

**SonarQube Integration with Jenkins Master**

Create a new instance with t2.medium for SonarQube server. Install java 11

T2.medium is chargeable

amazon-linux-extras install java-openjdk11 -y



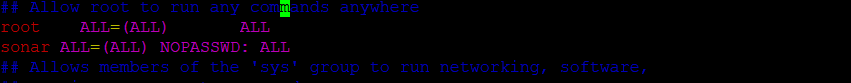
#Add the user sonar and Give sudo accecss to sonar

useradd sonar

visudo

sonar ALL=(ALL) NOPASSWD: ALL





#Download sonarqube

cd /opt

yum install wget unzip -y

wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.8.0.63668.zip

ls -ltr

unzip sonarqube-9.8.0.63668.zip

#Change the owner and group

chown -R sonar:sonar /opt/sonarqube-9.8.0.63668

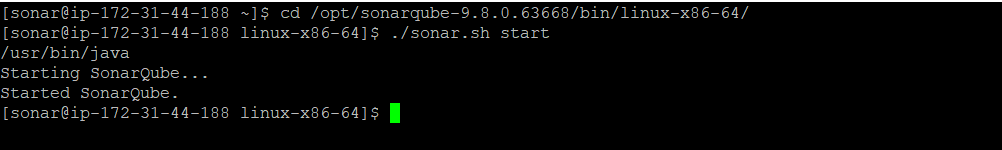
sudo su - sonar



#Start the service sonar

cd /opt/sonarqube-9.8.0.63668/bin/linux-x86-64/

./sonar.sh start

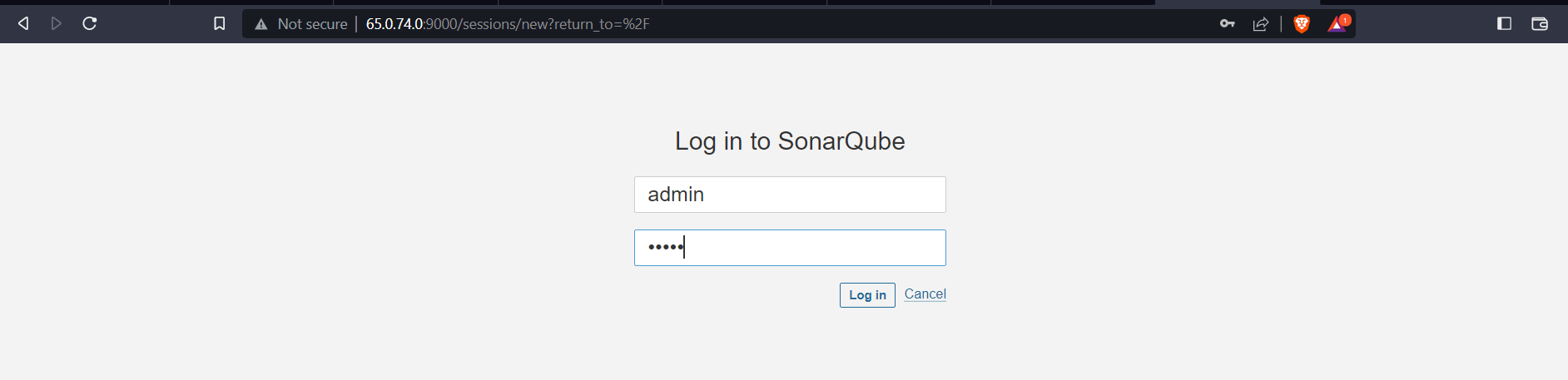


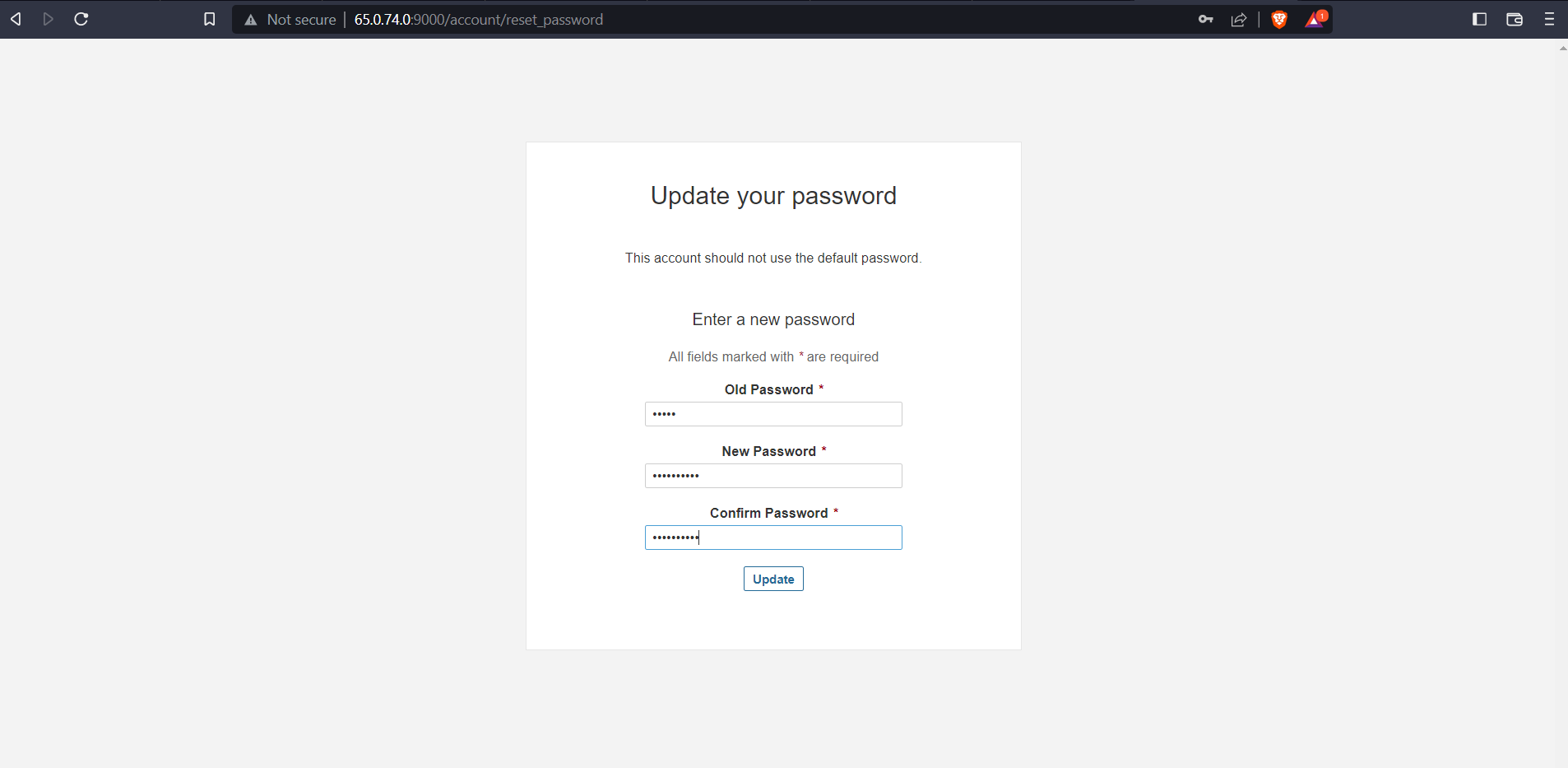
#login to sonarqube in browser

publicip:9000

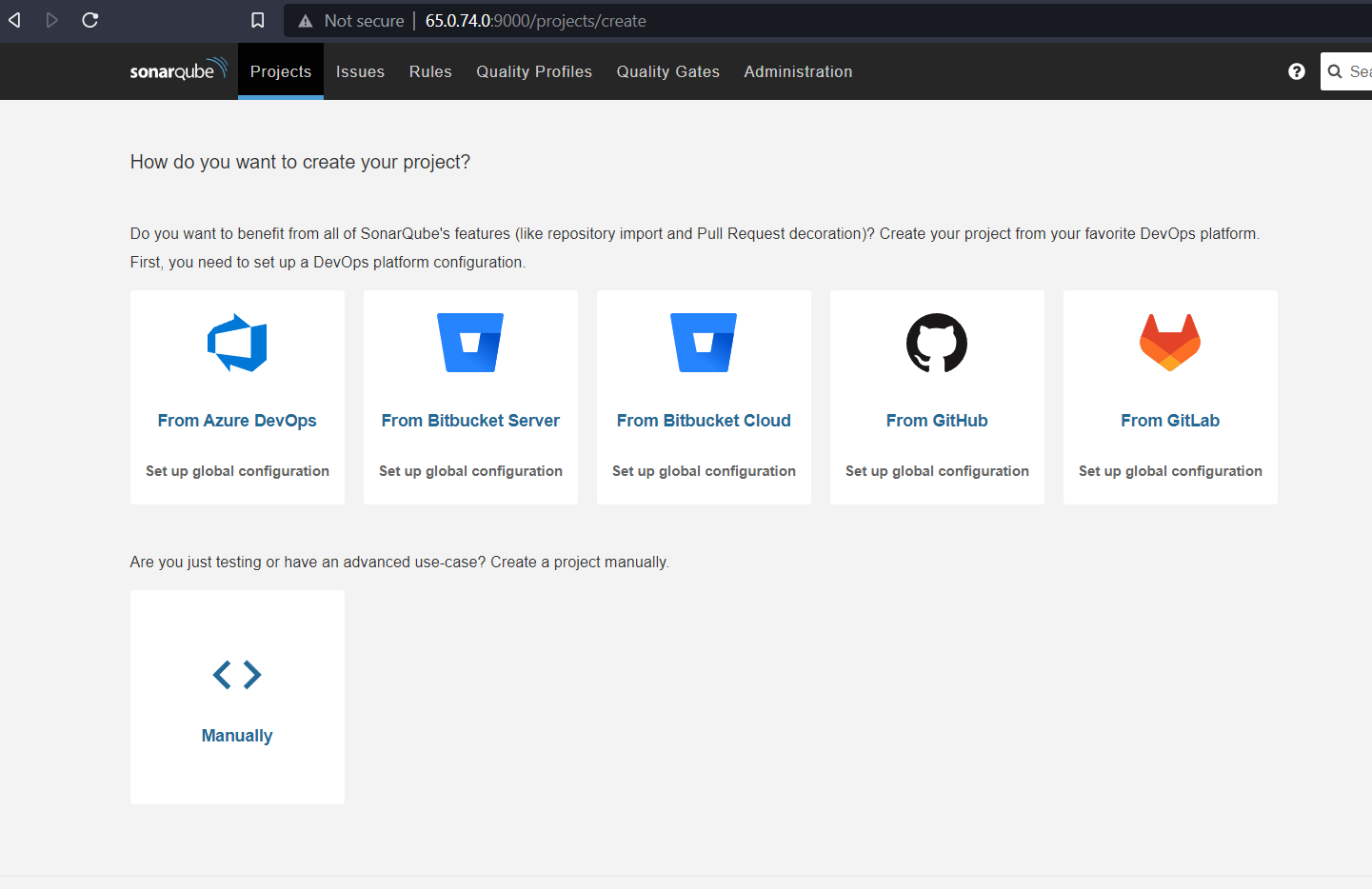
u:admin

p:admin

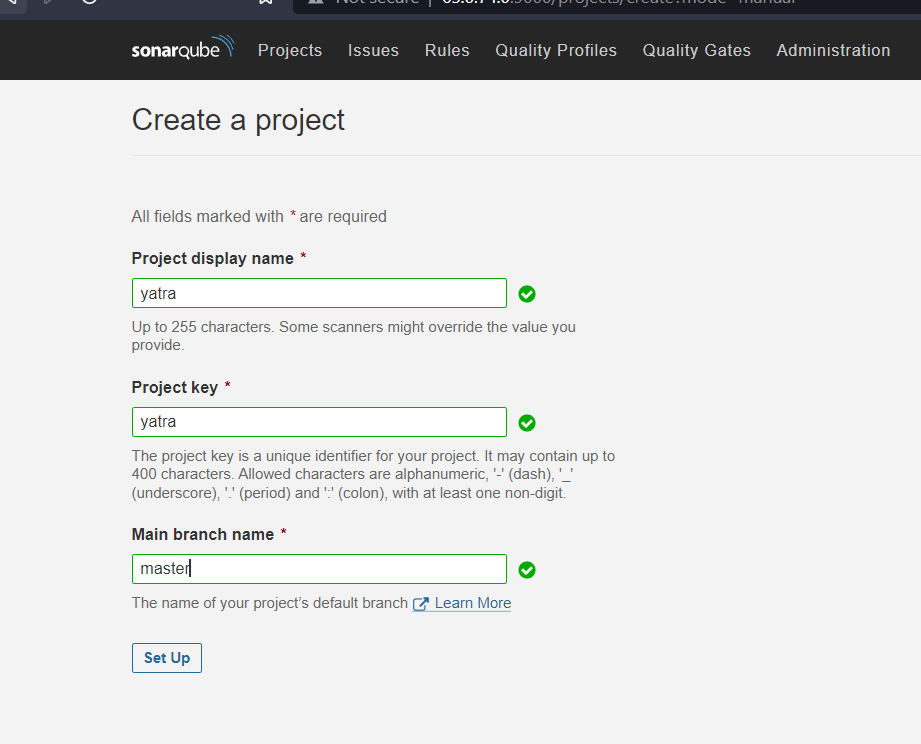




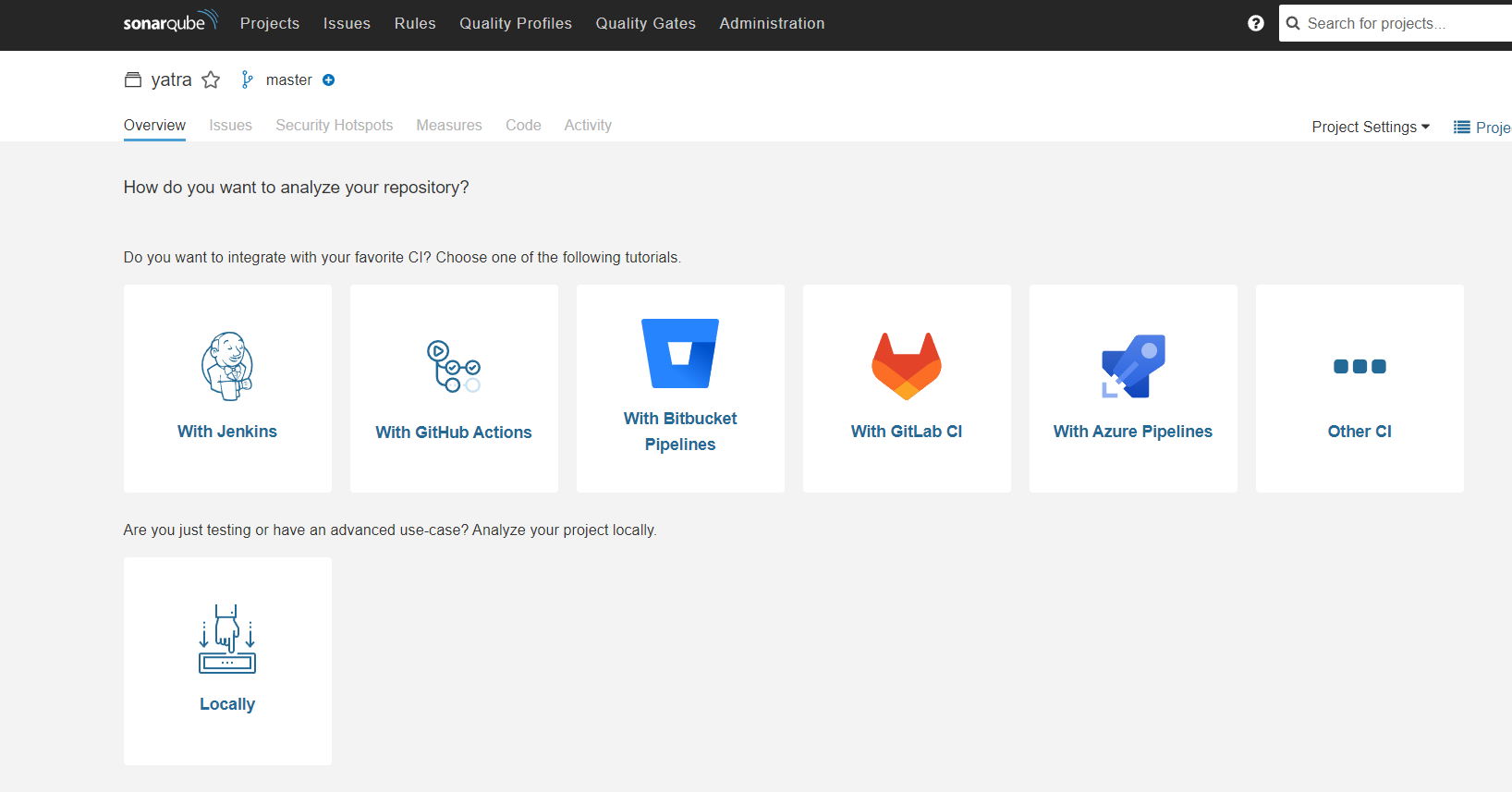
Click on projects and click on manually



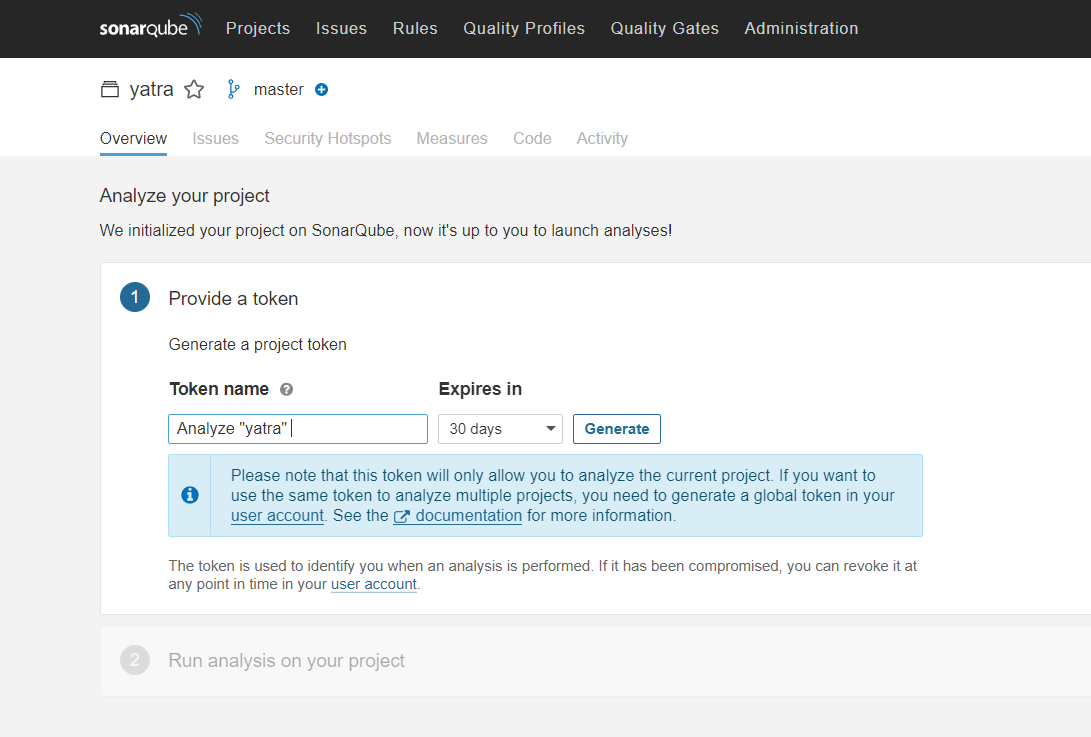
Type the details of your project and click on setup



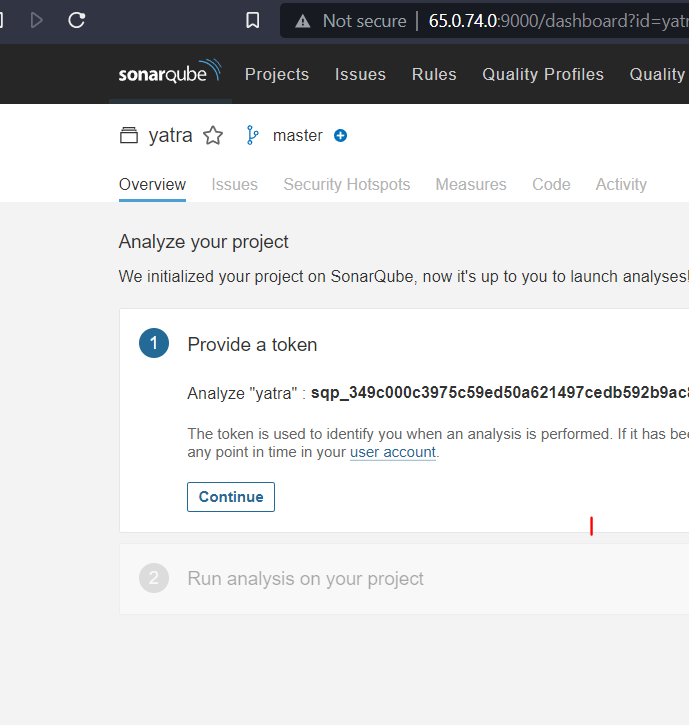
Click on locally

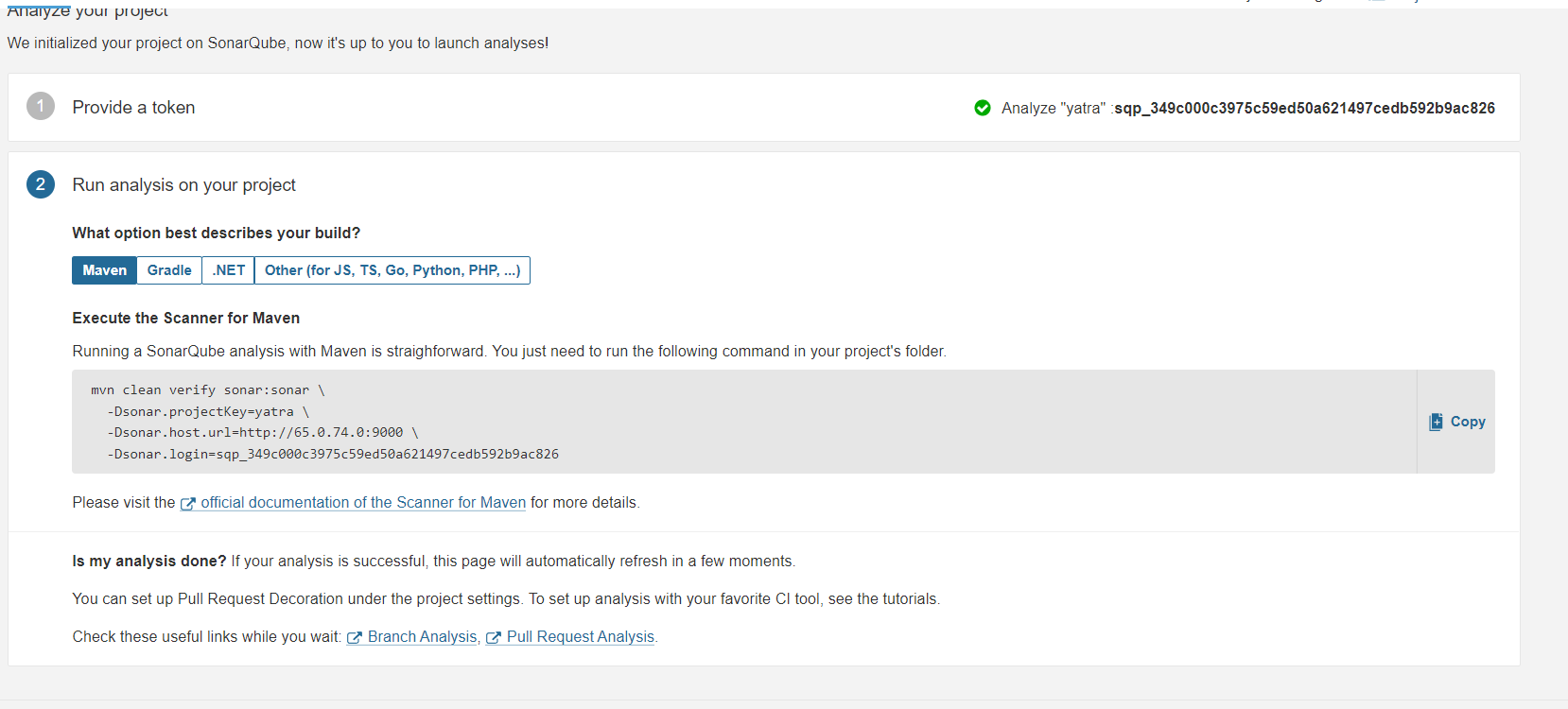


Click on generate



Click on continue



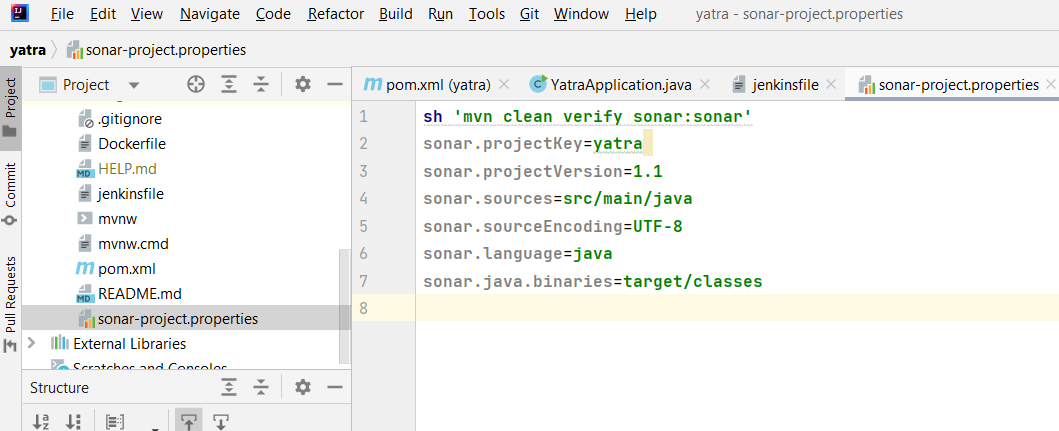


Go to intelij Idea and create a file named **sonar-project.properties**

Copy the first 2 lines from 2nd step of above screenshot. Remove the \ and -D

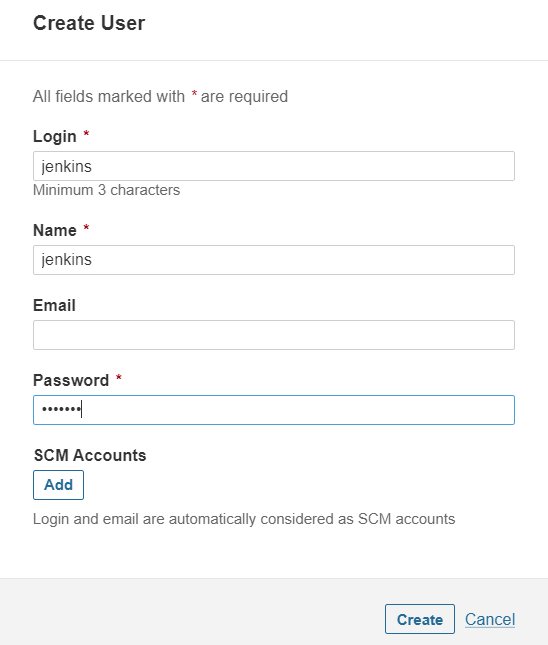
And then the below lines need to be pasted under the sonar.login property like the next screenshot.

**sonar.projectVersion**=**1.0  
sonar.sources**=**src/main/java  
sonar.sourceEncoding**=**UTF-8  
sonar.language**=**java  
sonar.java.binaries**=**target/classes**

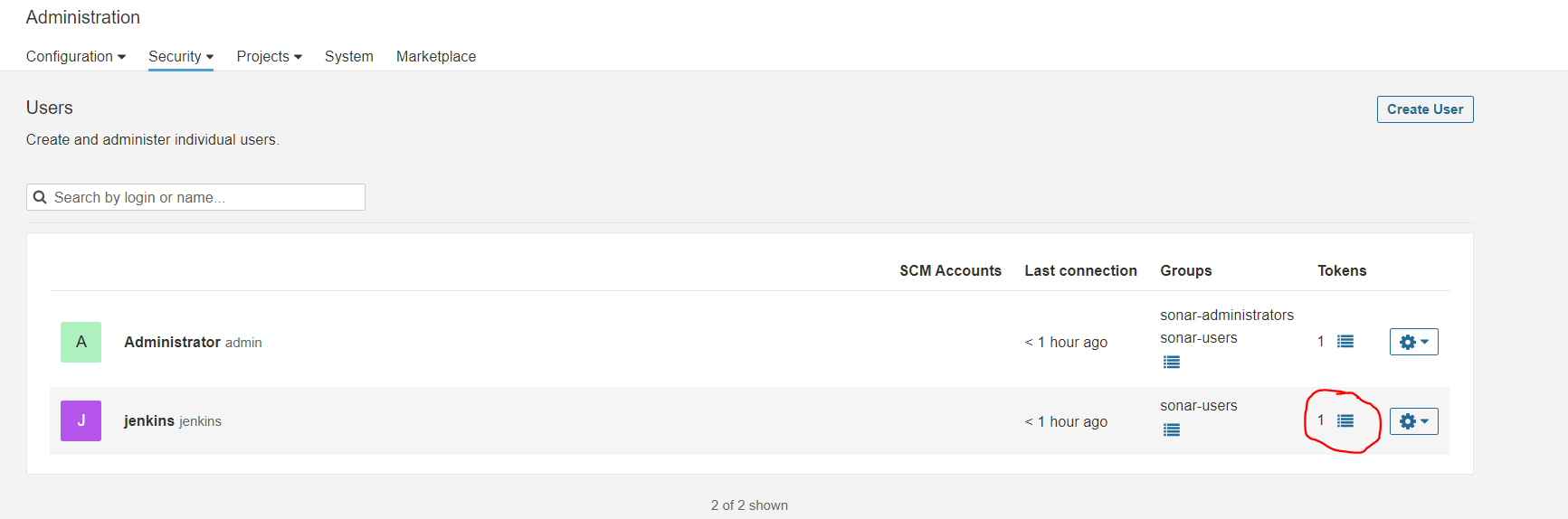


Go to Sonar qube page. Click on administration ->security->users.

Click on create user



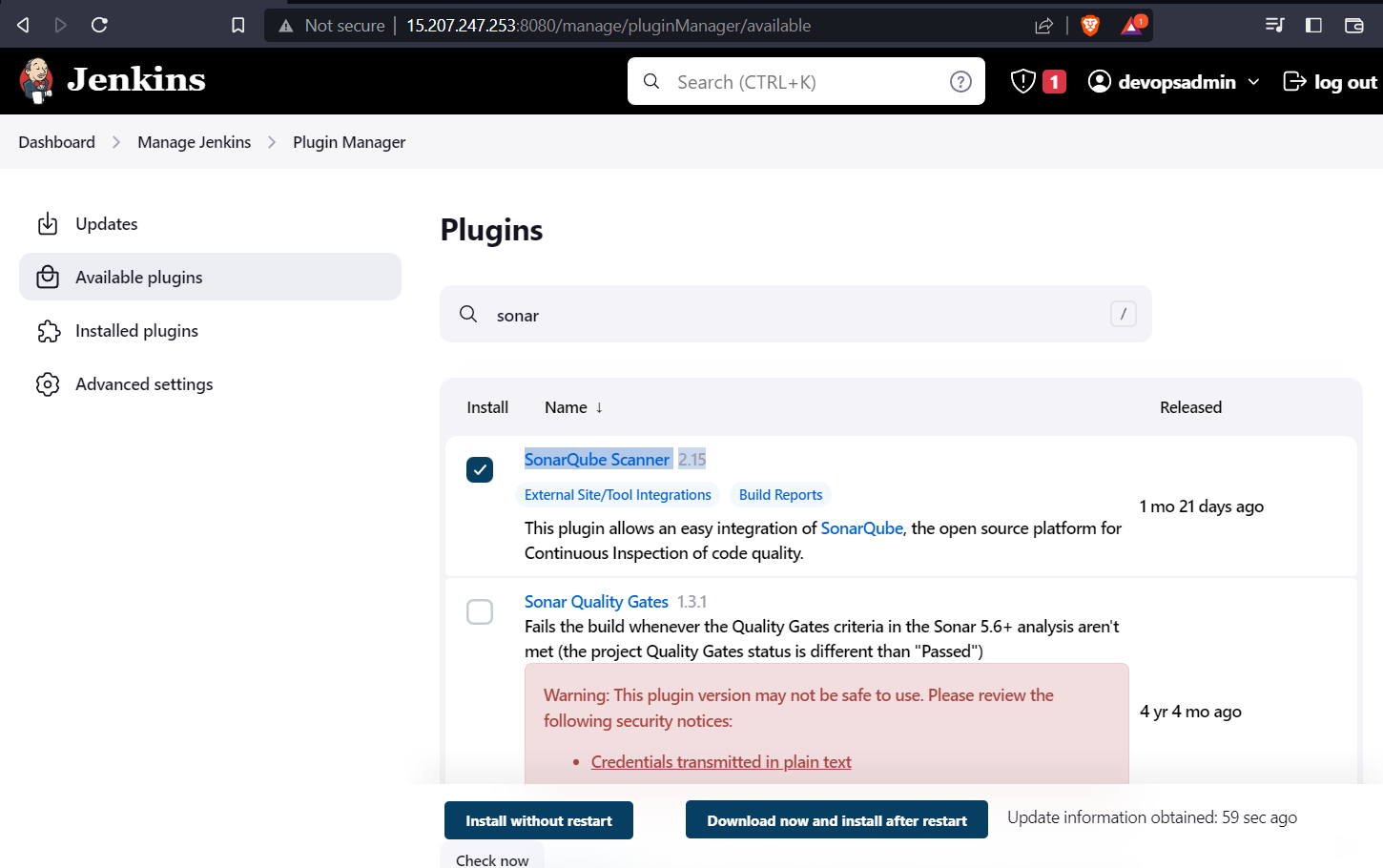
Click on create then click on the token(red rounded on the below screenshot) to generate token for Jenkins user



You can click on generate token then you can see on token has been generated. Copy the token to a note pad for future purpose

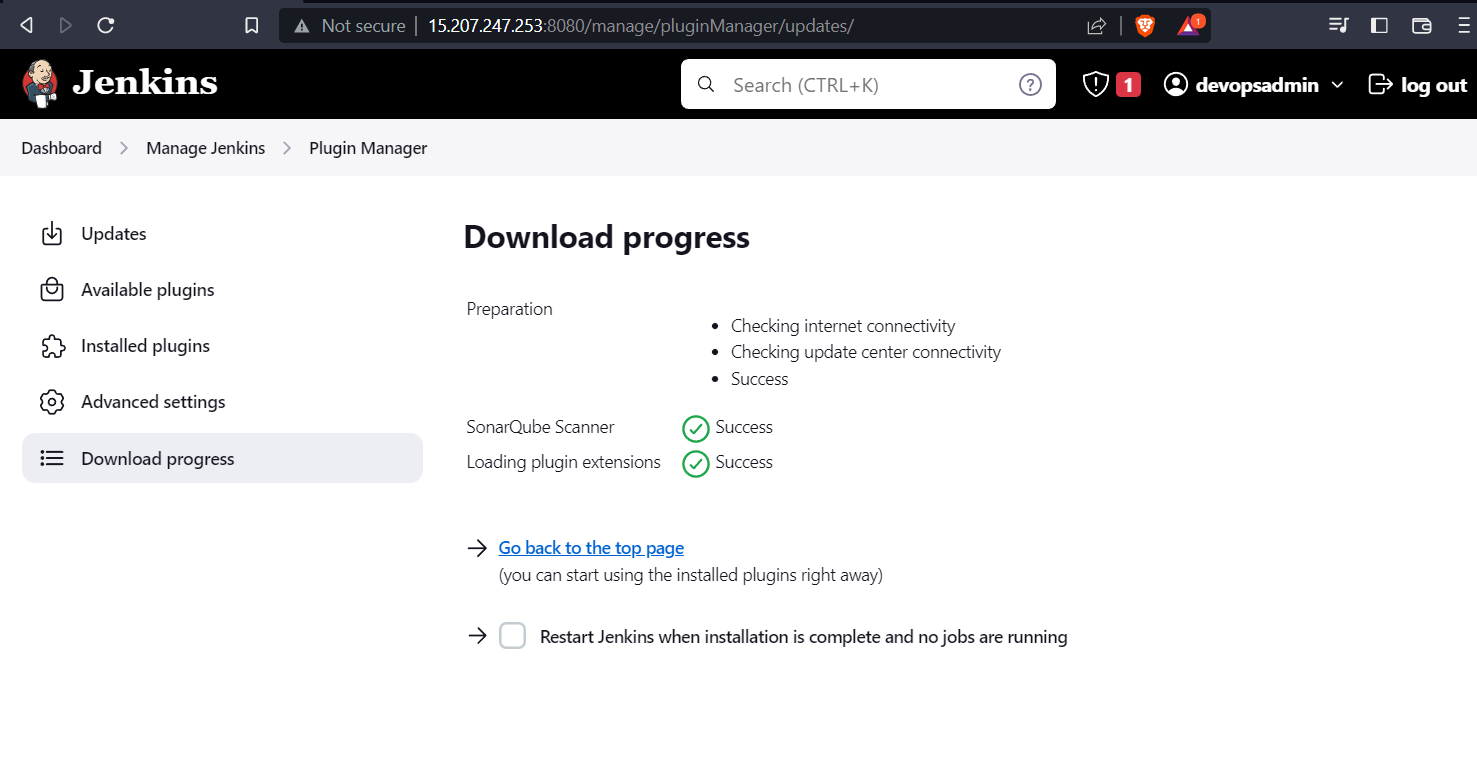


Go to the Jenkins and download the plugin **SonarQube Scanner**



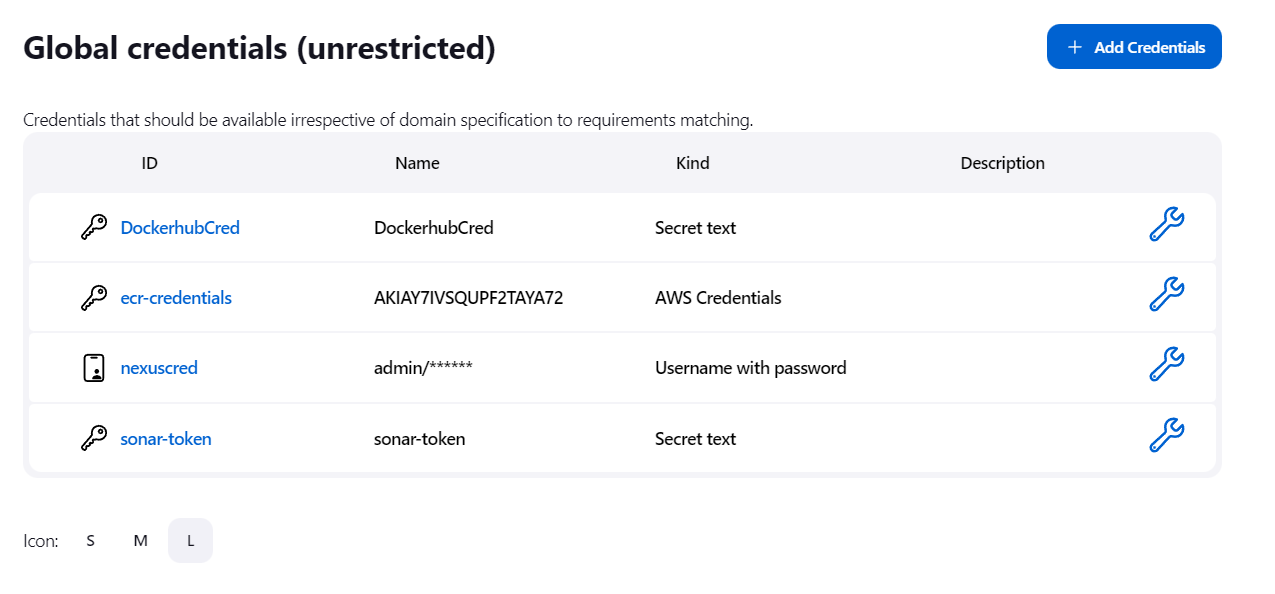
Click on install without restart.

Tick on the Restart Jenkins when installation is complete and no jobs are running



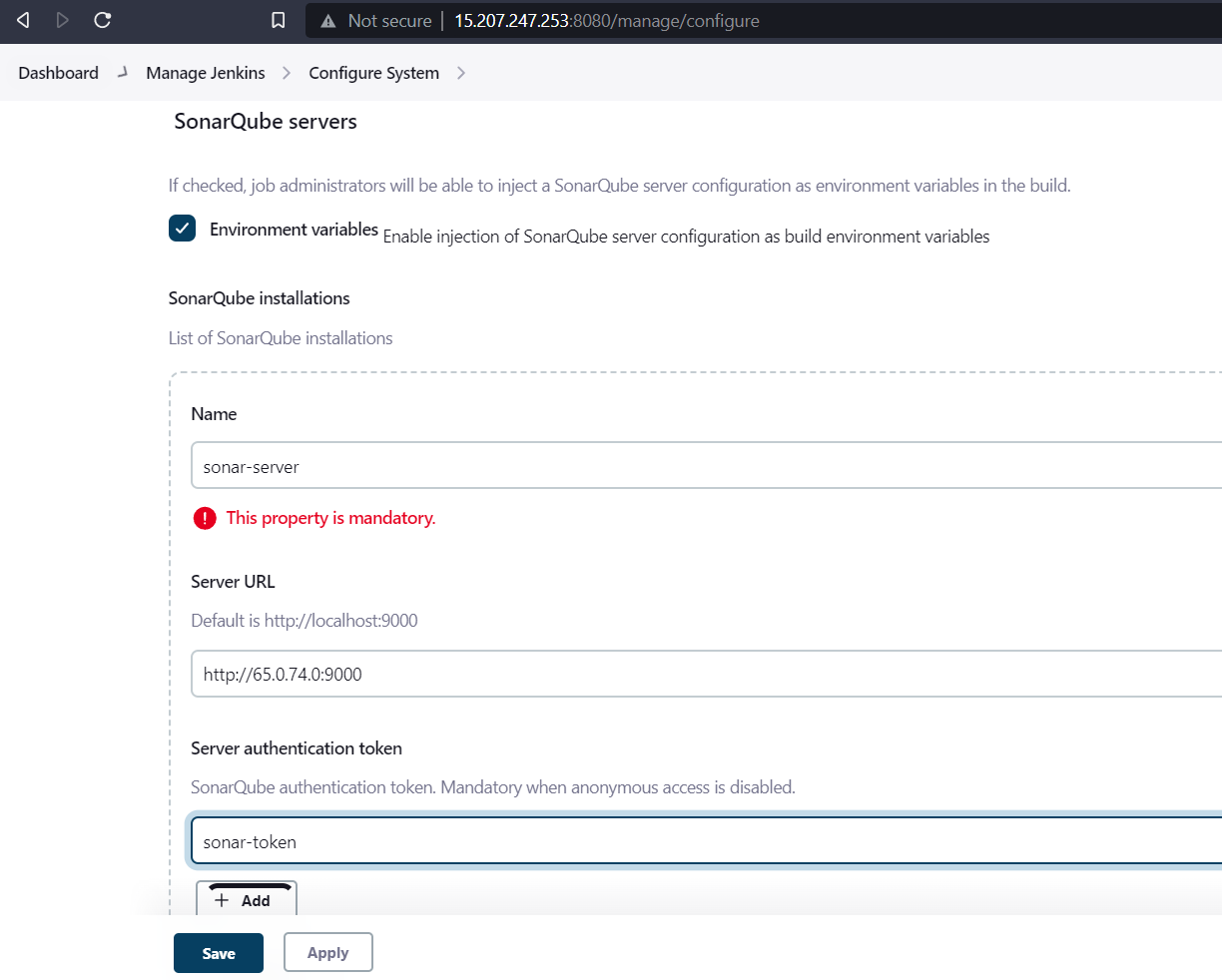
Add the global credentials with **the token** which is earlier generated while creating **the user jenkins in SonarQube.**





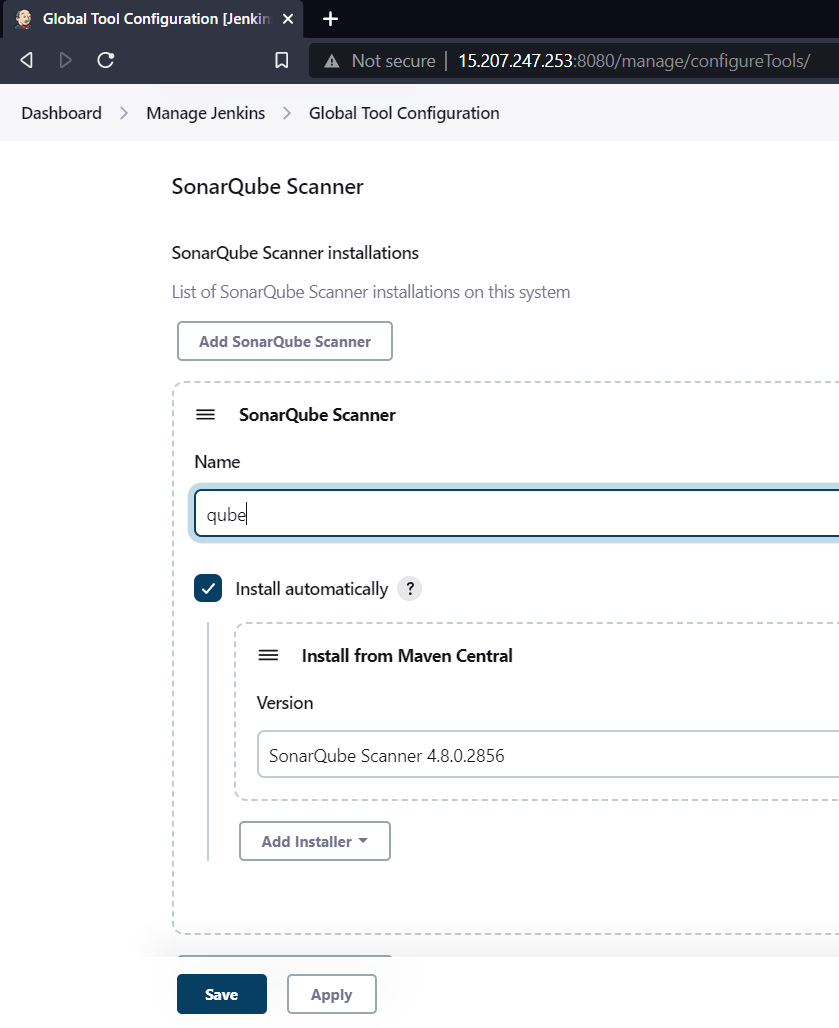
Go to dashboard-> Manage Jenkins ->Configure system

Go to SonarQube servers. Select Environment Variables. Type the name , Server URL of SonarQube. Select the token that we added. And click on save.



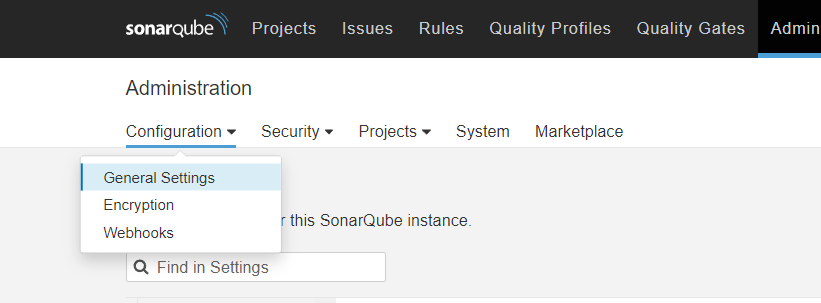
Go to Manage Jenkins->Global Tool Configuration

Go to Sonar qube scanner part. Click on Add SonarQube Scanner then type the name and click on save.

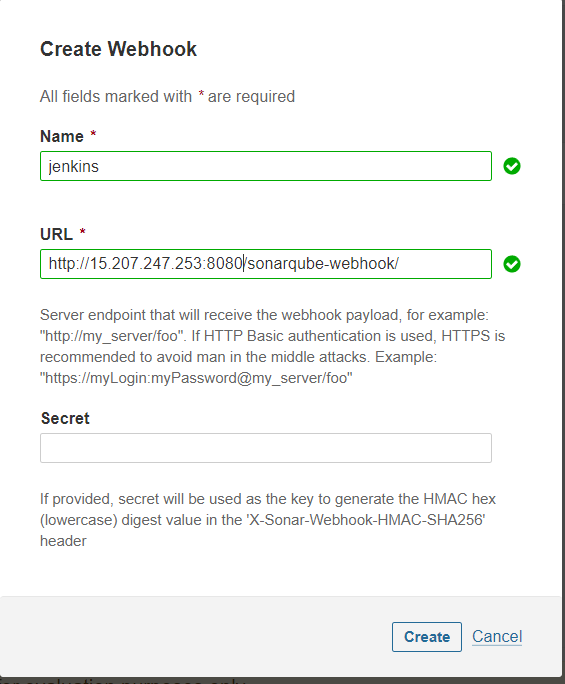


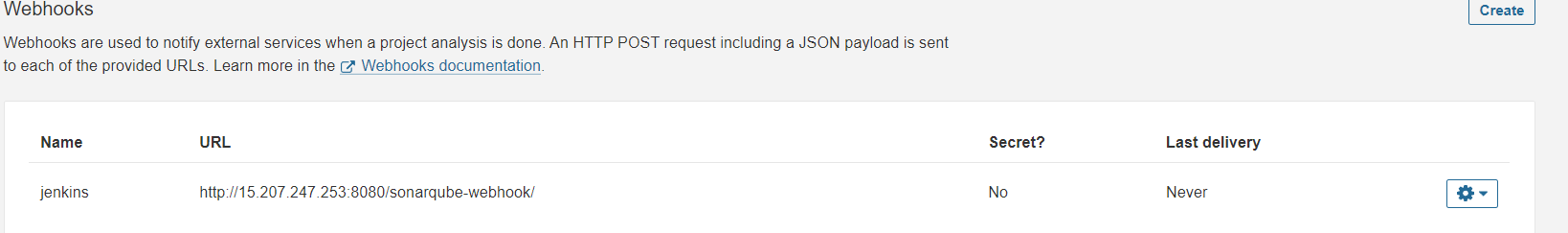
Create web hook for quality gate

Go to SonarQube Server -> Administration -> Configuration -> Webhooks



Click on create. Type the Jenkins URL like below screenshot and click on create.

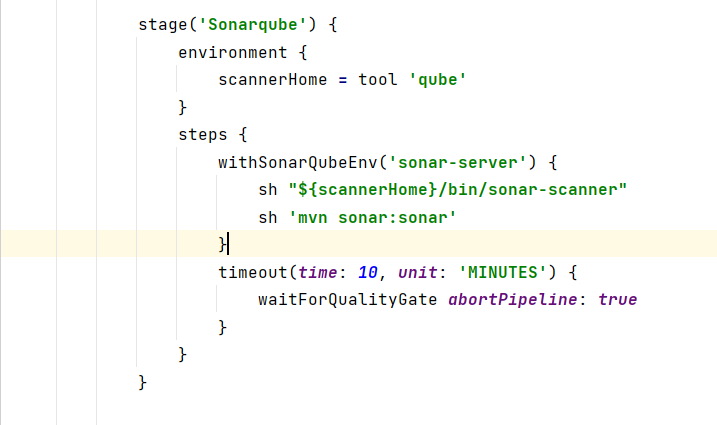




Go to Intellij Idea and open Jenkins file.

Type the script for sonarqube analysis. (Here **qube** is the name of sonarqube scanner which is mentioned in **Global tool configuration**, and **sonar-server** is the name of sonarqube installation from **Configure system**.)

stage(**'Sonarqube'**) {  
 environment {  
 scannerHome **=** tool **'qube'** }  
 steps {  
 withSonarQubeEnv(**'sonar-server'**) {  
 sh **"${scannerHome}/bin/sonar-scanner"** sh **'mvn sonar:sonar'** }  
 timeout(***time***: ***10***, ***unit***: **'MINUTES'**) {  
 waitForQualityGate ***abortPipeline***: ***true*** }  
 }  
}



Push the changes to Git hub. And run the project in Jenkins.

There you can see your project has do the sonarqube analysis success fully. You can see an option sonarqube on the left side. You can click and check the overview and code scanning results.

