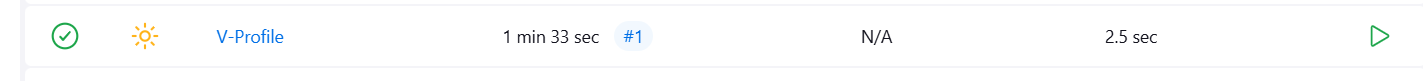
Jenkins task for Day-03.

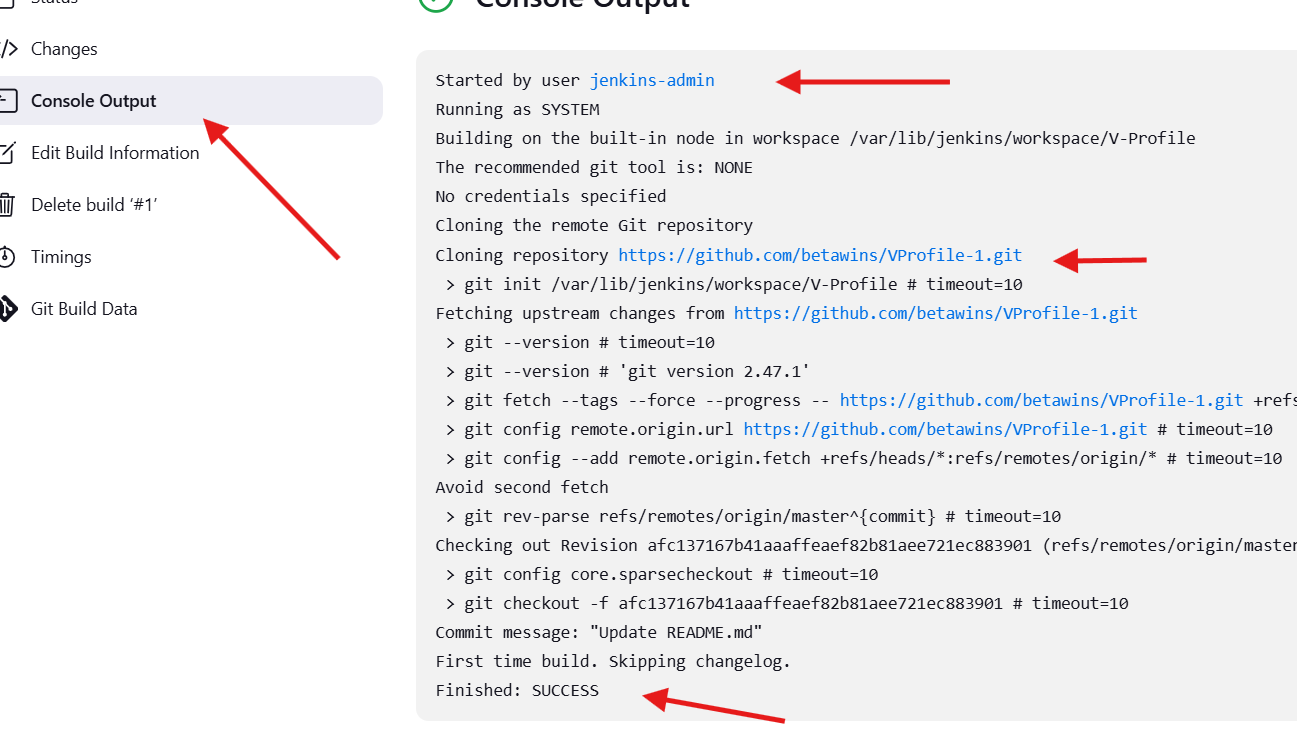
1) Create one jenkins job using the below code and create three stages.

stage1: Git clone to download the source code.

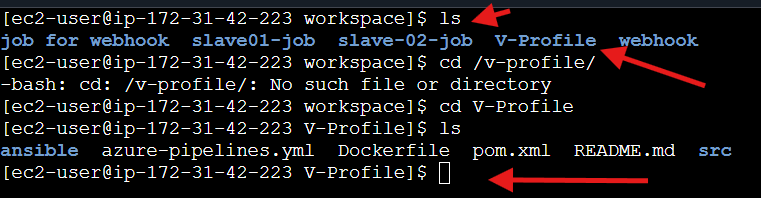
Here what we did , by using the below command ,

We created job name as v-profile and link clone .and save and success and need to check in local Jenkins server as well



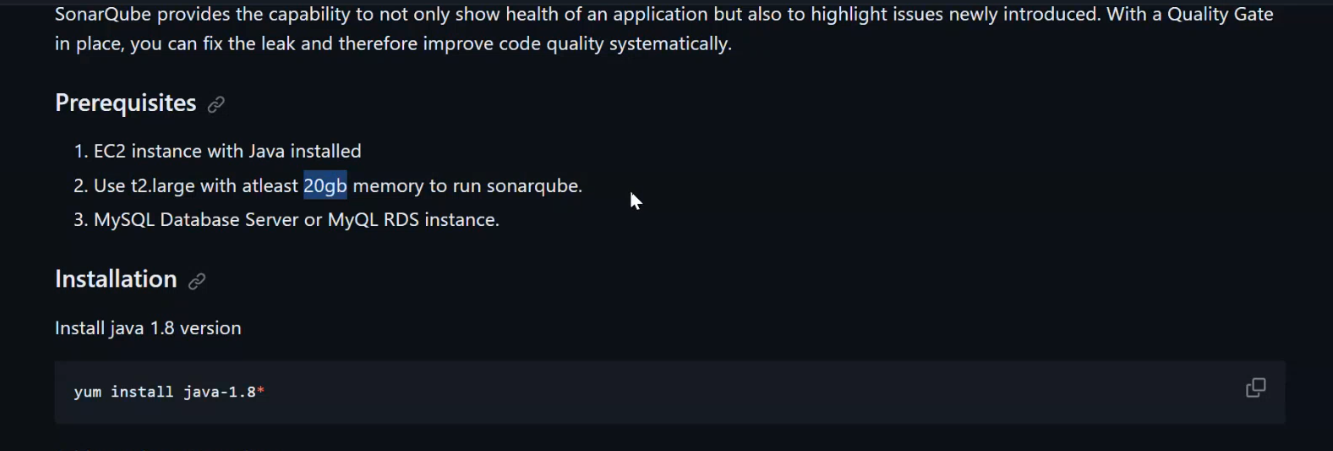


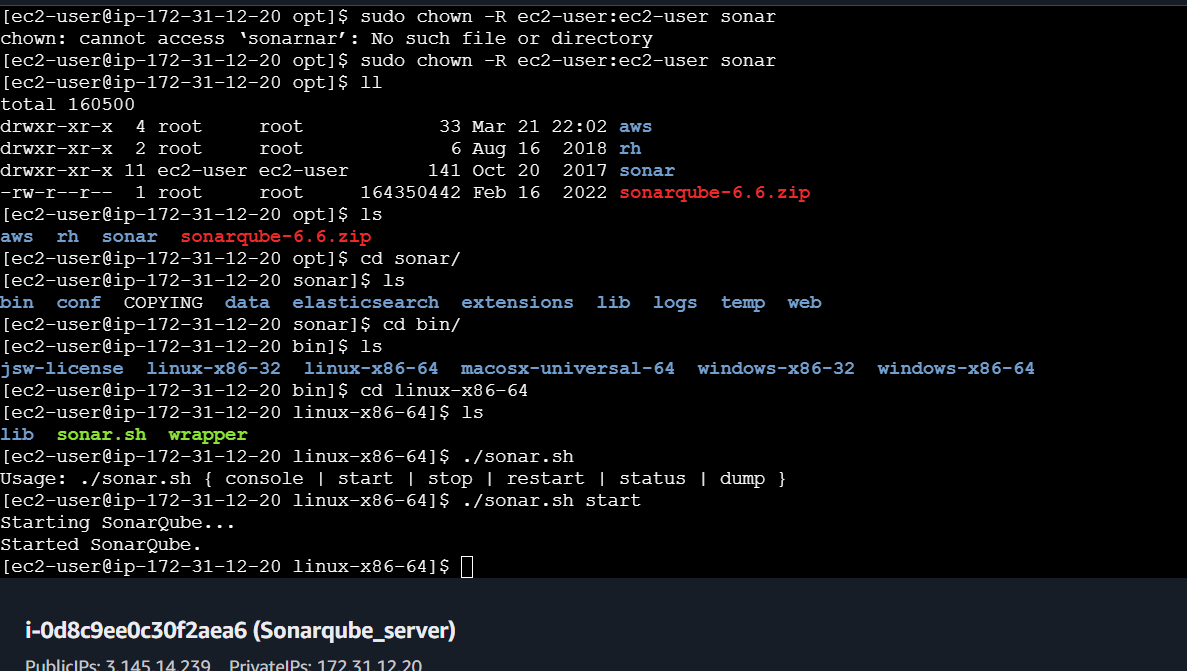
We created job name as v-profile and link clone .and save and success and need to check in local Jenkins server as well

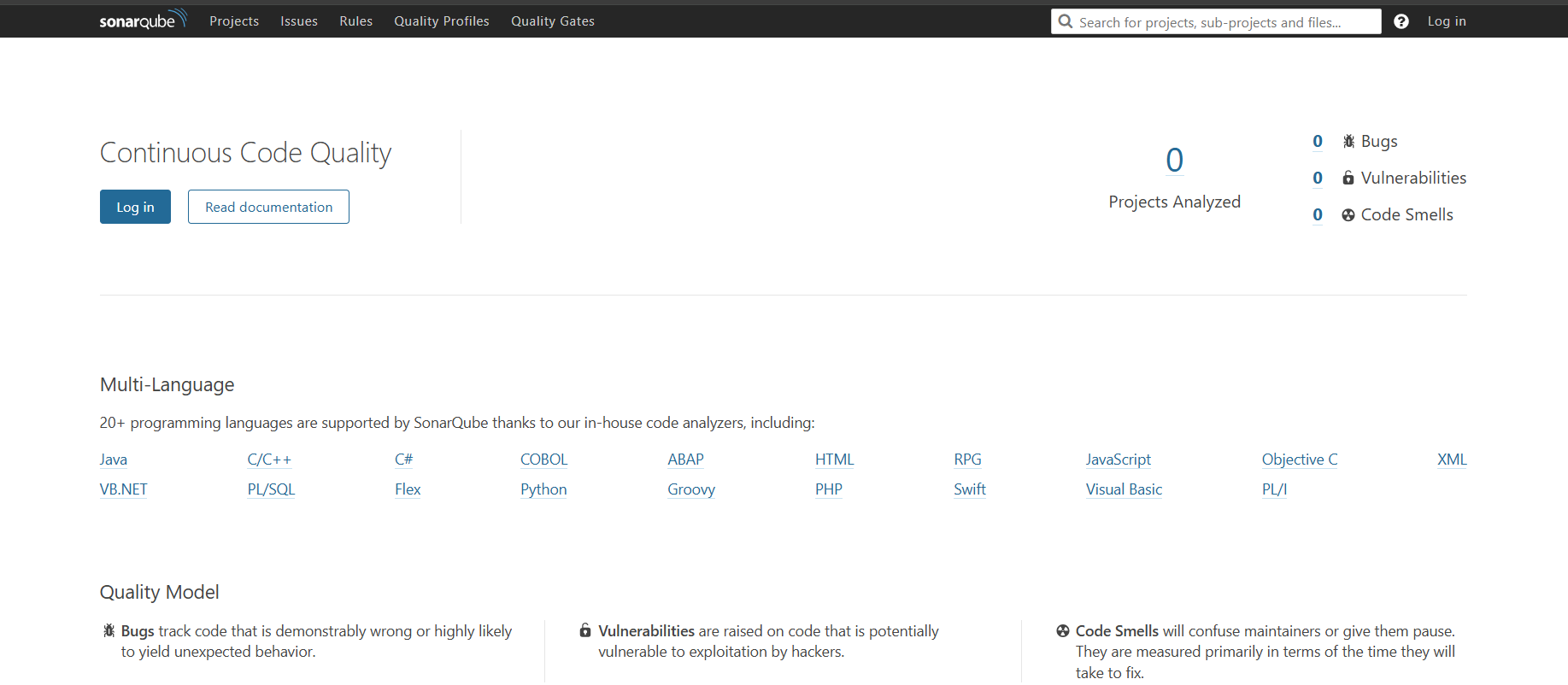


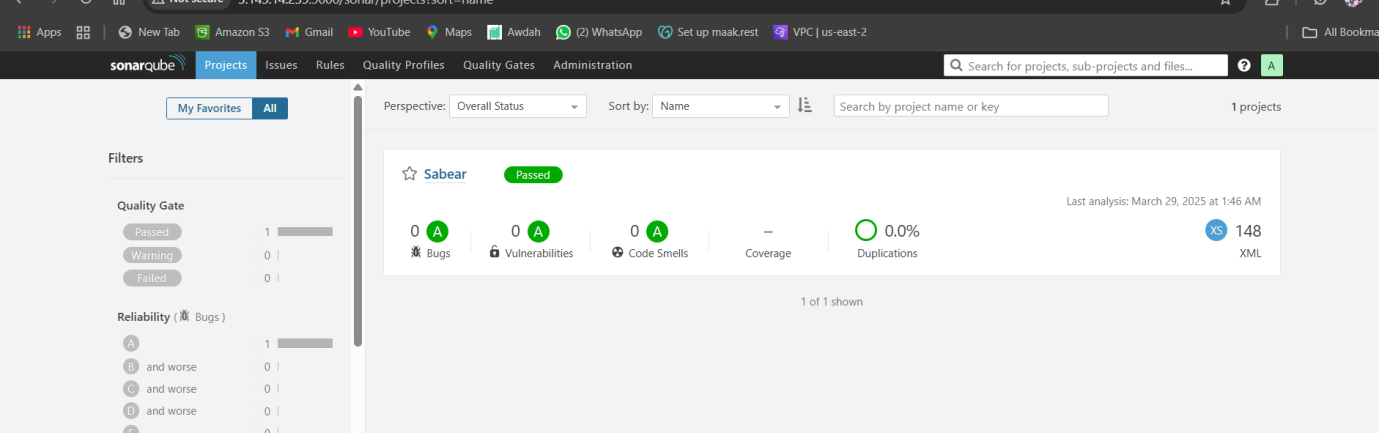
stage2: Sonarqube Integration to check the quality of code

note: sonarqube run on port 9000









stage3: Slack Integration to send the alerts to slack.

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

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

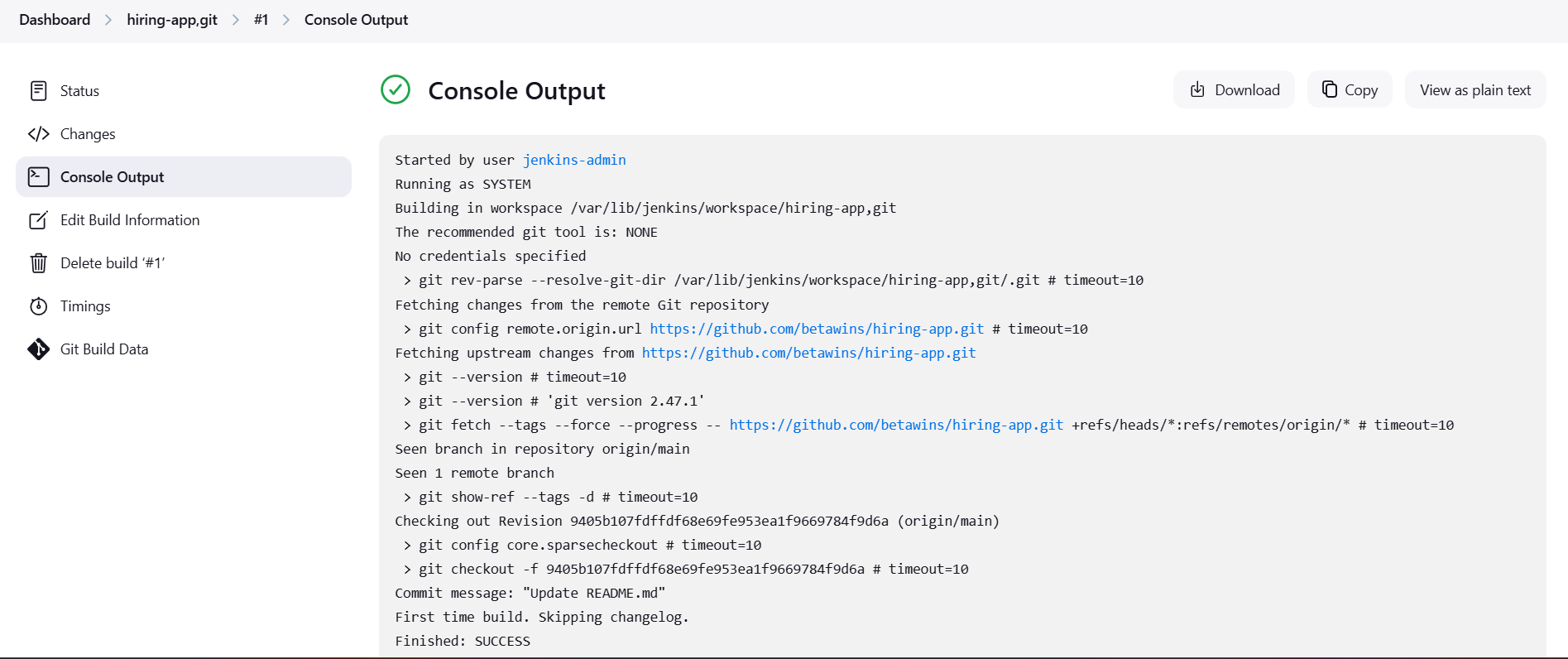
stage1: Git clone to download the source code.

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>

Git clone done



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

**Steps to Integrate SonarQube:**

### **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.

### **Installation**

yum install java-1.8\*

yum update

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

sudo yum install mysql-community-server

sudo systemctl start mysqld.service

sudo systemctl status mysqld.service

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

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;

**Test Using new password**

mysql -u root -p

**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

**Login to mysql**

mysql -u root -p

**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;

Exit

### **ON EC2 Instance (root user)**

ls

cd sonar/

ls

cd conf/

ls

vi sonar.properties

**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`

exit (switch to ec2 user)

cd /opt/

sudo chown -R ec2-user:ec2-user sonar

ll

cd sonar/bin/linux-x86-64/

ls

bash sonar.sh start

bash sonar.sh status

**SonarQube application uses port 9000. access SonarQube from browser**

[http://52.91.180.136:9000/sonar](http://52.91.180.136:9000/sonar/projects)

Default credentials for sonarqube.

Username : admin

Password: admin

Now install sonar qube scanner

Go to root user

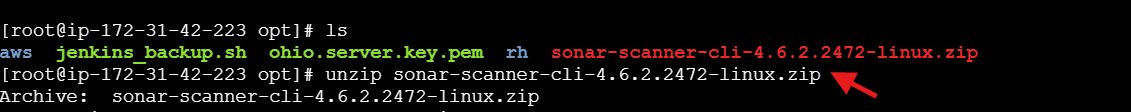
Cd /opt /

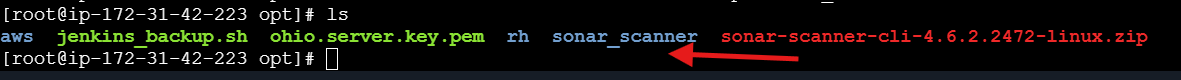
wget <https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.6.2.2472-linux.zip>

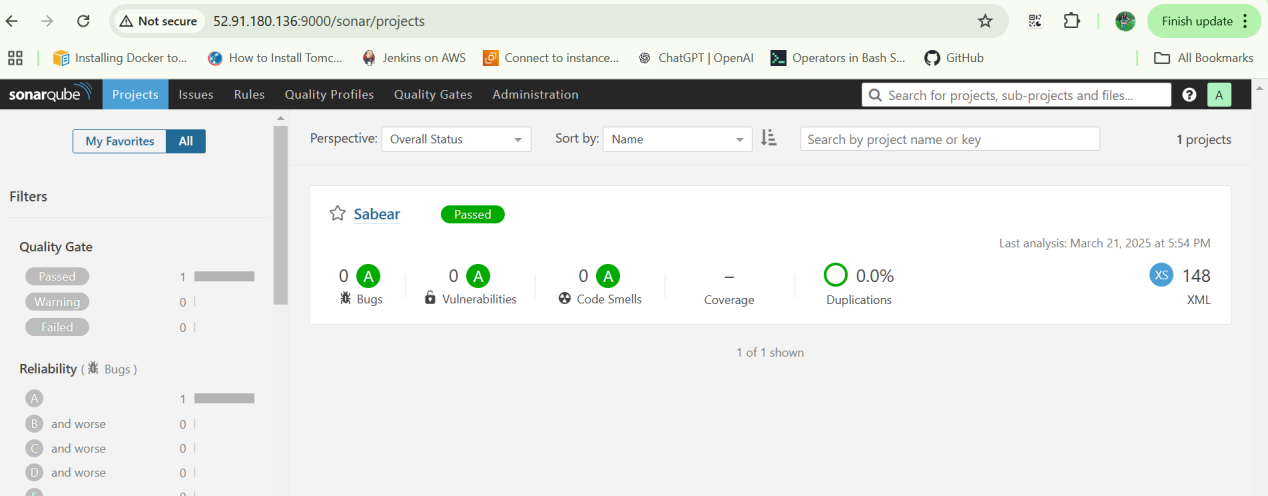
ls

use unzip sonar scanner

rename now [root@ip-172-31-42-223 opt]# mv sonar-scanner-4.6.2.2472-linux /opt/sonar\_scanner

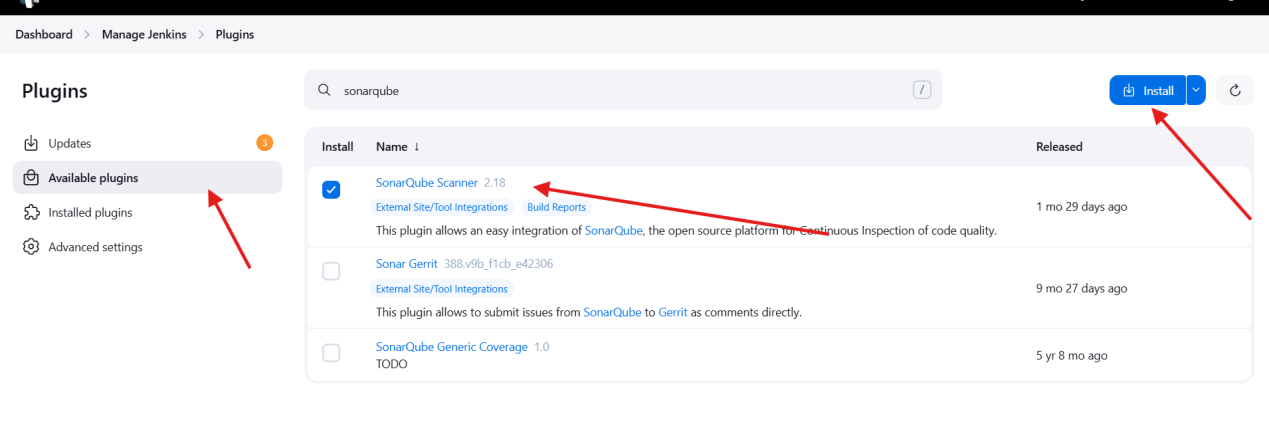






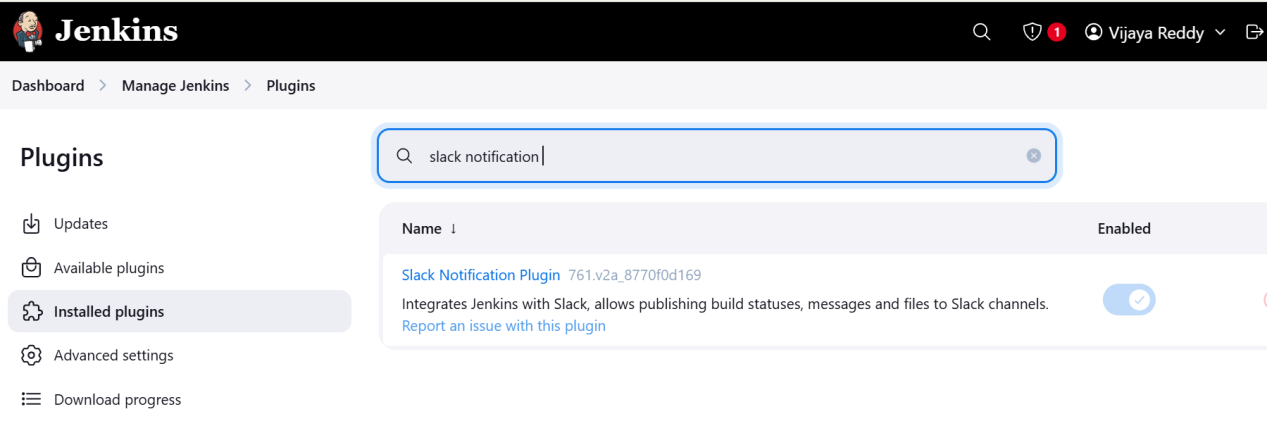
Now go Jenkins browser ,>> manage jeinkins>>plugins

Type in search bar (sonarqube)

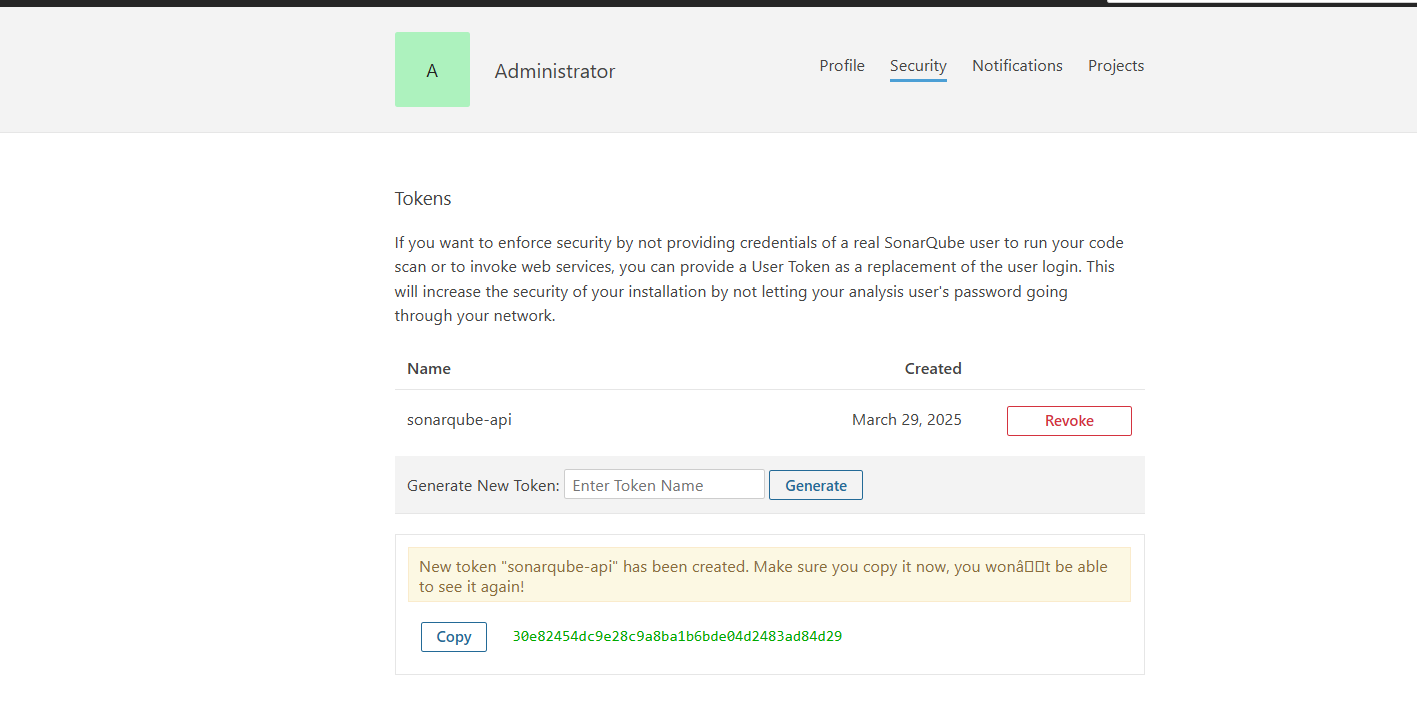


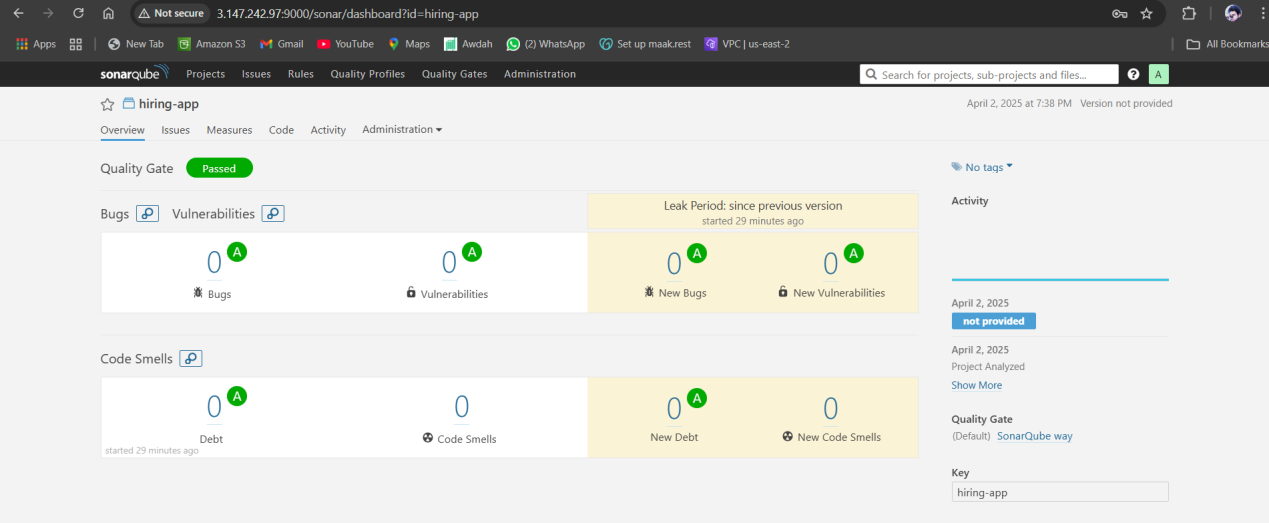
**stage3: Slack Integration to send the alerts to slack.**

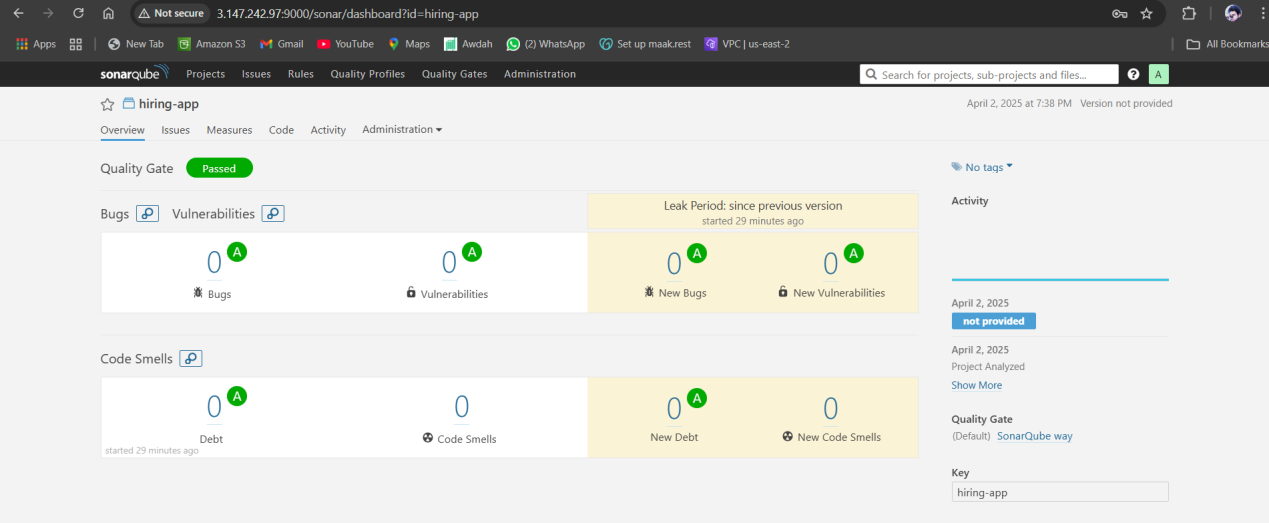
Install Slack Notification plugin in Jenkins Page under Manage Jenkins



Generat one token , ( sonar-qube.api







Go to slack, click on 3 dots (…) and select Automations option