**Jenkins tasks for Day-03**

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

stage1: Git clone to download the source code.

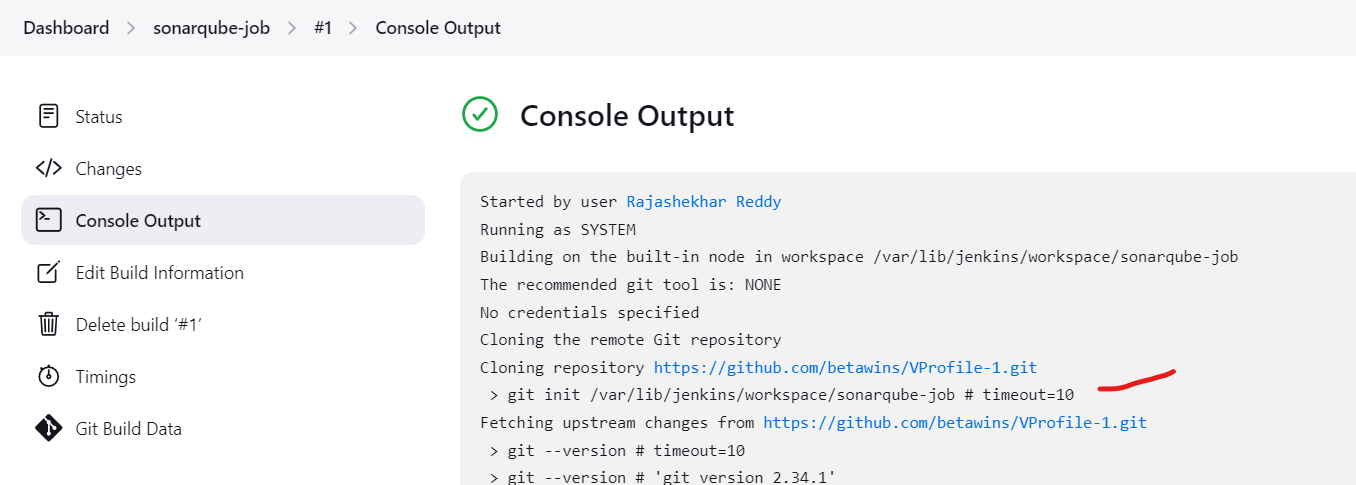
stage2: Sonarqube Integration to check the quality of code

stage3: Slack Integration to send the alerts to slack.

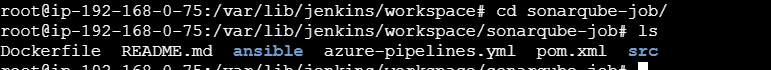
URL: https://github.com/betawins/VProfile-1.git

\* First Access to my Jenkins-master Dashboard then I created new job (sonarqube-job) in added a git URL

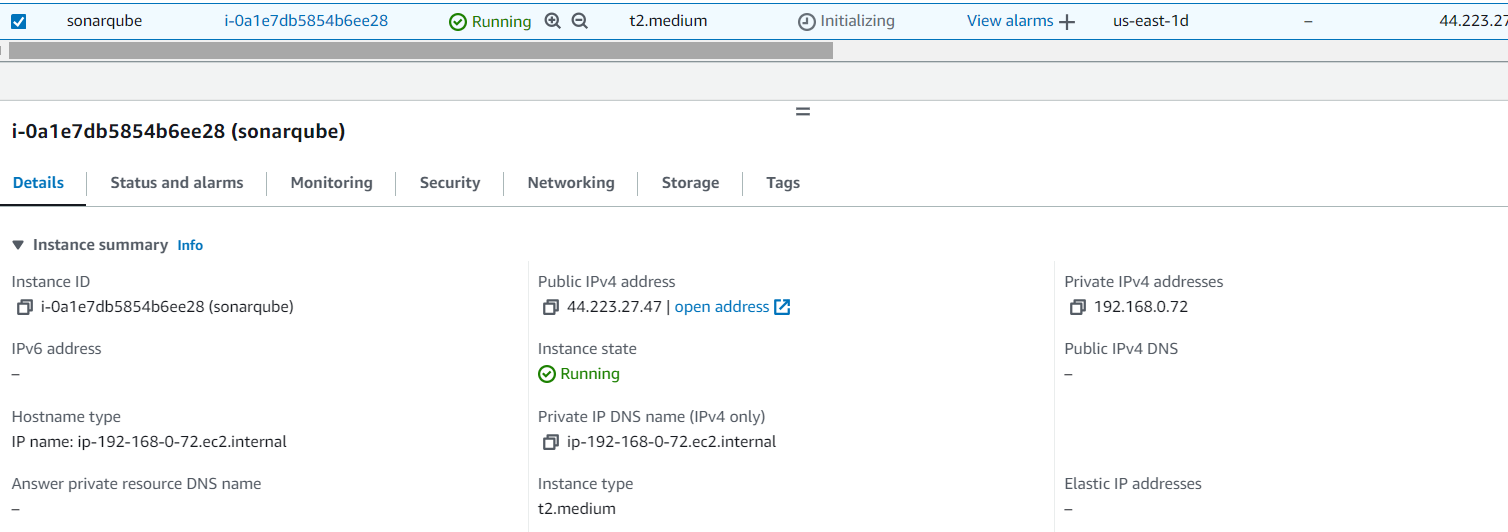
Url: “https://github.com/betawins/VProfile-1.git”



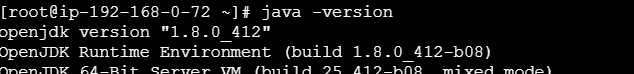
\* Once created the job build it, after build the job workspace will created



\* First created a new instance with t2.medium and with storage of 20 GB



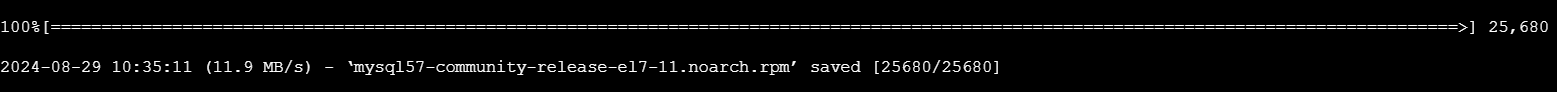
\* install Java using “ yum install java-1.8\*



\* Download mysql package using

sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm

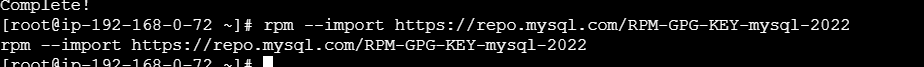
sudo yum localinstall mysql57-community-release-el7-11.noarch.rpm





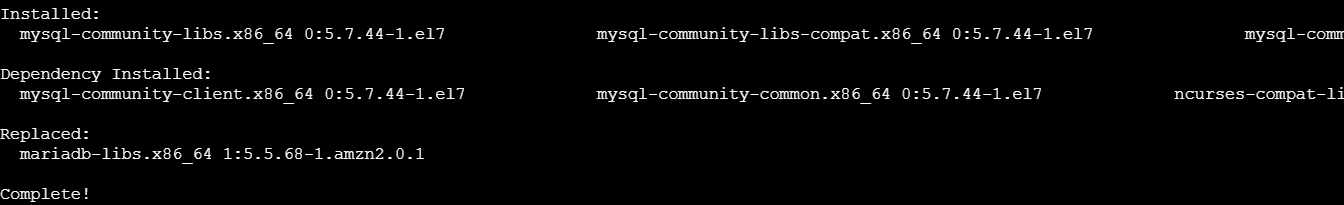
\* Import the key using

rpm --import <https://repo.mysql.com/RPM-GPG-KEY-mysql-2022>



\* Install the mysql community-server

sudo yum install mysql-community-server



\* Then start the mysql using

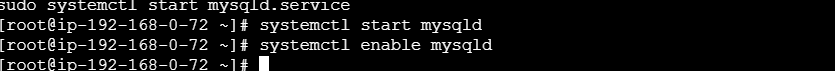
sudo systemctl start mysqld.service



Start MySQL and Enable Start at Boot Time using below commands

systemctl start mysqld

systemctl enable mysqld



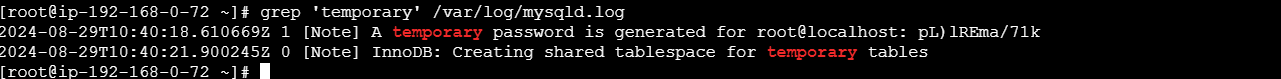
\* I have checked that mysql port is running or not using netstat -na | grep 3306



Configure the MySQL Root Password

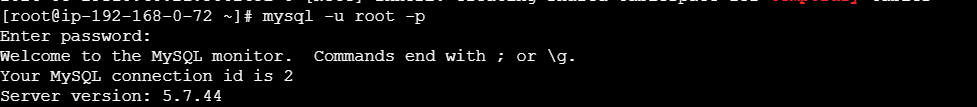
We will see default MySQL root password

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



Login to mysql using the default password

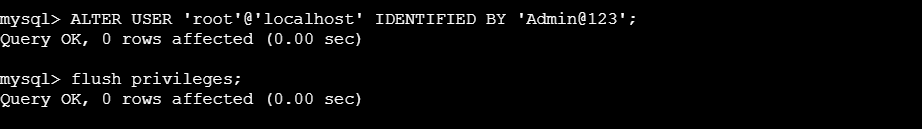
mysql -u root -p



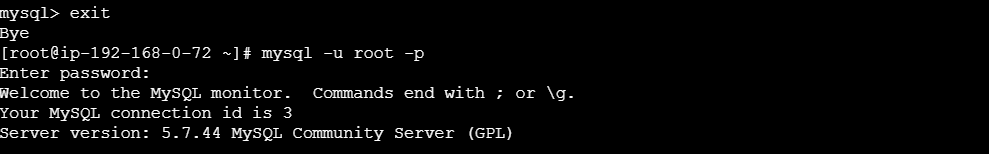
Now replace the default password with a new and strong password

ALTER USER 'root'@'localhost' IDENTIFIED BY 'Admin@123';

flush privileges;



Tested Using new password , alogged in with new password

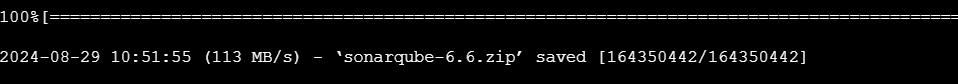


Download & unzip SonarQube 6.0

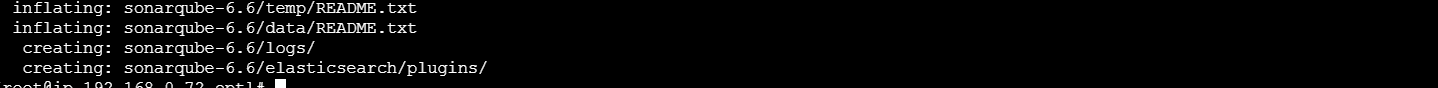
\* change to “**/opt”**

Then Download sonarQube

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

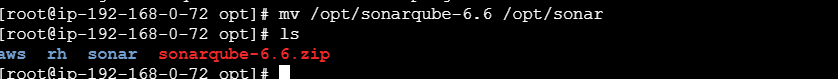


unzip sonarqube-6.6.zip



\* Rename the folder name

mv /opt/sonarqube-6.6 /opt/sonar



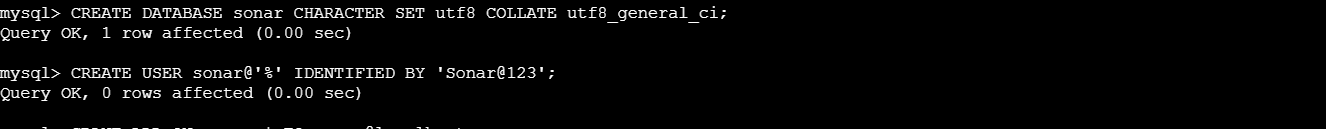
Login to mysql

mysql -u root -p

\* Create a local and a remote user

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

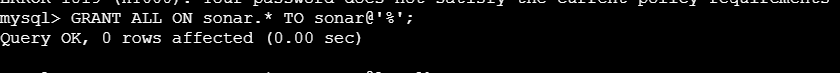
CREATE USER sonar@'%' IDENTIFIED BY 'Sonar@123';



Grant database access permissions to users

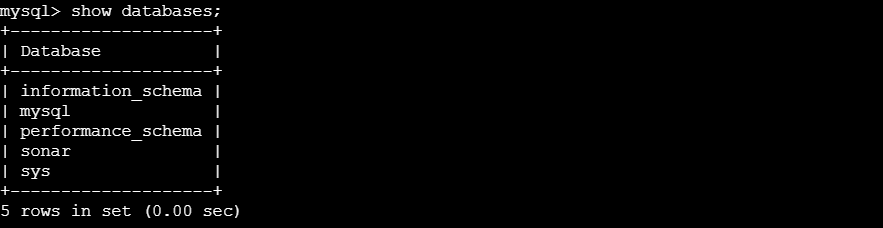
GRANT ALL ON sonar.\* TO sonar@localhost;

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

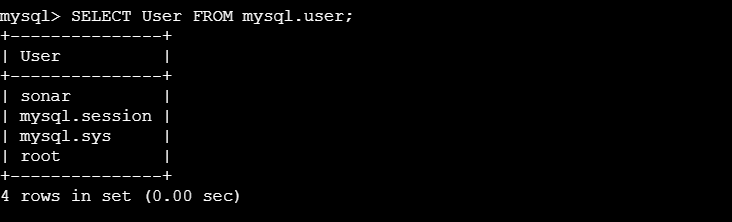


**check users and databases**

show databases;



SELECT User FROM mysql.user;



FLUSH PRIVILEGES;



**#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

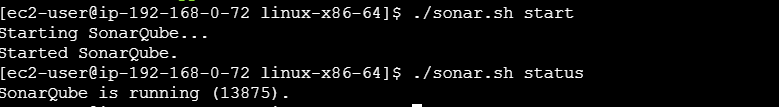
- sonar.web.host=`0.0.0.0`

- sonar.web.context=`/sonar

Start SonarQube service

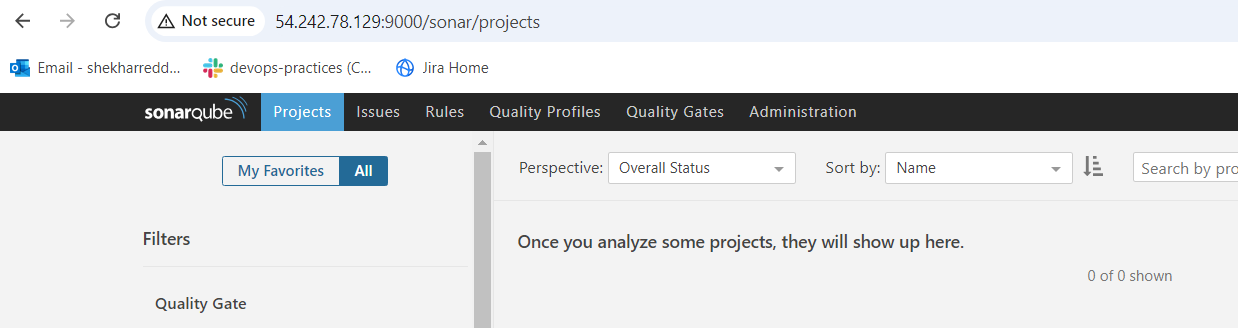
# cd /opt/sonar/bin/linux-x86-64/

# ./sonar.sh start

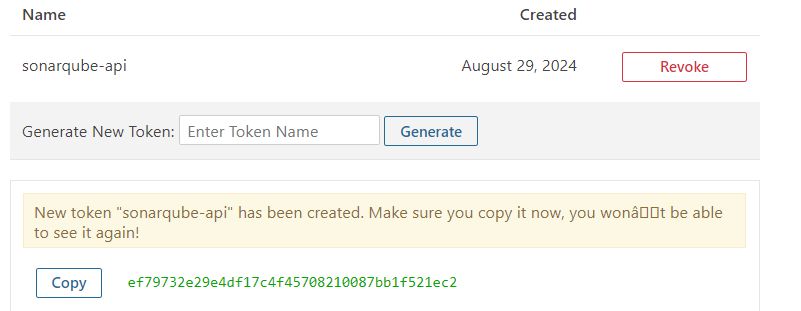


Logged into sonarqube

http://54.242.78.129:9000/sonar



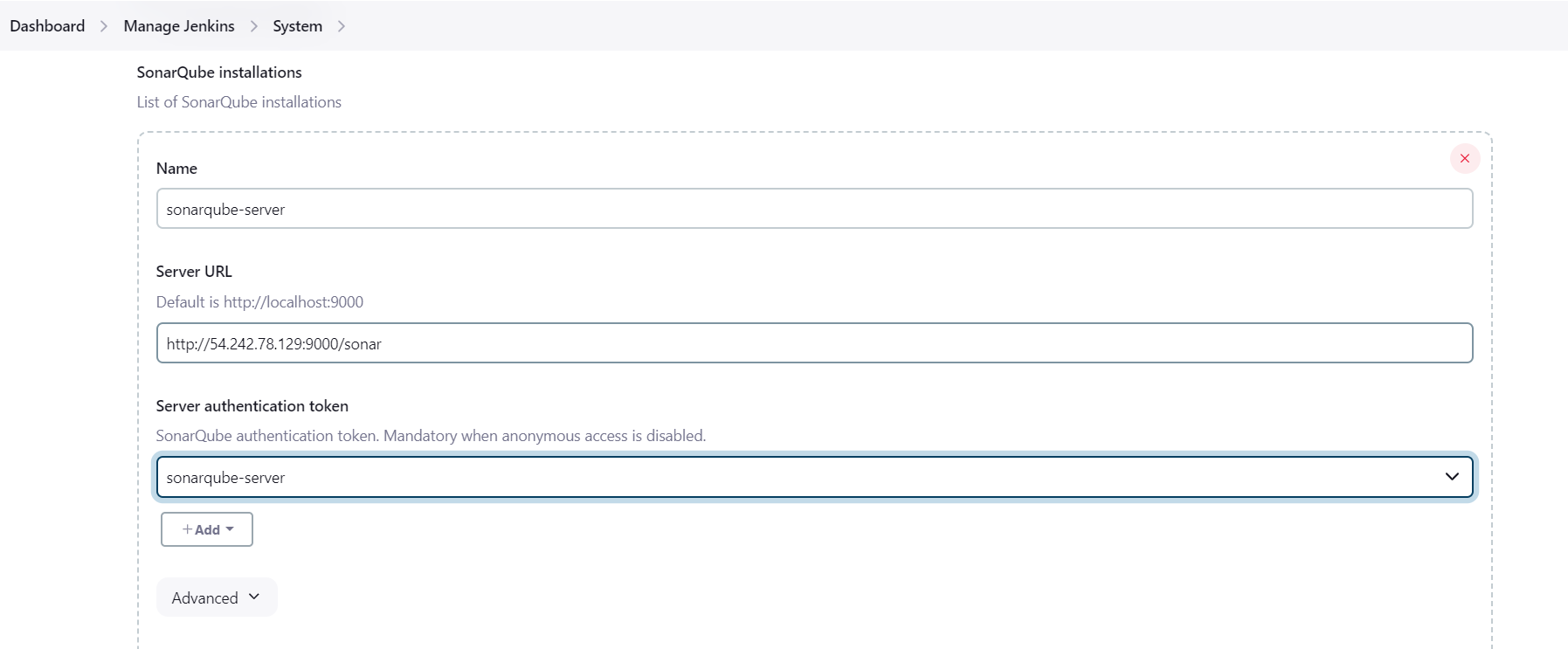
Then create a token in sonarqube



\* To add the Global credentilas of sonarqube

Jenkins [Dashboard](http://54.165.95.20:8081/)→[Manage Jenkins](http://54.165.95.20:8081/manage/)→[Credentials](http://54.165.95.20:8081/manage/credentials/)→[System](http://54.165.95.20:8081/manage/credentials/store/system/)→[Global credentials (unrestricted)](http://54.165.95.20:8081/manage/credentials/store/system/domain/_/)

\* Jenkins [Dashboard](http://54.165.95.20:8081/)→[Manage Jenkins](http://54.165.95.20:8081/manage/)→[Credentials](http://54.165.95.20:8081/manage/credentials/)→[System](http://54.165.95.20:8081/manage/credentials/store/system/)→[Global credentials (unrestricted)](http://54.165.95.20:8081/manage/credentials/store/system/domain/_/)



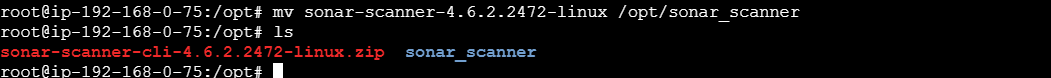
\* Download the sonarqube-scanner package at opt folder in jenkins-server

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



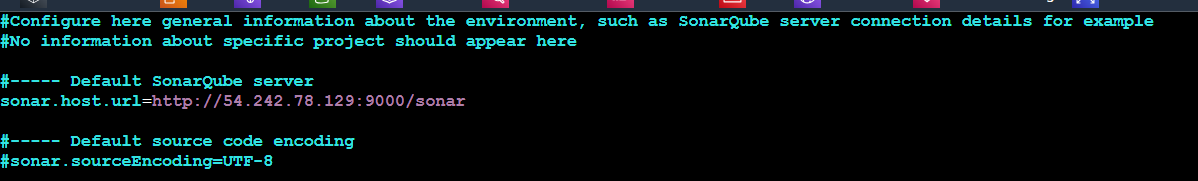
\* Renamed mv sonar-scanner-4.6.2.2472-linux /opt/sonar\_scanner



Set 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/sonar

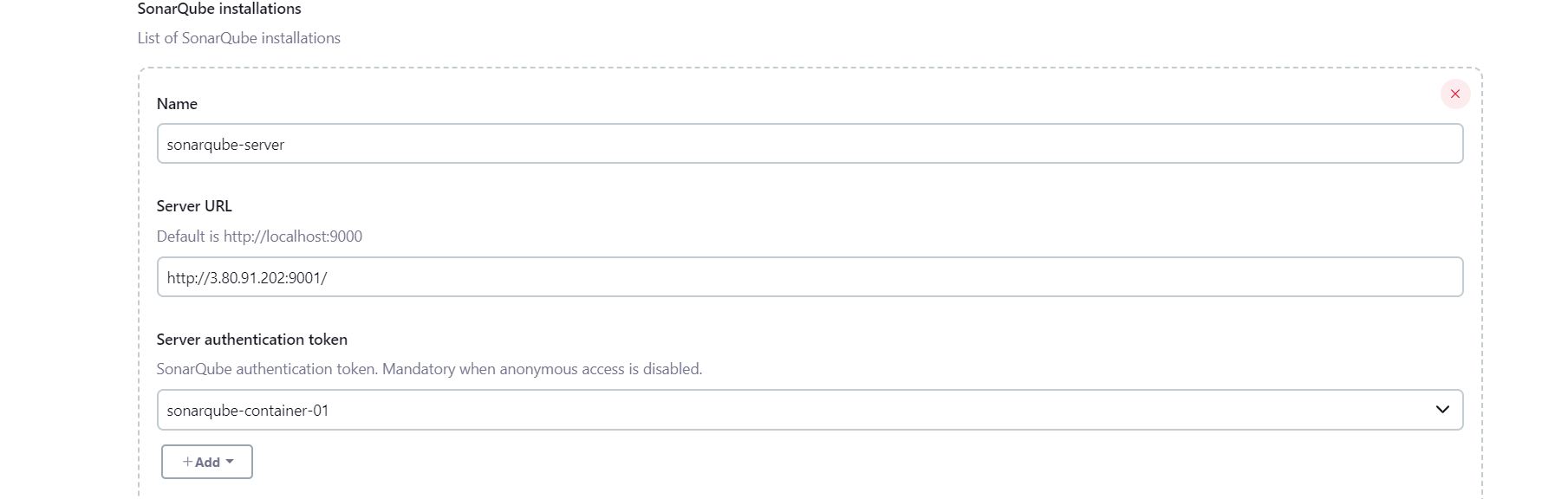


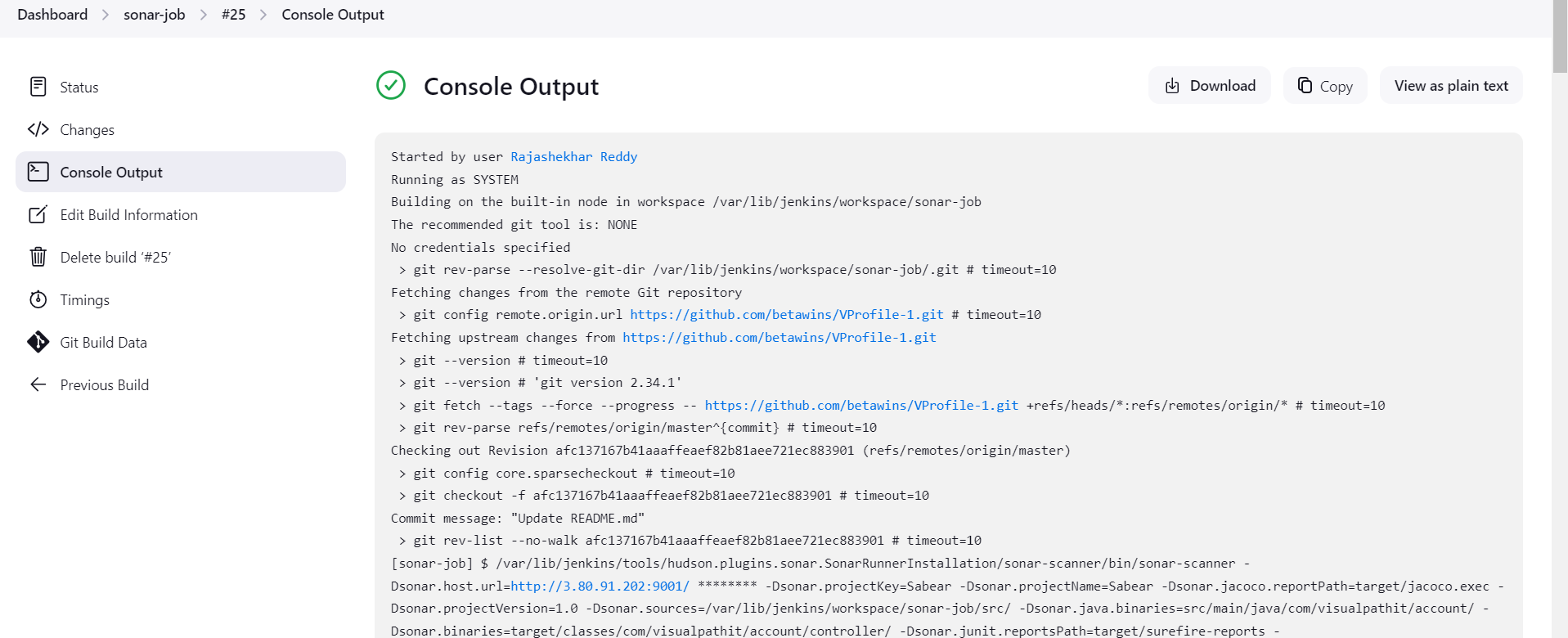
Configuring SonarQube scanner home path

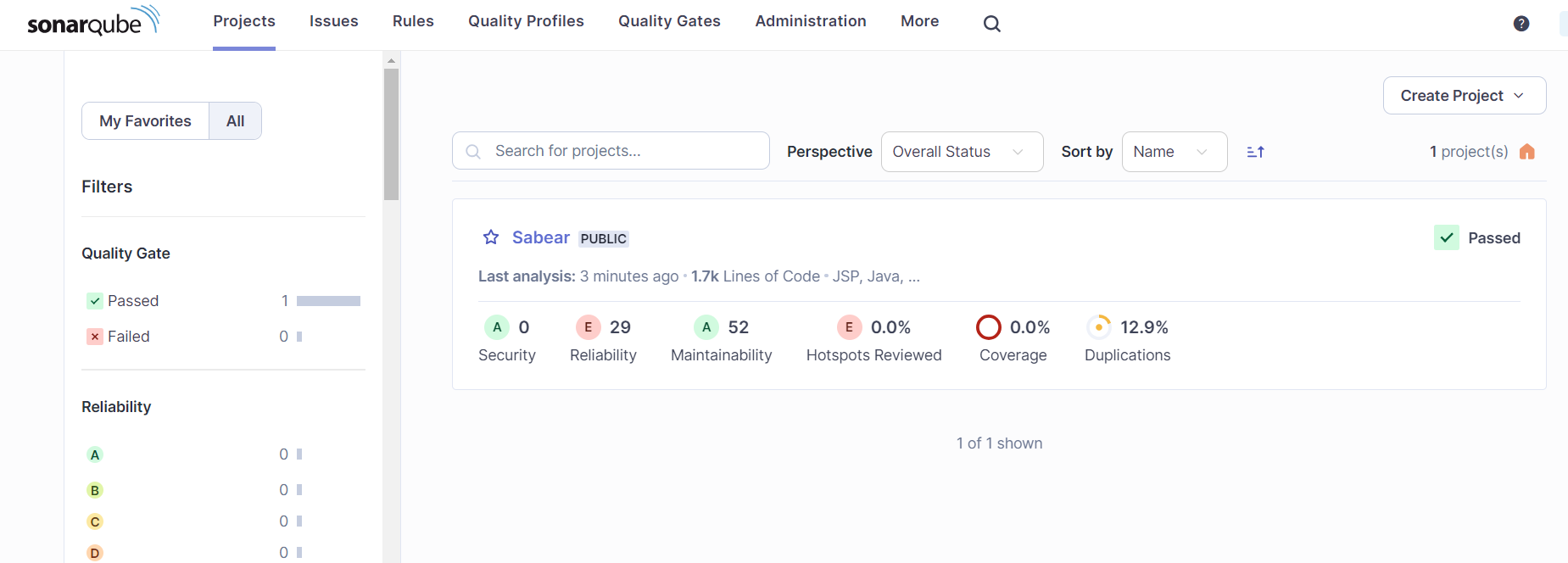
**Manage Jenkins` > `Global Tool Configuration` > `SonarQube Scanner`**

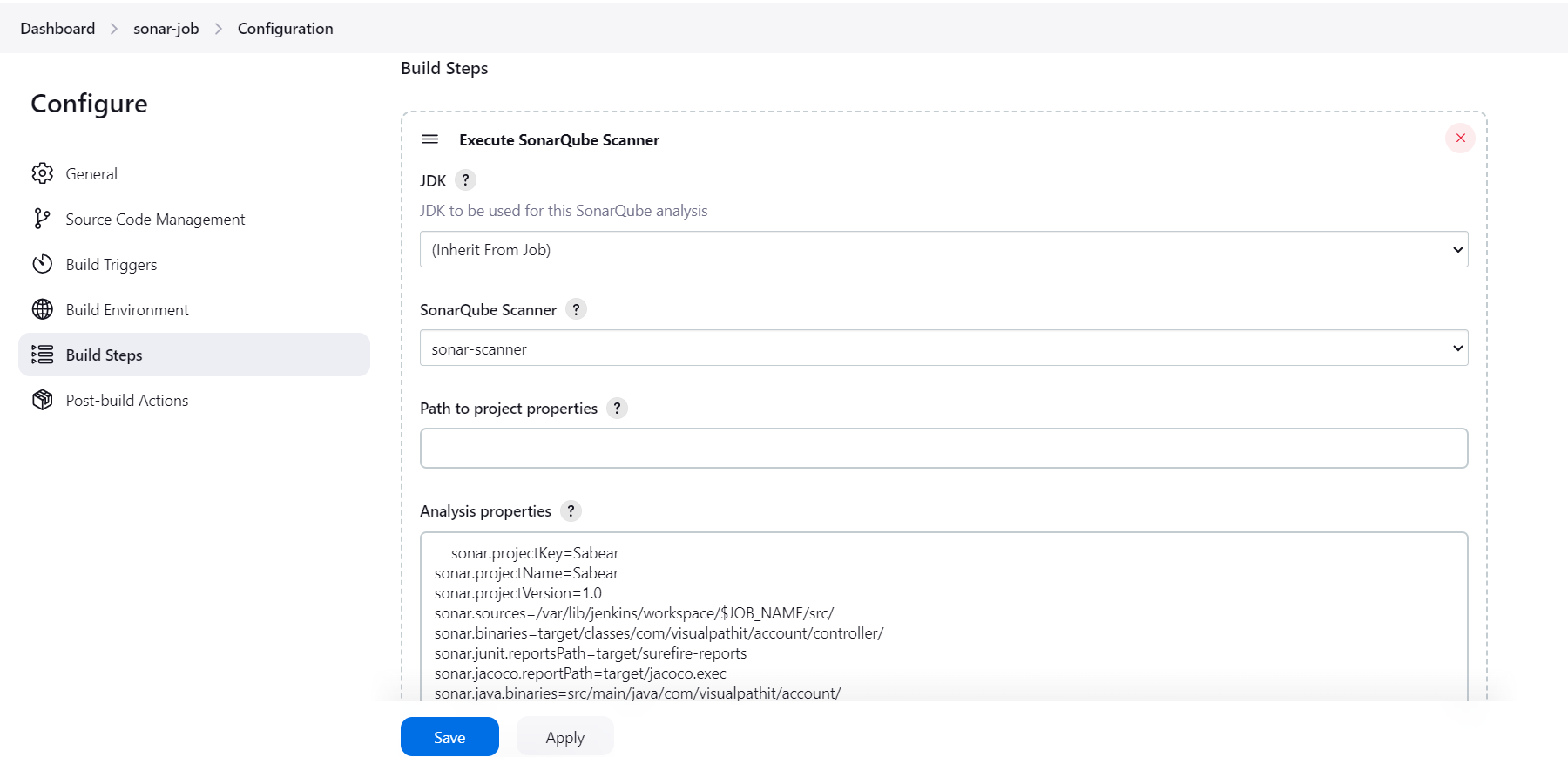
**-** Name  **: `sonar\_scanner`**

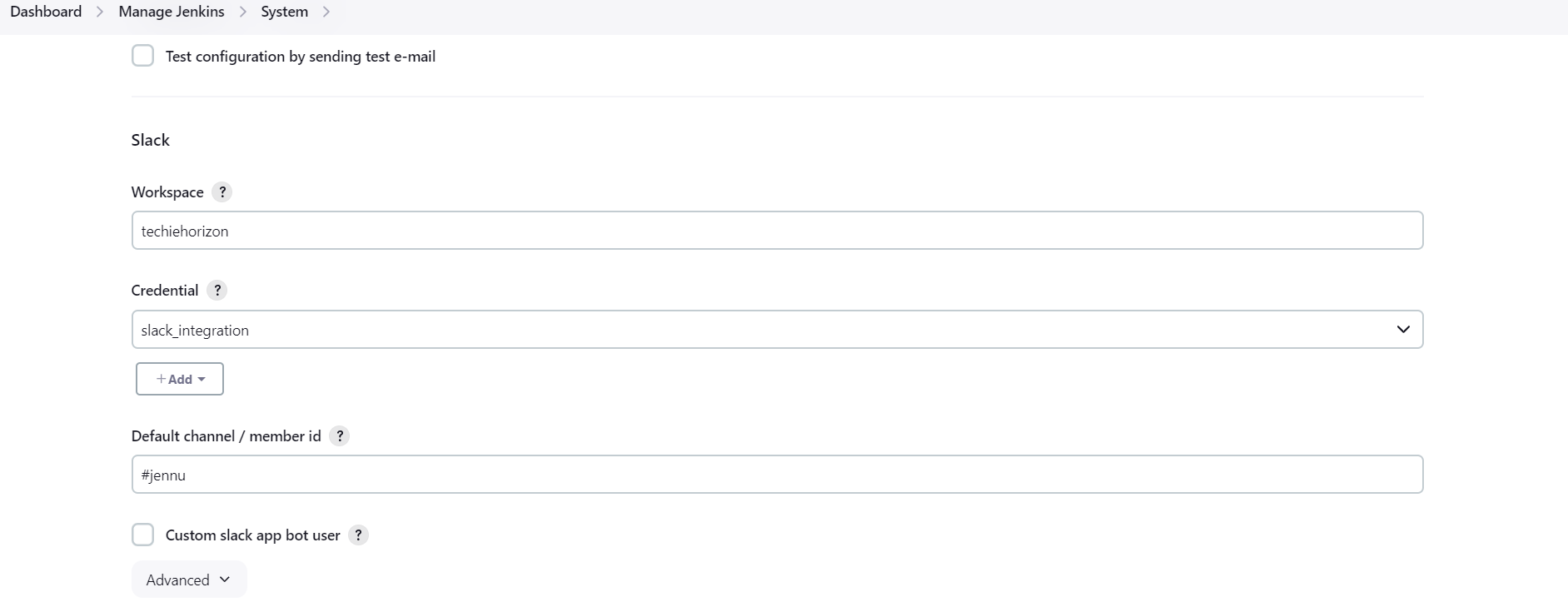
**-** SONAR\_RUNNER\_HOME **: `/opt/sonar\_scanner/bin`**











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

stage1: Git clone to download the source code.

stage2: Sonarqube Integration to check the quality of code

stage3: Slack Integration to send the alerts to slack.

URL: <https://github.com/betawins/hiring-app.git>

