**TASK-03 [JENKINS CICD –PIPELINE]**

**1) Create one Jenkins job using the below code and create three stages.**

Steps to follow:

Launch ec2 for Jenkins server

After launching , Install all the prerequisites

* sudo wget -O /etc/yum.repos.d/jenkins.repo <https://pkg.jenkins.io/redhat-stable/jenkins.repo>
* sudo rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key>
* sudo yum install java-17-amazon-corretto-devel
* yum install Jenkins

After installing all the prerequisites,

Start the Jenkins service

* systemctl start Jenkins
* systemctl status Jenkins

Now in browser(publicip:8080)

We can see the Jenkins Default page

Login to that with initial administrator password

Install all the plugins

Sign in as admin

CICD PIPELINE

**stage1: Git clone to download the source code.**

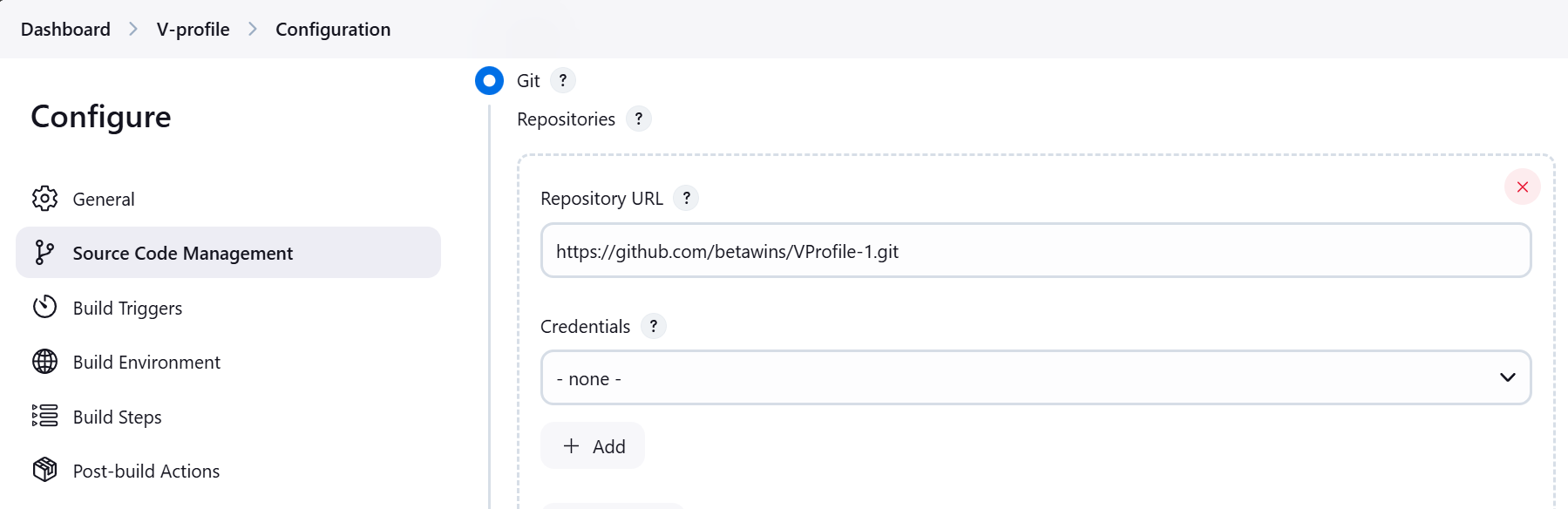
Go to the Jenkins Page

Create a job with freestyle

Go to Dashboard>Source code management

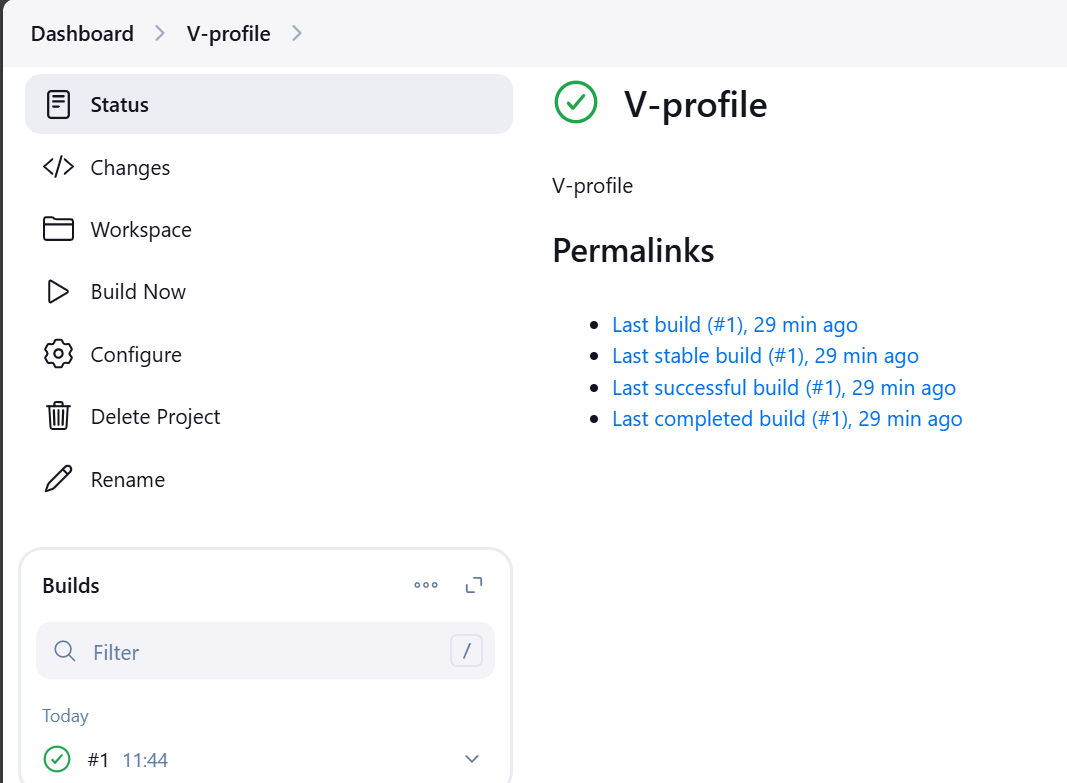
* (Download git in the local machine of Jenkins)

As we are attaching the git repo url



Save the details

Execute the job(Build Now)



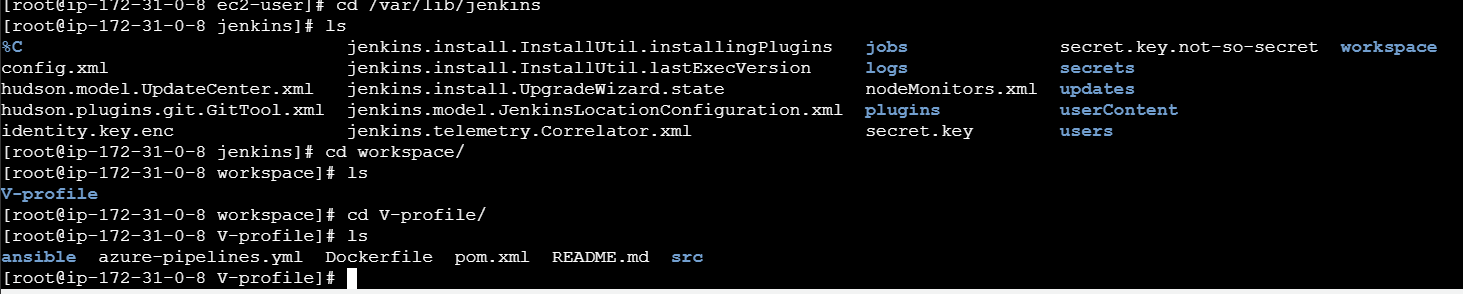
Now go to the Jenkins server(Instance)

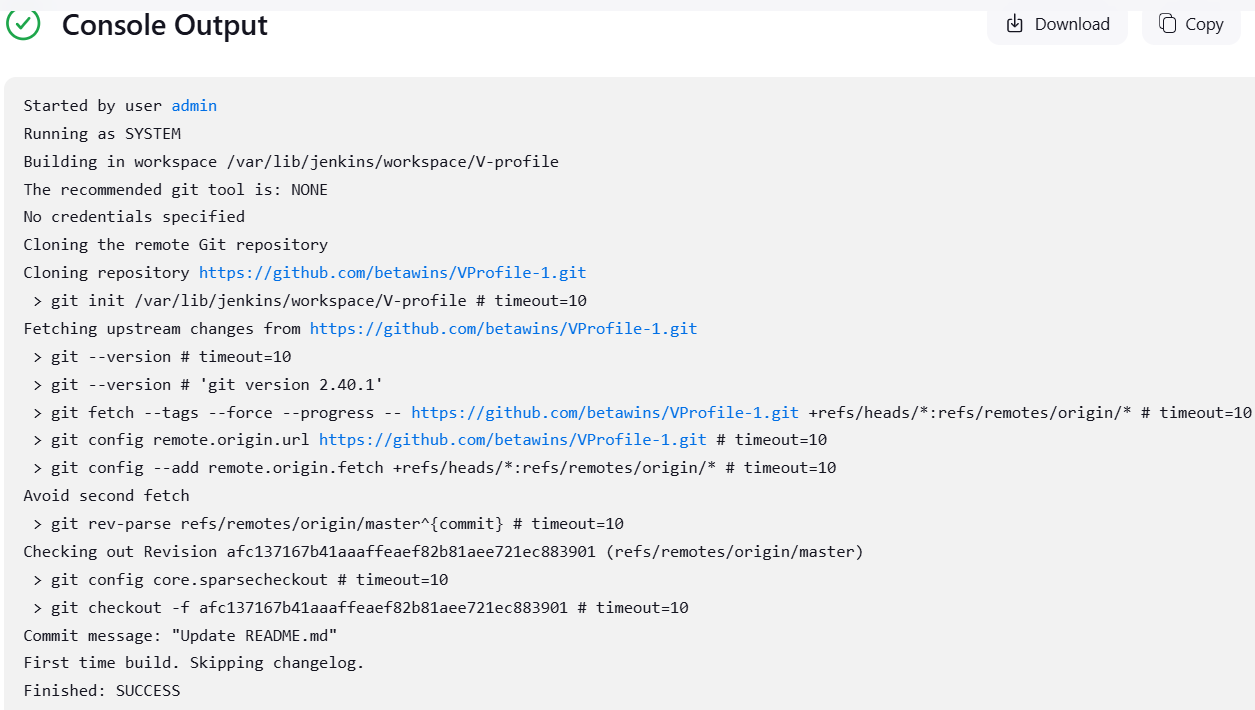
We can use the command

* cd /var/lib/Jenkins [we will find all the files of Jenkins]

we need to check for workspace as we executed a job in Jenkins

* cd workspace/ [we will find details of job]



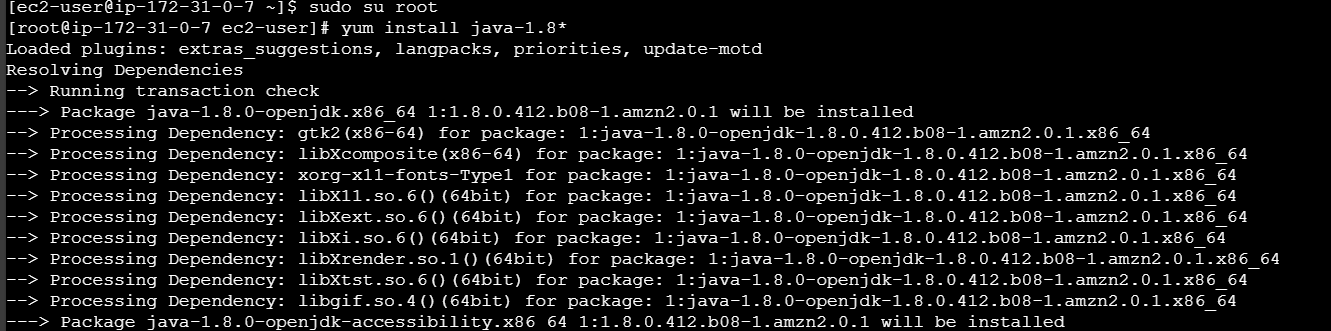
Successfully , we can able to clone

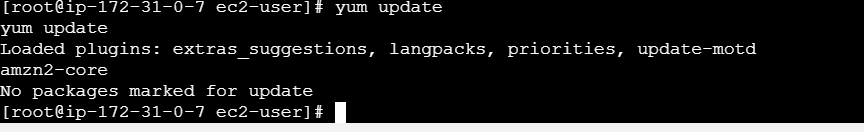
**stage2: SonarQube Integration to check the quality of code**

*Prerequisites:*

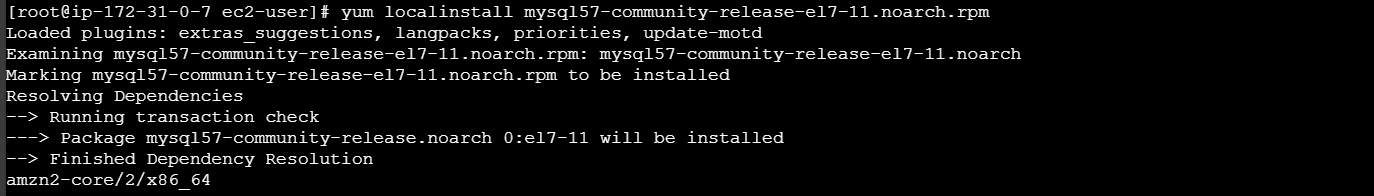
* Ec2 instance with java installed
* Use t2.large with atleast 20gb memory to run sonarQube
* Mysql DB

**Steps:**

* Installing the java
* Yum update

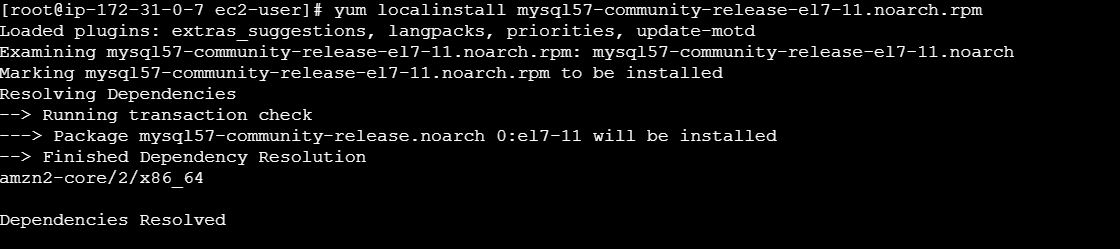


* Downloading the mysql
* wget <https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm>
* yum localinstall mysql57-community-release-el7-11.noarch.rpm
* rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022

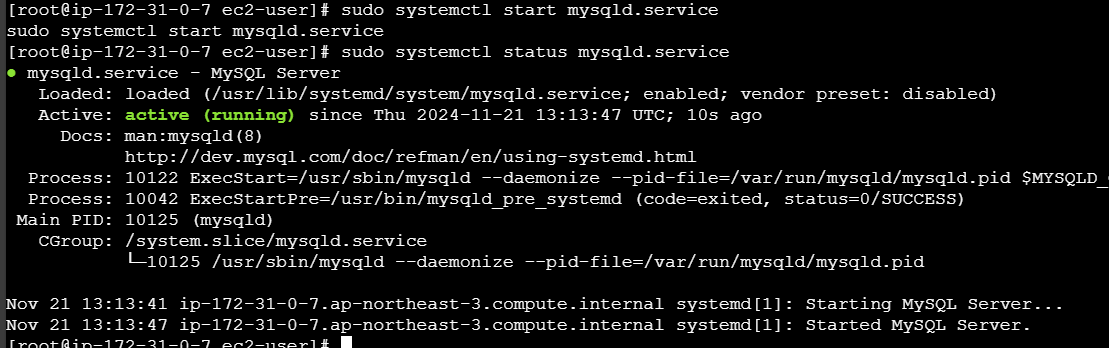




Installing the mysql

sudo yum install mysql-community-server

sudo systemctl start mysqld.service

sudo systemctl status mysqld.service 

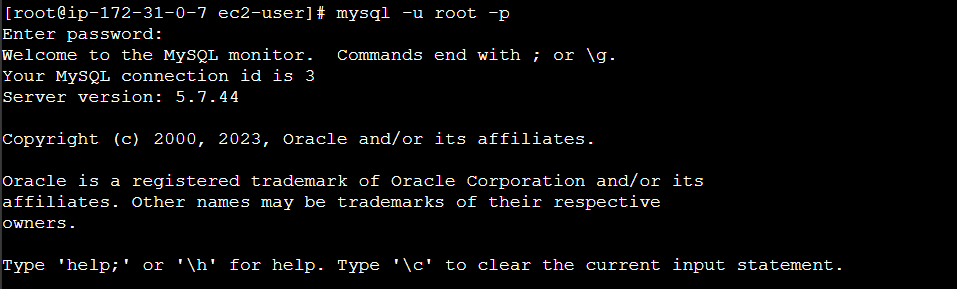
Configure the root password as we see the default root password

* grep 'temporary' /var/log/mysqld.log

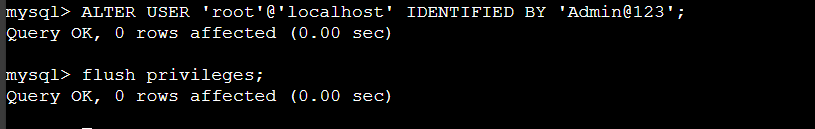


Login to mysql using the default password

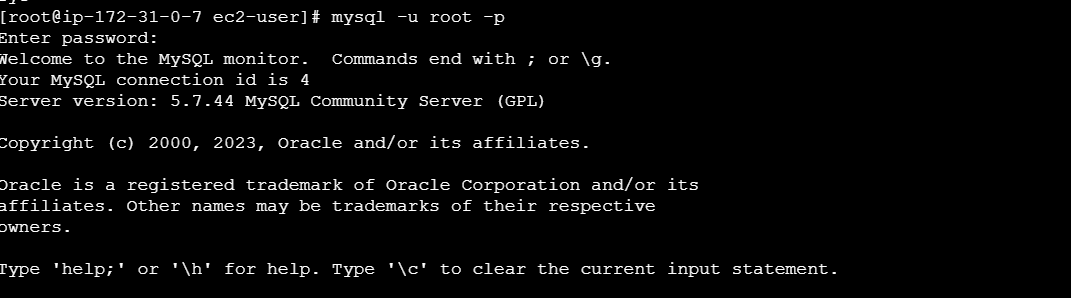
* mysql -u root –p
* (Here the password: Fer?Sgdhr6C%)



Now replace the default password with a new and strong password



Test Using new password

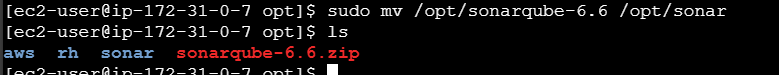
 Download & unzip SonarQube 6.0

# cd /opt

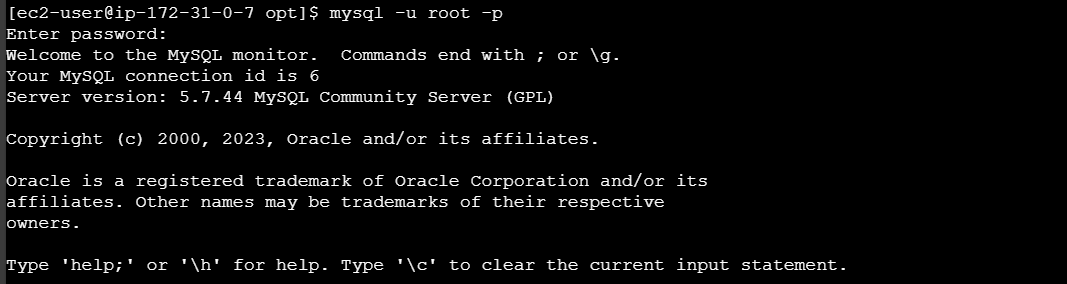
# wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-6.6.zip

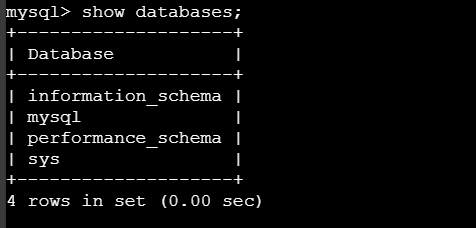
# unzip sonarqube-6.6.zip

# mv /opt/sonarqube-6.6 /opt/sonar

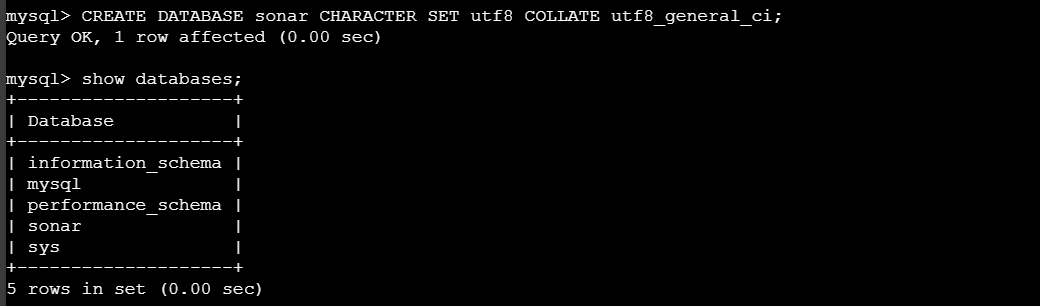


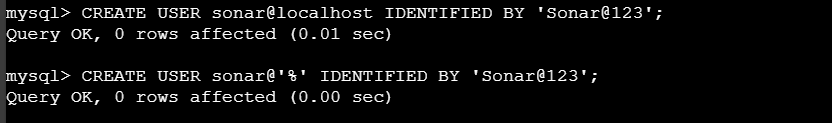
Now go to mysql



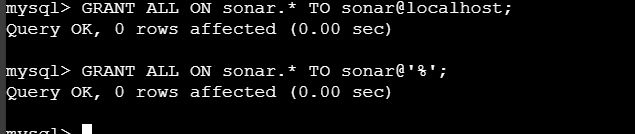


CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8\_general\_ci;

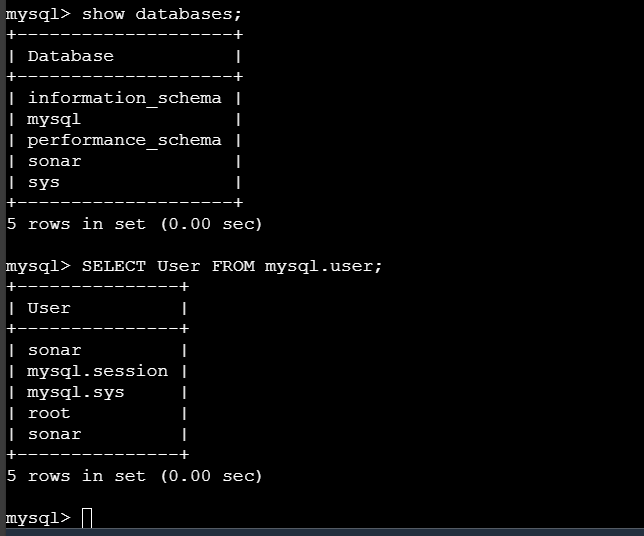


Create a local and a remote user 

Grant database access permissions to users



check users and databases



ON EC2 Instance

Edit sonar properties file to uncomment and provide required information for below properties.

Going EC2 instance and enable SonarQube properties file to connect his Database.

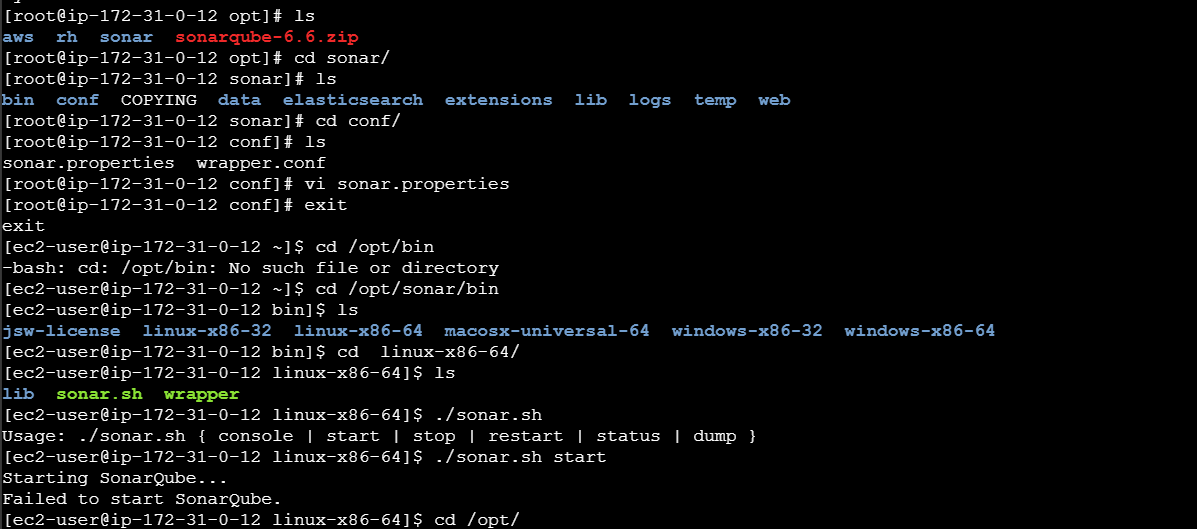
- File Name: /opt/sonar/conf/sonar.properties

- sonar.jdbc.username=`sonar`

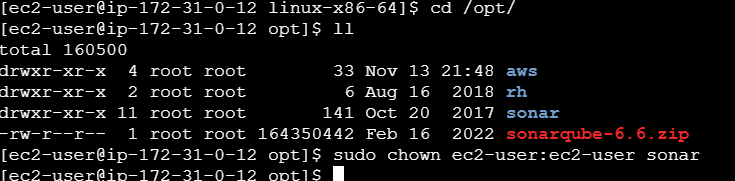
- sonar.jdbc.password=`Sonar@123`

- sonar.jdbc.url=jdbc:mysql://`localhost:3306`/sonar?useUnicode=true&characterEncoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance&useSSL=false

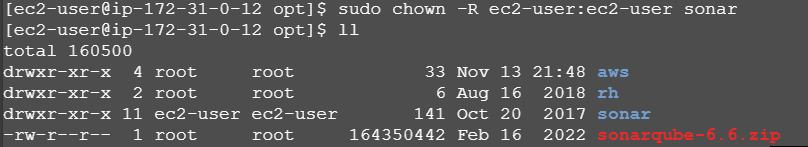
- sonar.web.host=`0.0.0.0`

- sonar.web.context=`/sonar`

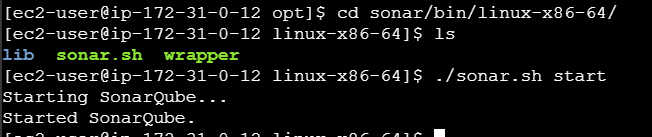
As we can see that sonarQube failed to start as ‘sonar’ directory does not have user access

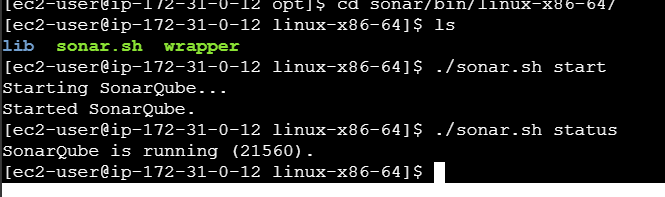


Changing the permissions



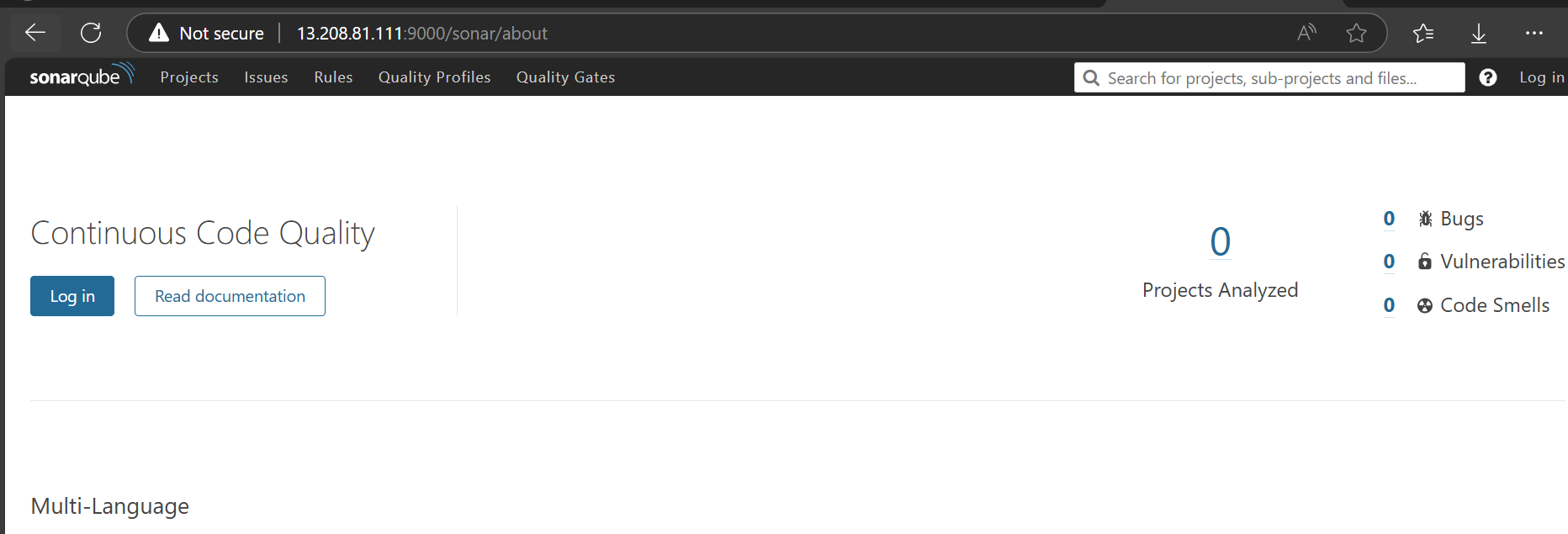
After changing the permissions we can see that sonarqube was started and running



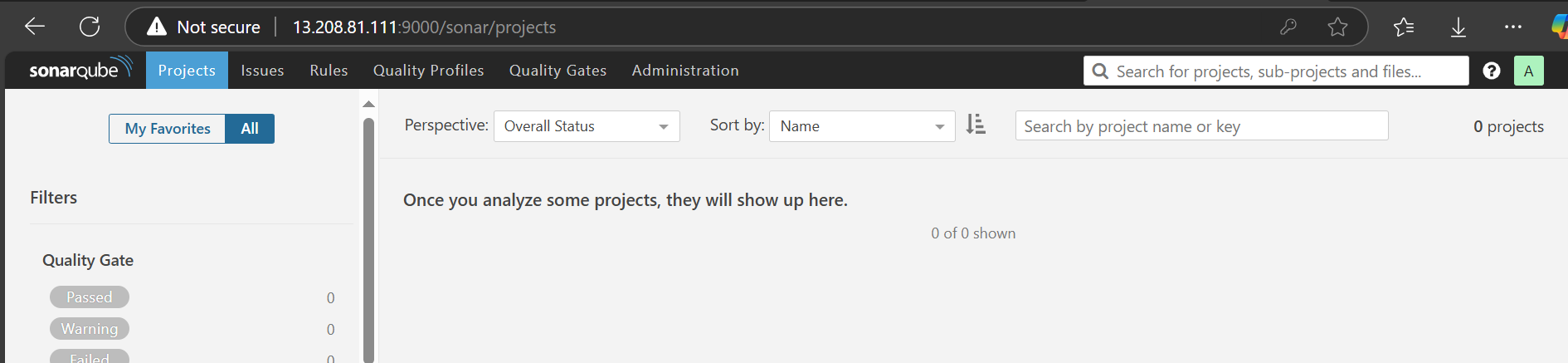


Checking in the browser

We are using :public ip:9000/sonar/



Login with the admin account



Login to Jenkins server and install sonarqube scanner.

- SonarQube scanner URL : https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner

- Package : https://sonarsource.bintray.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-3.2.0.1227-linux.zip

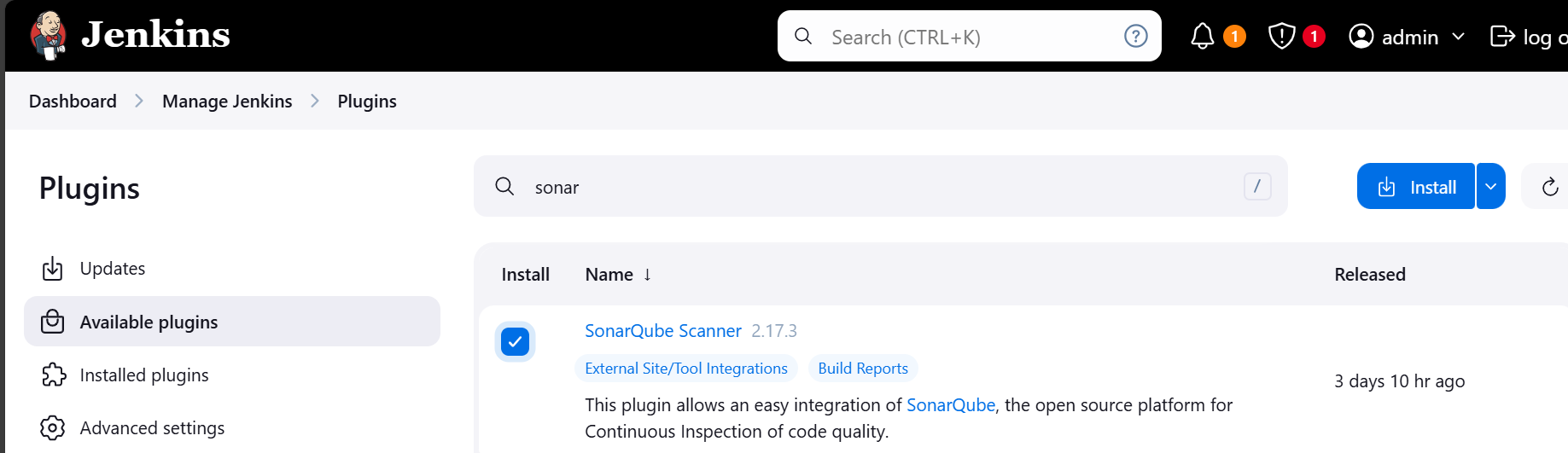
-

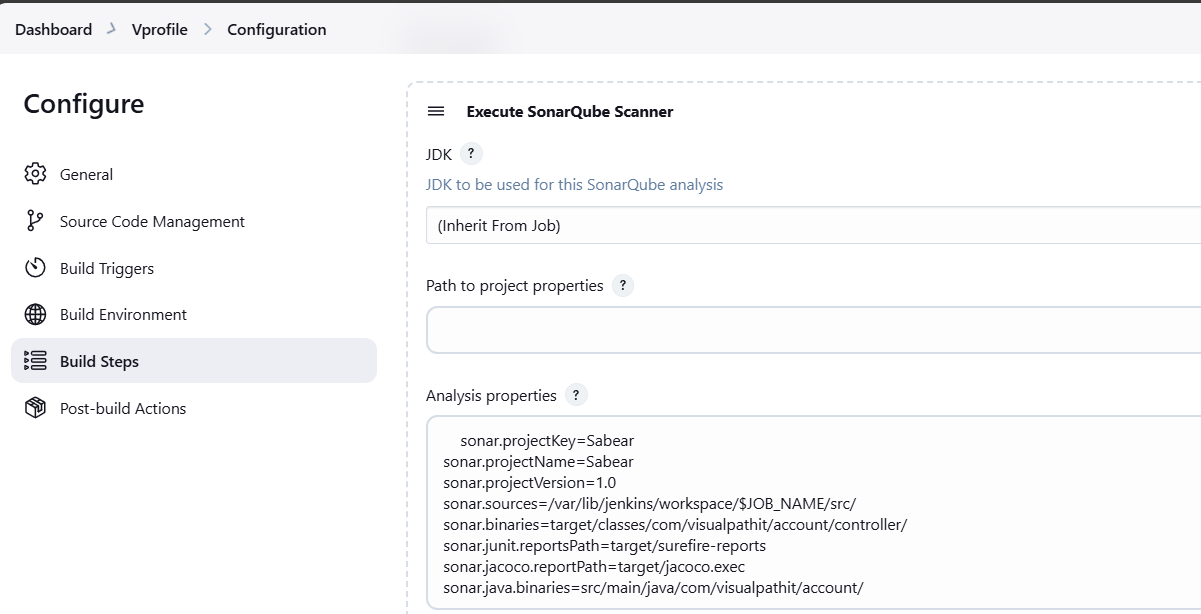
# wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.6.2.2472-linux.zip

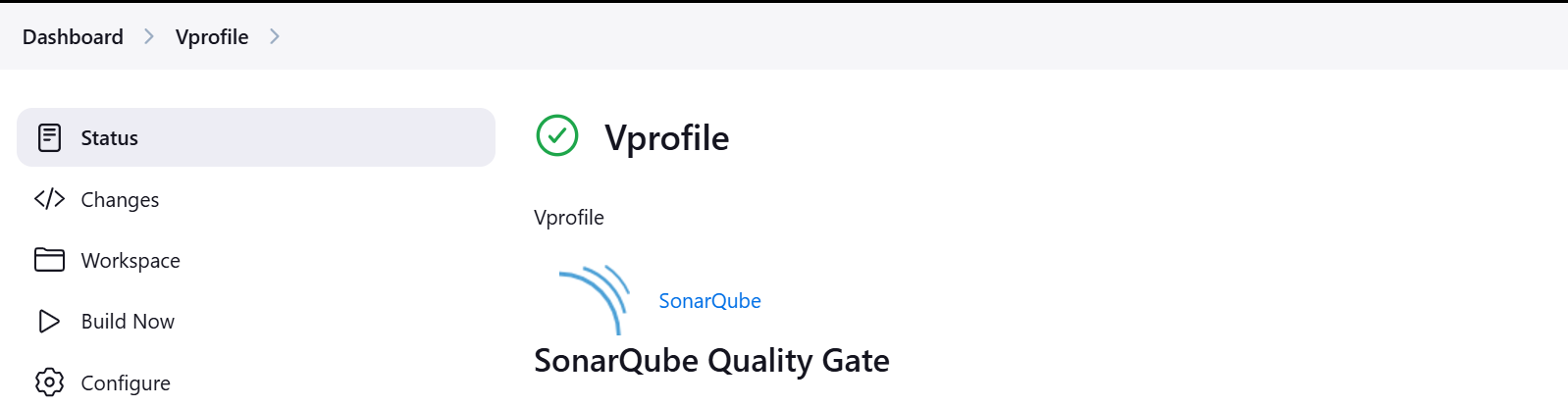
# unzip sonar-scanner-cli-4.6.2.2472-linux.zip

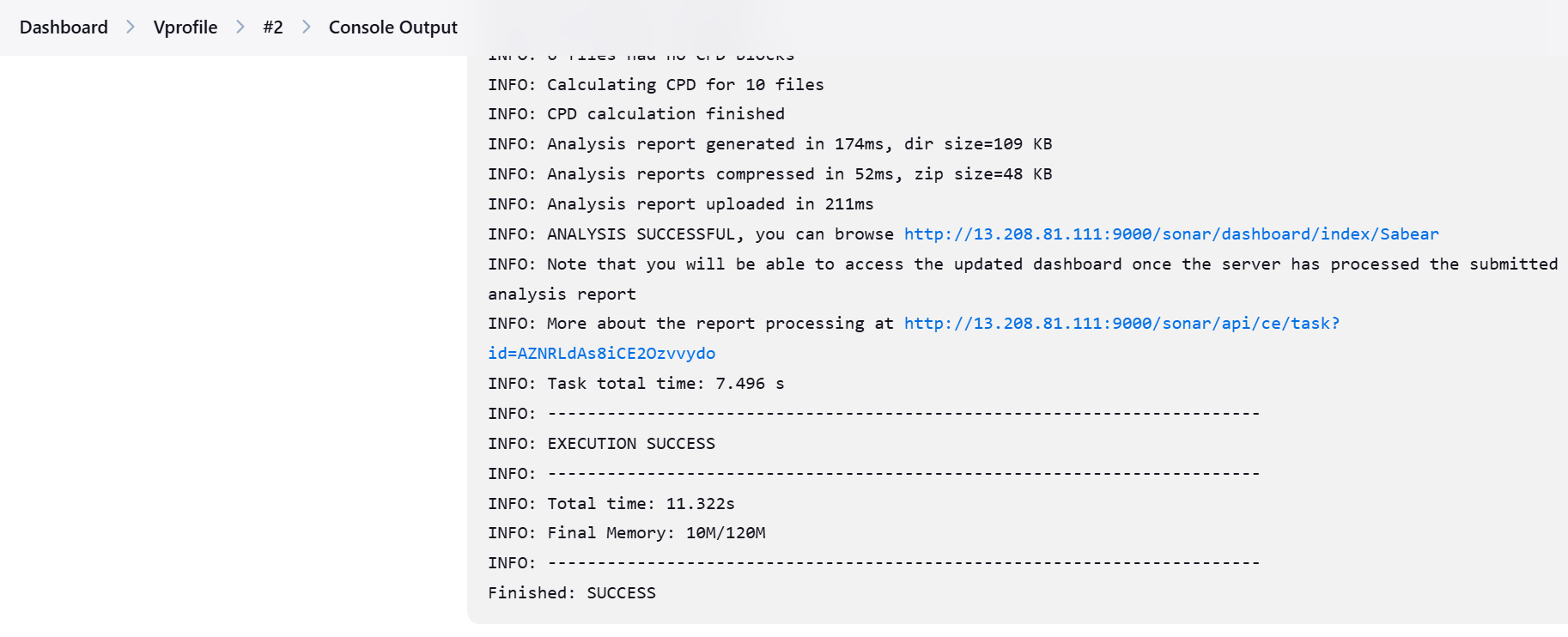
# mv sonar-scanner-4.6.2.2472-linux /opt/sonar\_scanner

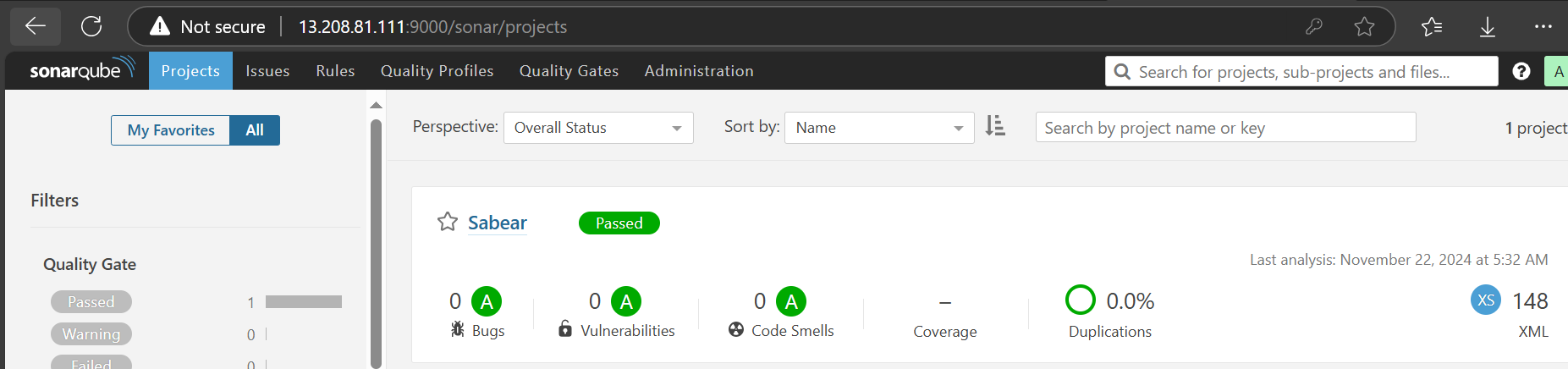
Downloading the plugin[sonarQube Scanner]









We can also check the report in sonarqube

**stage3: Slack Integration to send the alerts to slack.**

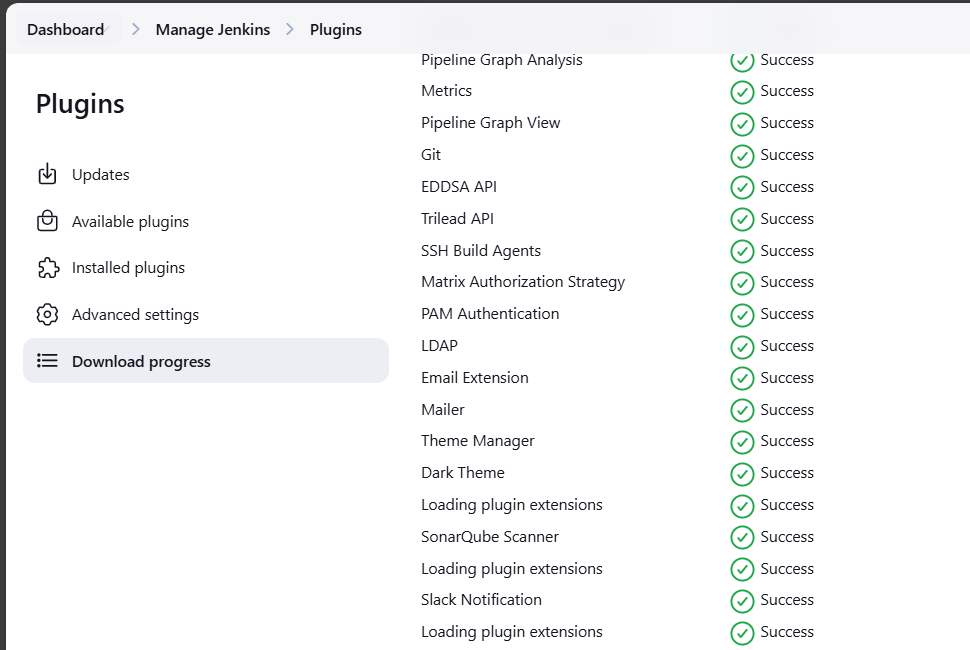
**URL:** [**https://github.com/betawins/VProfile-1.git**](https://github.com/betawins/VProfile-1.git)

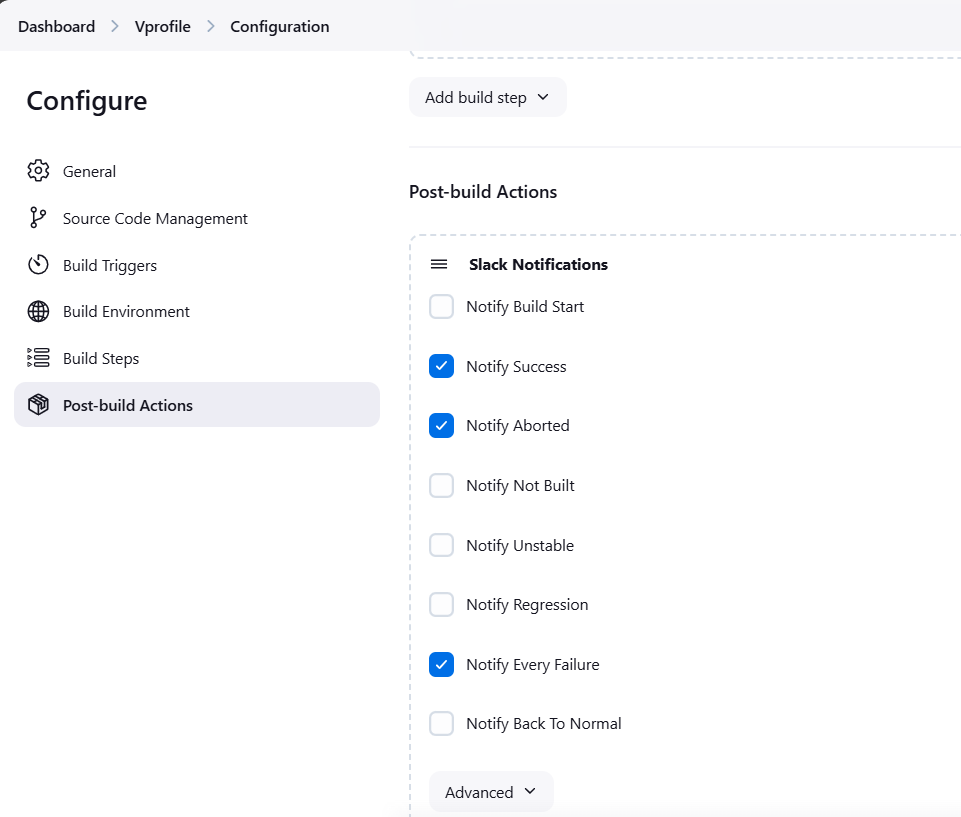
Downloading the plugin for slack Notification

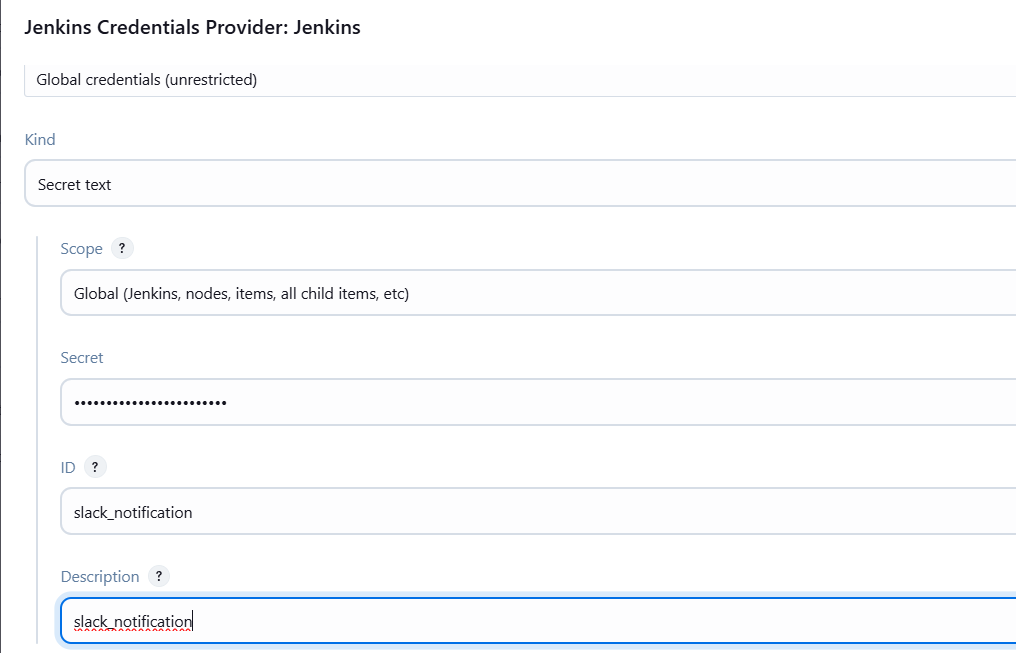
Dashboard>Manage Jenkins>Plugins

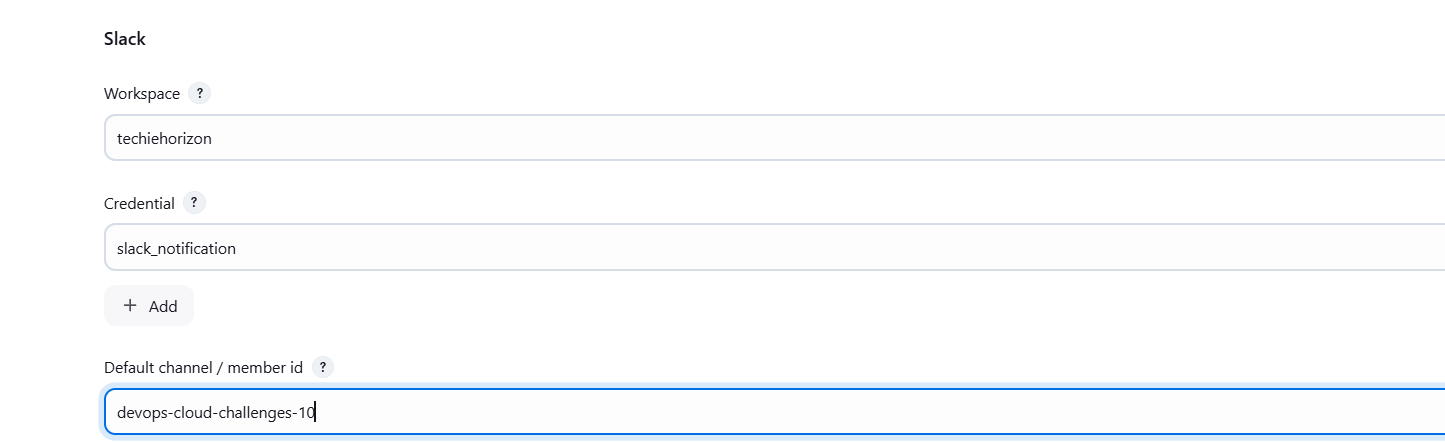
In available plugins>slack Notification

Install the plugin

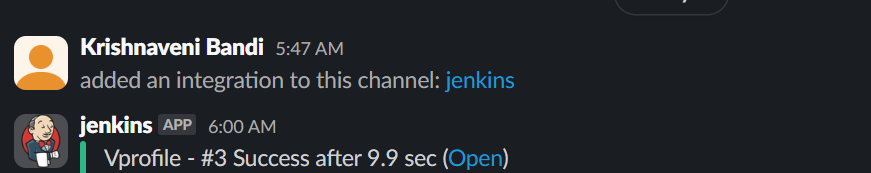






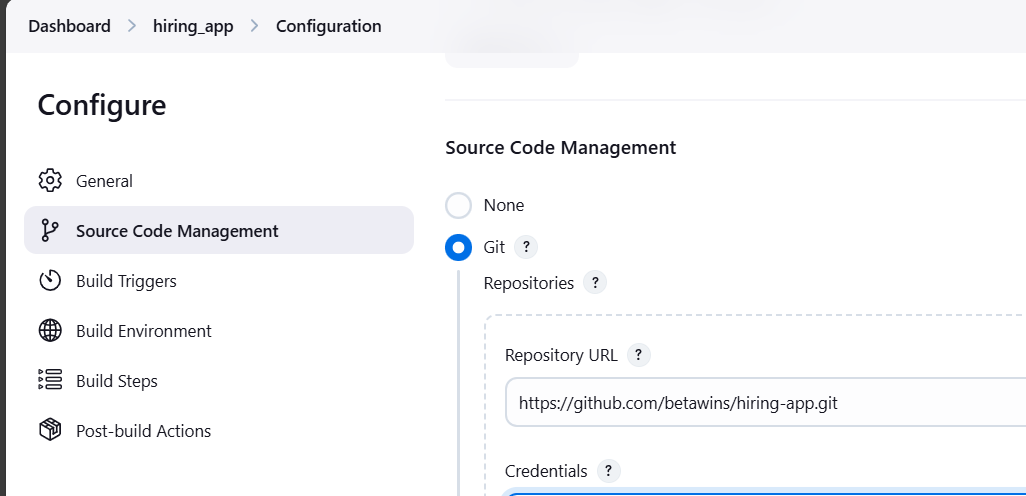


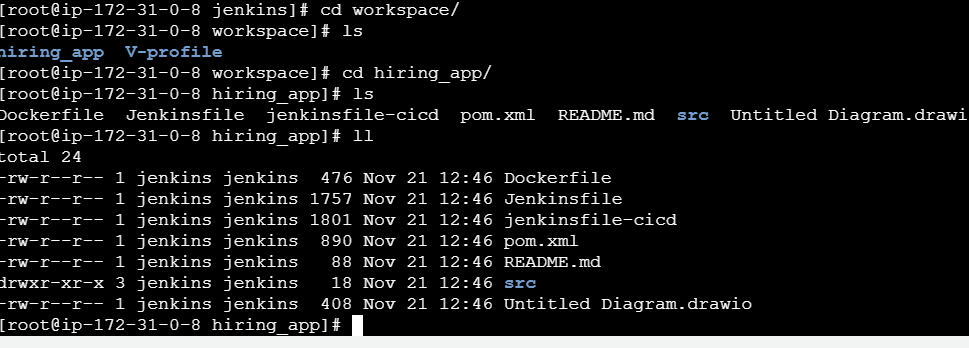
Output Generated in the slack



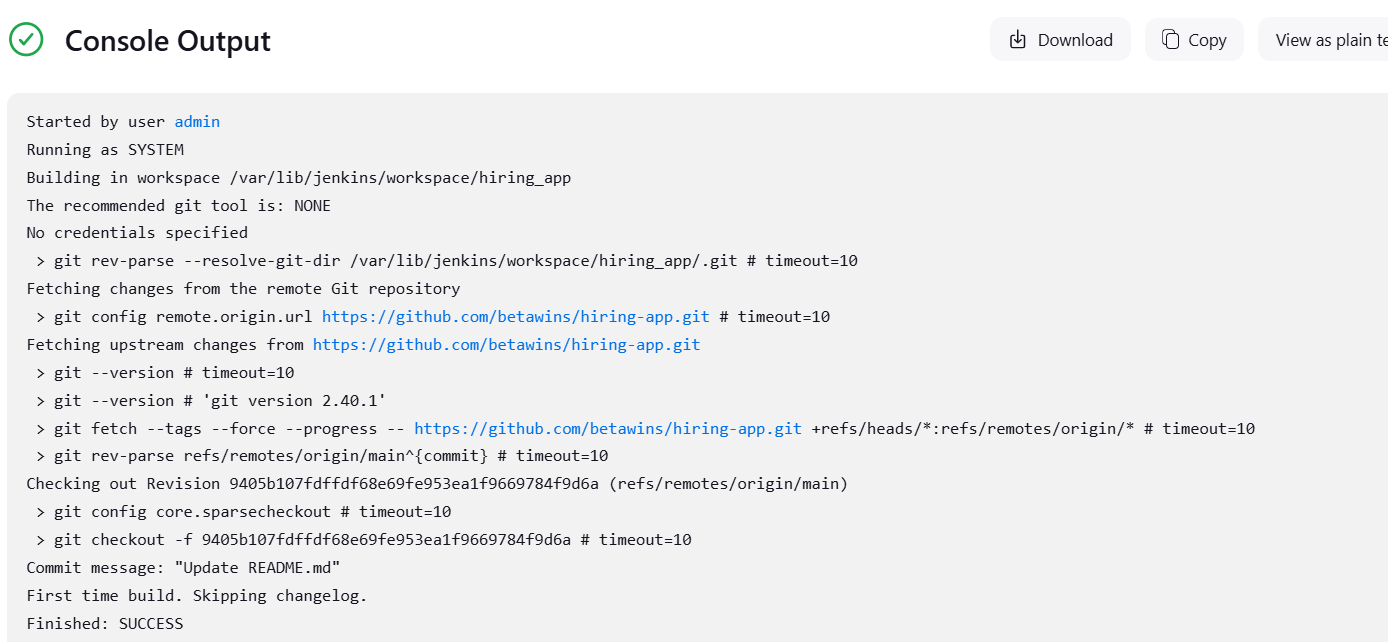
**2) 1) Create one Jenkins job using the below code and create three stages.**

**stage1: Git clone to download the source code.**

Creating the Job and adding the configurations

Now we can 

Console output



**stage2: Sonarqube Integration to check the quality of code**

**Prerequisites**

1. EC2 instance with Java installed

2. Use t2.large with atleast 20gb memory to run sonarqube.

1. MySQL Database Server or MyQL RDS instance**.**

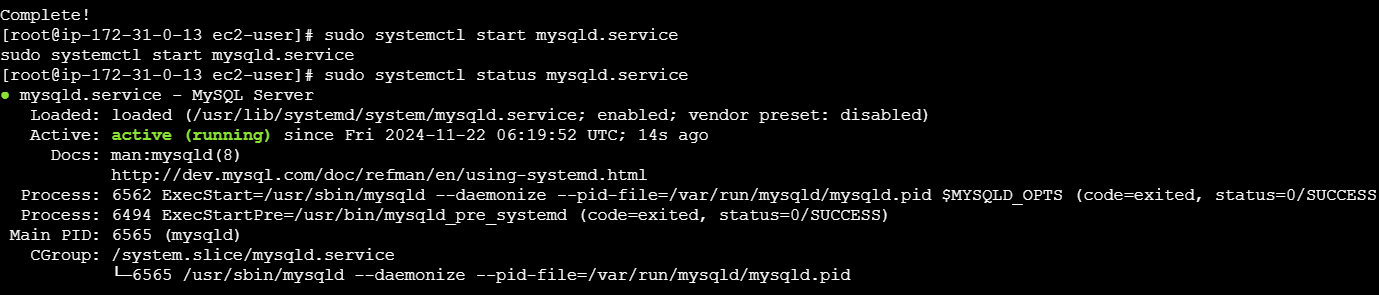
**Steps:**

Launching one ec2 instance with t2.large and 20 gb storage

Installed java

yum –y install java-1.8\*

* Downloading mysql sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
* sudo yum localinstall mysql57-community-release-el7-11.noarch.rpm
* rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022

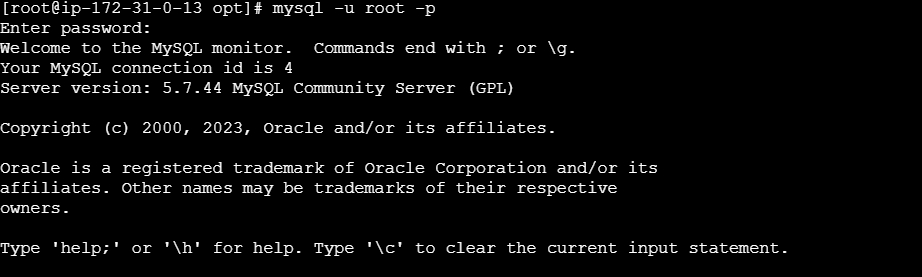


Configure the MySQL Root Password

You will see default MySQL root password

* grep 'temporary' /var/log/mysqld.log

Login to mysql using the default password

mysql -u root –pNow replace the default password with a new and strong password

* ALTER USER 'root'@'localhost' IDENTIFIED BY 'Admin@123';
* flush privileges;
* Create a new sonar database

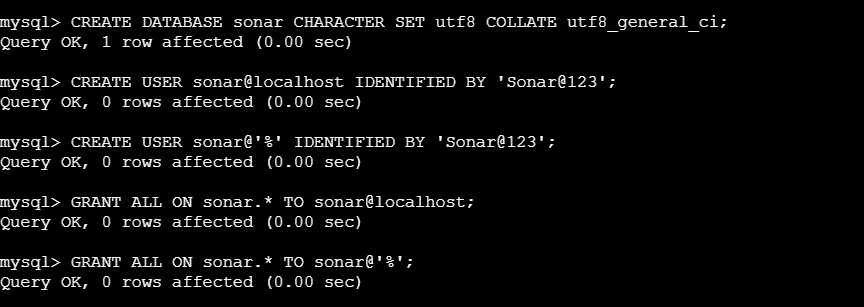
CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8\_general\_ci

* Create a local and a remote user
* CREATE USER sonar@localhost IDENTIFIED BY 'Sonar@123';
* CREATE USER sonar@'%' IDENTIFIED BY 'Sonar@123';

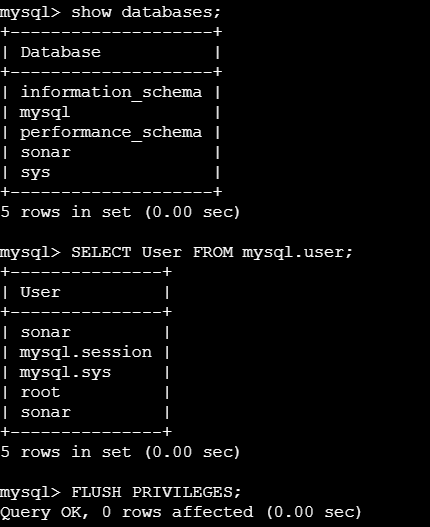
Grant database access permissions to users

GRANT ALL ON sonar.\* TO sonar@localhost;

GRANT ALL ON sonar.\* TO sonar@'%';



* check users and databases
* show databases;
* SELECT User FROM mysql.user;
* FLUSH PRIVILEGES;
* QUIT



ON EC2 Instance

Edit sonar properties file to uncomment and provide required information for below properties.

- File Name: /opt/sonar/conf/sonar.properties

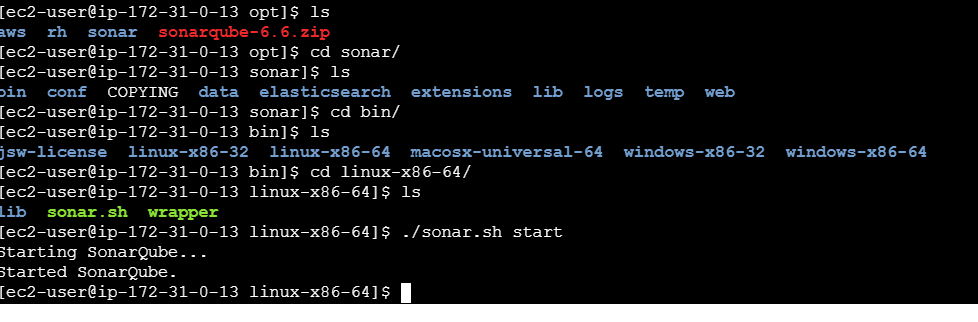
- sonar.jdbc.username=`sonar`

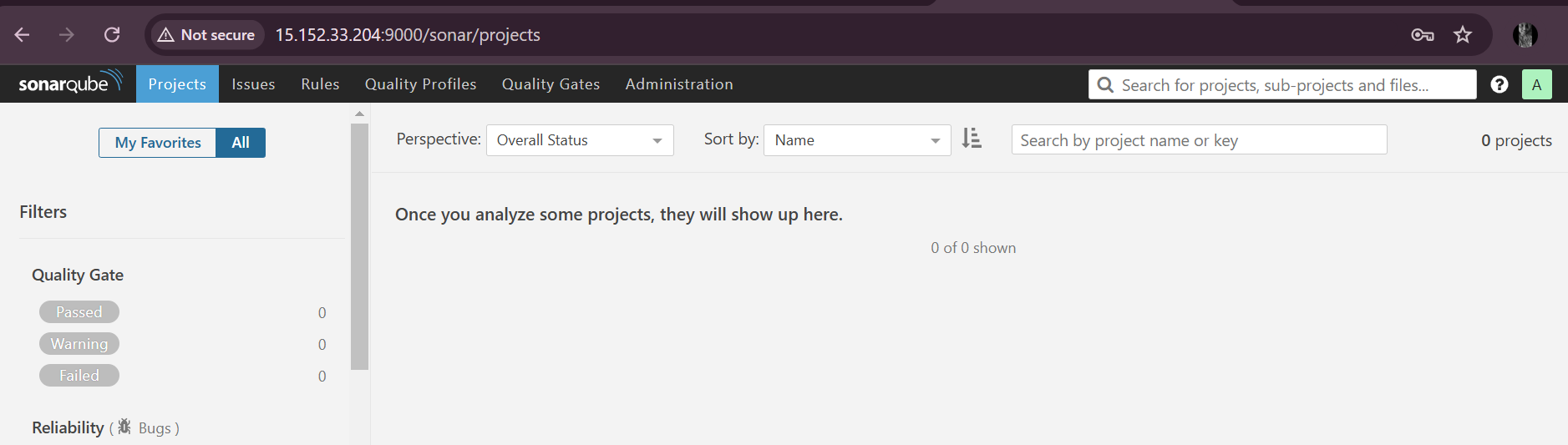
- sonar.jdbc.password=`Sonar@123`

- sonar.jdbc.url=jdbc:mysql://`localhost:3306`/sonar?useUnicode=true&characterEncoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance&useSSL=false

- sonar.web.host=`0.0.0.0`

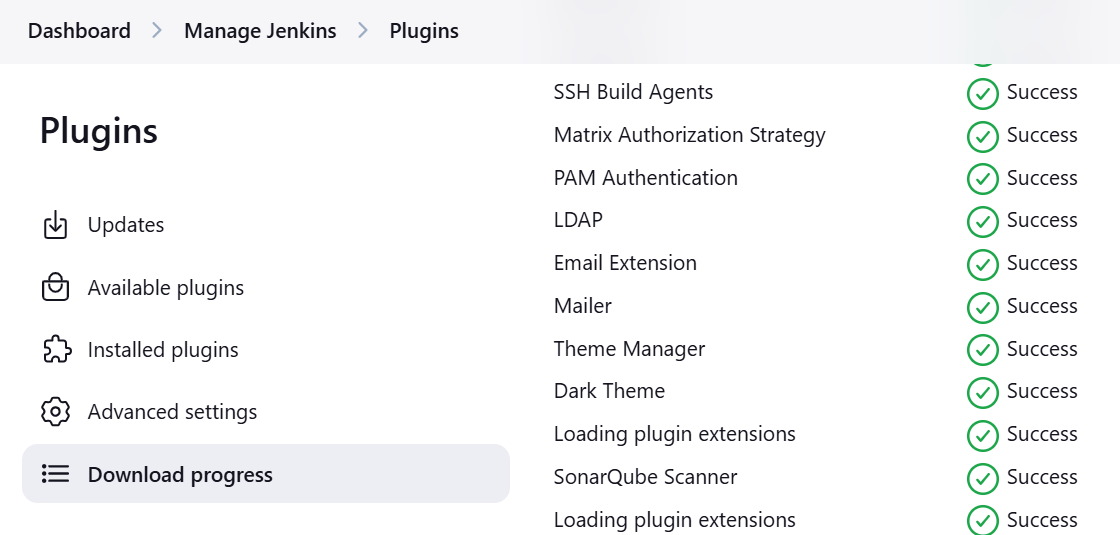
- sonar.web.context=`/sonar`





Now go to Jenkins browser and Install Sonarqube Scanner Pluggin Login to Jenkins GUI console and install " SonarQube scanner" plugin

`Manage Jenkins` > `Manage Plugins` > `Avalable` > `SonarQube scanner



* Next ,

Go to Manage Jenkins>system>sonarqube Installations



Login to Jenkins server and install sonarqube scanner.

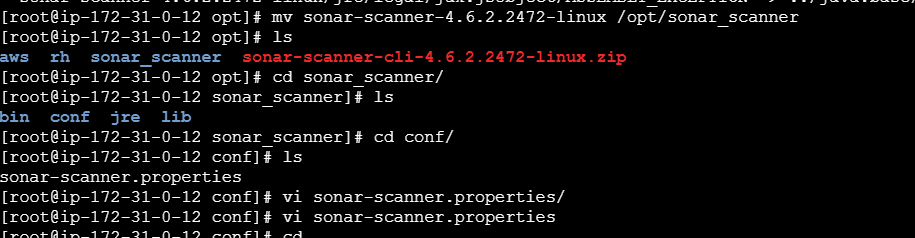
# wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.6.2.2472-linux.zip

# unzip sonar-scanner-cli-4.6.2.2472-linux.zip

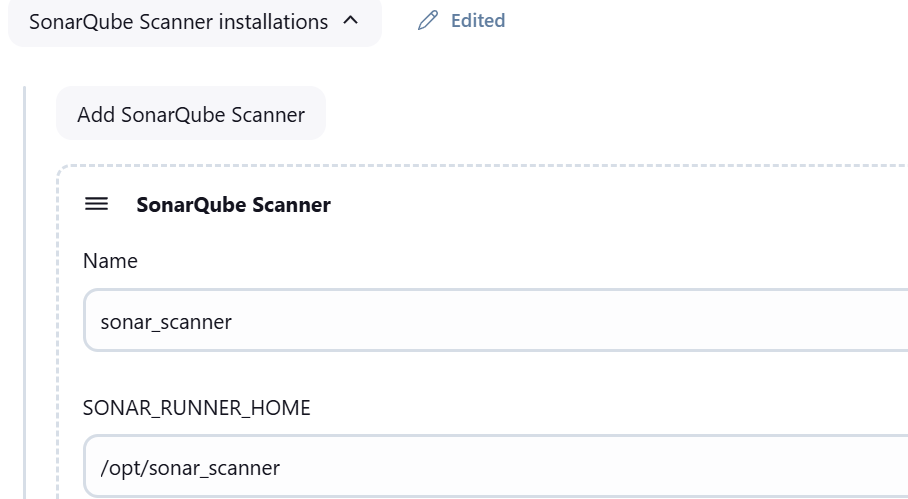
# mv sonar-scanner-4.6.2.2472-linux /opt/sonar\_scannerSet SonarQube server details in sonar-scanner property file

- Sonar properties file: /opt/sonar\_scanner/conf/sonar-scanner.properties

- sonar.host.url=http://`<SONAR\_SERVER\_IP>`:9000



Next,

Jenkins>Manage Jenkins>Tools

It is a traditional method

`

**USING DOCKER**

**In sonarqube server**

* Installing the java package

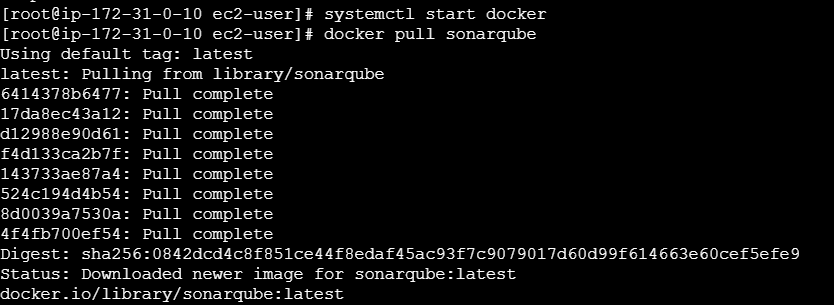
sudo yum install java-17-amazon-corretto-devel

* Installing docker

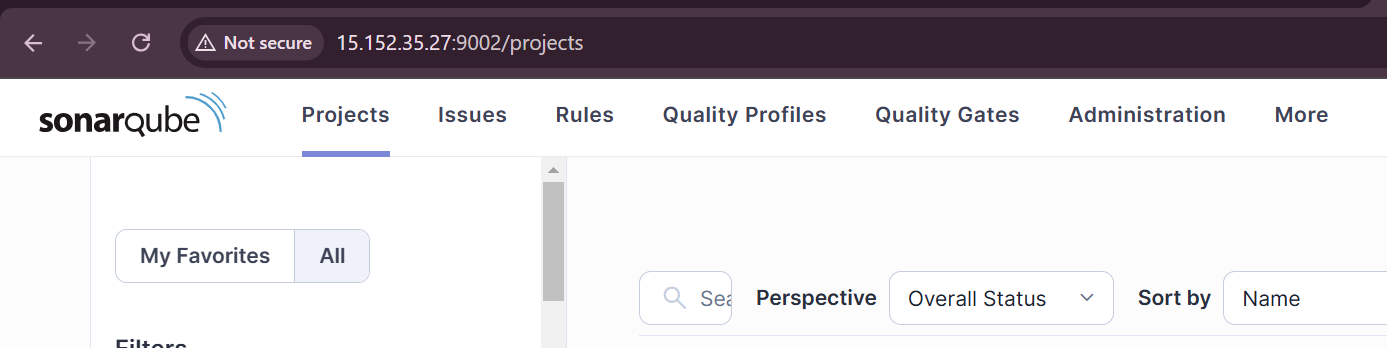
yum install docker –y

* Starting the docker

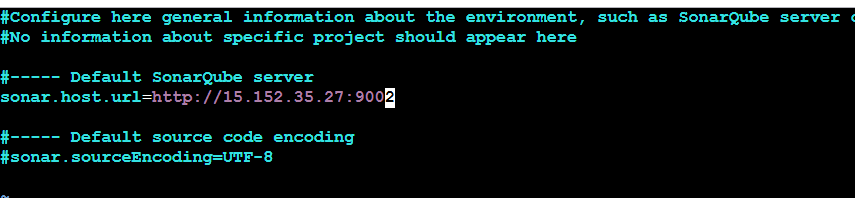
Systemctl start docker

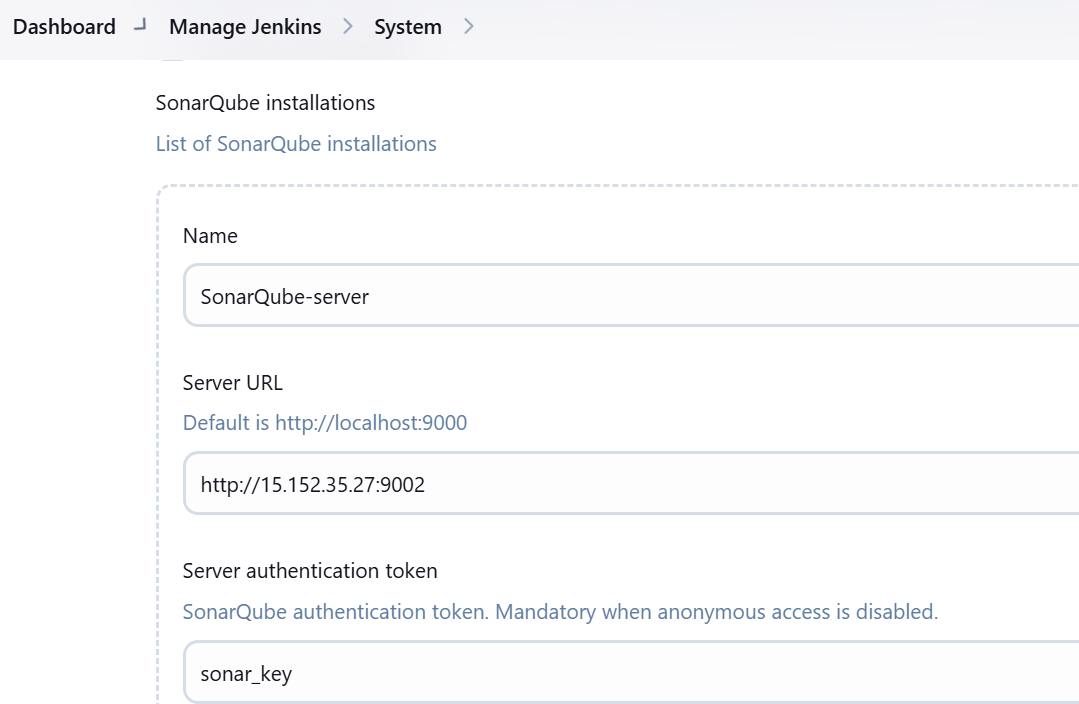
* docker pull sonarqube
* docker images
* docker run –itd –p 9002:9000
* docker run ps



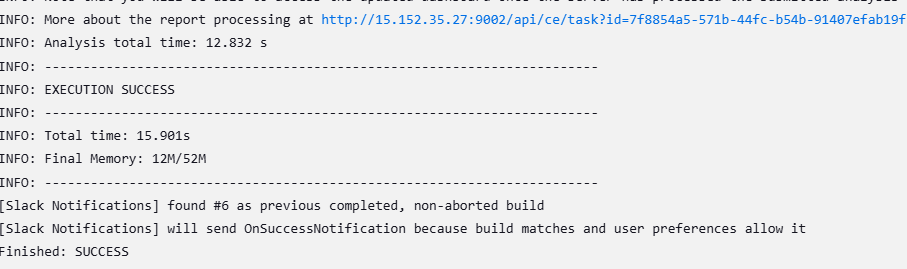
Checking the output

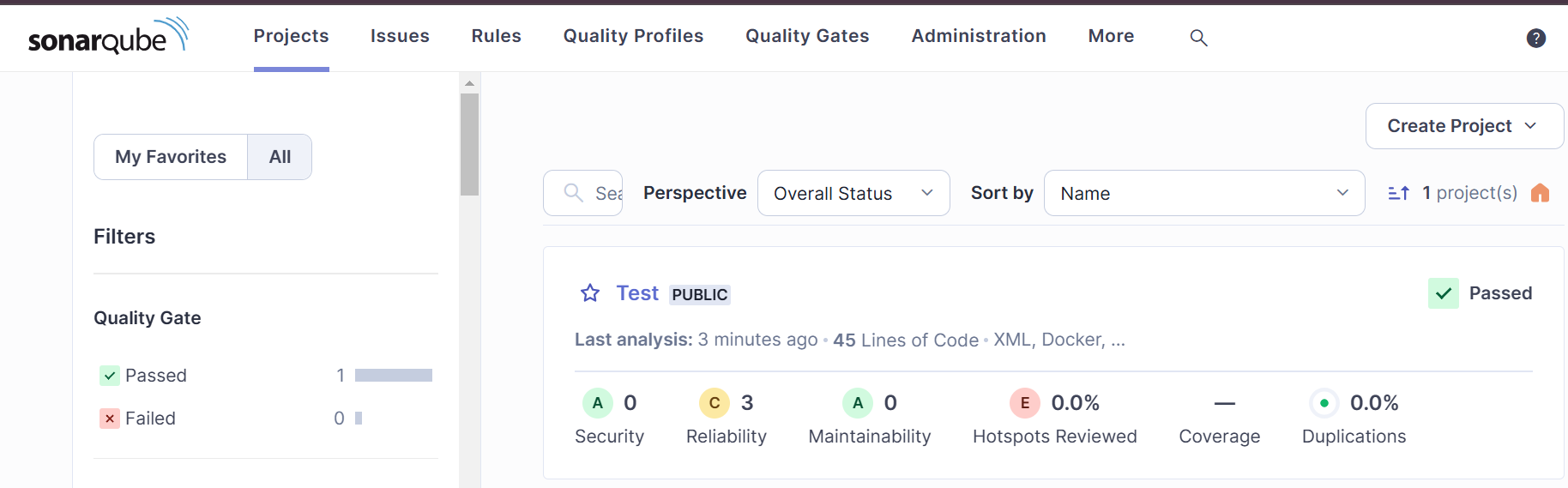
Also Need to edit the sonar properties in the Jenkins



In JENKINS PAGE,we need to add the details and credentials

Execute the job

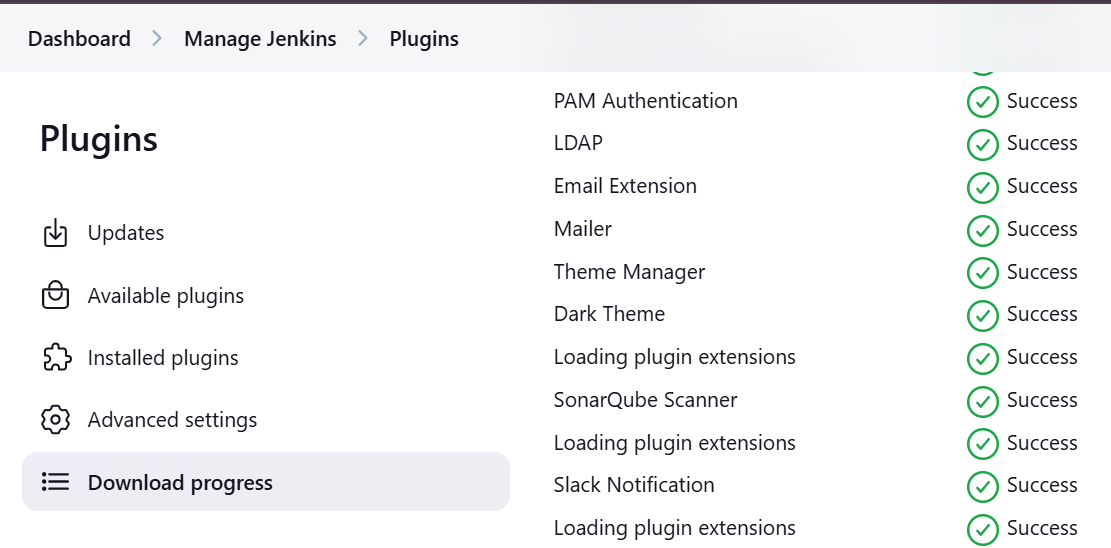


In sonarqube , 

**stage3: Slack Integration to send the alerts to slack.**

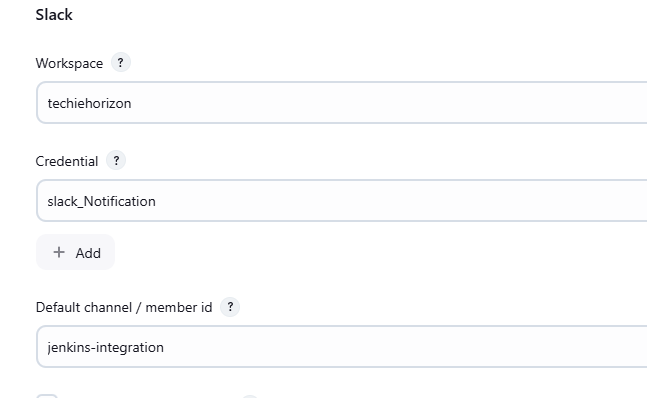
**URL:** [**https://github.com/betawins/hiring-app.git**](https://github.com/betawins/hiring-app.git)

Downloading the plugin for slack notification



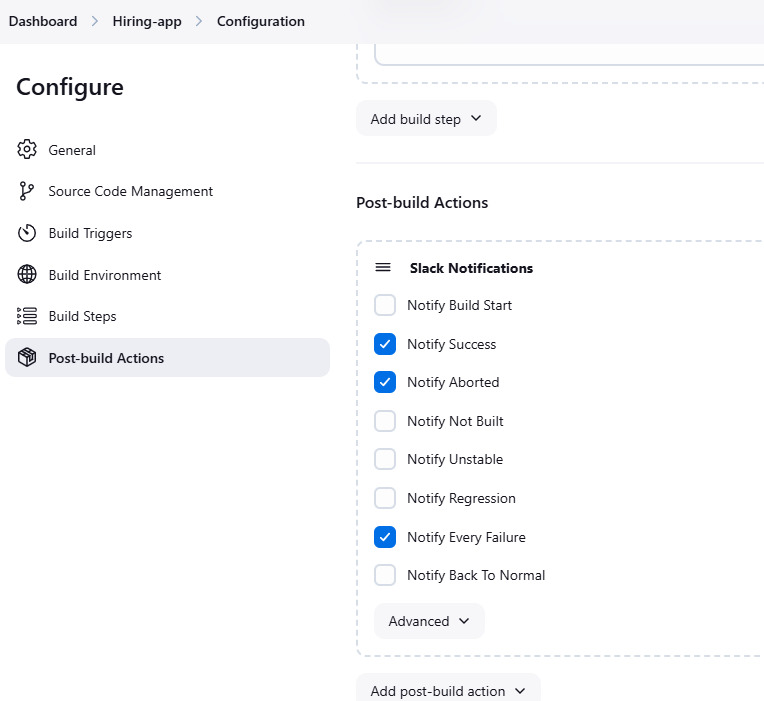
In slack,Go to apps>Jenkins>configure

Now go the manage Jenkins>system>slack



In jenkins page

Go to jobs>Hiring\_app>configure>post-build actions



We can check the output in the slack

