out
```

```
deployment "catalog-operator" successfully rolled out
```

```
Package server phase: InstallReady
```

```
Package server phase: Installing
```

```
Package server phase: Succeeded
```

```
deployment "packageserver" successfully rolled out
```

rgukt@pushpaPC MINGW64 ~

```
$
```

```
rgukt@pushpaPC MINGW64 ~
```

```
$ kubectl create -f https://operatorhub.io/install/argocd-operator.yaml  
subscription.operators.coreos.com/my-argocd-operator created
```

```
rgukt@pushpaPC MINGW64 ~
```

```
$ |
```

rgukt@pushpaPC MINGW64 ~

\$ kubectl get csv -n operators

NAME	DISPLAY	VERSION	REPLACES	PHASE
argocd-operator.v0.12.0	Argo CD	0.12.0	argocd-operator.v0.11.0	Installing

rgukt@pushpaPC MINGW64 ~

\$ |

Now add the dockerhub credentials in the jenkins so that the docker image will be stored inside the dockerhub of the respective repository that we gave in the pipeline syntax

The screenshot shows the Jenkins web interface for creating a new credential. The browser address bar indicates the URL is `3.219.219.199:8080/manage/credentials/store/system/domain/_/newCredentials`. The breadcrumb navigation shows the path: `Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >`. The form is titled 'Kind' and has a dropdown menu set to 'Username with password'. Below this, the 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Username' field contains the text 'pushpa867'. There is an unchecked checkbox labeled 'Treat username as secret'. The 'Password' field is masked with dots. The 'ID' field contains the text 'docker-cred'. The 'Description' field is empty. A blue 'Create' button is at the bottom left of the form.

Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?

pushpa867

☐ Treat username as secret ?

Password ?

.....

ID ?

docker-cred

Description ?


Create

Now provide the github credentials in the jenkins , create a token in the github

← → ↺

Not secure 3.219.219.199:8080/manage/credentials/store/system/domain/\_/newCredentials

🔍 ☆ 📁 📌 📄 📄 📄

 **Jenkins**

🔍 Search (CTRL+K) ⓘ

🛡️ 1 👤 Bellamkonda Pushpa Shree ▾ 🚪 log out

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

## New credentials

Kind

Secret text ▾

Scope ⓘ

Global (Jenkins, nodes, items, all child items, etc) ▾

Secret

.....

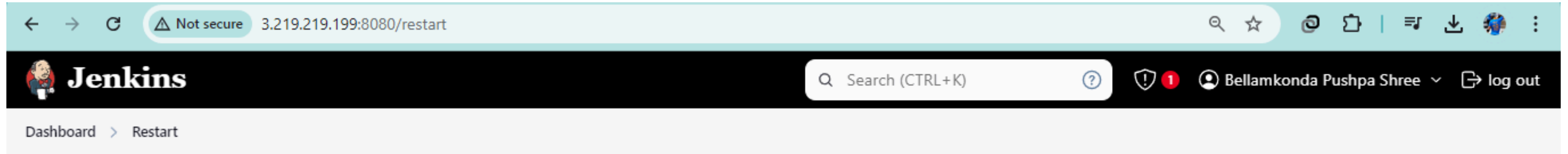
ID ⓘ

github

Description ⓘ

Create


# Now again restart the jenkins interface





## step:16 Now build the pipeline

← → ↻ ⚠ Not secure 3.219.219.199:8080/job/pushpa%20shree/multi-pipeline-graph/ 🔍 ☆ 🔄 📁 | 🎵 ⬇️ 🌐 ⋮

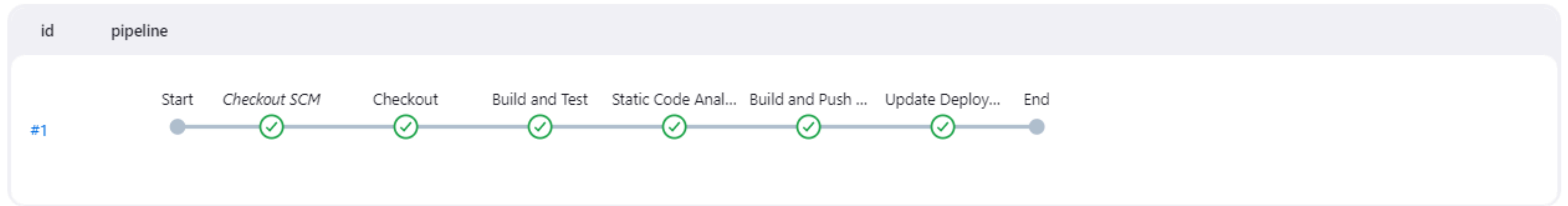
 **Jenkins** 🔍 Search (CTRL+K) ? 🛡️ 1 👤 Bellamkonda Pushpa Shree ▾ 🚪 log out

Dashboard > pushpa shree > Stages

## Build pushpa shree

▶ Build

Configure



We can observe that our application is deployed in the sonarqube server

The screenshot displays the SonarQube web interface. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. A search bar is present on the right. A warning message at the top indicates that the version of SonarQube is past its end of life.

The main content area shows a list of projects. The first project, 'spring-boot-demo', is highlighted. It has a 'Passed' status and a last analysis time of 8 minutes ago. The project's metrics are displayed below the status:

Metric	Value	Quality Gate
Bugs	0	A
Vulnerabilities	0	A
Hotspots Reviewed	-	A
Code Smells	0	A
Coverage	0.0%	
Duplications	0.0%	
Lines	79	XS

The project is associated with XML and Java files.

On the left side, there are filters for Quality Gate (Passed: 1, Failed: 0), Reliability (A: 1, B: 0, C: 0, D: 0, E: 0), and Security (A: 1, B: 0, C: 0, D: 0, E: 0).

A warning message at the bottom states: 'Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.'

The footer indicates that SonarQube™ technology is powered by SonarSource SA, and it is the Community Edition, Version 9.4 (build 54424), under the LGPL v3 license. Links for Community, Documentation, Plugins, and Web API are provided.

And our docker image is also pushed inside our dockerhub repository

hub.docker.com/repository/docker/pushpa867/cicd-repo/tags

dockerhub

Explore

Repositories

Organizations

Usage

Search Docker Hub

ctrl+K

?

⚙

🔄

⋮

P

pushpa867 / [Repositories](#) / [cicd-repo](#) / [Tags](#)

Using 0 of 1 private repositories.

General

Tags

Builds

Collaborators

Webhooks

Settings

☒

Sort by

Newest ▾

Delete

TAG

1

Last pushed 10 minutes ago by [pushpa867](#)

Digest

OS/ARCH

Last pull

Compressed Size ⓘ

[1666a38937e9](#)

linux/amd64

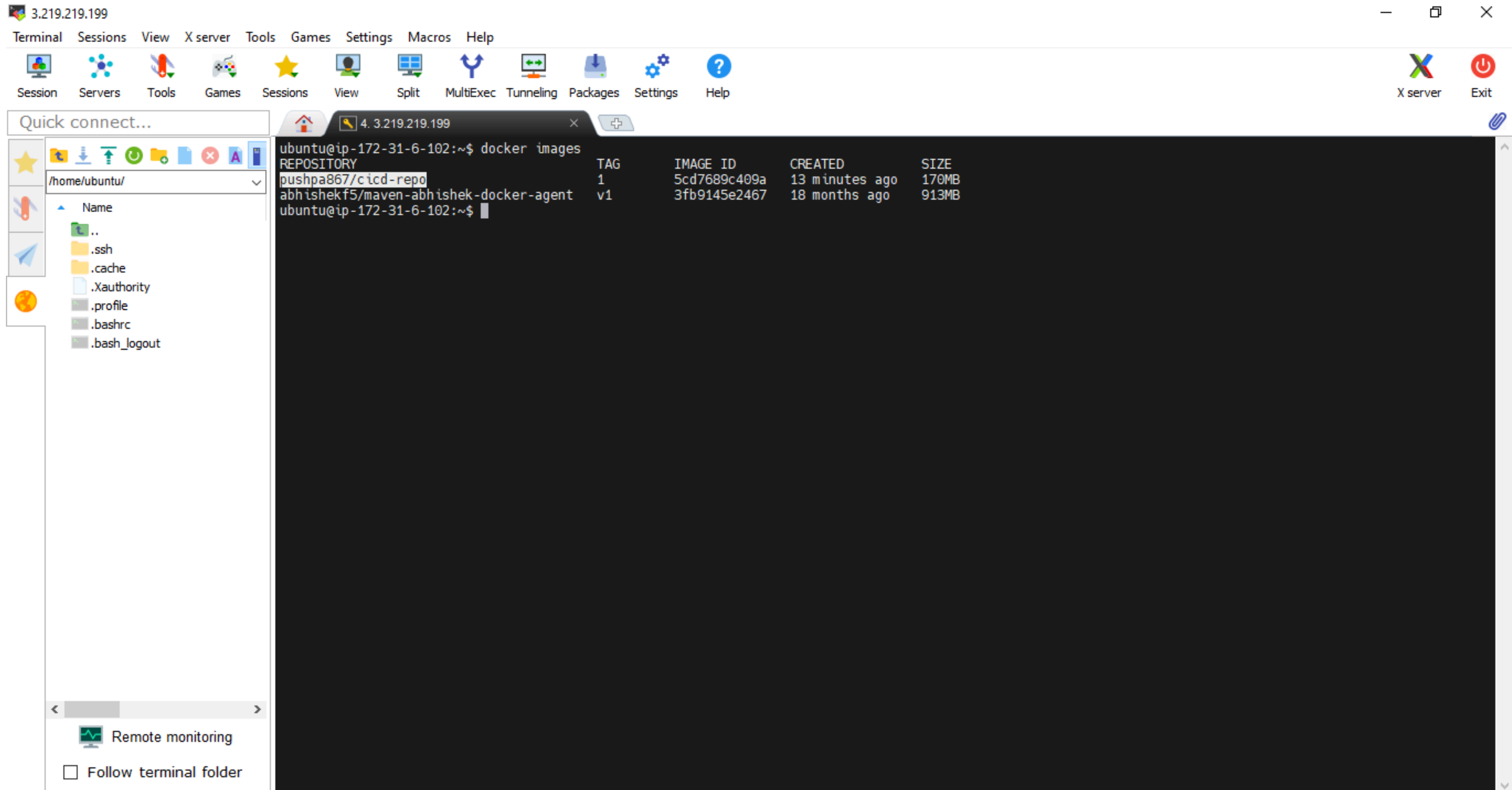
10 minutes ago

67.43 MB

docker pull pushpa867/cicd-repo:1

Copy

# Run this command to see the docker image in the ec2 instance



The screenshot shows a terminal window titled "3.219.219.199" with a menu bar (Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help) and a toolbar. The left sidebar shows a file explorer for "/home/ubuntu/" with a list of files and folders: .., .ssh, .cache, .Xauthority, .profile, .bashrc, and .bash\_logout. The main terminal area shows the command `docker images` being executed, resulting in a table of Docker images.

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
pushpa867/cicd-repo	1	5cd7689c409a	13 minutes ago	170MB
abhishekf5/maven-abhishek-docker-agent	v1	3fb9145e2467	18 months ago	913MB

Below the table, the prompt `ubuntu@ip-172-31-6-102:~$` is visible. At the bottom of the sidebar, there are options for "Remote monitoring" (checked) and "Follow terminal folder" (unchecked).

# Steps for Continuous Deployment

step:1 create a new Argo CD cluster with the default configuration

- Create a yaml file called "argocd-basic.yaml" which is used to create argocd cluster

- Configure this yaml file i.e we are giving previliges to the file

- We can observe that our "argocd-basic" service is in "ClusterIP" mode , it means only people who has access with cluster can access this service so to make it accessable to our friends or colleagues we will make it into "NodePort" mode

- For that

**kubectrl edit svc example-argocd-server**

**\*\*Now in the editor , change the type from ClusterIP to "NodePort"**

MINGW64:/c/Users/rgukt/desktop/MY\_THINGS/AWS/Jenkins-Zero-to-Hero

```
rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ vim argocd-basic.yml

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl apply -f argocd-basic.yml
Warning: ArgoCD v1alpha1 version is deprecated and will be converted to v1beta1
argocd.argoproj.io/example-argocd configured

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ vim argocd-basic.yml

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl apply -f argocd-basic.yml
argocd.argoproj.io/example-argocd configured

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl get pods
NAME                                READY   STATUS    RESTARTS
example-argocd-application-controller-0  1/1     Running   1
example-argocd-redis-6545fd6d6c-hg4hv    1/1     Running   1 (15m ago)
example-argocd-repo-server-869d5757c7-zz7b8  1/1     Running   20 (20h ago)
example-argocd-server-76bb84cddc-vdrjq    0/1     Running   17 (106s ago)

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl get svc
NAME                                TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)
example-argocd-metrics              ClusterIP    10.109.57.142  <none>          8082/TCP
example-argocd-redis                ClusterIP    10.99.242.202  <none>          6379/TCP
example-argocd-repo-server           ClusterIP    10.100.55.107  <none>          8081/TCP
example-argocd-server                ClusterIP    10.109.33.109  <none>          80/TCP
example-argocd-server-metrics        ClusterIP    10.107.100.137 <none>          8083/TCP
kubernetes                           ClusterIP    10.96.0.1      <none>          443/TCP

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl edit svc example-argocd-server
```

\*kubectl.exe-edit-1158243830.yaml - Notepad

File Edit Format View Help

```
- 10.109.33.109
internalTrafficPolicy: Cluster
ipFamilies:
- IPv4
ipFamilyPolicy: SingleStack
ports:
- name: http
  port: 80
  protocol: TCP
  targetPort: 8080
- name: https
  port: 443
  protocol: TCP
  targetPort: 8080
selector:
  app.kubernetes.io/name: example-argocd-server
sessionAffinity: None
type: NodePort
status:
  loadBalancer: {}
```

Ln 45, Col 17 100% Windows (CRLF) UTF-8



# Now we can observe that our service is changed to NodePort mode

```
MINGW64:/c/Users/rgukt/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero
rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ vim argocd-basic.yml

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl apply -f argocd-basic.yml
Warning: ArgoCD v1alpha1 version is deprecated and will be converted to v1beta1 automatically. Moving forward, please use v1beta1 as the ArgoCD API version.
argocd.argoproj.io/example-argocd configured

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ vim argocd-basic.yml

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl apply -f argocd-basic.yml
argocd.argoproj.io/example-argocd configured

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
example-argocd-application-controller-0  1/1     Running   1          23h
example-argocd-redis-6545fd6d6c-hg4hv    1/1     Running   1 (15m ago) 23h
example-argocd-repo-server-869d5757c7-zz7b8 1/1     Running   20 (20h ago) 23h
example-argocd-server-76bb84cddc-vdrjq    0/1     Running   17 (106s ago) 23h

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl get svc
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)          AGE
example-argocd-metrics              ClusterIP    10.109.57.142   <none>       8082/TCP          23h
example-argocd-redis                 ClusterIP    10.99.242.202   <none>       6379/TCP          23h
example-argocd-repo-server            ClusterIP    10.100.55.107   <none>       8081/TCP,8084/TCP 23h
example-argocd-server                 ClusterIP    10.109.33.109   <none>       80/TCP,443/TCP    23h
example-argocd-server-metrics         ClusterIP    10.107.100.137  <none>       8083/TCP          23h
kubernetes                           ClusterIP    10.96.0.1       <none>       443/TCP           23h

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl edit svc example-argocd-server
service/example-argocd-server edited

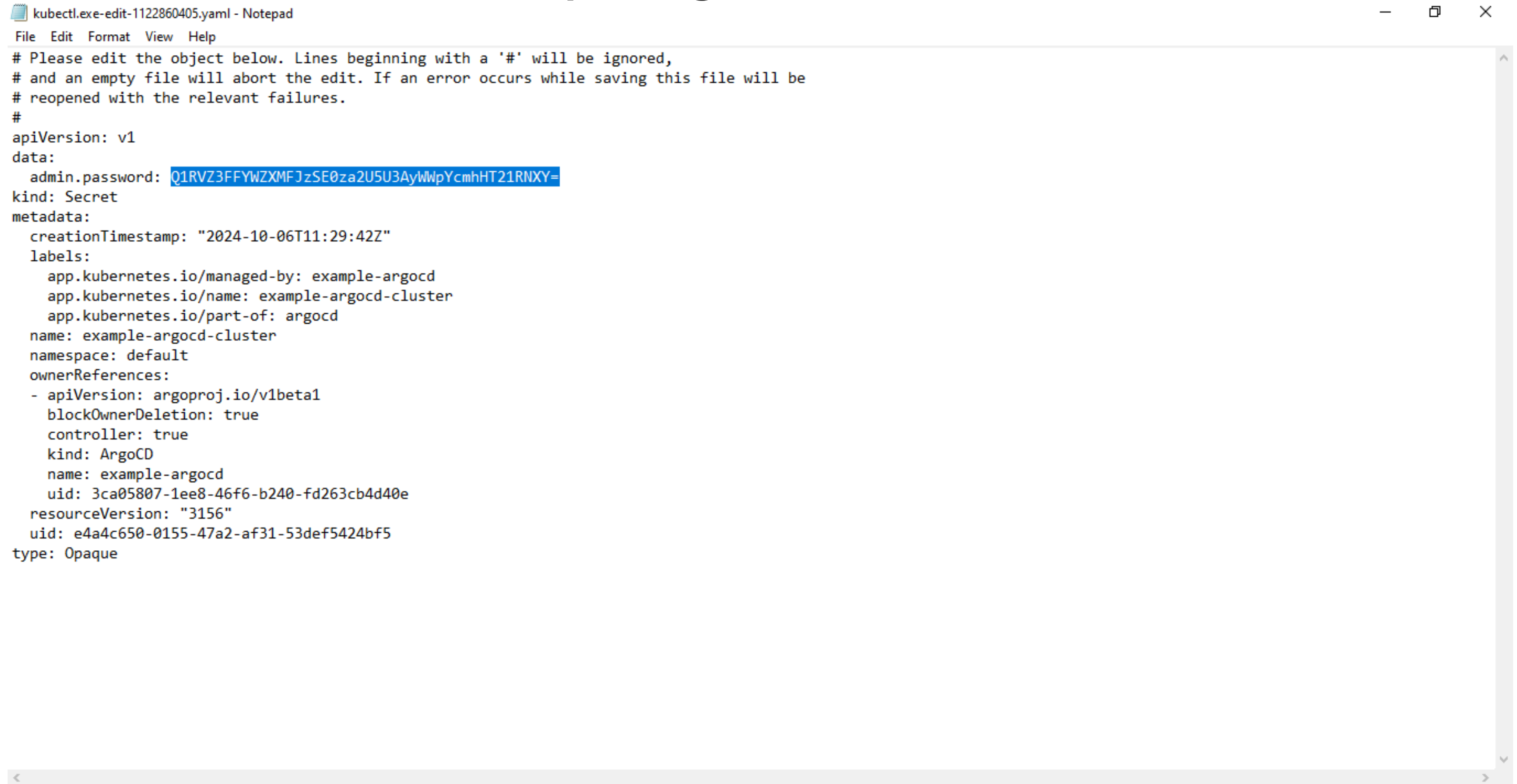
rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl get svc
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)          AGE
example-argocd-metrics              ClusterIP    10.109.57.142   <none>       8082/TCP          23h
example-argocd-redis                 ClusterIP    10.99.242.202   <none>       6379/TCP          23h
example-argocd-repo-server            ClusterIP    10.100.55.107   <none>       8081/TCP,8084/TCP 23h
example-argocd-server                 NodePort     10.109.33.109   <none>       80:31498/TCP,443:31382/TCP 23h
example-argocd-server-metrics         ClusterIP    10.107.100.137  <none>       8083/TCP          23h
kubernetes                           ClusterIP    10.96.0.1       <none>       443/TCP           23h

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ minikube service argocd-server
```



Run the following command to get secret code in the editor

## kubectl edit secret example-argocd-server



```
kubectl.exe-edit-1122860405.yaml - Notepad
File Edit Format View Help
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: v1
data:
  admin.password: Q1RVZ3FFYWZXMFIzSE0za2U5U3AyWpYcmhHT21RNXY=
kind: Secret
metadata:
  creationTimestamp: "2024-10-06T11:29:42Z"
  labels:
    app.kubernetes.io/managed-by: example-argocd
    app.kubernetes.io/name: example-argocd-cluster
    app.kubernetes.io/part-of: argocd
  name: example-argocd-cluster
  namespace: default
  ownerReferences:
  - apiVersion: argoproj.io/v1beta1
    blockOwnerDeletion: true
    controller: true
    kind: ArgoCD
    name: example-argocd
    uid: 3ca05807-1ee8-46f6-b240-fd263cb4d40e
  resourceVersion: "3156"
  uid: e4a4c650-0155-47a2-af31-53def5424bf5
type: Opaque
```

Now use the command to get the encrypted code of this secret

**echo "your\_code" | base64 -d**

```
rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ kubectl edit secret example-argocd-cluster
Edit cancelled, no changes made.

rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ echo Q1RVZ3FFYWZXMFIzSE0za2U5U3AyWWpYcmhHT21RNXY= | base64 -d
CTUgqEafW0RsHM3ke9Sp2YjXrhG0mQ5v
rgukt@pushpaPC MINGW64 ~/desktop/MY_THINGS/AWS/Jenkins-Zero-to-Hero (main)
$ |
```

step:2 Now run the below command , so that we get default url to access our cluster

**minikube service example-argocd-server**

```
rgukt@pushpaPC MINGW64 ~/desktop/Jenkins-Zero-to-Hero (main)
$ minikube service example-argocd-server
```

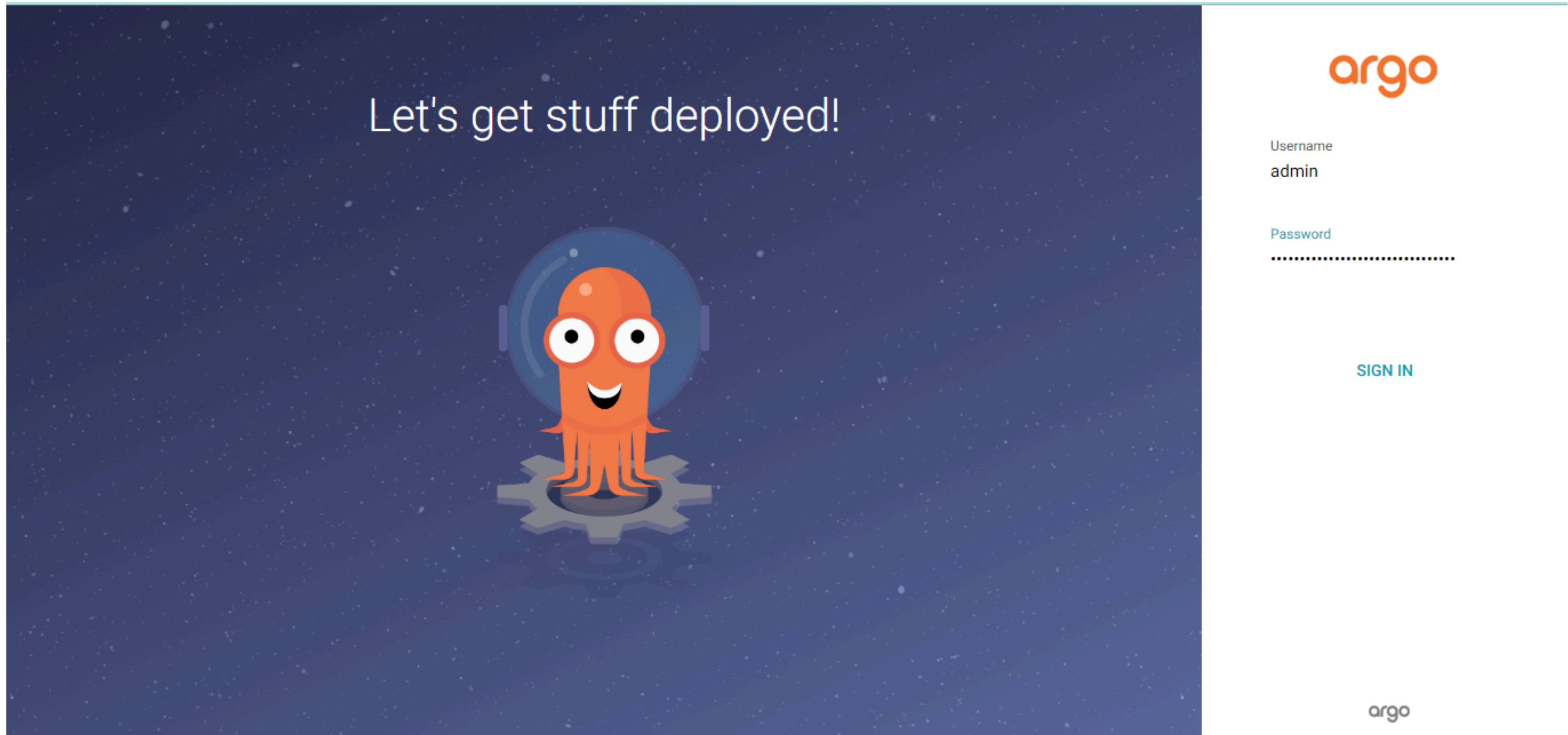
NAMESPACE	NAME	TARGET PORT	URL
default	example-argocd-server	http/80	http://192.168.49.2:31498
		https/443	http://192.168.49.2:31382

\* Starting tunnel for service example-argocd-server.

NAMESPACE	NAME	TARGET PORT	URL
default	example-argocd-server		http://127.0.0.1:51934
			http://127.0.0.1:51935

```
[default example-argocd-server http://127.0.0.1:51934
http://127.0.0.1:51935]
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
```

step:3 Now edit one of the url and provide the password that we got and give username as your choise



The image shows the Argo CD login interface. On the left, a dark blue panel with a starry background features the text "Let's get stuff deployed!" and a cartoon orange octopus wearing a blue space helmet, standing on a grey gear. On the right, a white panel contains the Argo logo, a "Username" field with the value "admin", a "Password" field with a masked input, a "SIGN IN" button, and a small Argo logo at the bottom right.

Let's get stuff deployed!

argo

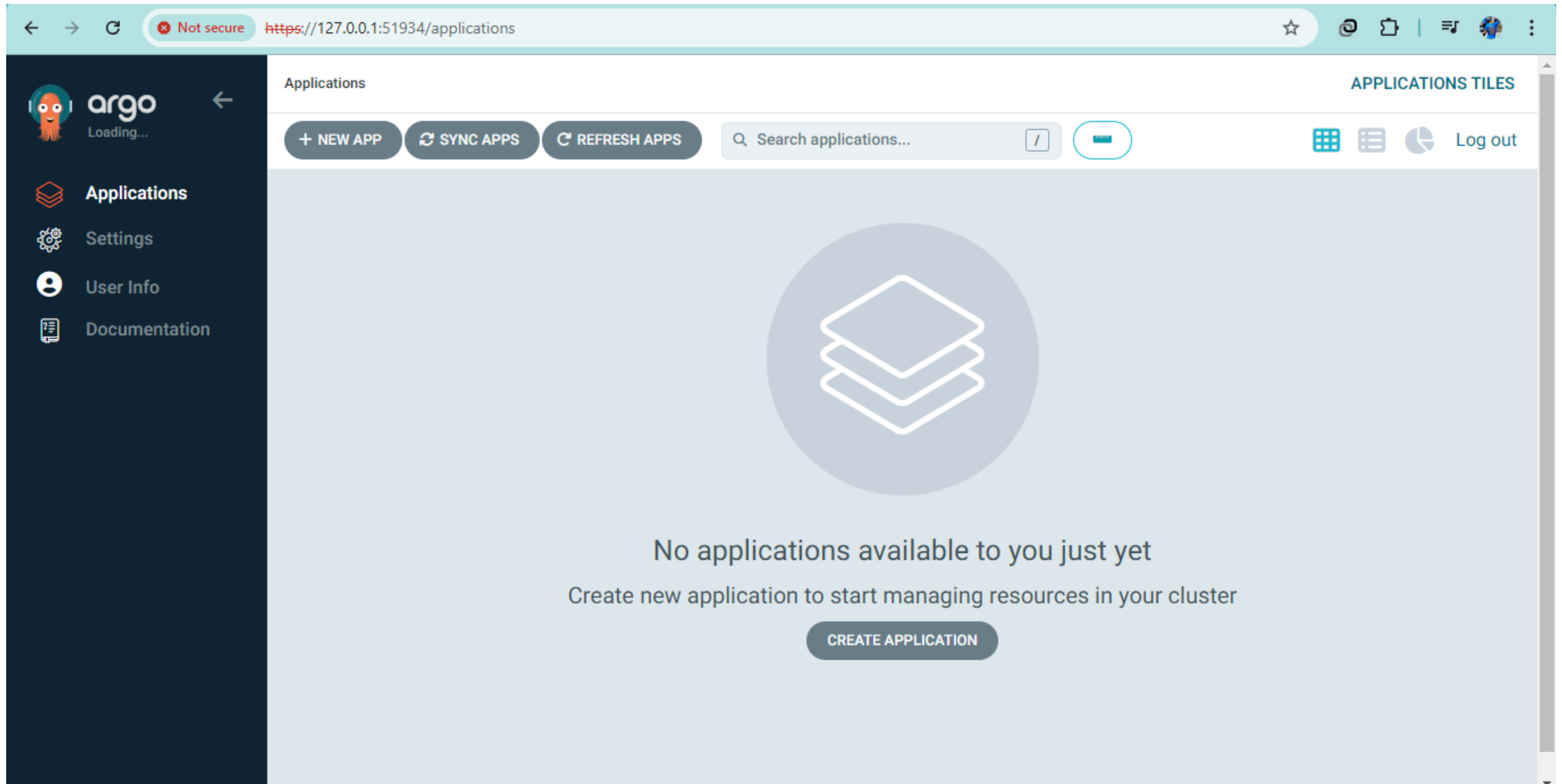
Username  
admin

Password  
.....

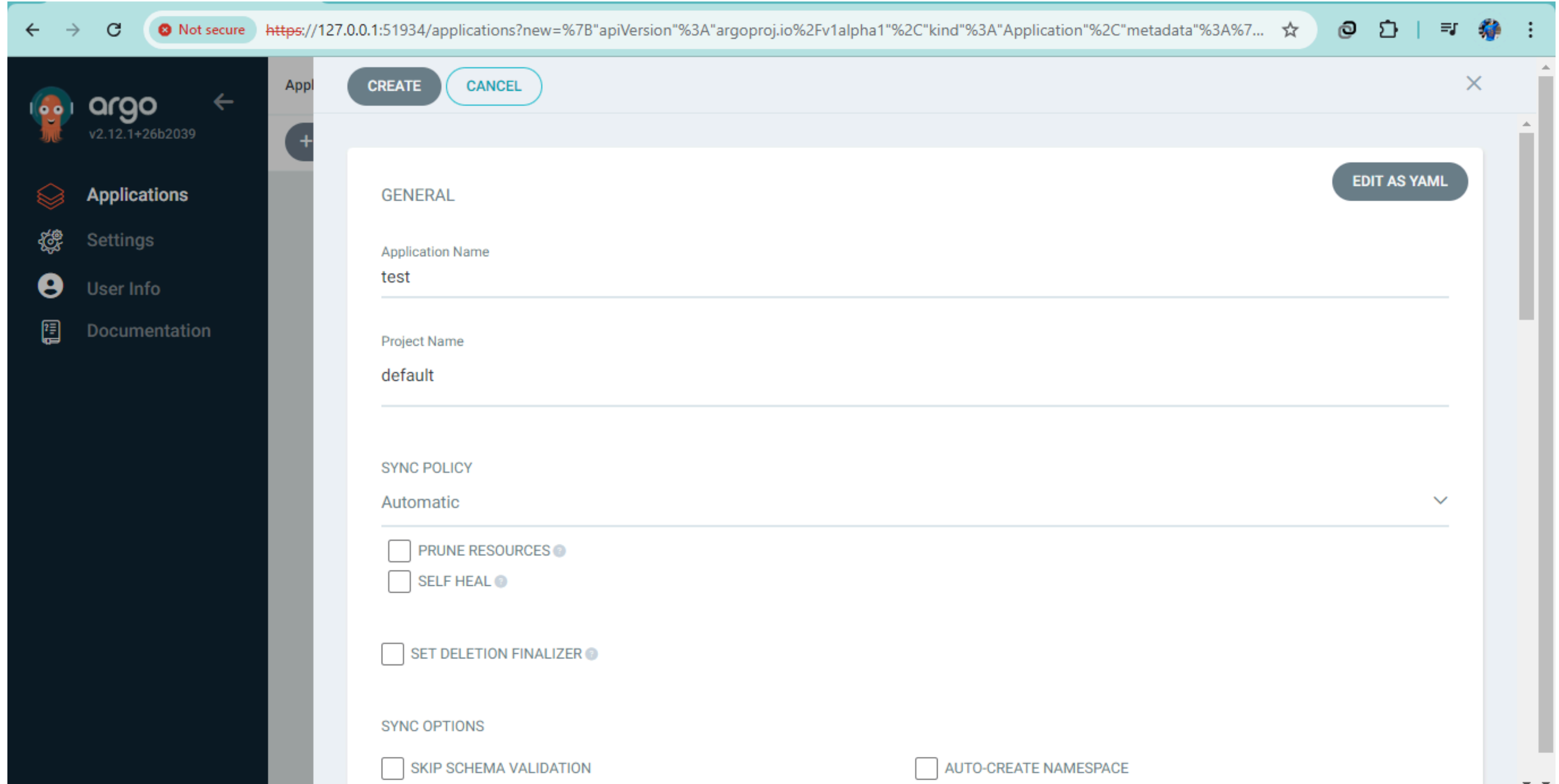
SIGN IN

argo

After logging in this is the interface we see



step:4 Now click on "new app" and click on "create" and give application name , project name , destination url , namespace , repository url and dockerfile path and click on create



The screenshot shows the Argo CD web interface in a browser. The address bar indicates a local development environment at `https://127.0.0.1:51934/applications?new=%7B%22apiVersion%3A%22argoproj.io%2Fv1alpha1%2C%22kind%3A%22Application%2C%22metadata%3A%7B%22name%3A%22test%22%2C%22namespace%3A%22default%22%7D%7D%7D`. The left sidebar contains the Argo logo and navigation links: Applications, Settings, User Info, and Documentation. The main content area is titled 'CREATE' and 'CANCEL' buttons. The form is divided into sections: GENERAL, SYNC POLICY, and SYNC OPTIONS. The 'GENERAL' section has input fields for 'Application Name' (filled with 'test') and 'Project Name' (filled with 'default'). The 'SYNC POLICY' section shows 'Automatic' as the selected policy. The 'SYNC OPTIONS' section includes checkboxes for 'PRUNE RESOURCES', 'SELF HEAL', 'SET DELETION FINALIZER', 'SKIP SCHEMA VALIDATION', and 'AUTO-CREATE NAMESPACE'. An 'EDIT AS YAML' button is located in the top right corner of the form.

Not secure <https://127.0.0.1:51934/applications?new=%7B%22apiVersion%3A%22argoproj.io%2Fv1alpha1%2C%22kind%3A%22Application%2C%22metadata%3A%7B%22name%3A%22test%22%2C%22namespace%3A%22default%22%7D%7D%7D>

argo v2.12.1+26b2039

Applications

Settings

User Info

Documentation

CREATE CANCEL

EDIT AS YAML

GENERAL

Application Name

test

Project Name

default

SYNC POLICY

Automatic

☐ PRUNE RESOURCES ?

☐ SELF HEAL ?

☐ SET DELETION FINALIZER ?

SYNC OPTIONS

☐ SKIP SCHEMA VALIDATION

☐ AUTO-CREATE NAMESPACE





- Applications
- Settings
- User Info
- Documentation

Application

CREATE

CANCEL



+ NEW

### DESTINATION

Cluster URL

https://kubernetes.default.svc

URL ▼

Namespace

default

Directory ▼

### DIRECTORY


DIRECTORY RECURSE

☐


TOP-LEVEL ARGUMENTS


No items





 **argo**  
v2.12.1+26b2039

←

 Applications

 Settings

 User Info

 Documentation

Application

CREATE

CANCEL

×

☐ REPLACE ⚠

☐ RETRY

SOURCE

Repository URL

https://github.com/PushpaShreeBellamkonda/Jenkins-Zero-To-Hero

GIT ▼

Revision

HEAD

Branches ▼

Path

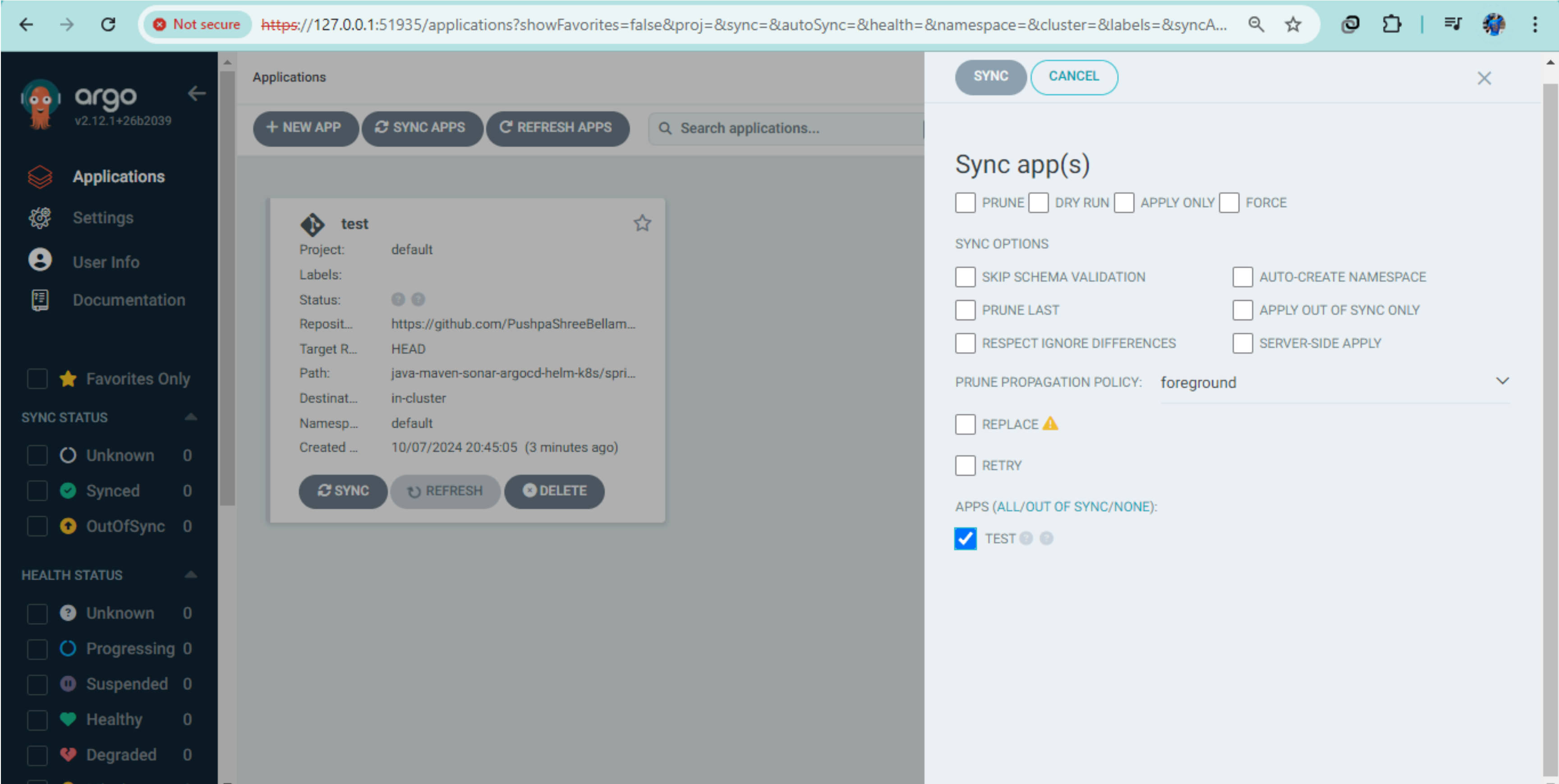
java-maven-sonar-argocd-helm-k8s/spring-boot-app-manifests

DESTINATION

This is the application we have created

The screenshot displays the Argo CD web interface in a browser. The address bar shows the URL `https://127.0.0.1:51935/applications?showFavorites=false&proj=&sync=&autoSync=&health=&namespace=&cluster=&labels=`. The interface includes a dark sidebar on the left with the Argo logo and version `v2.12.1+26b2039`, and navigation links for Applications, Settings, User Info, and Documentation. Below these are filters for 'Favorites Only' and status counts for SYNC STATUS (Unknown, Synced, OutOfSync) and HEALTH STATUS (Unknown, Progressing, Suspended, Healthy, Degraded). The main content area is titled 'Applications' and features buttons for '+ NEW APP', 'SYNC APPS', and 'REFRESH APPS', along with a search bar. A modal window for an application named 'test' is open, showing details: Project: default, Labels: (empty), Status: (unknown), Repository: `https://github.com/PushpaShreeBellam...`, Target R...: HEAD, Path: `java-maven-sonar-argocd-helm-k8s/spri...`, Destination: in-cluster, Namespace: default, and Created: 10/07/2024 20:45:05. At the bottom of the modal are buttons for SYNC, REFRESH, and DELETE. The top right corner shows 'APPLICATIONS TILES' and a 'Log out' button.

step:5 Now click on "sync apps" and check the application "test" and sync it



argo  
v2.12.1+26b2039

Applications

Settings

User Info

Documentation

☐ ★ Favorites Only

SYNC STATUS

☐ ⌚ Unknown 0

☐ ✓ Synced 0

☐ ⬆ OutOfSync 0

HEALTH STATUS

☐ ? Unknown 0

☐ 🔄 Progressing 0

☐ ⏸ Suspended 0

☐ 🟢 Healthy 0

☐ 🟡 Degraded 0

☐ 🟠 Missing 0

Applications

+ NEW APP

↻ SYNC APPS

🔄 REFRESH APPS

🔍 Search applications...

test

Project: default

Labels:

Status: ● ●

Reposit... https://github.com/PushpaShreeBellam...

SYNC

CANCEL

Sync app(s)

☐ PRUNE

☐ DRY RUN

☐ APPLY ONLY

☐ FORCE

SYNC OPTIONS

☐ SKIP SCHEMA VALIDATION

☐ AUTO-CREATE NAMESPACE

☐ PRUNE LAST

☐ APPLY OUT OF SYNC ONLY

OVER-SIDE APPLY

Complete

CLOSE

☒ TEST ● ●

step:6 Now check the pods and deployment , we see that our pods are deployed and running

```
rgukt@pushpaPC MINGW64 ~
```

```
$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
example-argocd-application-controller-0	0/1	Error	1	43m
example-argocd-redis-6545fd6d6c-hg4hv	0/1	Error	7	47h
example-argocd-repo-server-869d5757c7-zz7b8	0/1	Pending	44	47h
example-argocd-server-76bb84cddc-vdrjq	0/1	Error	44	47h
spring-boot-app-7f67db6bf-ggsnt	1/1	Running	2 (4m10s ago)	18h
spring-boot-app-7f67db6bf-mr2bz	1/1	Running	2	18h

```
rgukt@pushpaPC MINGW64 ~
```

```
$ kubectl get deploy
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
example-argocd-redis	1/1	1	1	47h
example-argocd-repo-server	0/1	1	0	47h
example-argocd-server	0/1	1	0	47h
spring-boot-app	2/2	2	2	18h

```
rgukt@pushpaPC MINGW64 ~
```

```
$ |
```