SonarQube with Jenkins

This section will guide you to:

* Integrate SonarQube with Jenkins

This guide has three subsections, namely:

1. Installing SonarQube 7.8
2. Installing and configuring SonarQube plugin in Jenkins
3. Creating a Jenkins job and running Sonarqube Scanner

**Note**: SonarQube 7.5 is already installed in the lab. We will need SonarQube 7.8 for this demo. Install SonarQube 7.8 using the steps given below. You need to have *JDK 1.8, Maven* configured in order to proceed with this demo.

**Step 1:** Installing SonarQube 7.8

* Open the browser in the lab and navigate to https://www.sonarqube.org/downloads/
* From the *historical downloads* section at the bottom of the page download the community edition of sonarqube7.8
* Run the following command in the terminal to extract the sonarqube-7.8 zip file:

*cd Downloads*

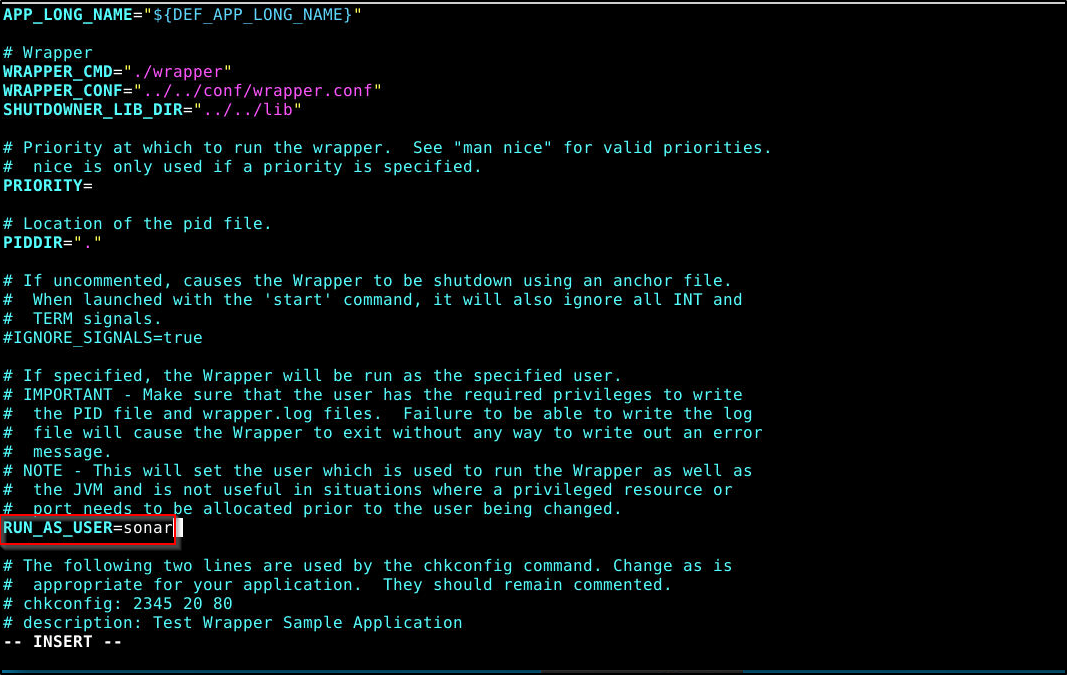
*sudo unzip sonarqube-7.8.zip -d /opt/sonarqube*

*cd*

*sudo useradd sonar*

*sudo -i*

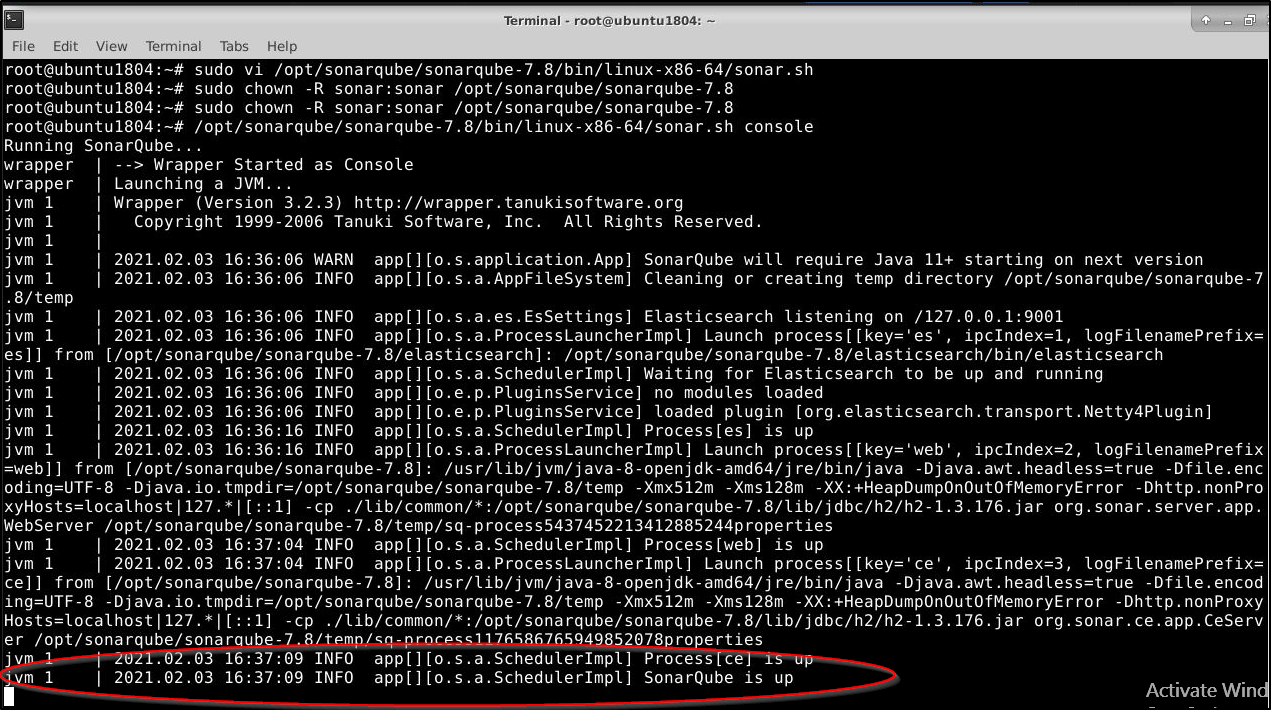
*sudo vi /opt/sonarqube/sonarqube-7.8/bin/linux-x86-64/sonar.sh*



* Save and exit the file using [esc] shift: wq
* Run the following commands to start the sonarqube server

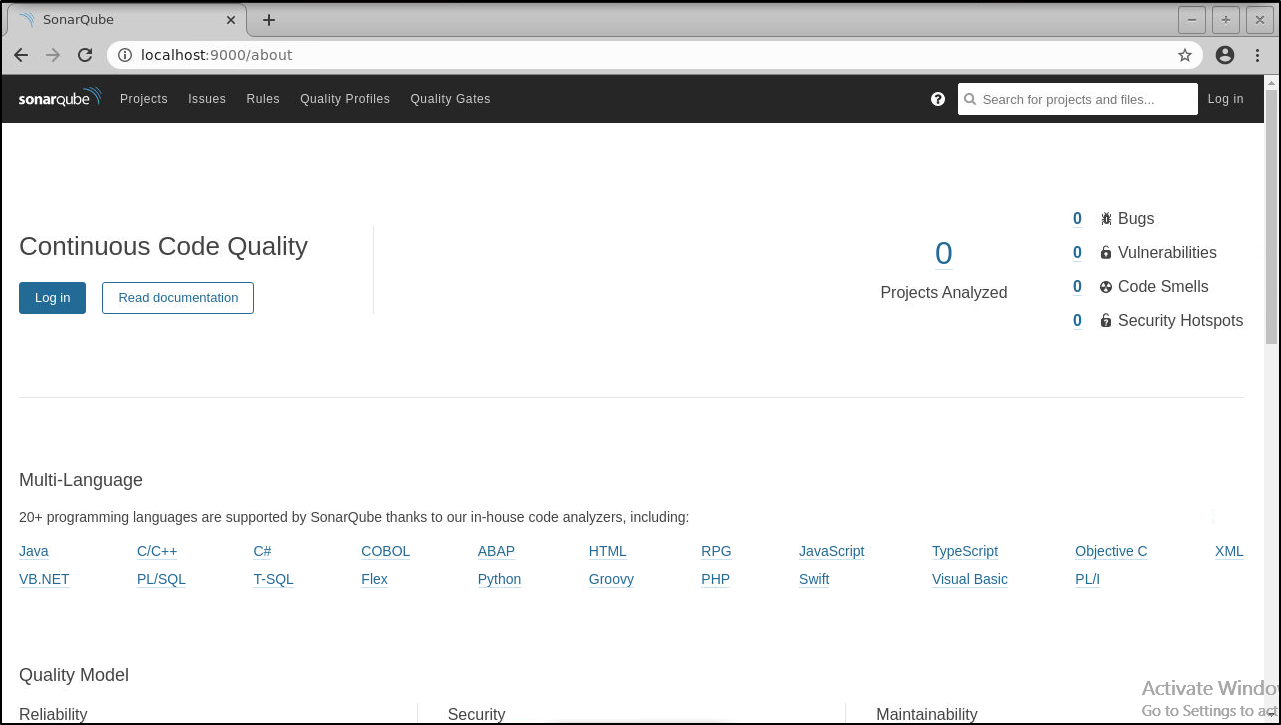
*sudo chown -R sonar:sonar /opt/sonarqube/sonarqube-7.8*

*/opt/sonarqube/sonarqube-7.8/bin/linux-x86-64/sonar.sh console*

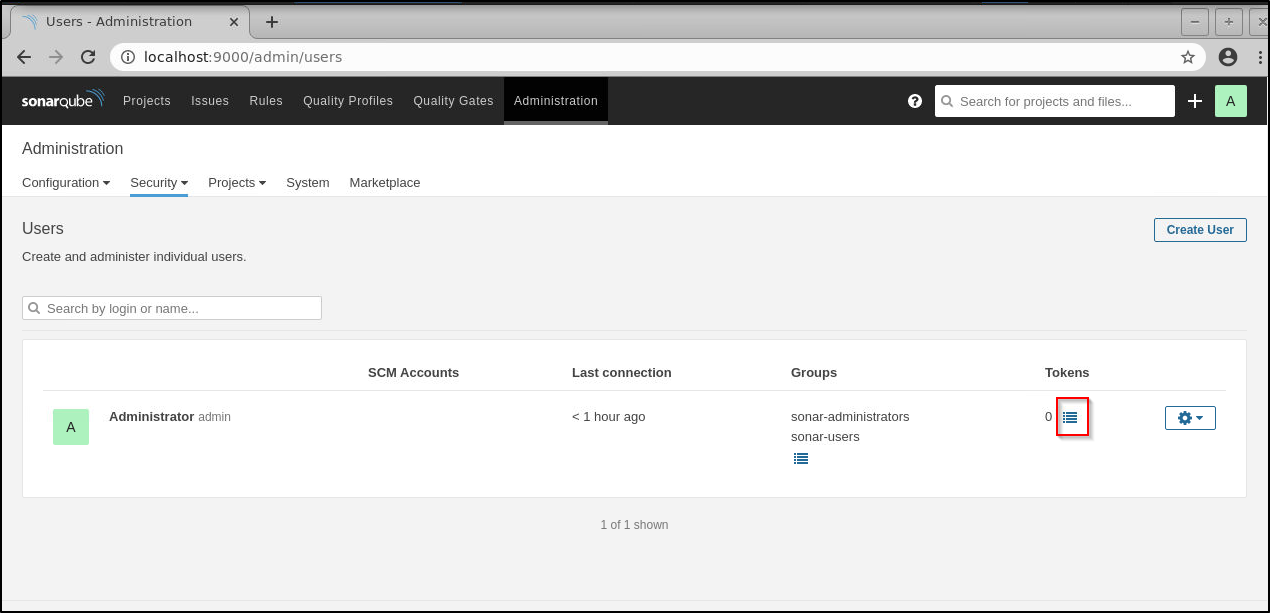
**

**Note:** Closing this terminal window will stop/kill the sonarqube process. Do not close this terminal window till you complete the demo.

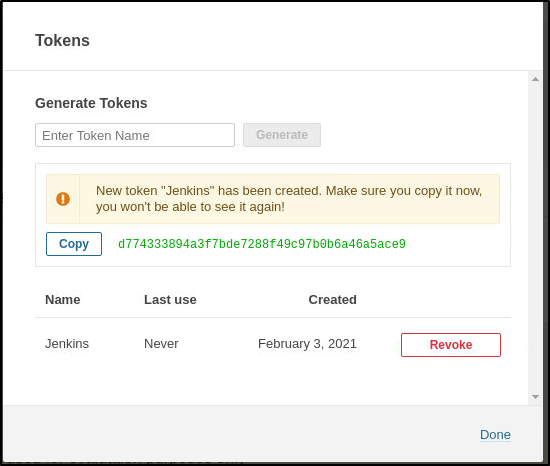
* Open the browser and navigate to **http://localhost:9000**

****

* Log in to sonarqube server with System Administrator credentials (admin/admin)
* Go to *Administration > Security > Users > Tokens*



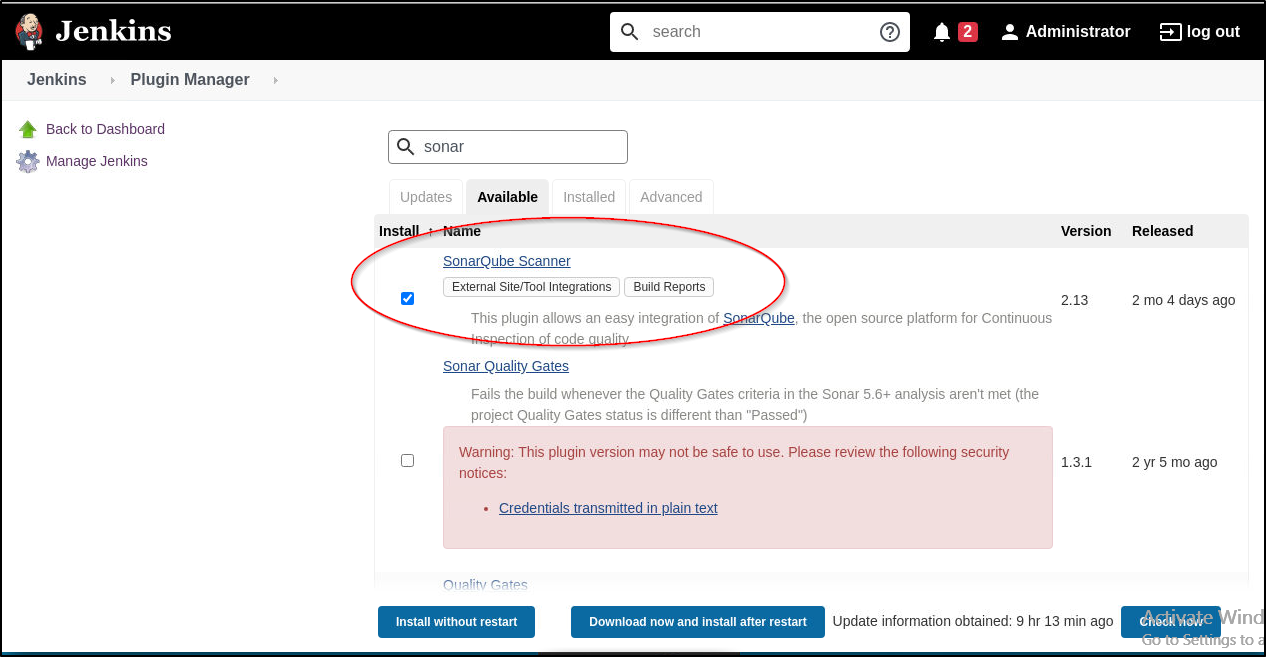
* Click on token and generate a token with name: *Jenkins* as shown below:

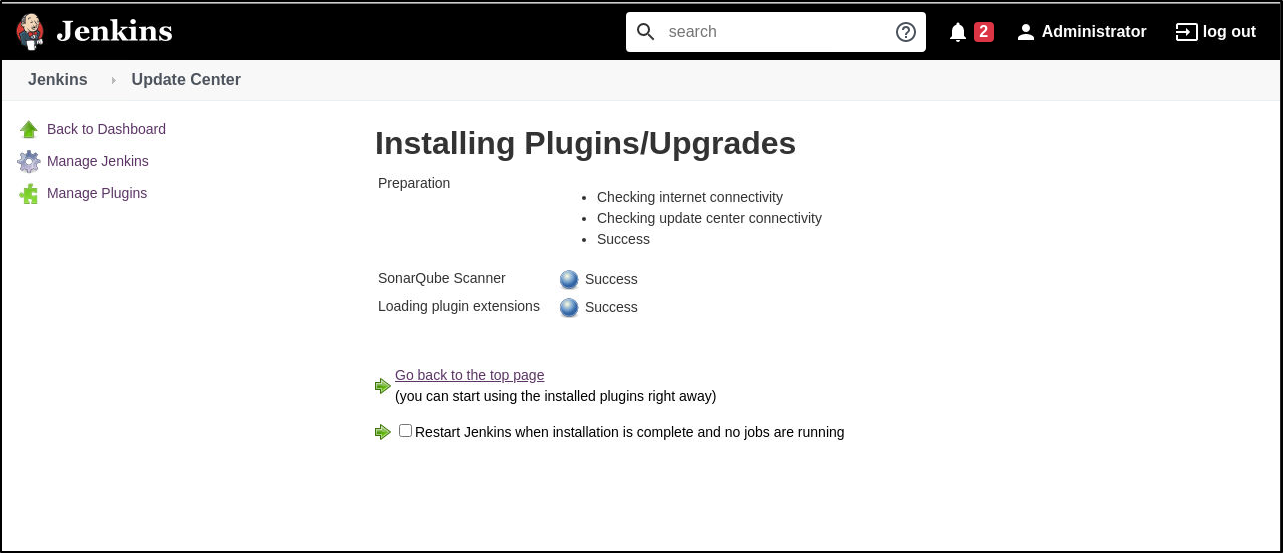


* Copy the generated token and note it down. It will be used in Jenkins for Sonar authentication

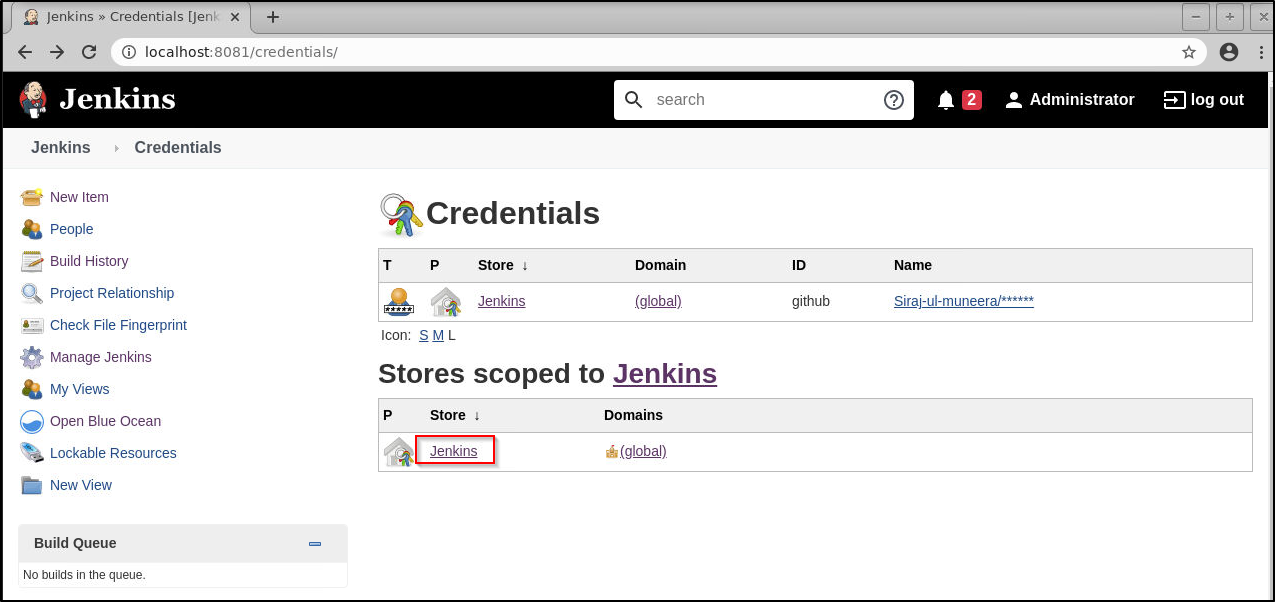
**Step 2:** Installing and configuring SonarQube plugin in Jenkins

* Go to *Manage Jenkins > Manage Plugins > Available* > search for *SonarQube Scanner*> Click on *install without restart*

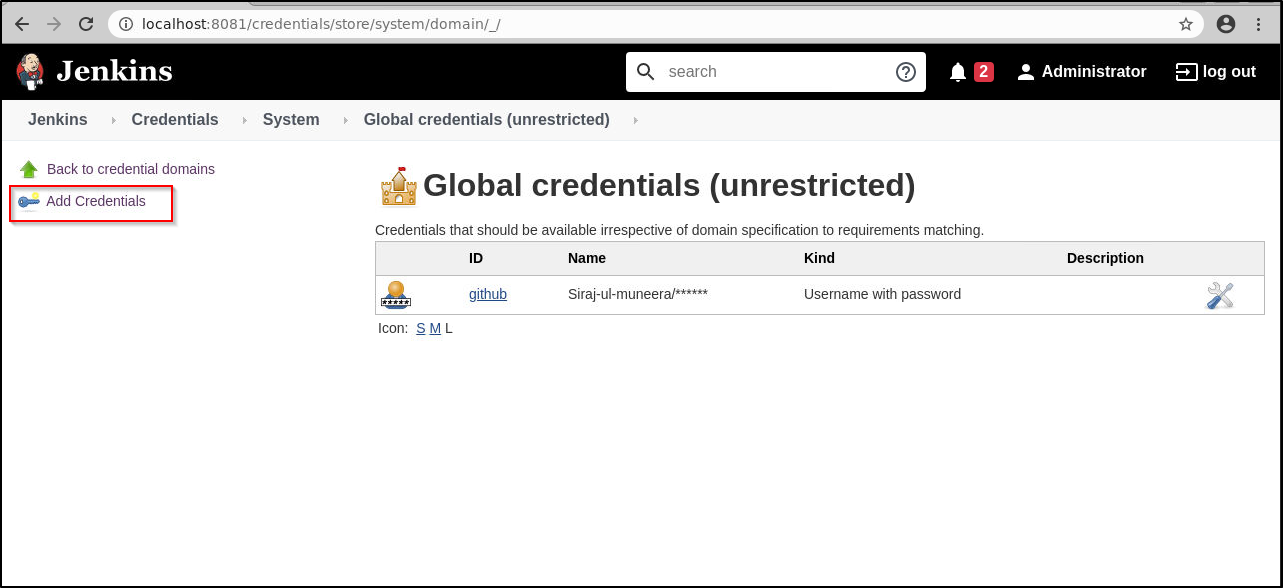




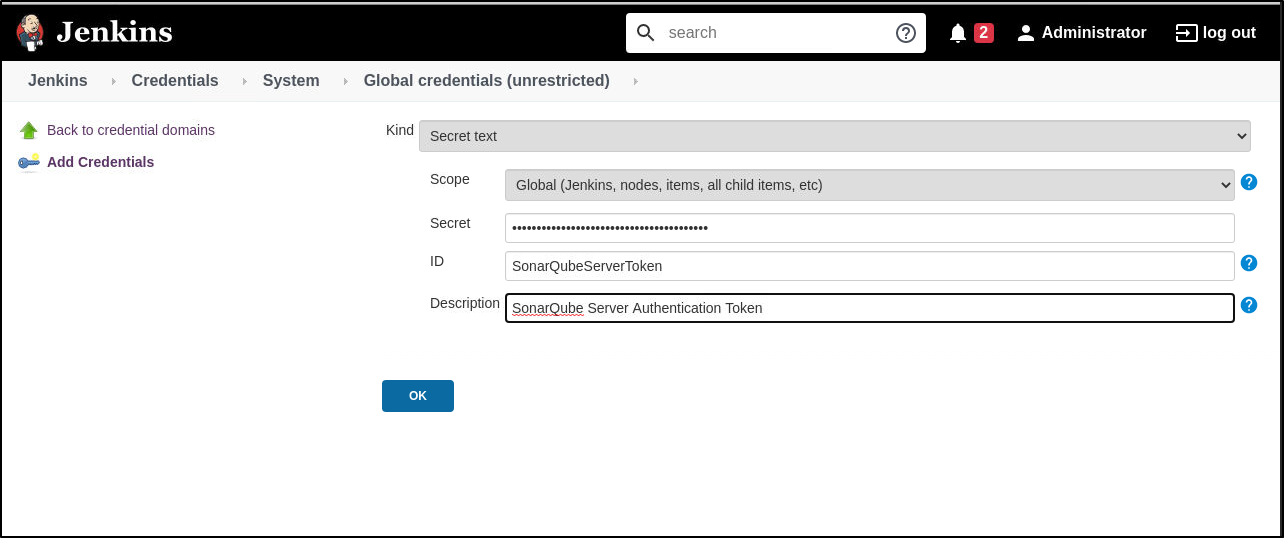
* Go to Jenkins dashboard > *Manage Jenkins* > *Manage Credentials*



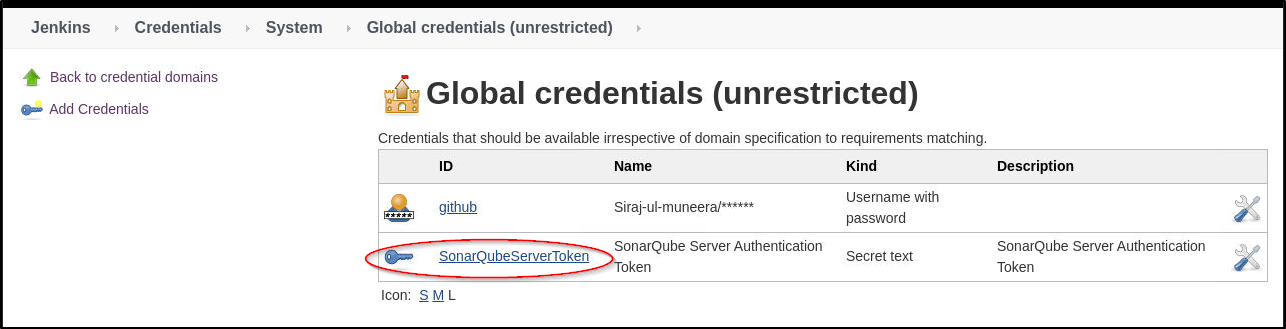
* Click on Jenkins as shown above and in the *Global credentials unrestricted* page, click on *Add Credentials*

**

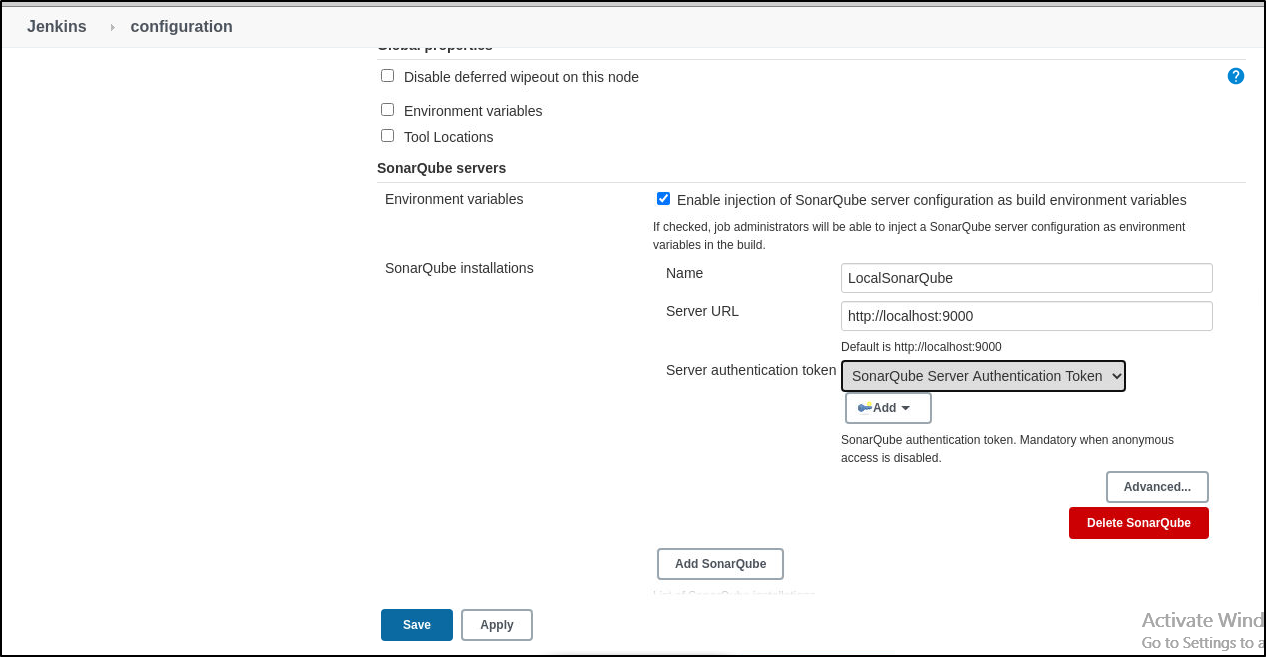
* Select the kind as***Secret text***from the drop-down. Paste the token that you had earlier copied from the sonarqube server into the ***Secret*** field. Give the ID and description as shown below and click on ok.



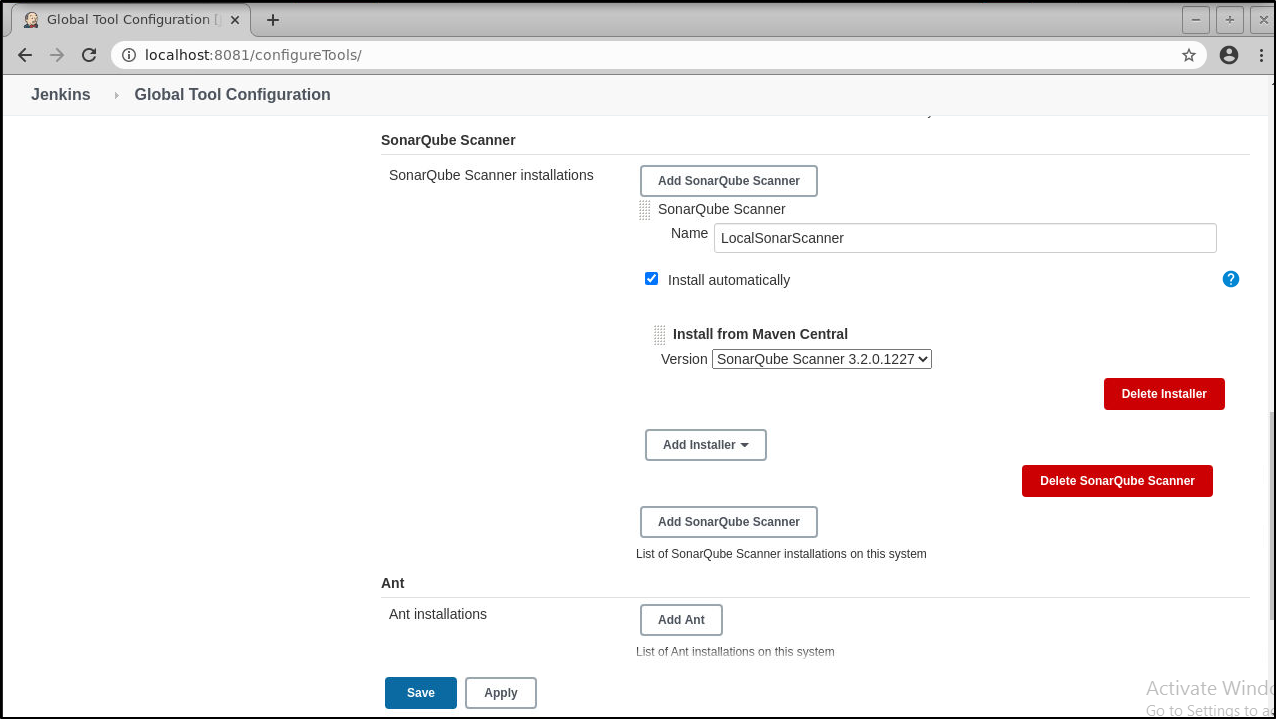
You will see the credentials added in the *Global credentials* page



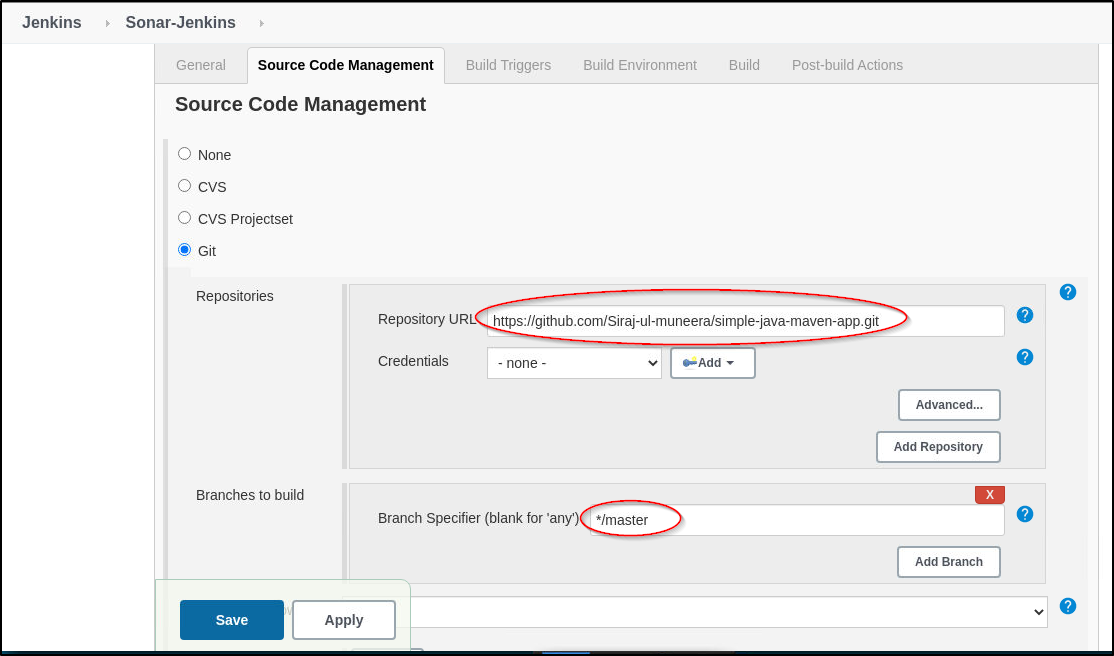
* Go to Jenkins dashboard > *Manage Jenkins* > *Configure system* > *SonarQube servers* section > Click on the checkbox *Enable injection of sonarqube server configuration as build environment variable*
* Click on *Add SonarQube* > provide a name (ex: LocalSonarQube) and Server URL as <http://localhost:9000>. Select the authentication token from the list and click on *Apply and Save* as shown below:

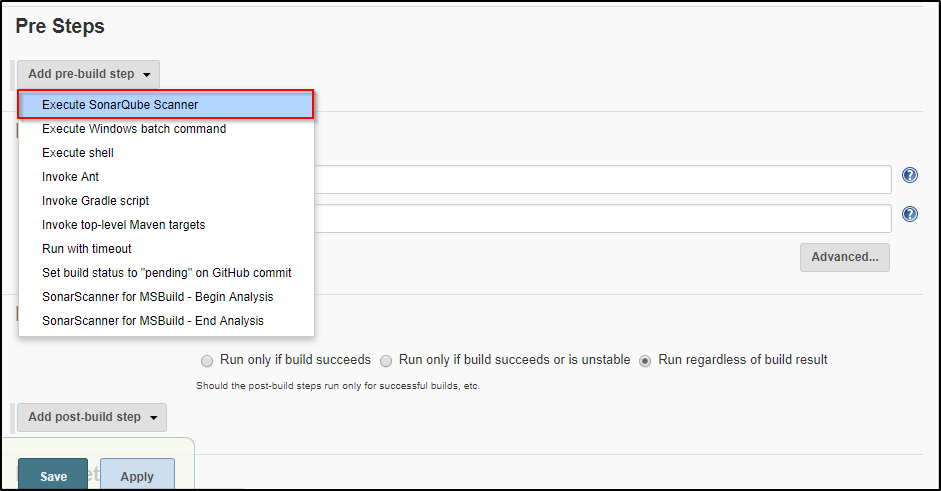


* Go to *Manage Jenkins > Global Tool Configuration >* Scroll for *SonarQube Scanner > Add SonarQube Scanner* > provide a name (ex: LocalSonarScanner), check **Install automatically** and select the version 3.2.0 from the drop-down list as shown below:



**Step 3**: Creating a Jenkins job and running Sonarqube Scanner

* Create a new job > provide a name (ex: Sonar-Jenkins), and select project type as *freestyle*
* Under SCM select *Git* and enter the git repository of the simple-java-maven-app that we had created earlier in the demo 4 of lesson 3  
    
  
* In the build section click on *Add a build step* and select the option *Execute SonarQube Scanner* from the drop-down list.



* Enter the details in the *Analysis properties* section as shown below:

*#Required metadata*

*sonar.projectKey=com.mycompany.app:my-app*

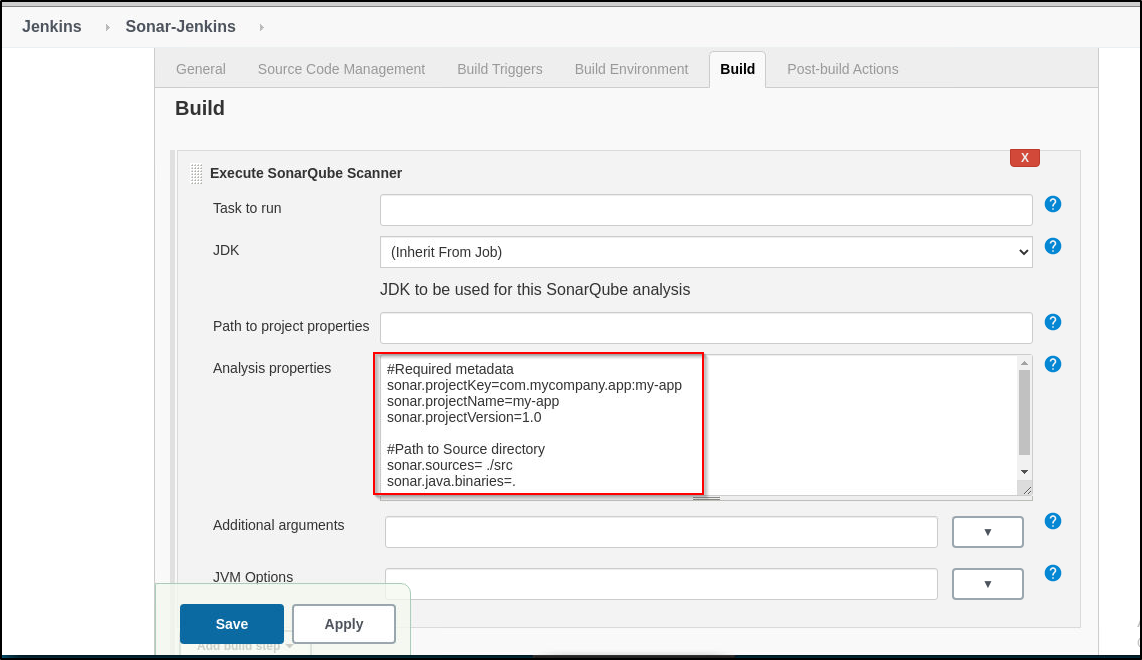
*sonar.projectName=my-app*

*sonar.projectVersion=1.0*

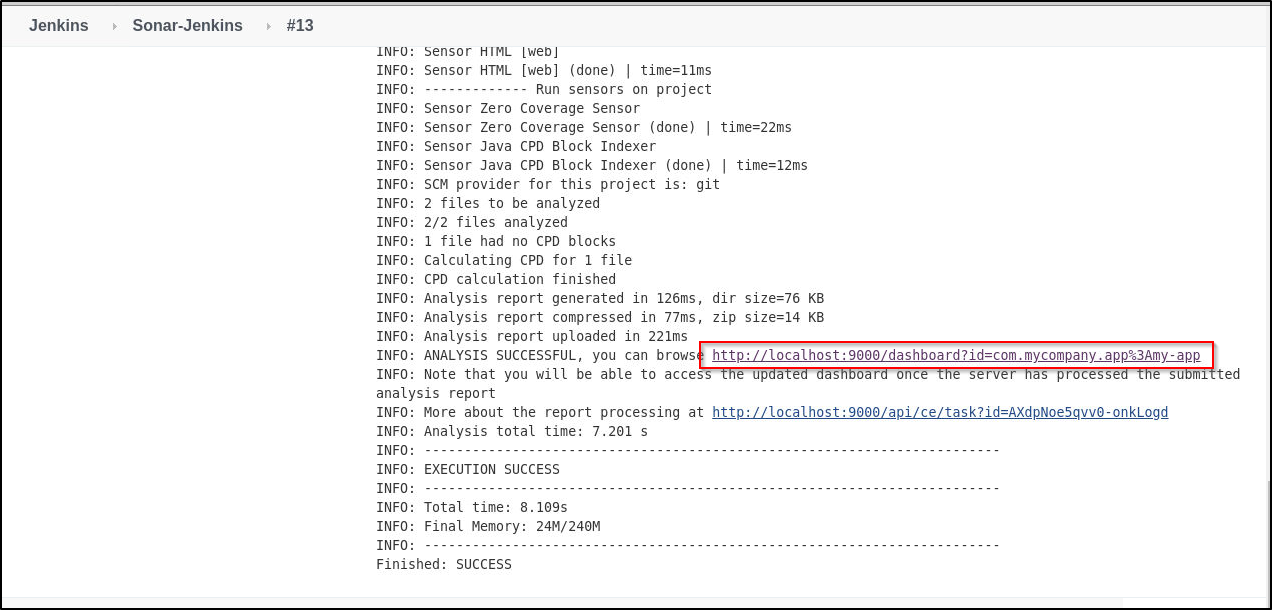
*#Path to Source directory*

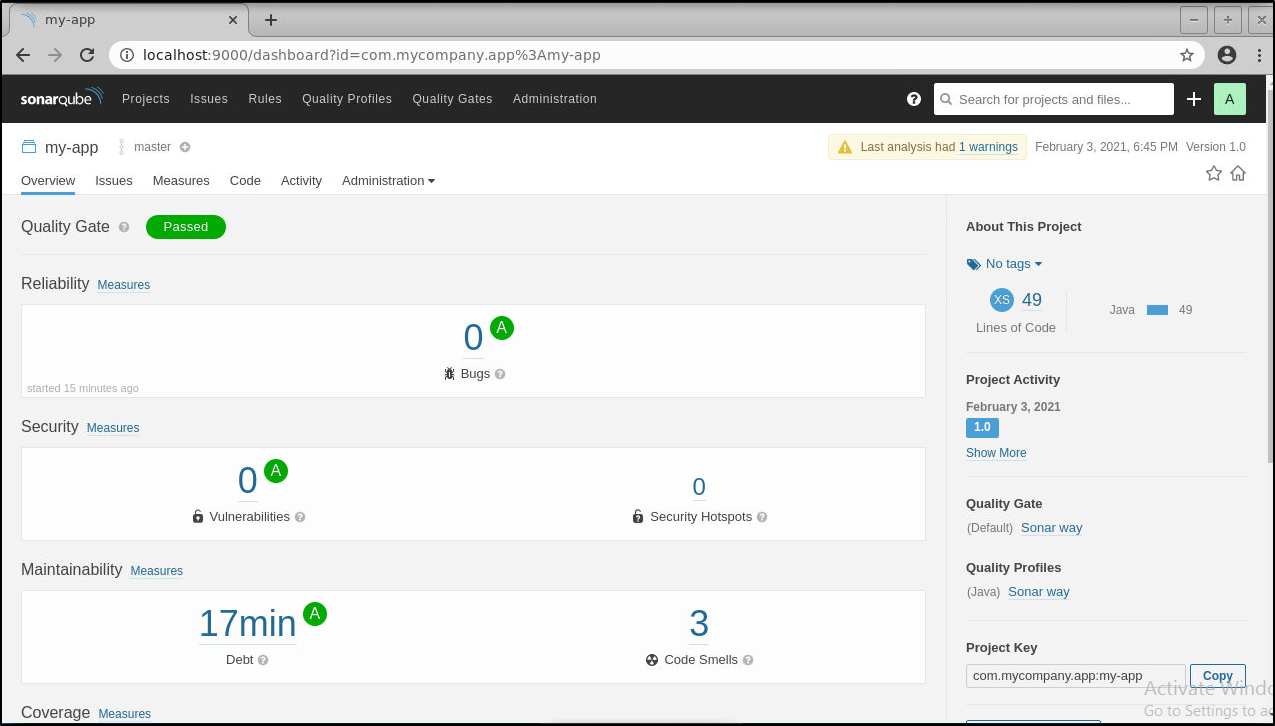
*sonar.sources= ./src*

*sonar.java.binaries=.*

**

* Click on *Apply and Save*
* Build the job
* On successful completion of the build from the console output you can see the project in the sonarqube server by clicking the link as shown in the output



You can also check the report on the sonarqube server. From the job dashboard, click on the sonarqube icon, and thenclick on *Projects.*You can see the report as shown below:

