Assignment 4

Automate the process of managing binary files and load balance

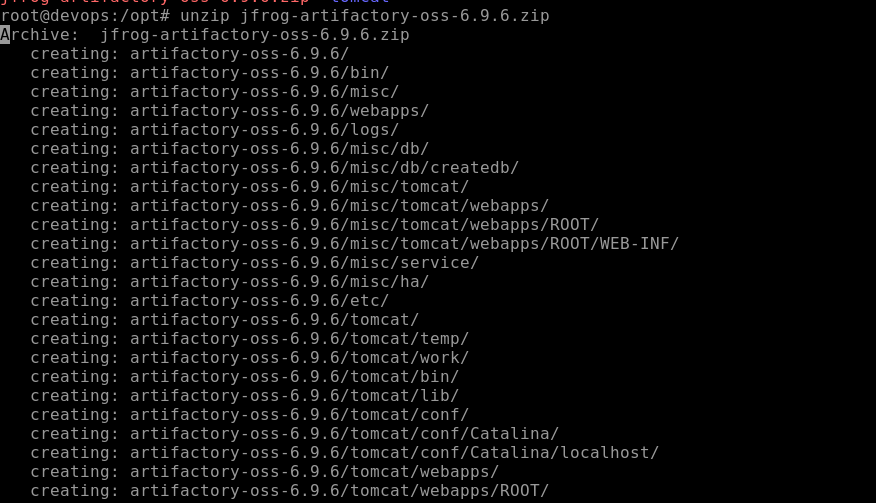
1.Install JFrog plugin in Jenkins and configure

2.Install SonarQube server on plain VM

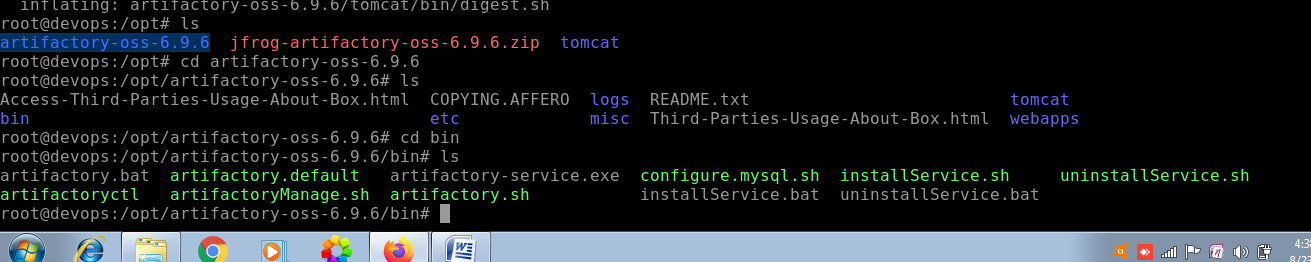
3.Configure SonarQube server with Jenkins

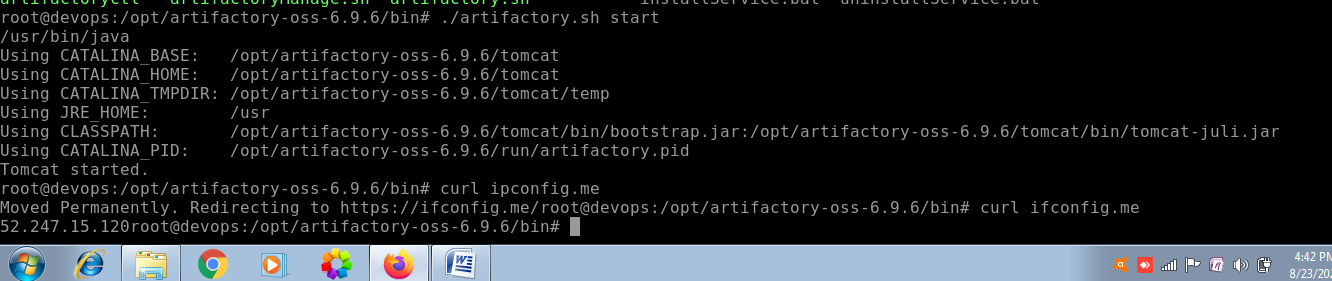
Now we will set up the JFrog server in our VM





After unzipping now we will go to bin directory and start the services

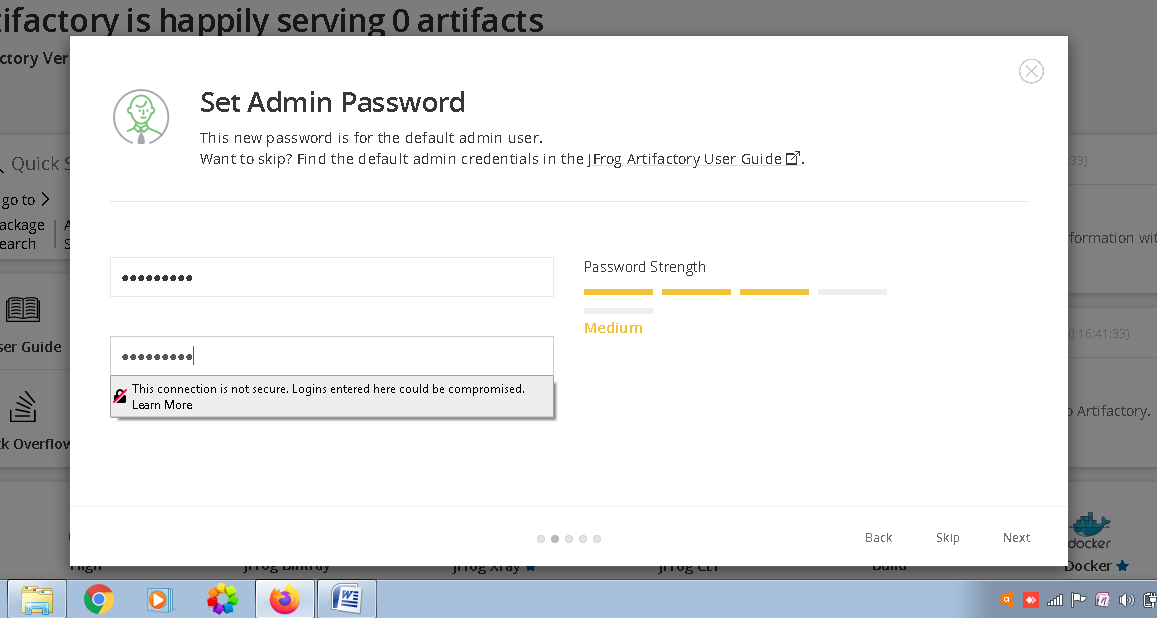


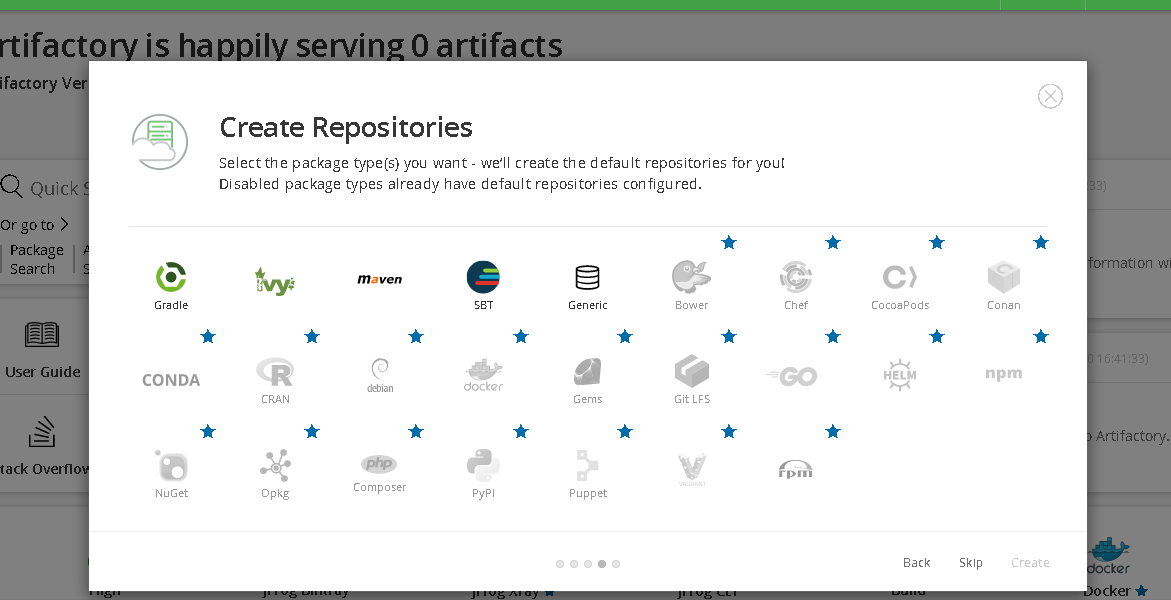


Now we could see that the JFROG Artifactory server is setup at port 8081 as follows

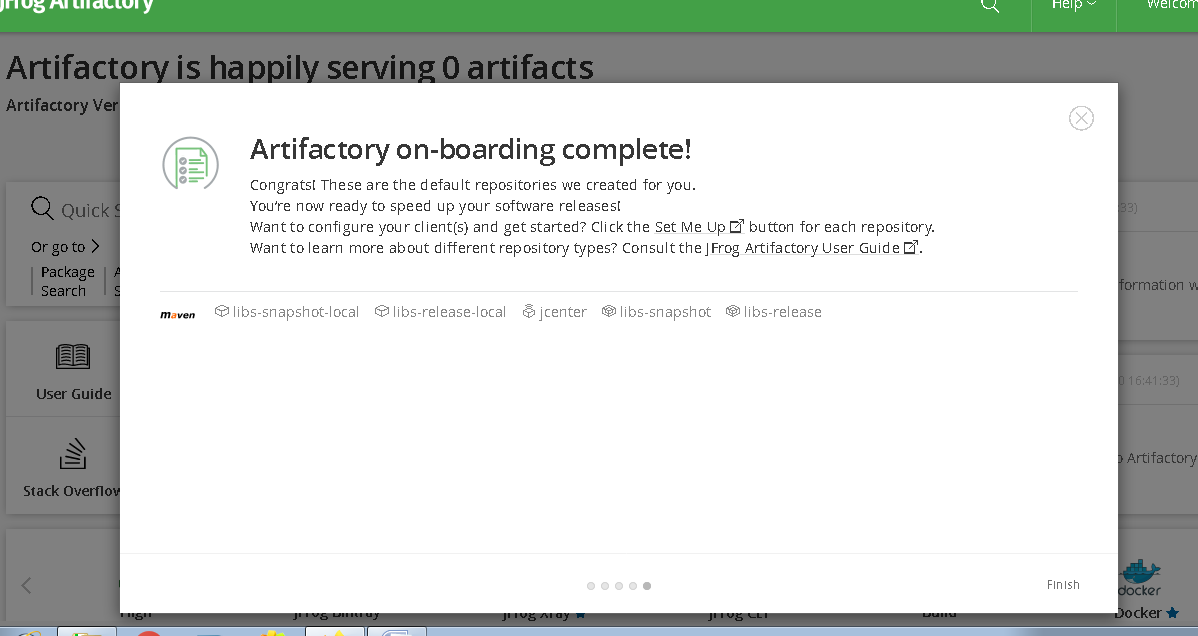


Then click next and setup the admin password as follows





Since we are going to build maven project hence I will be selecting Maven and then click on create



Then click on finish

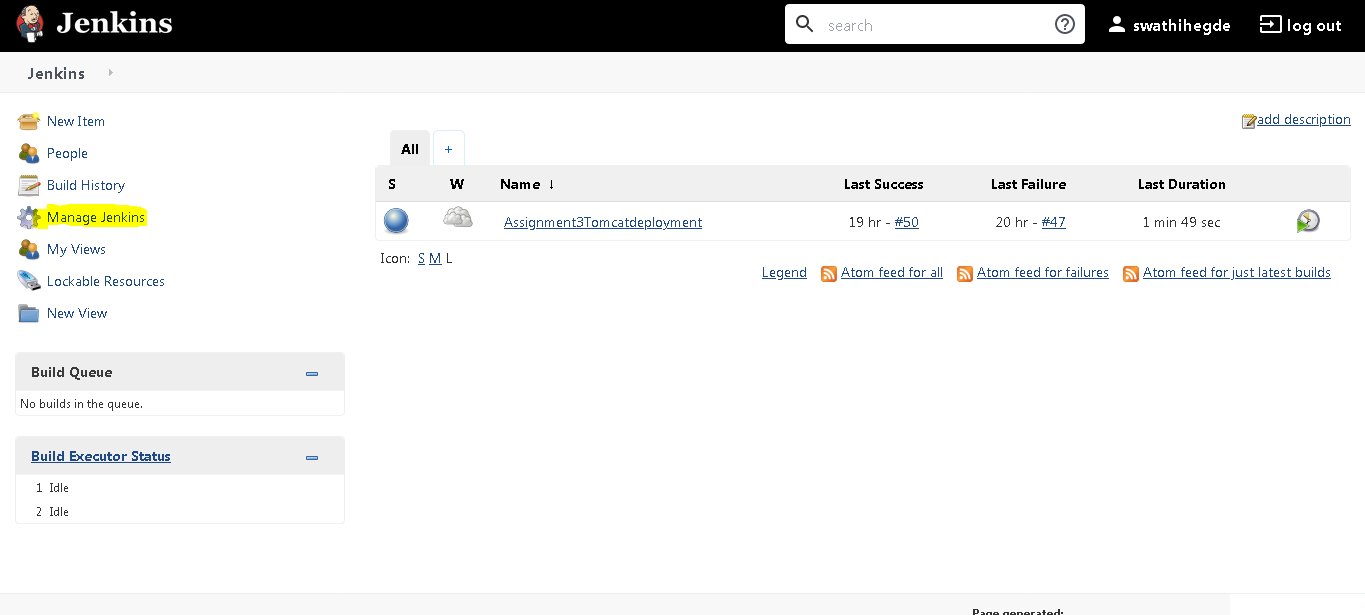
Now we will be building our project and will be storing our artifactory over here.

**Intergrating Artifactory with Jenkins**

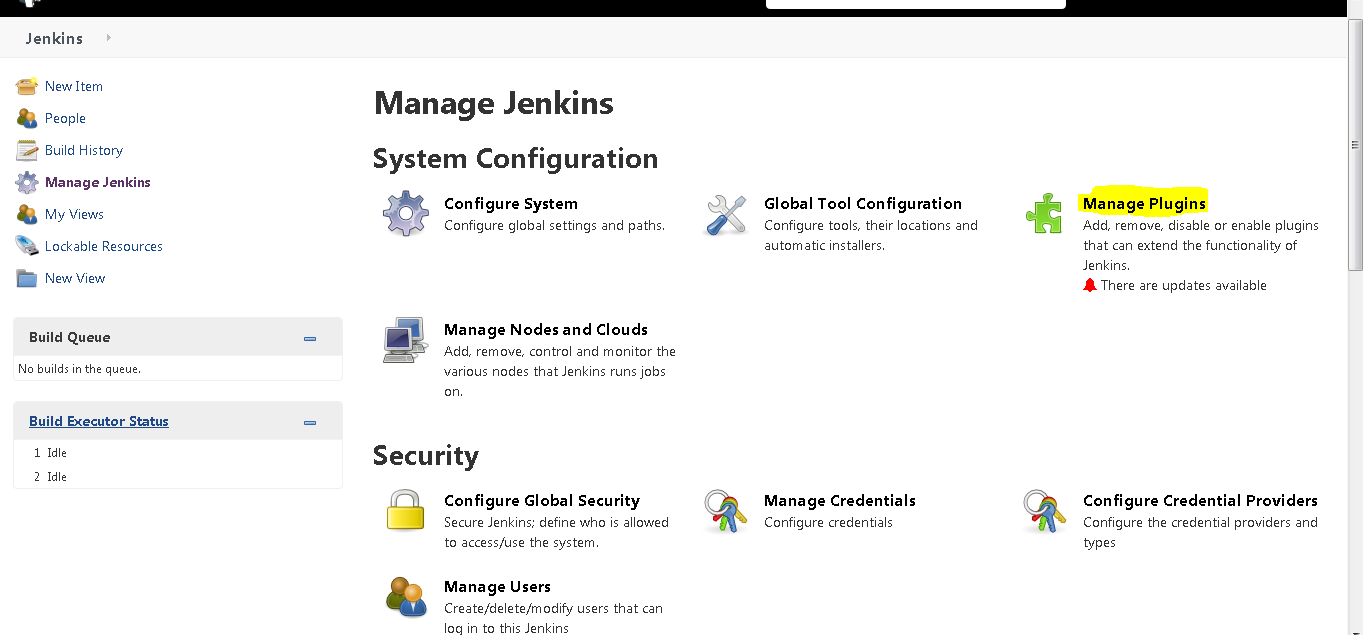
For that first login to Jenkins server on port 8080

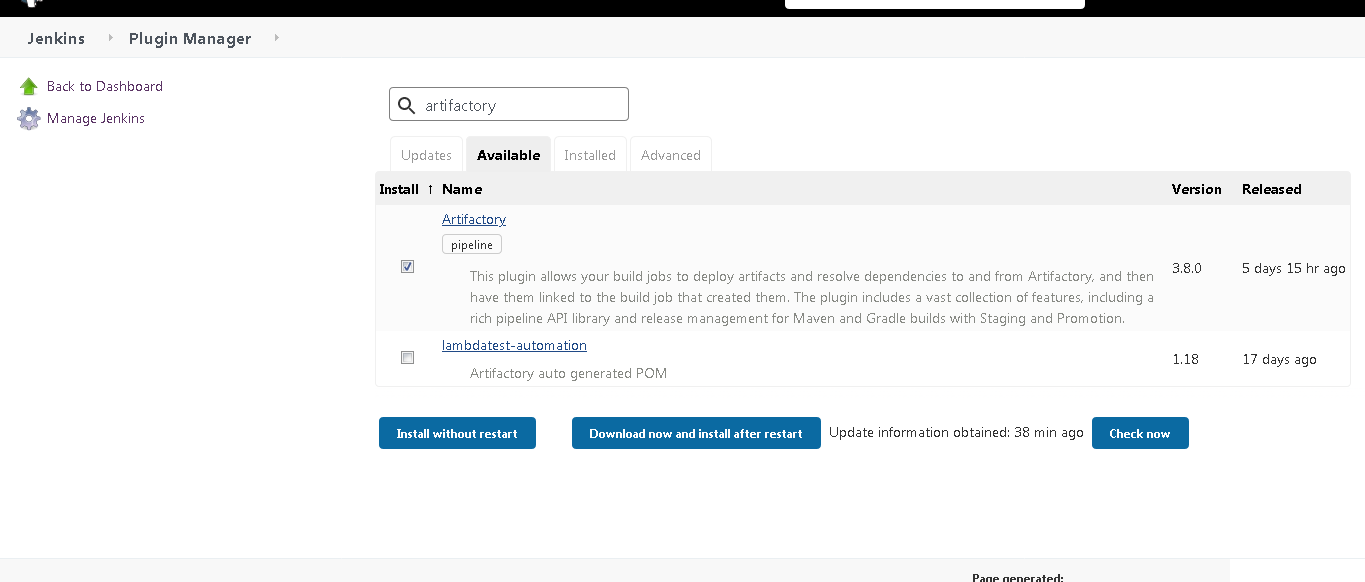
Now we will be installing artifactory plugin as follows

click on manage Jenkins

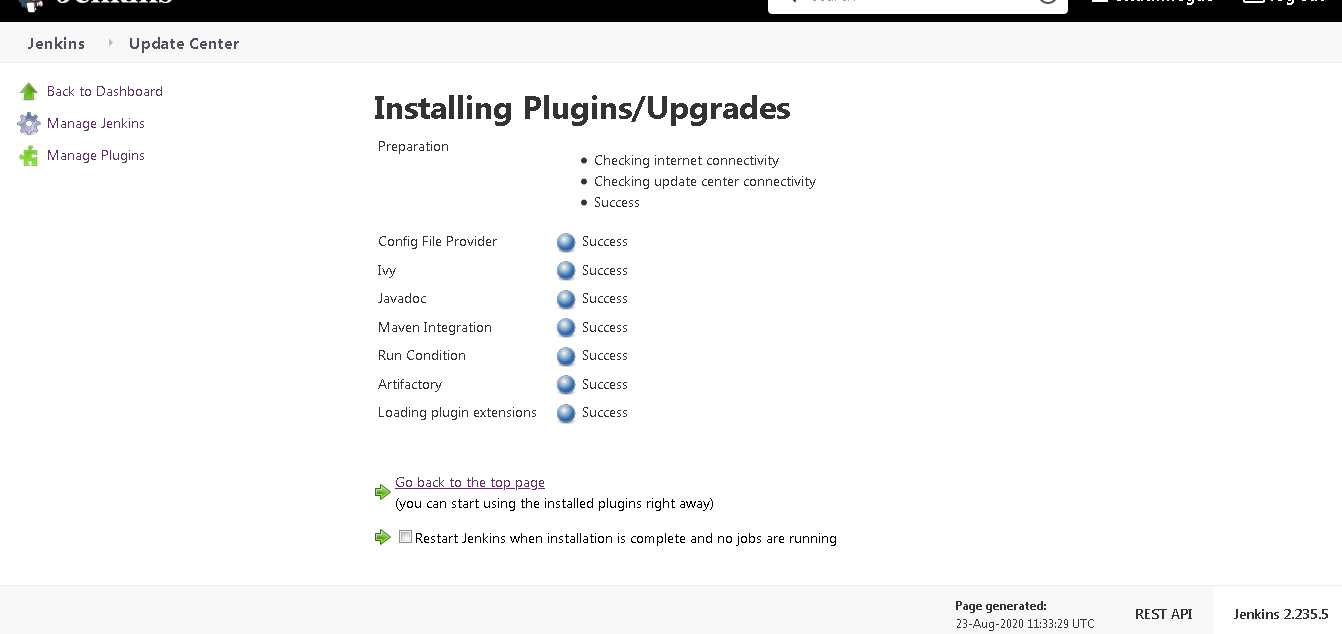


Then Manage plugins



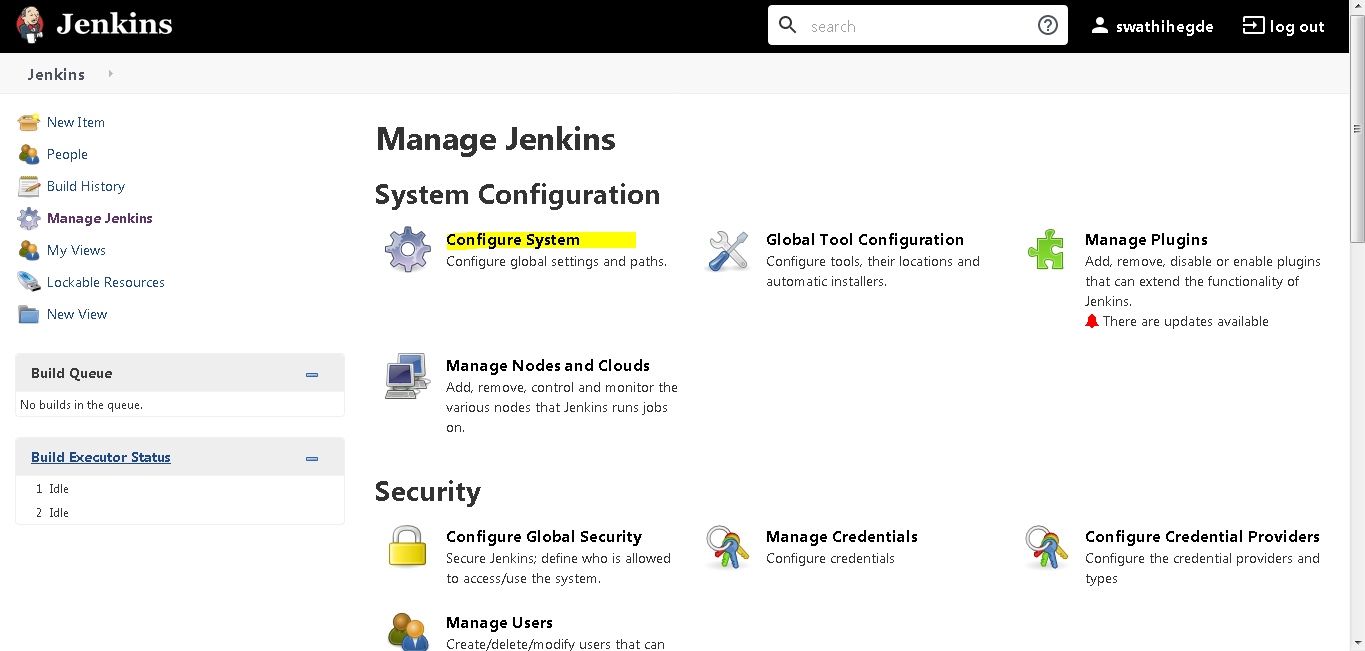


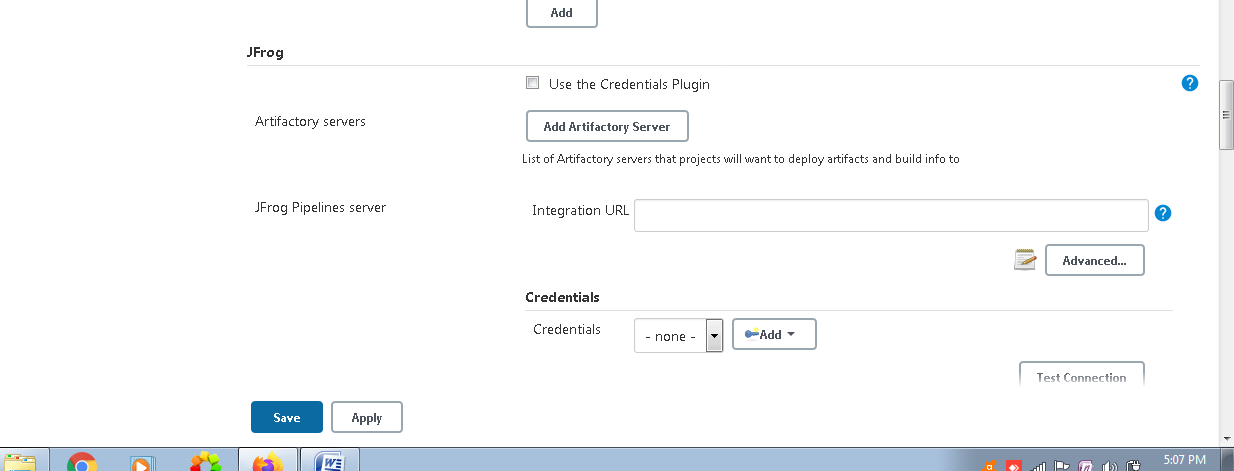
Then go to available tab and search artifactory ,select artifactory and Install without restart.



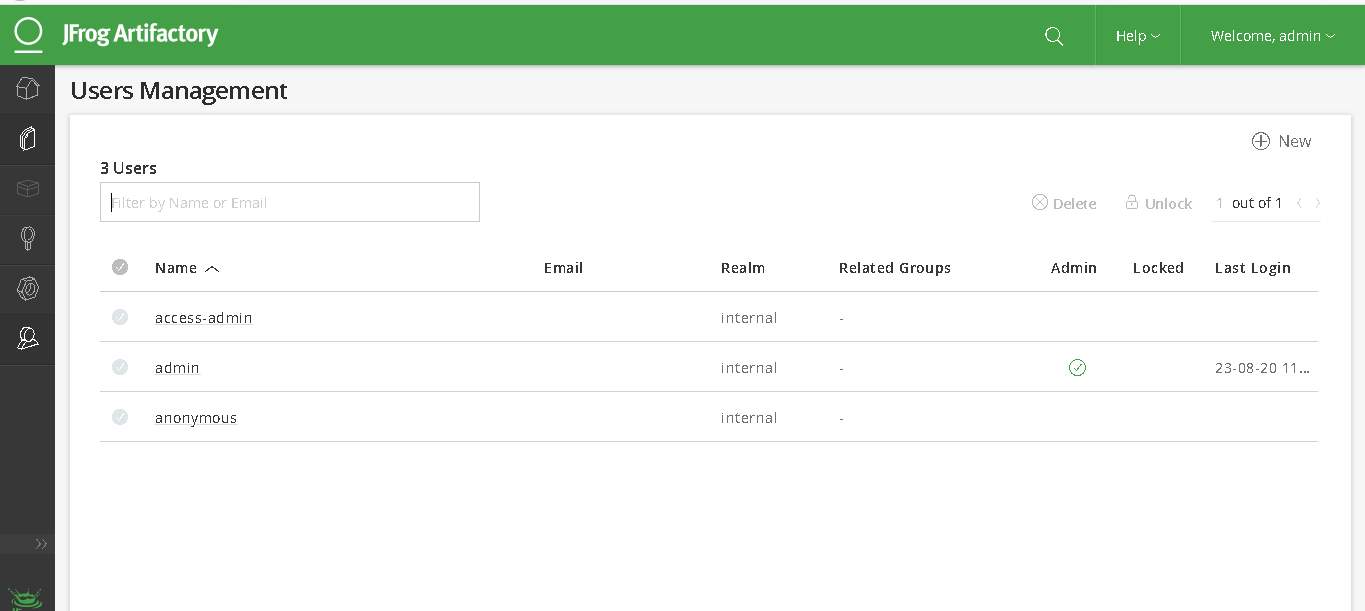
We could see the plugins are installed successfully.

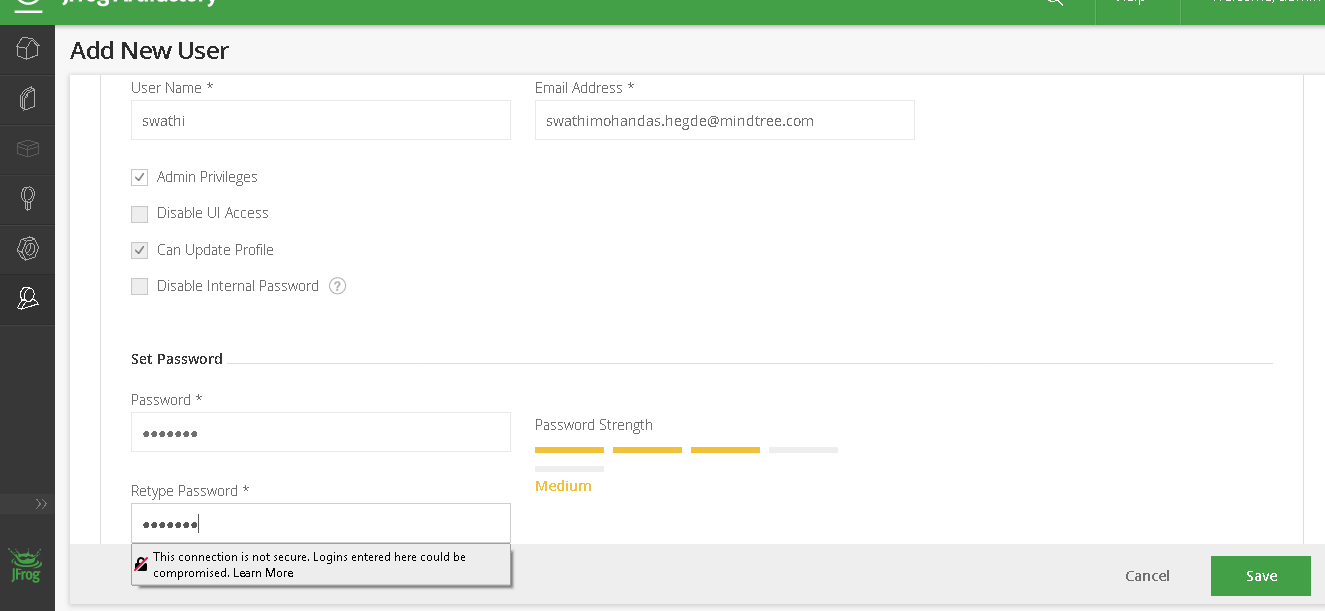
Now we will go back then manage Jenkins and Configure system and there we will be setting up our artifactory server details



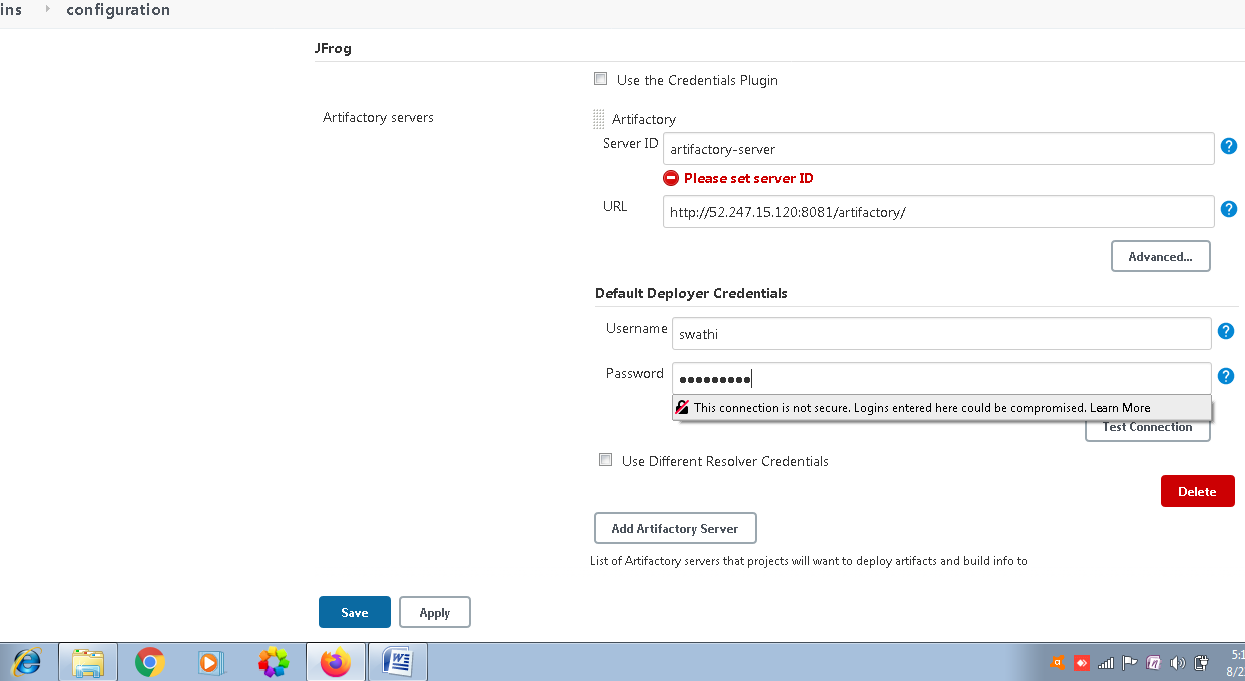


We will be creating 1 new user in jfrog atrifactory server and then we will provide the same credentials in the above,as providing the admin credentials to the Jenkins server is a bad practice.

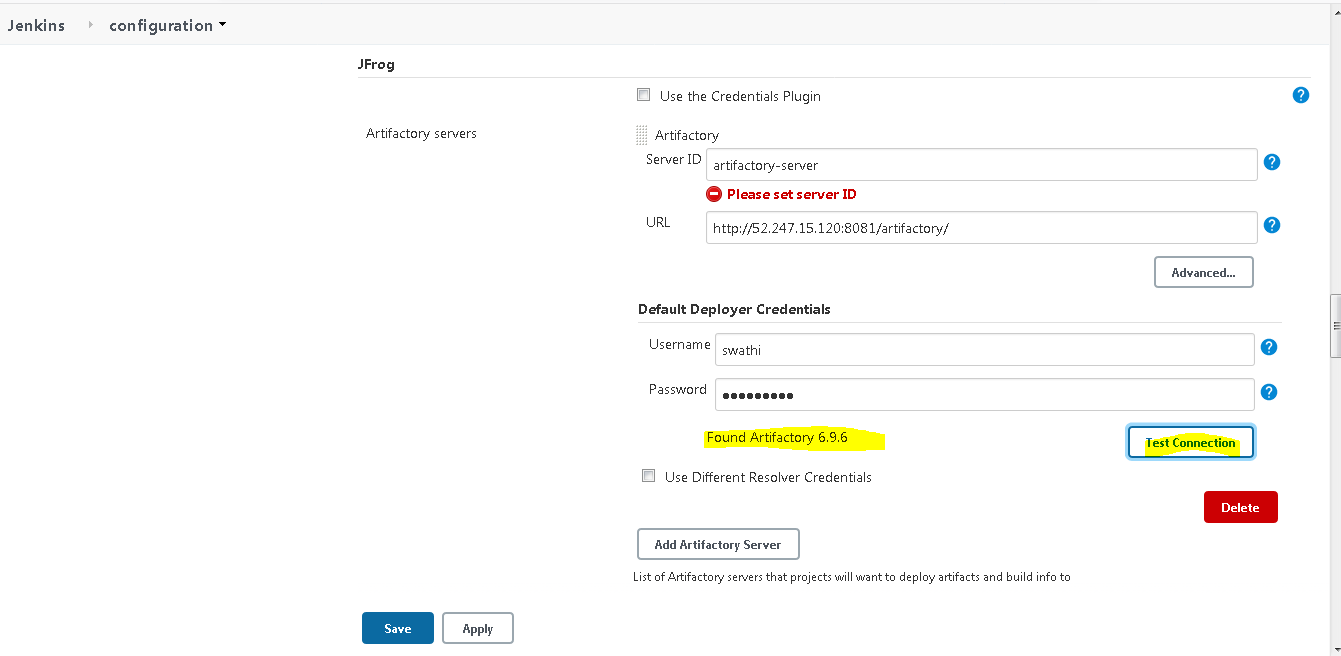




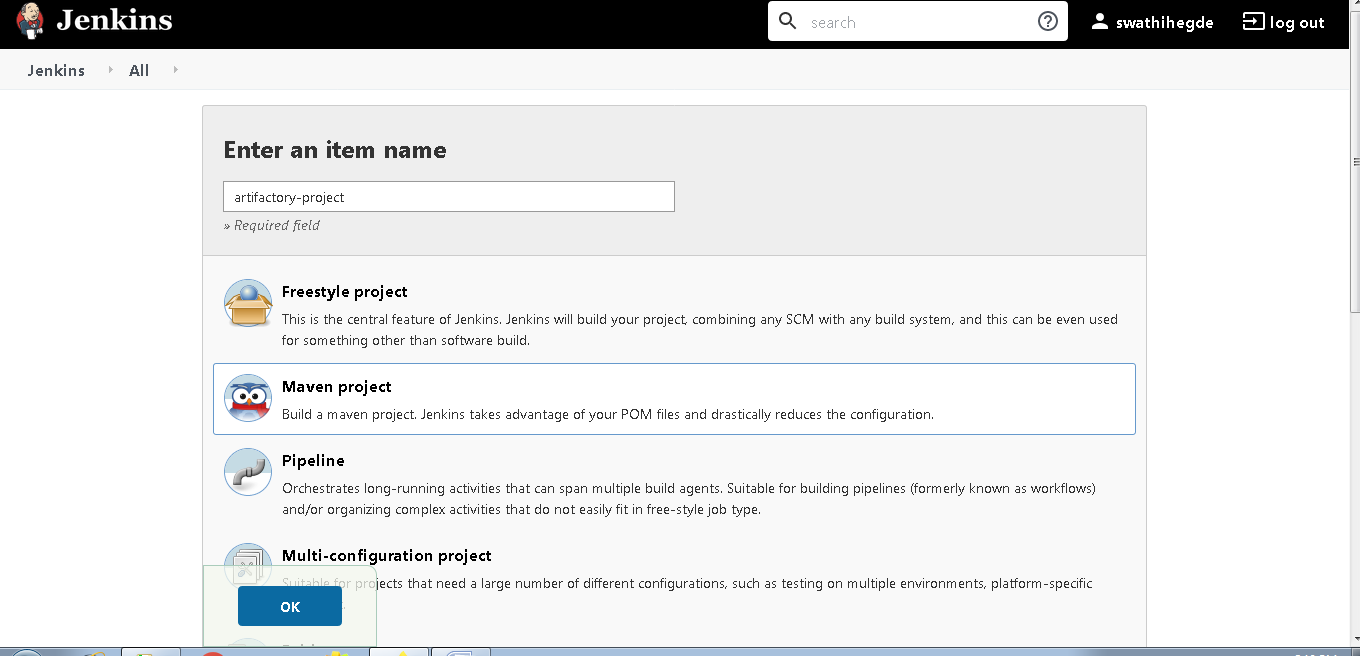
After that provide those credentials in the Jenkins

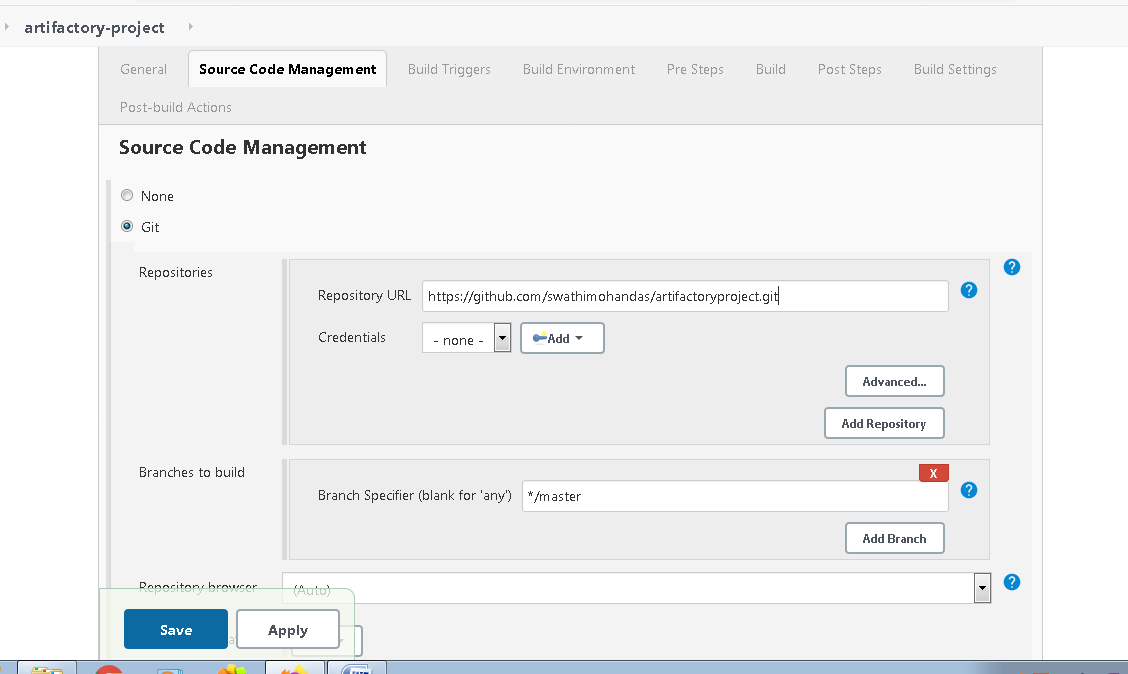


After clicking on test conncetion we get the output as follows



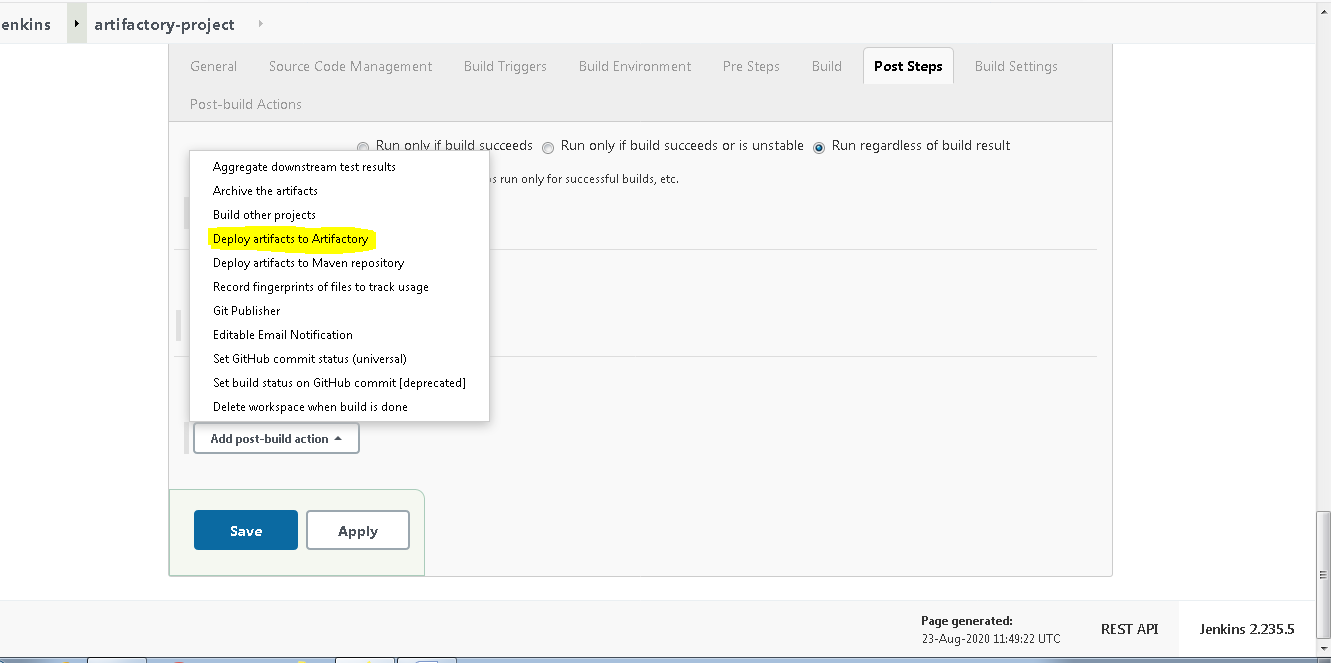
Now we will be creating 1 maven project as follows

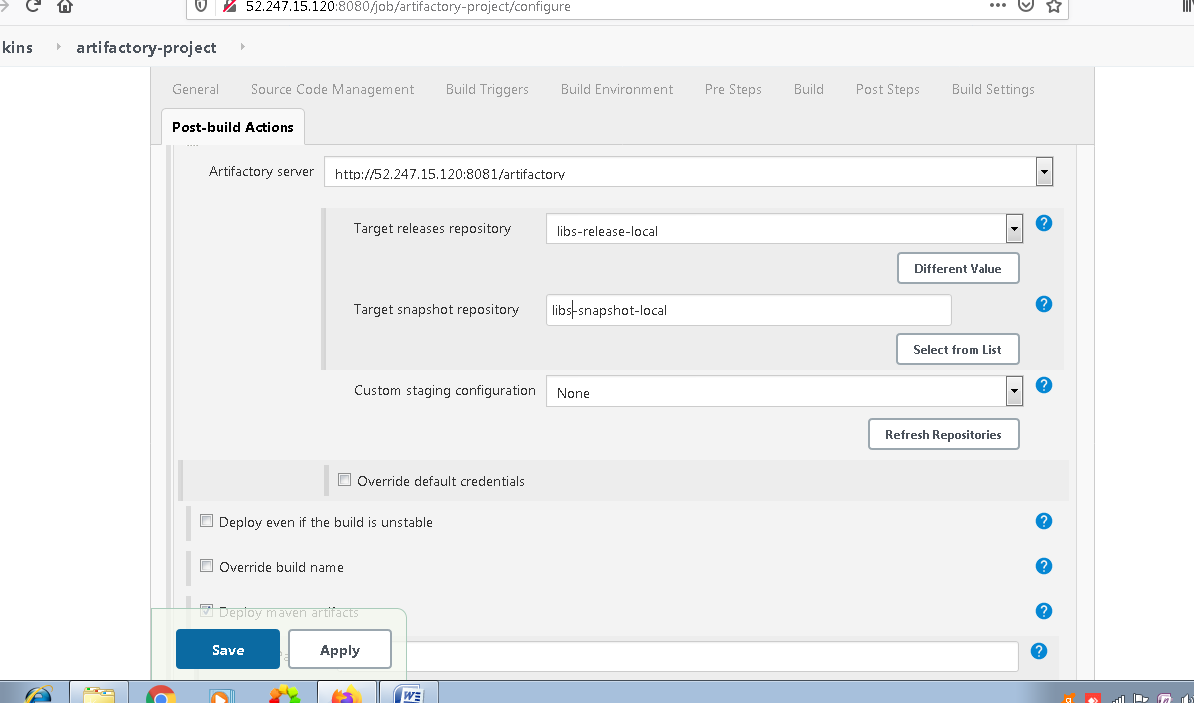






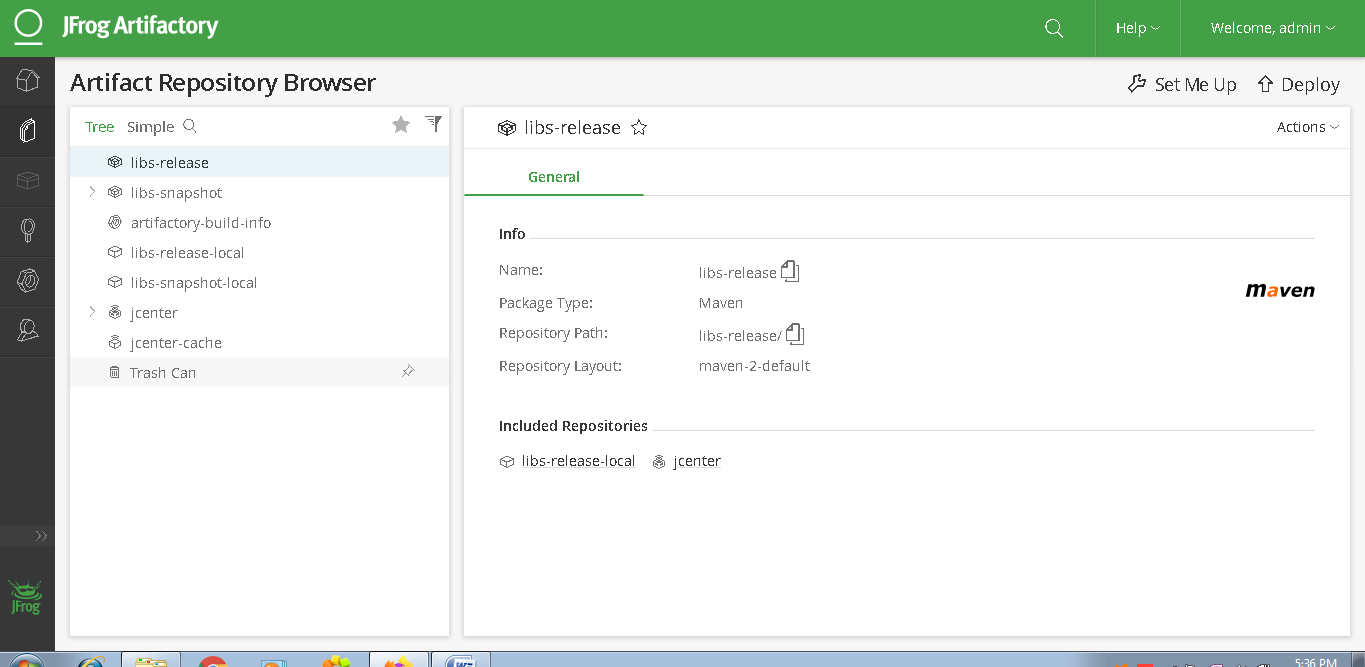
Then post build



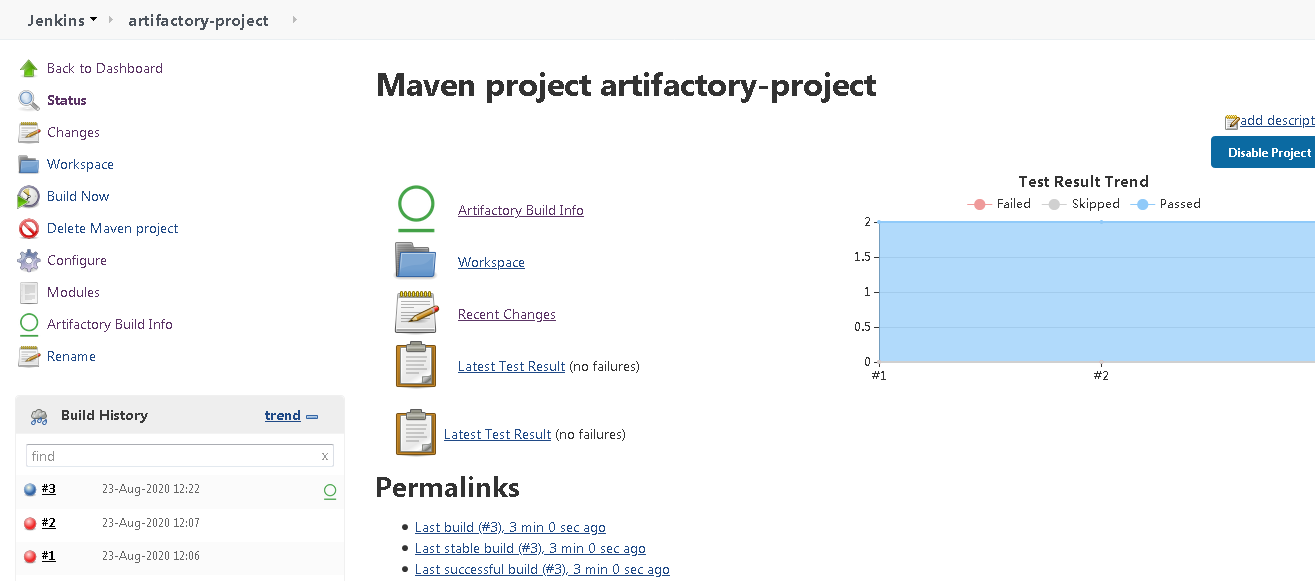


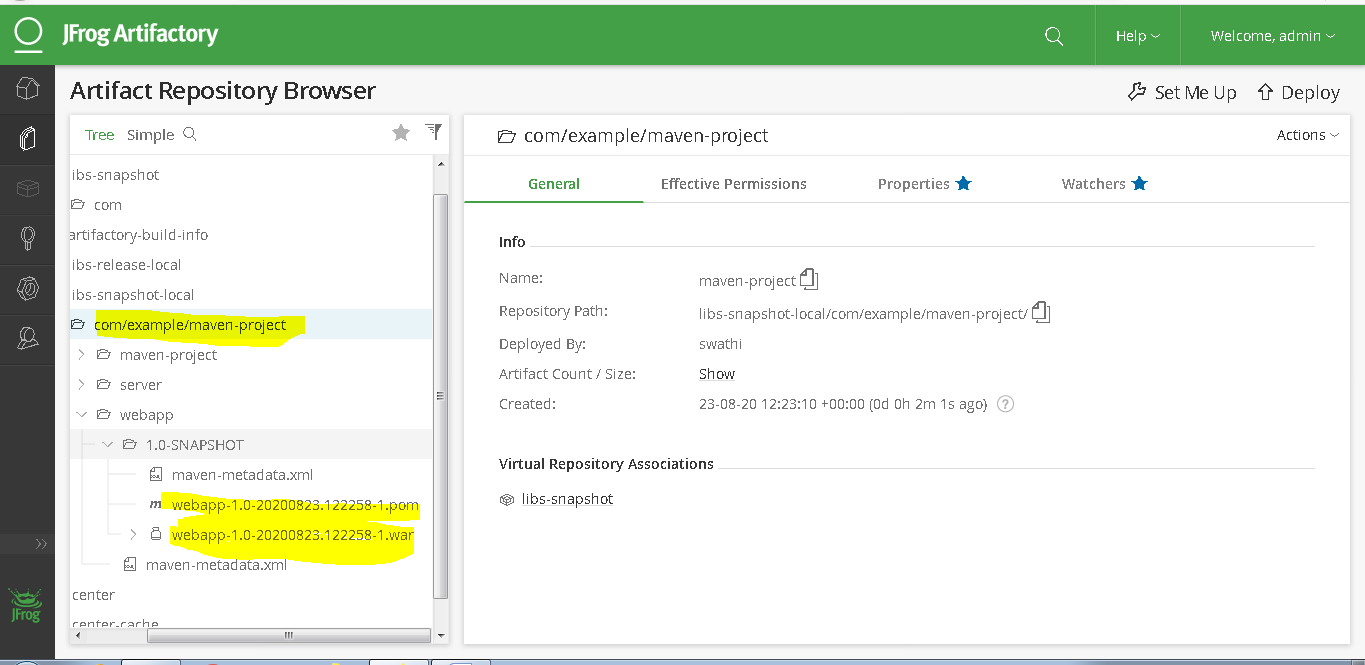
These are by be default setups

Before build you could see that there is no artfactory in the repository in jfrog server.



Now we will be building our project as follows

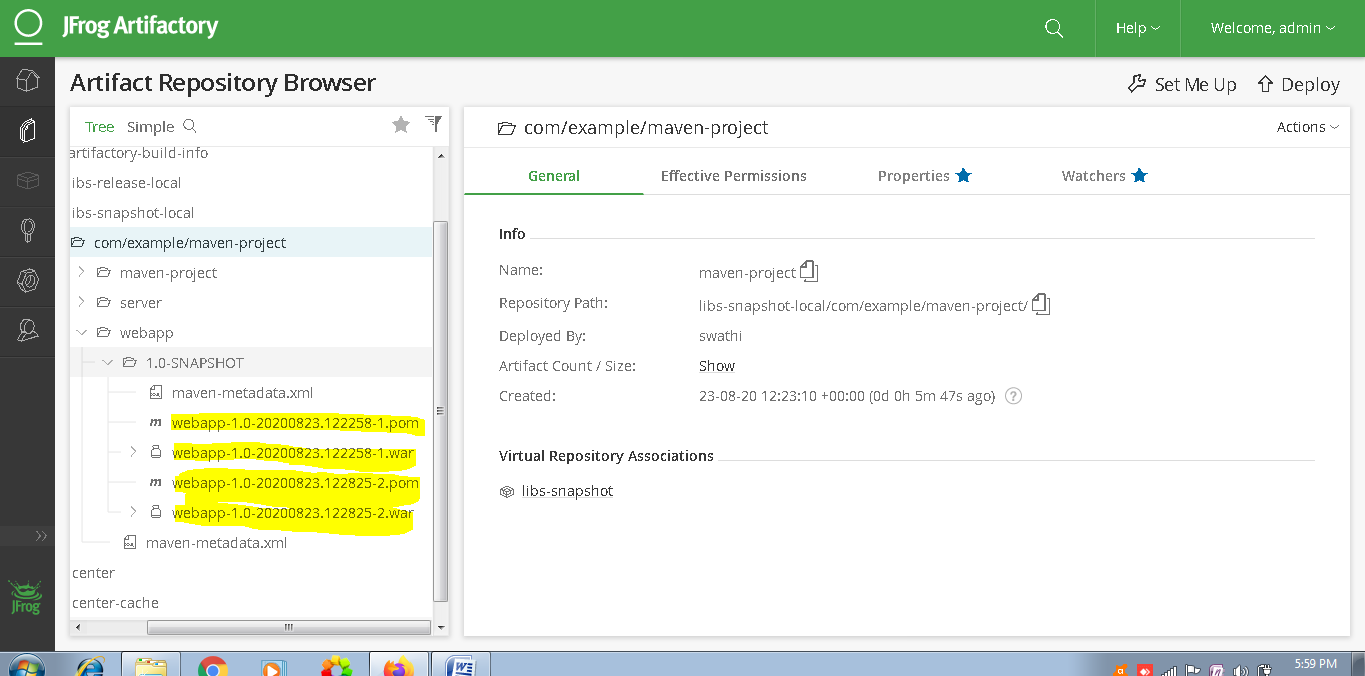


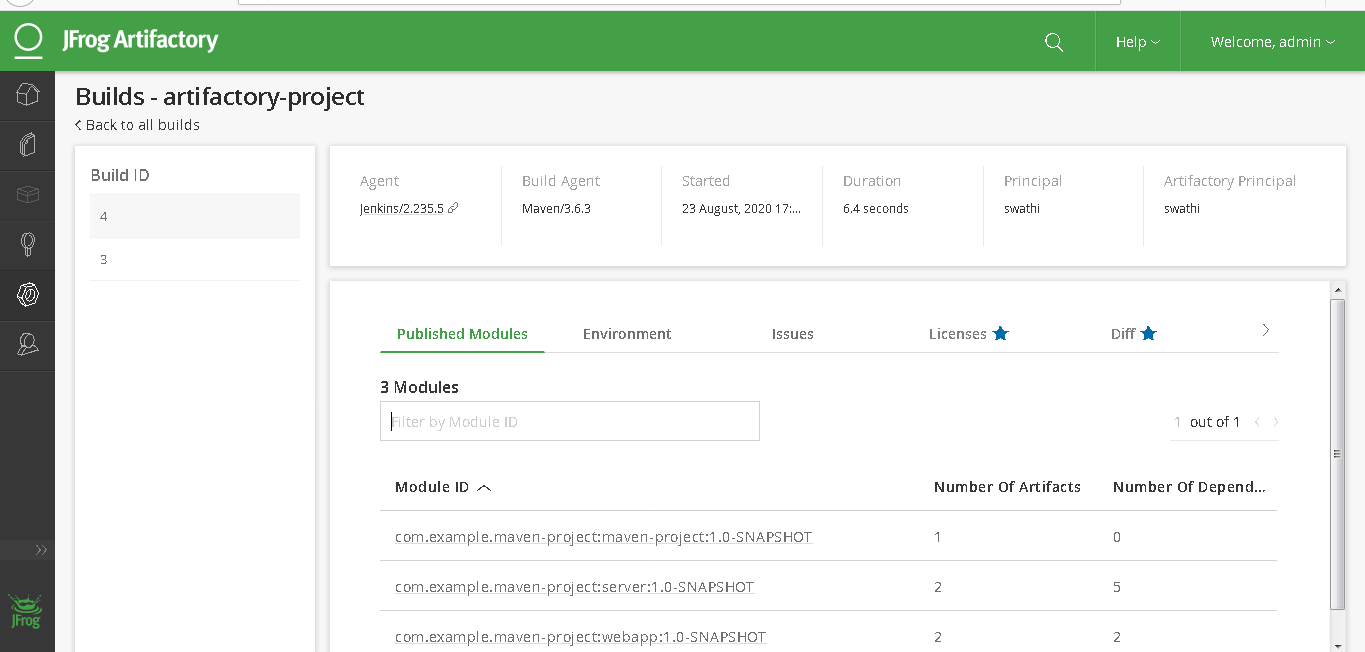


So we could see the snapshots above

Now we will build the project again and check in jfrog server

And we could see 2 versions as follows



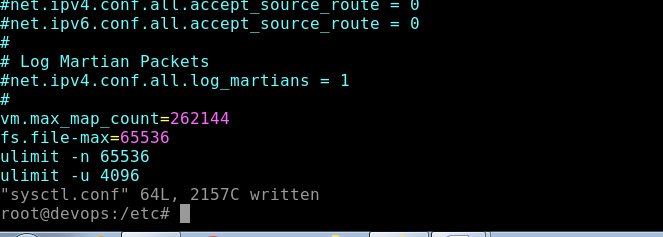


Now we will be setting up sonarqube server in our VM and will be integrating with Jenkins

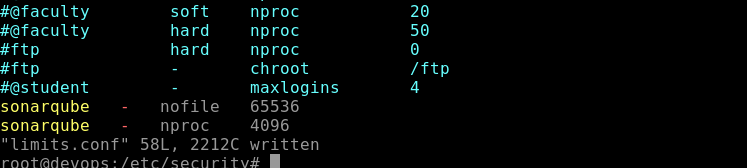
Sonarqube basically comes with 2 components

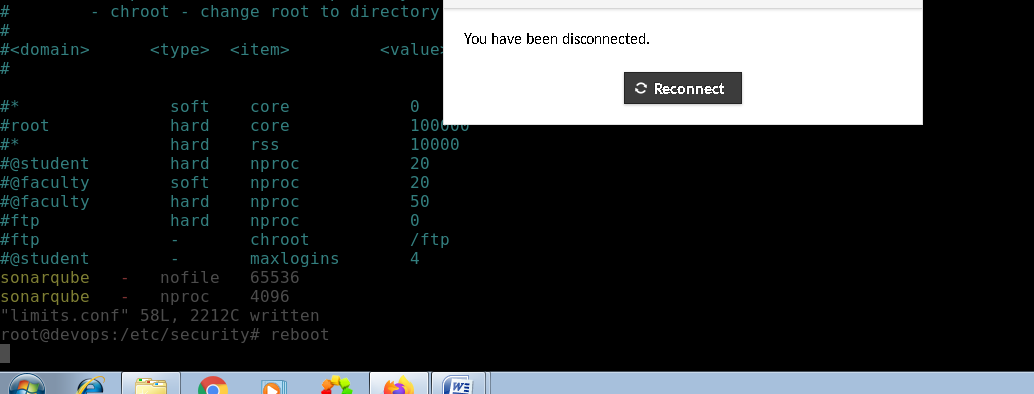
Sonarqube Installation

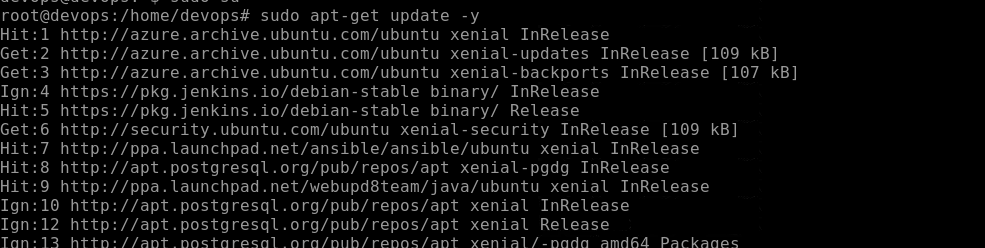
First edit sysctl.conf in /etc folder as follows

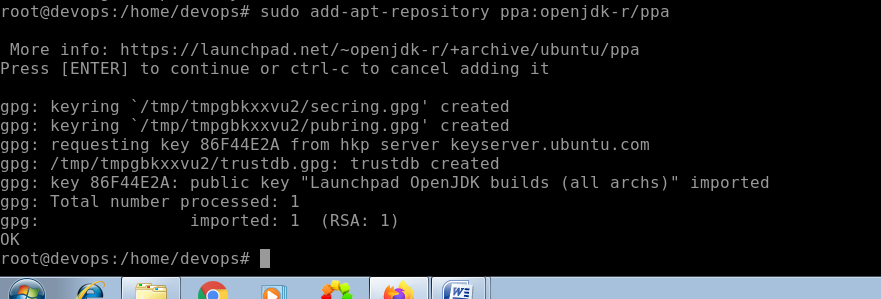


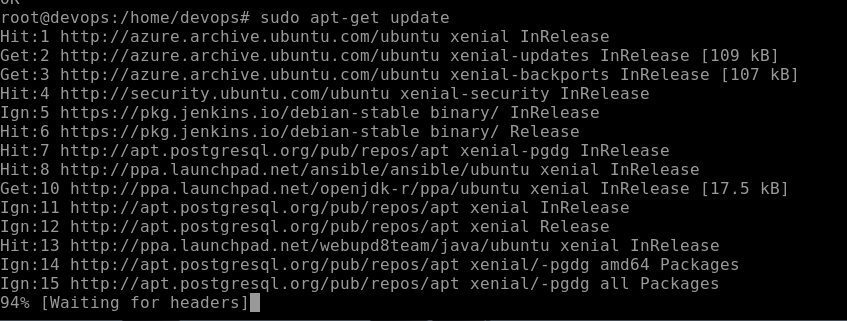
Edit /etc/security/limits.conf

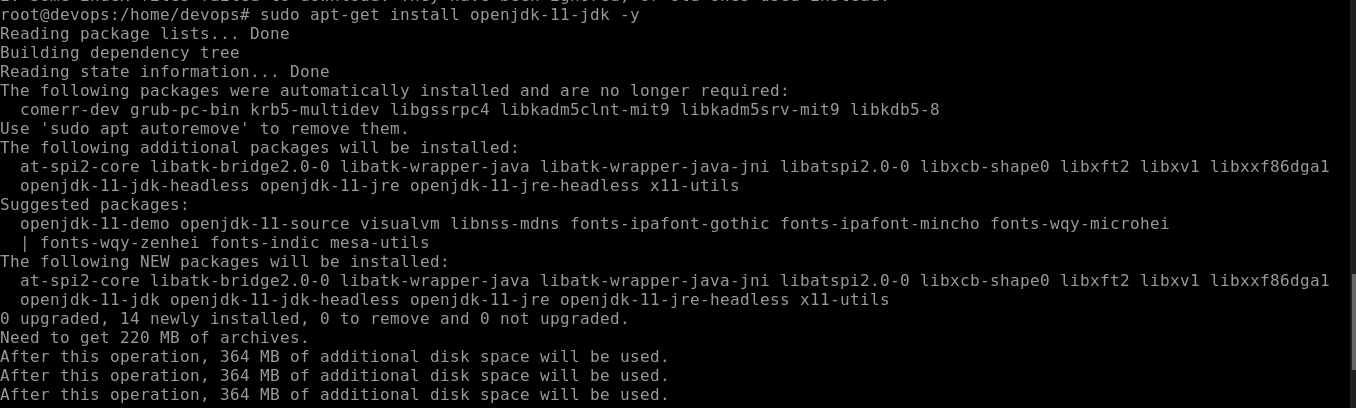


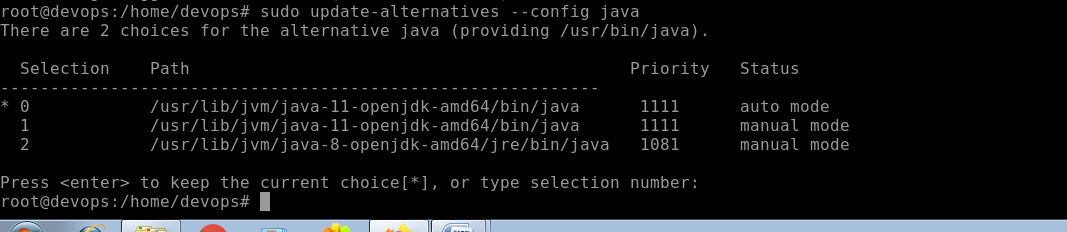


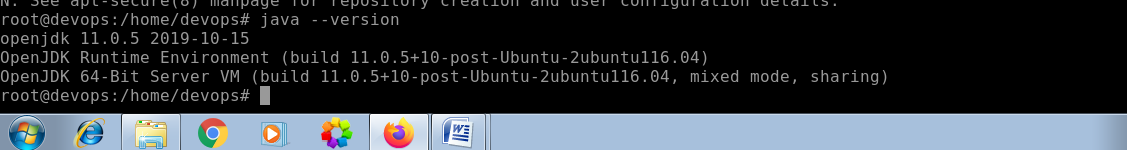


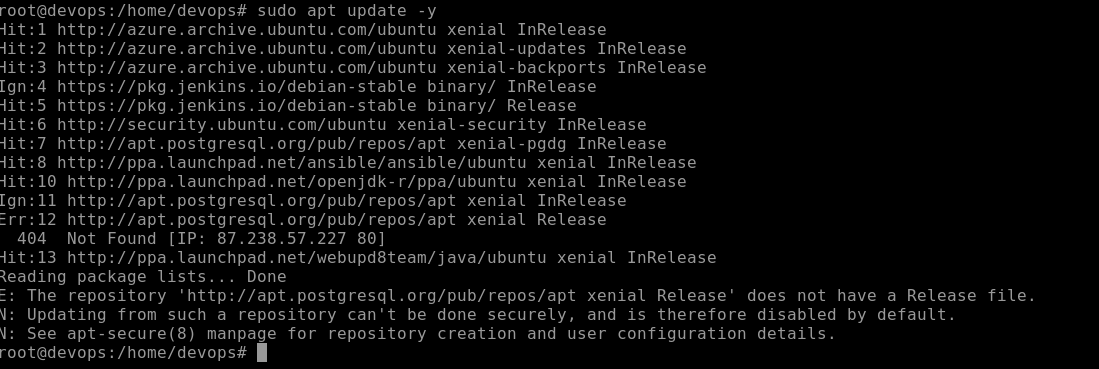


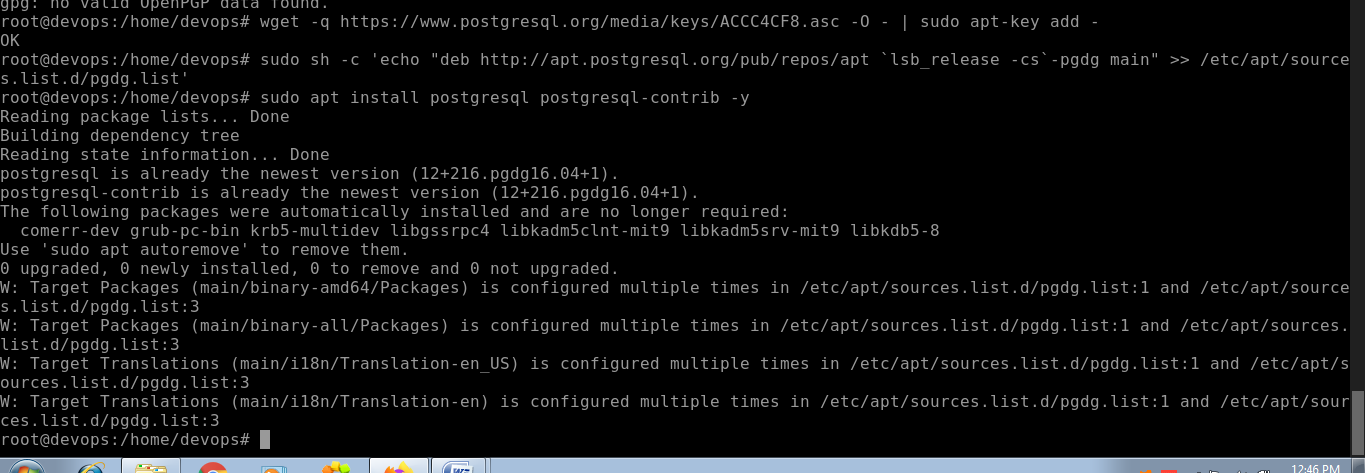


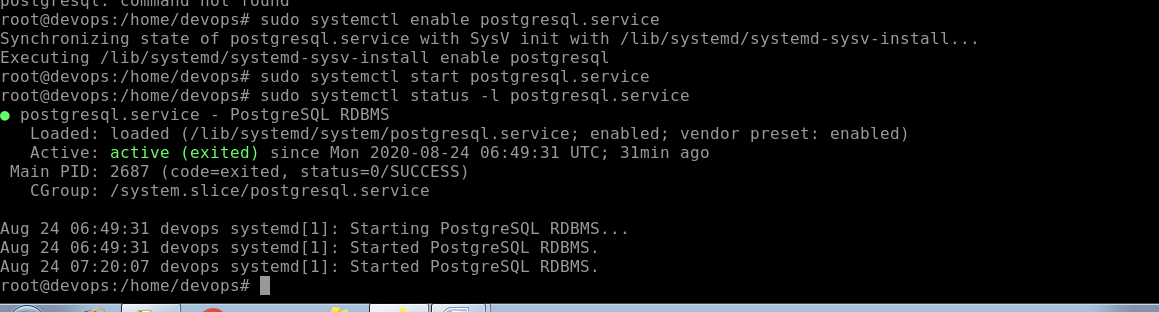


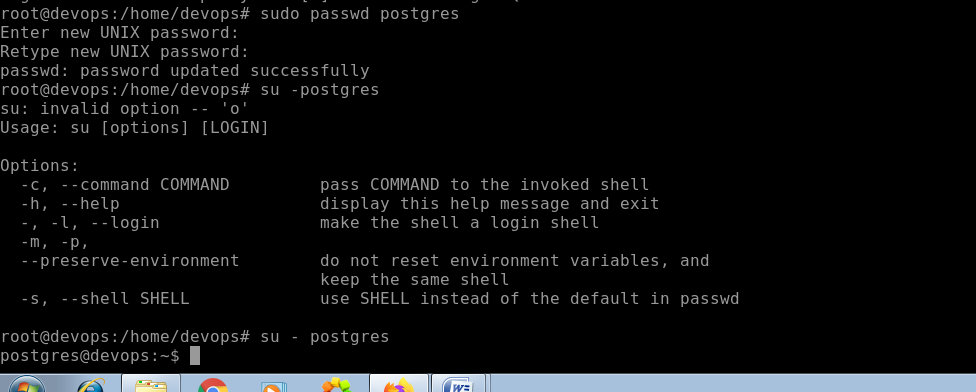




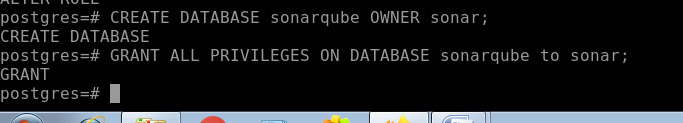




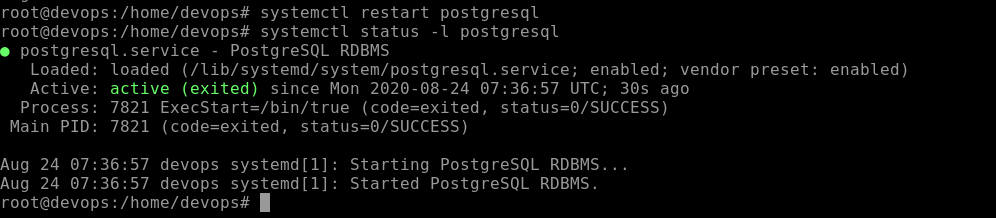


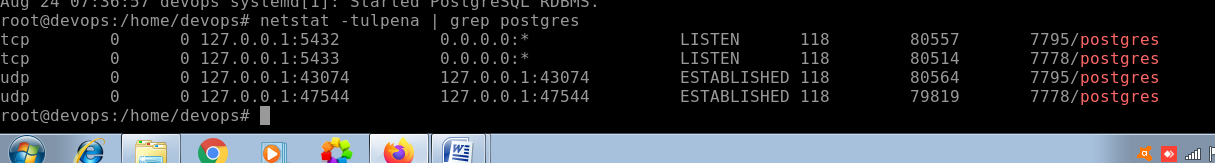


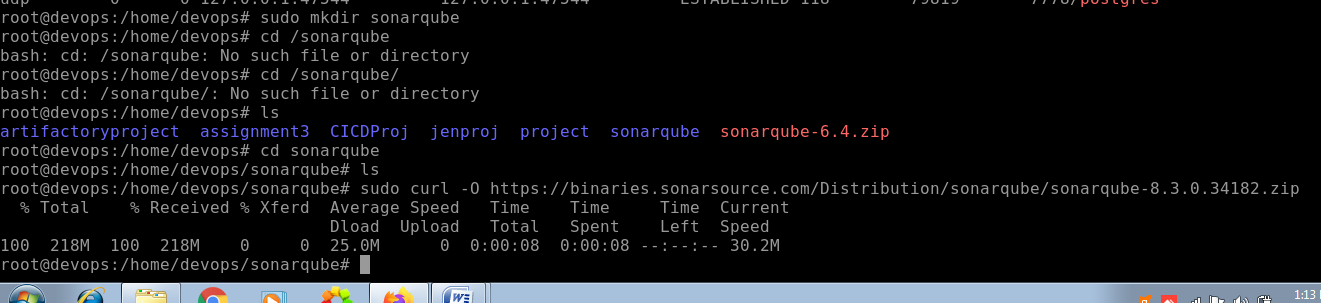




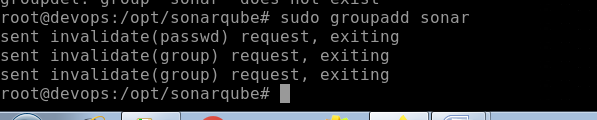


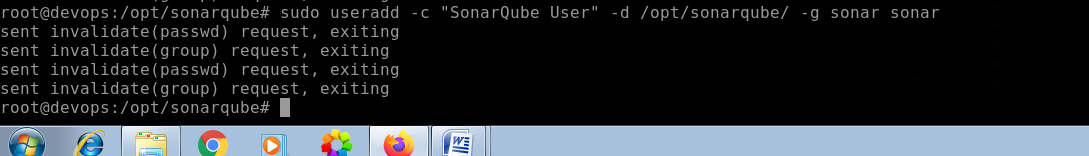






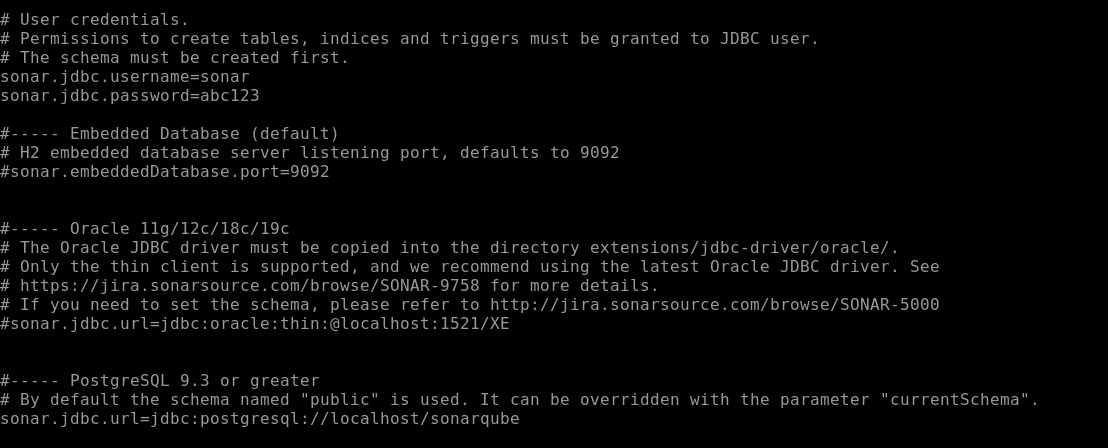


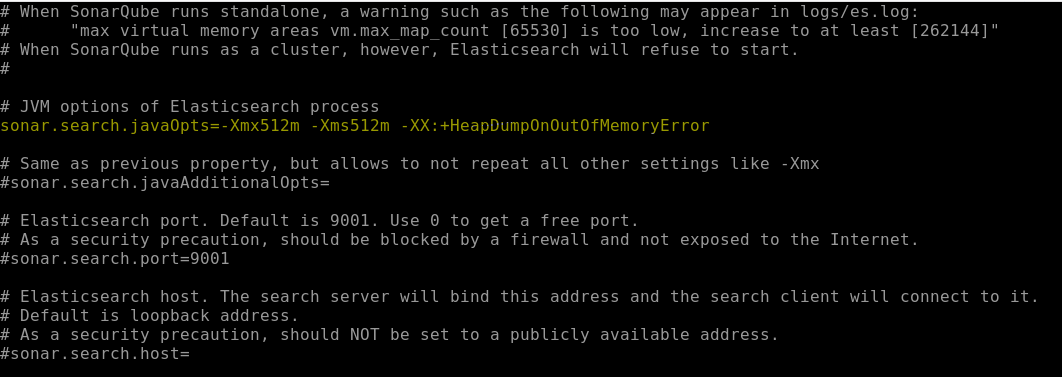




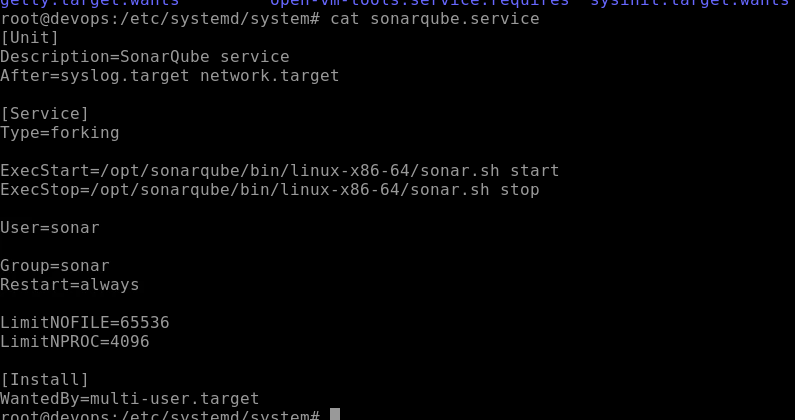


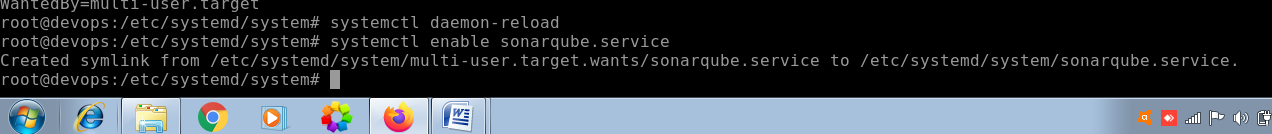
For sonar configuration,we have uncommented few things in sonar.properties file inside /opt/sonarqube/conf/ directory

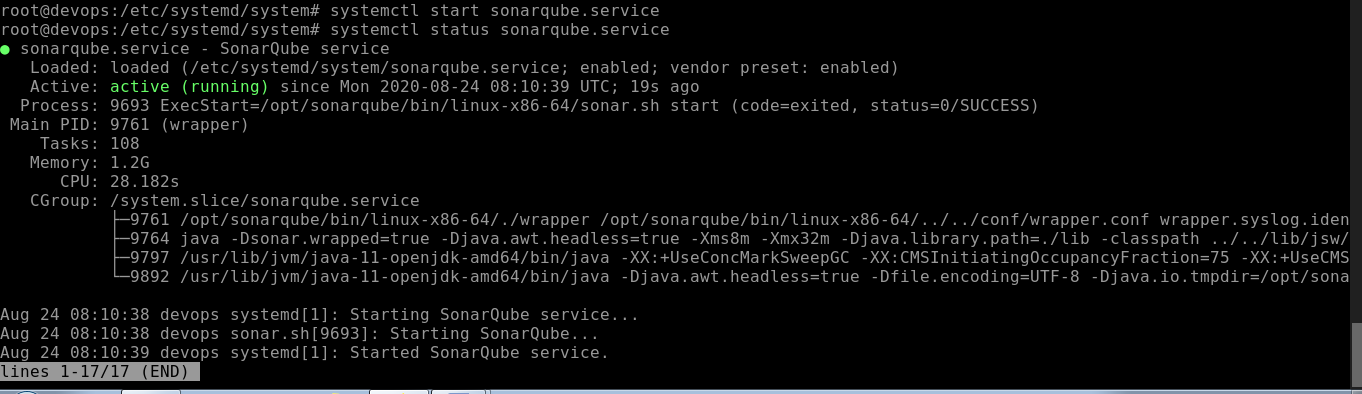


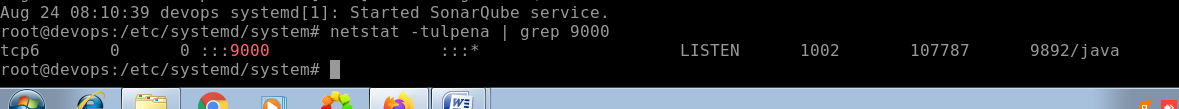


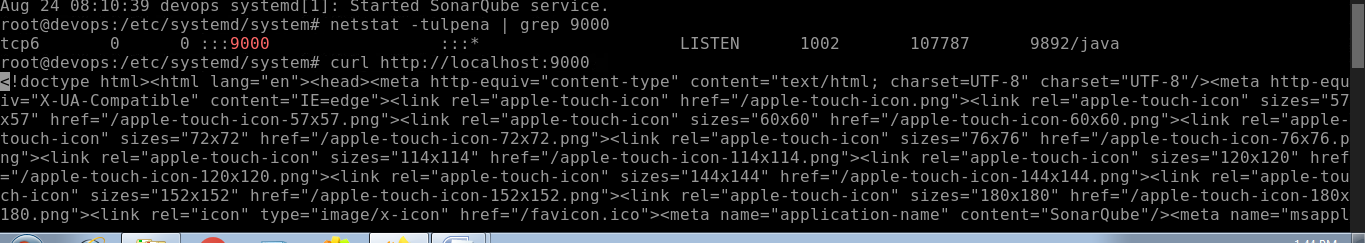
Then /etc/system/system

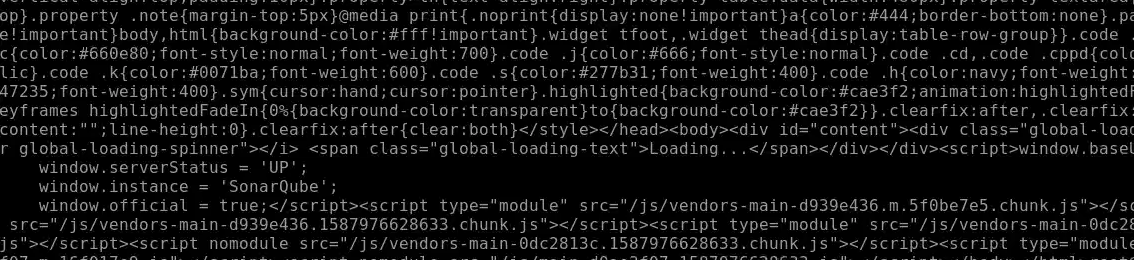






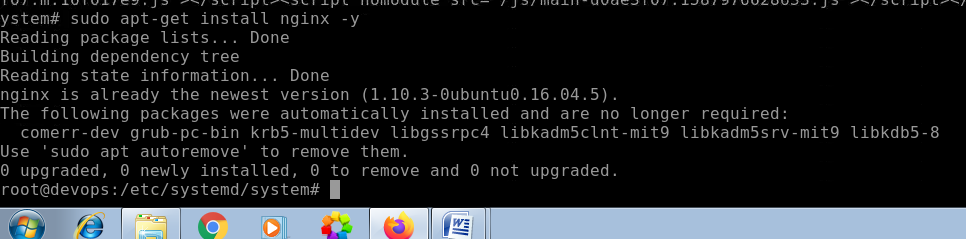






We could see Instance is running

To make the sonarqube accessible to outside world we will do the following

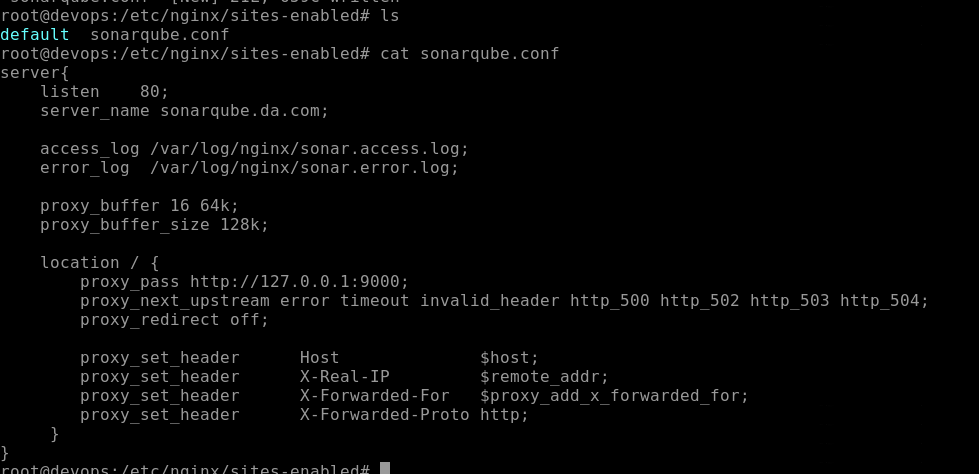


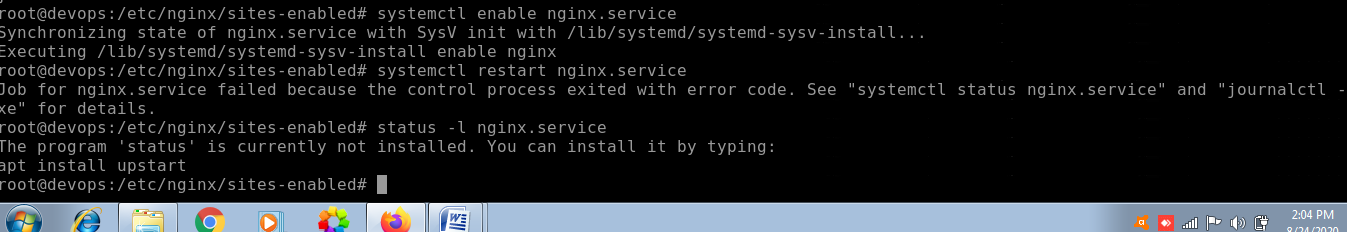
Check if these are enabled in /etc/nginx/nginx.conf

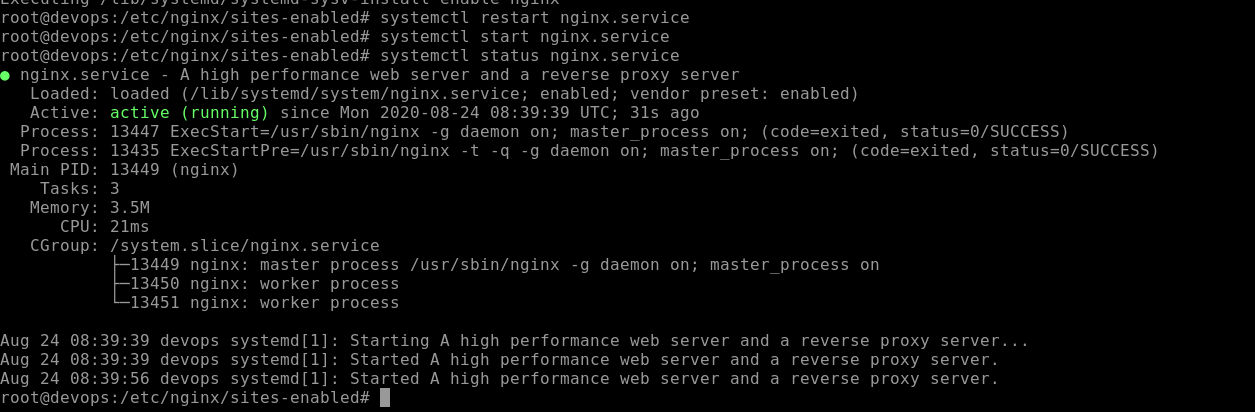


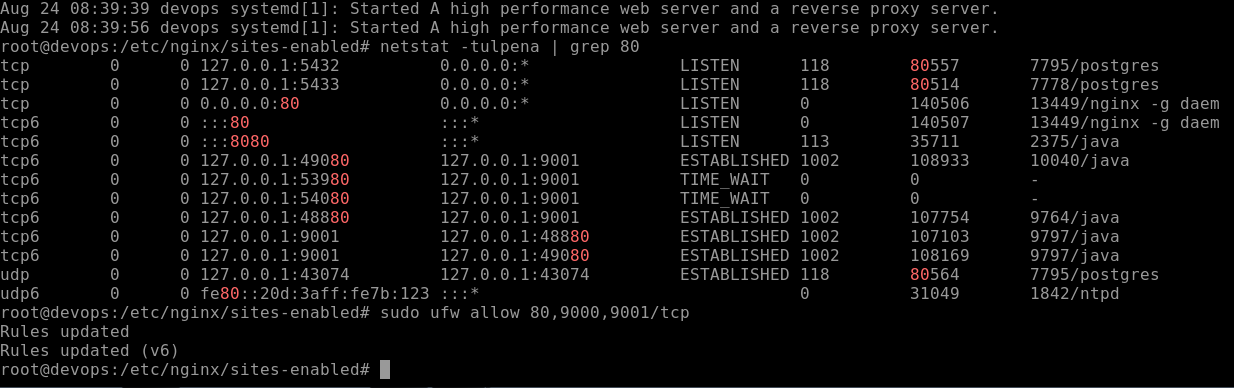
Now we will be creating virtual host

We will be creating a file under sites-enabled directory

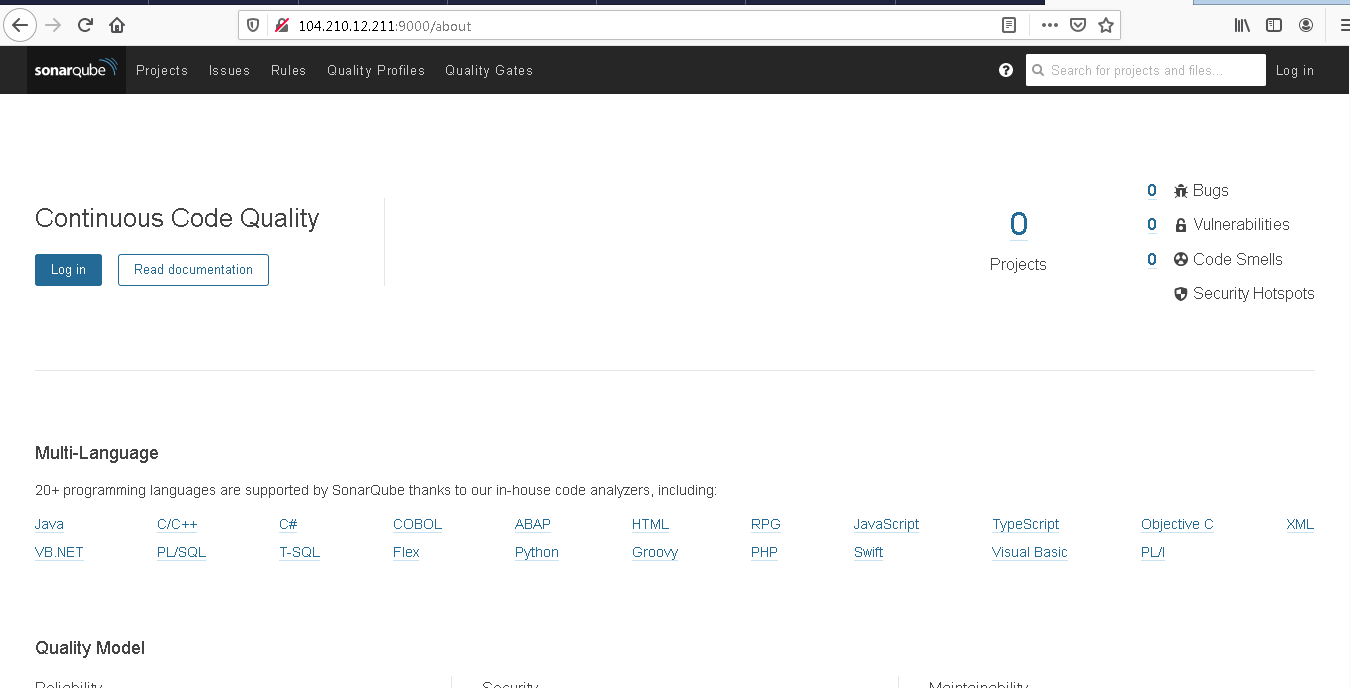


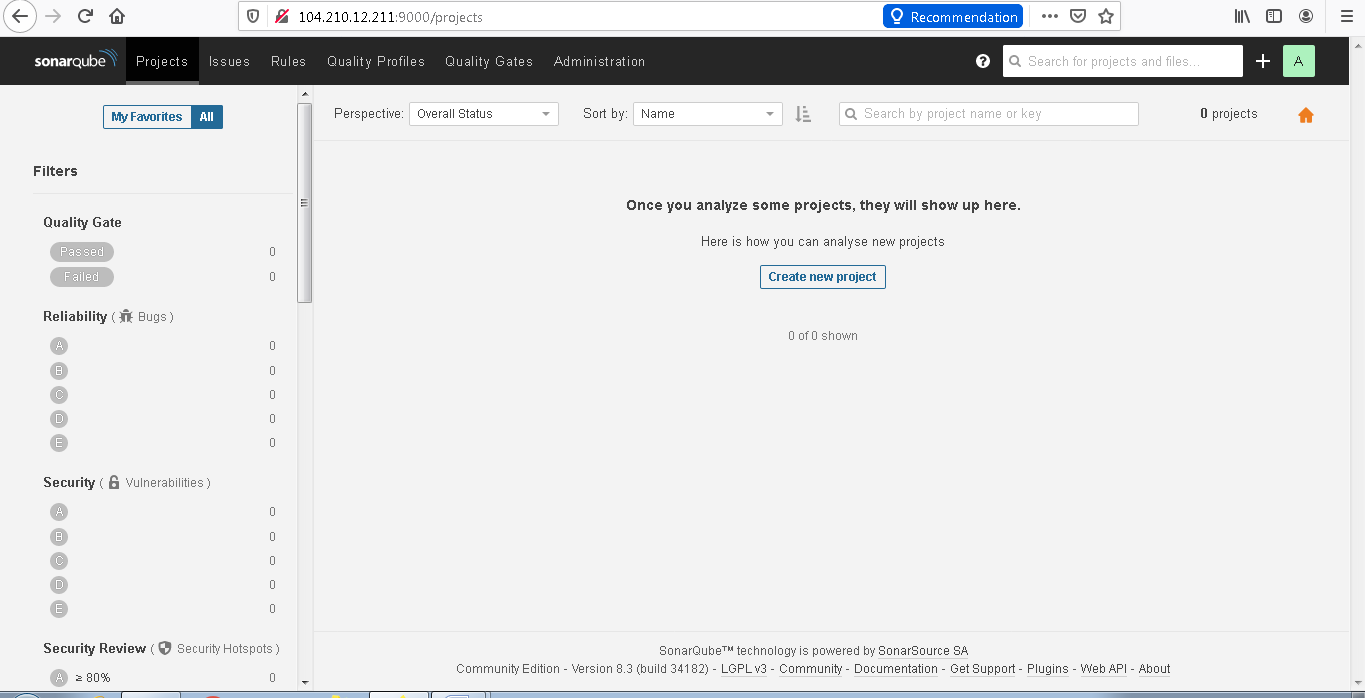




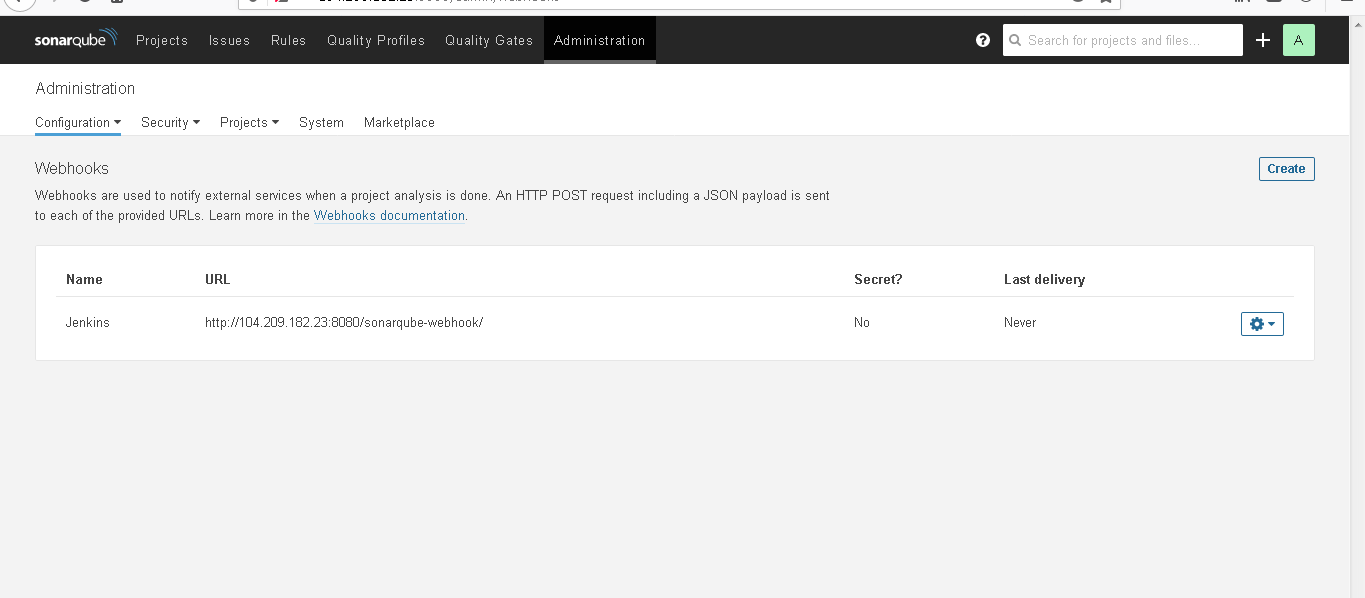


Now go to browser and check on port 9000



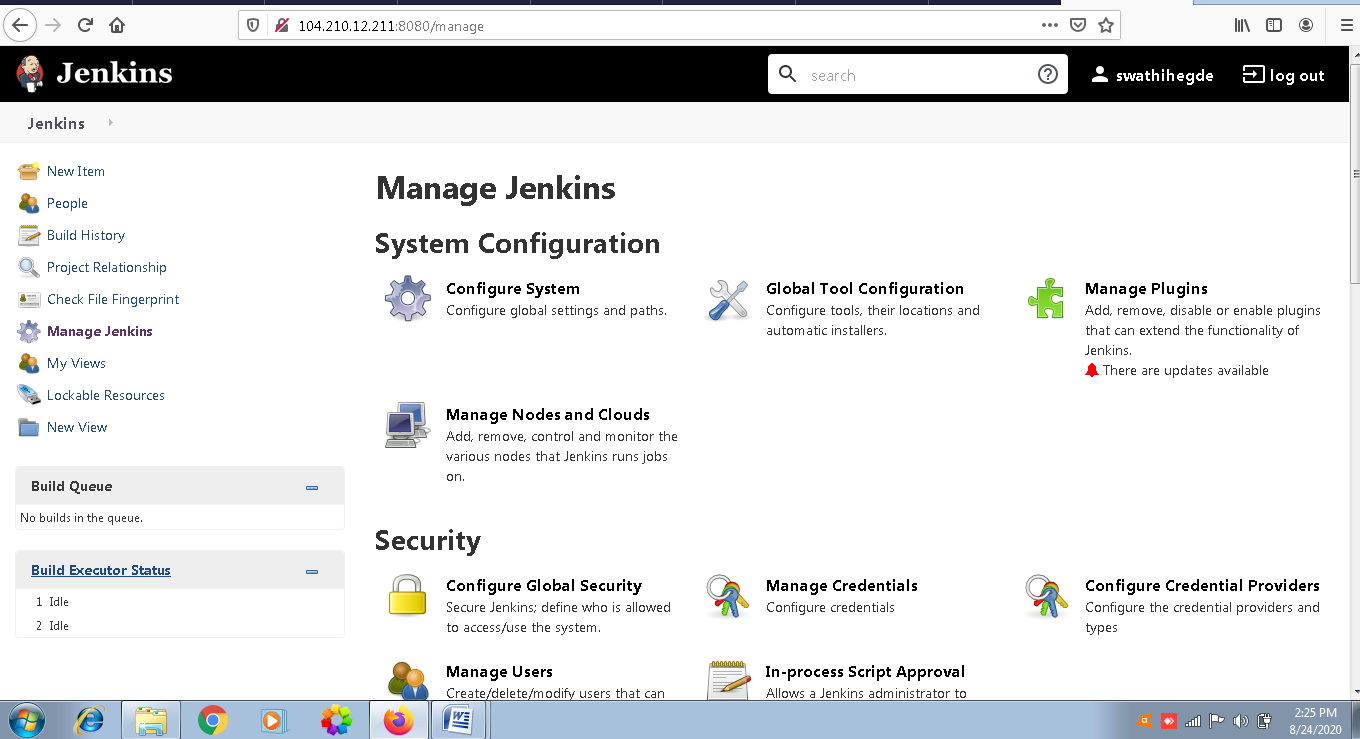


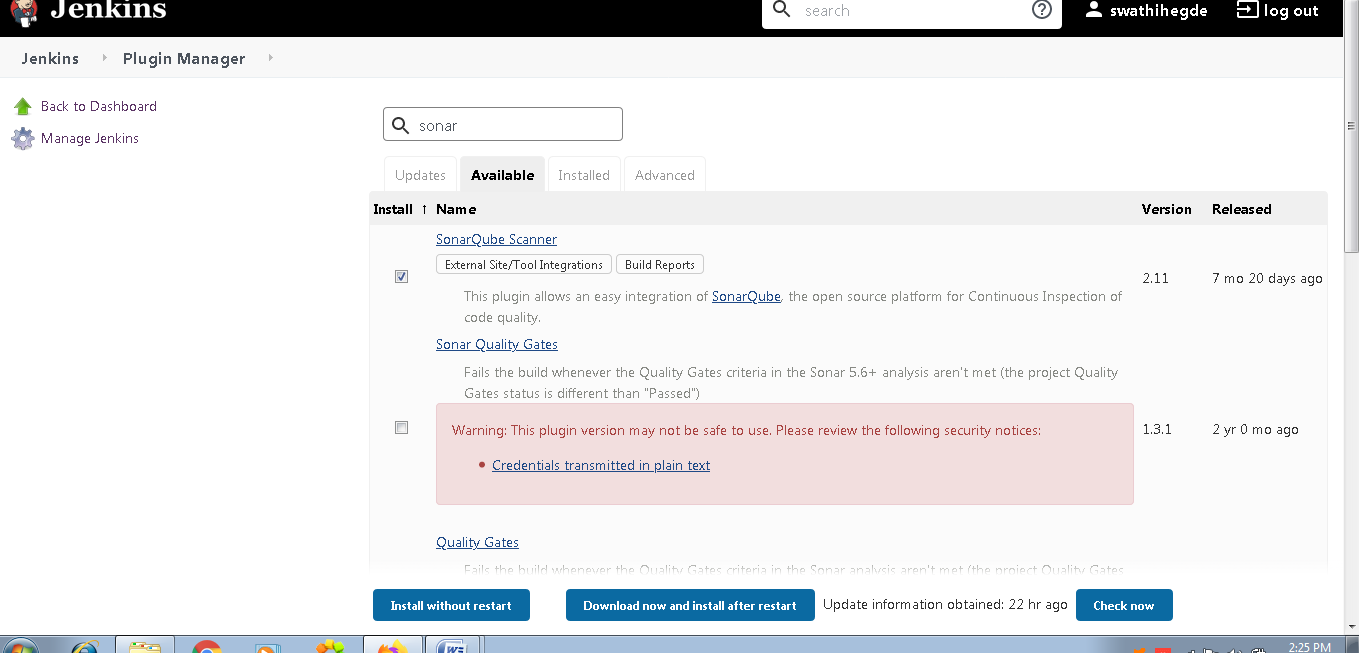




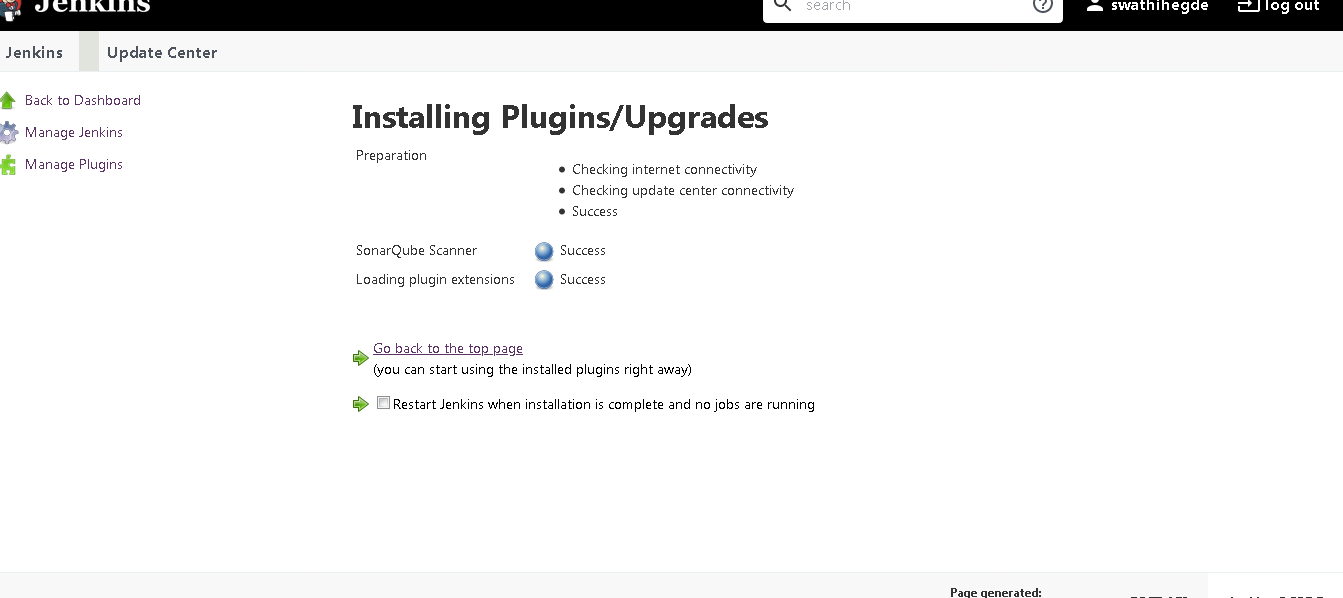
Now we will be integrating sonarqube with Jenkins

For that we need to install sonarqube plugins in Jenkins as follows



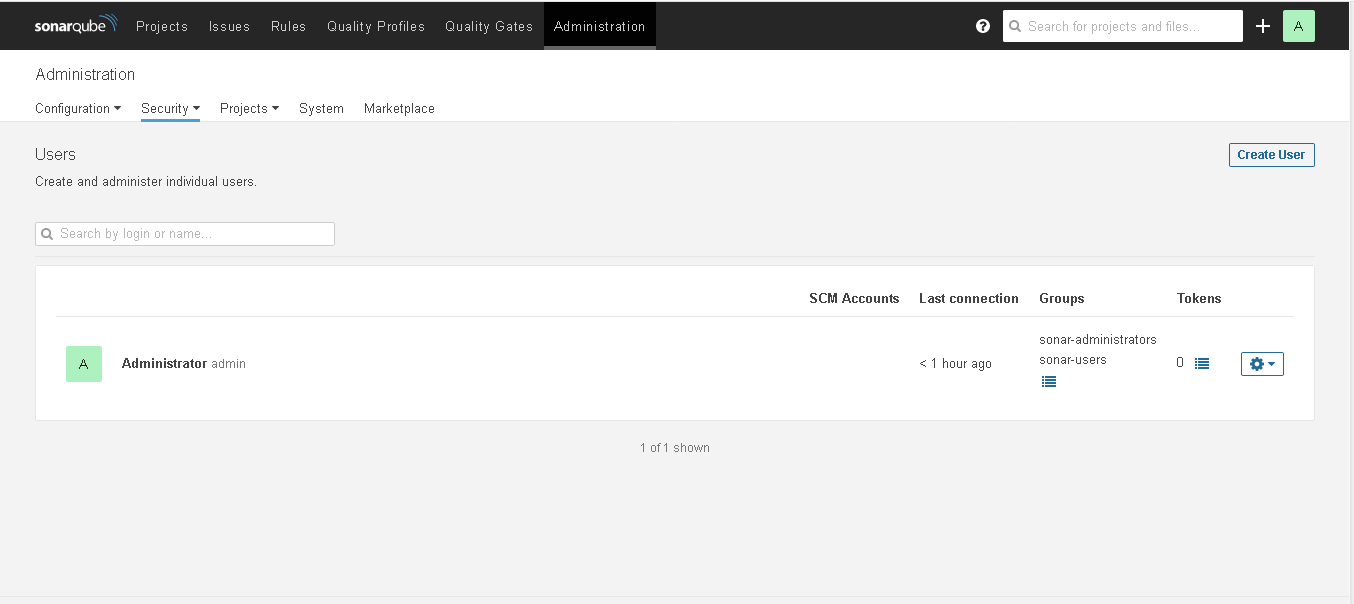


Click on install without restart

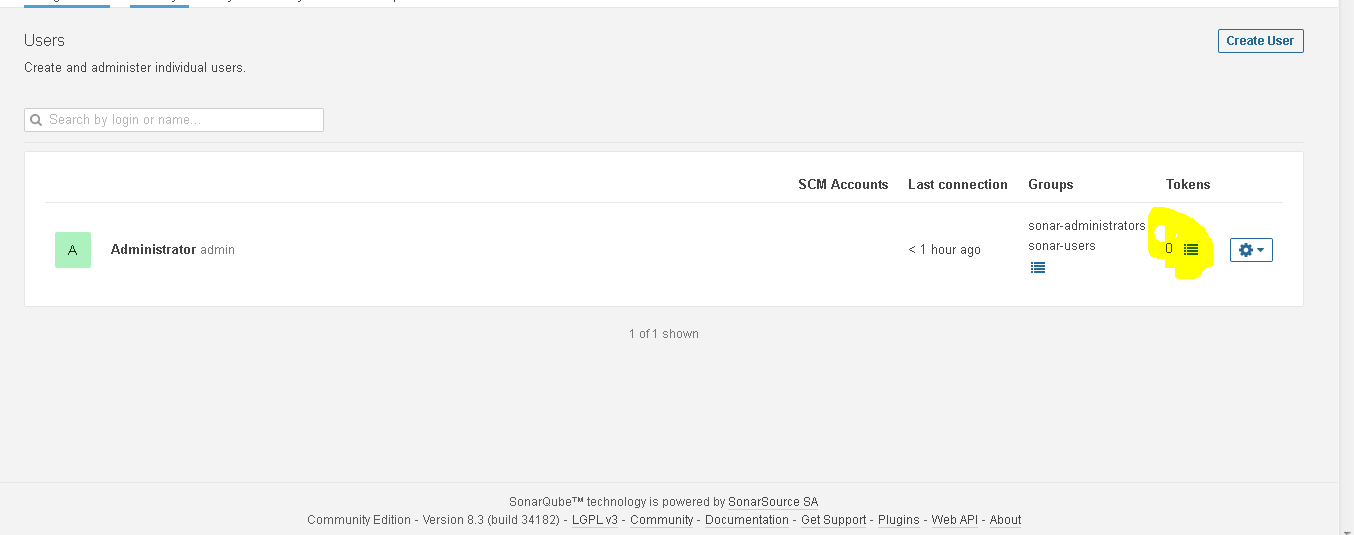


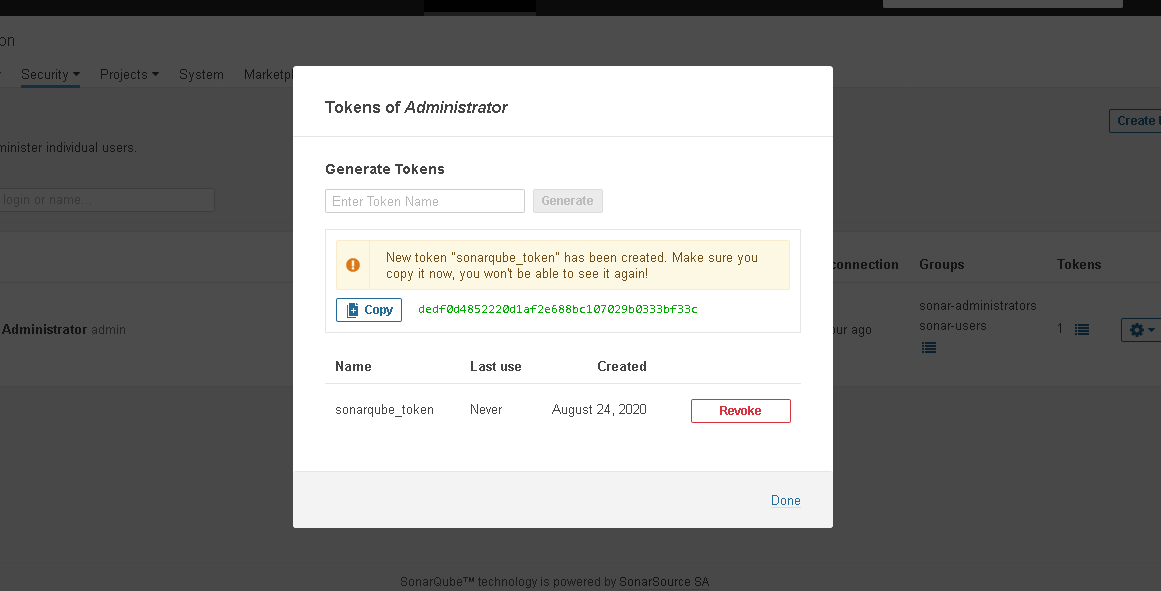
Then go to sonarqube site

Then click on administration tab



Now we will be creating a token





Now we could see the token generated

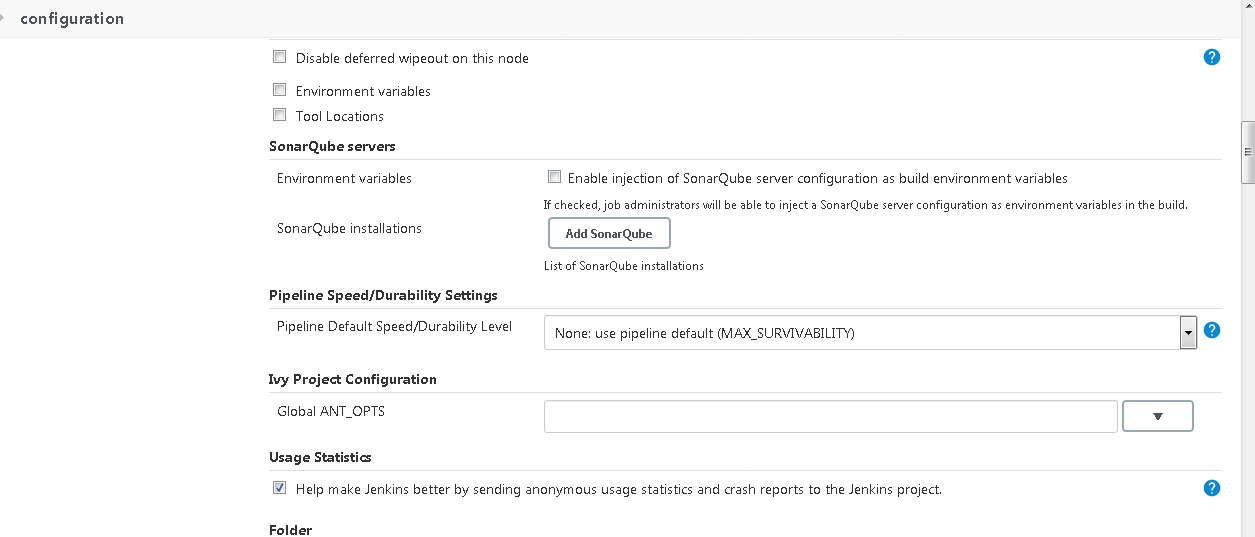
dedf0d4852220d1af2e688bc107029b0333bf33c

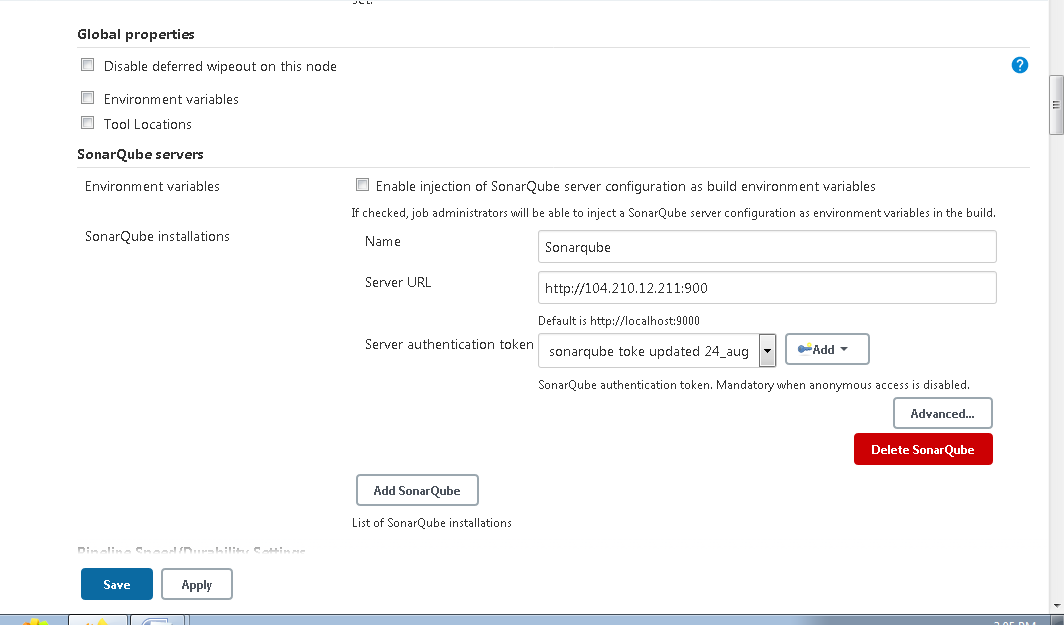
Then copy that token and then go to Jenkins page

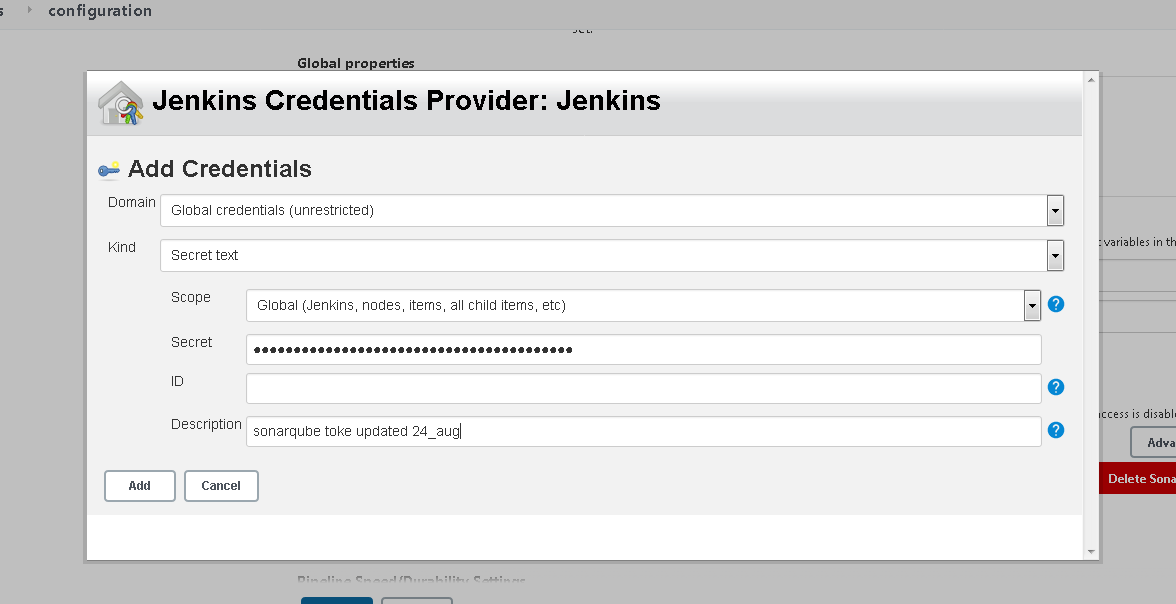
Manage Jenkins 🡪 Configure Jenkins

Go to sonarqube section

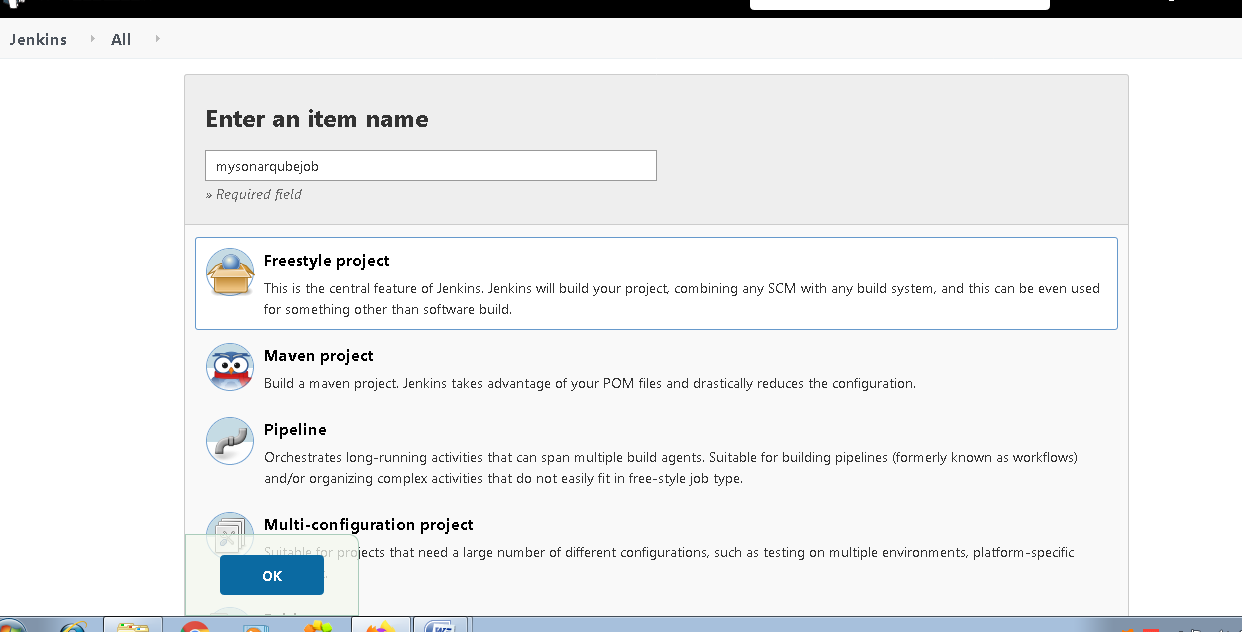
Then add sonarqube

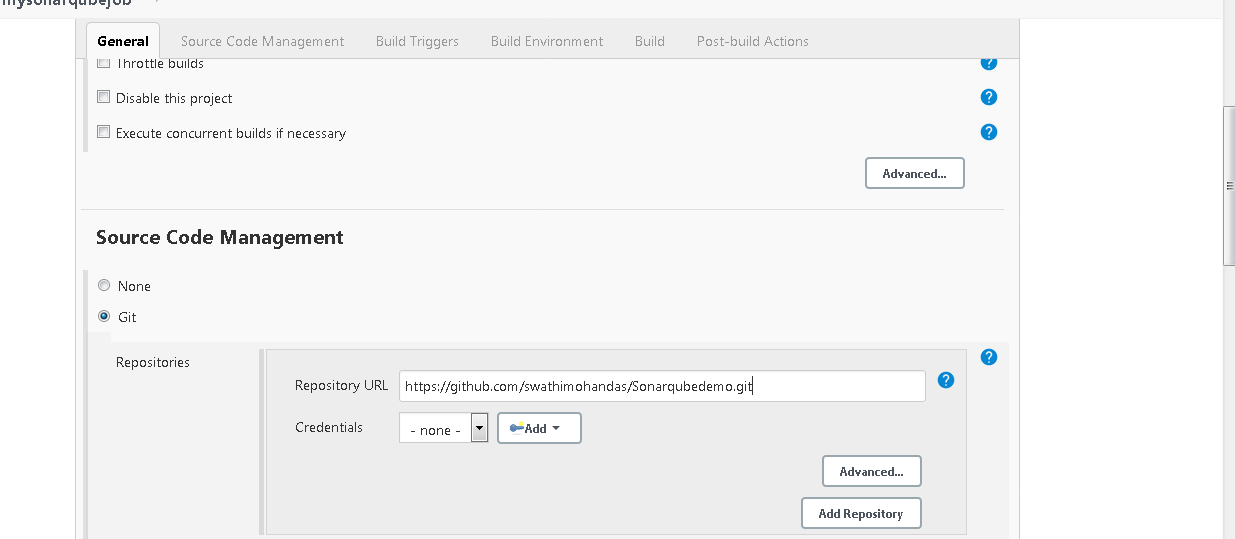




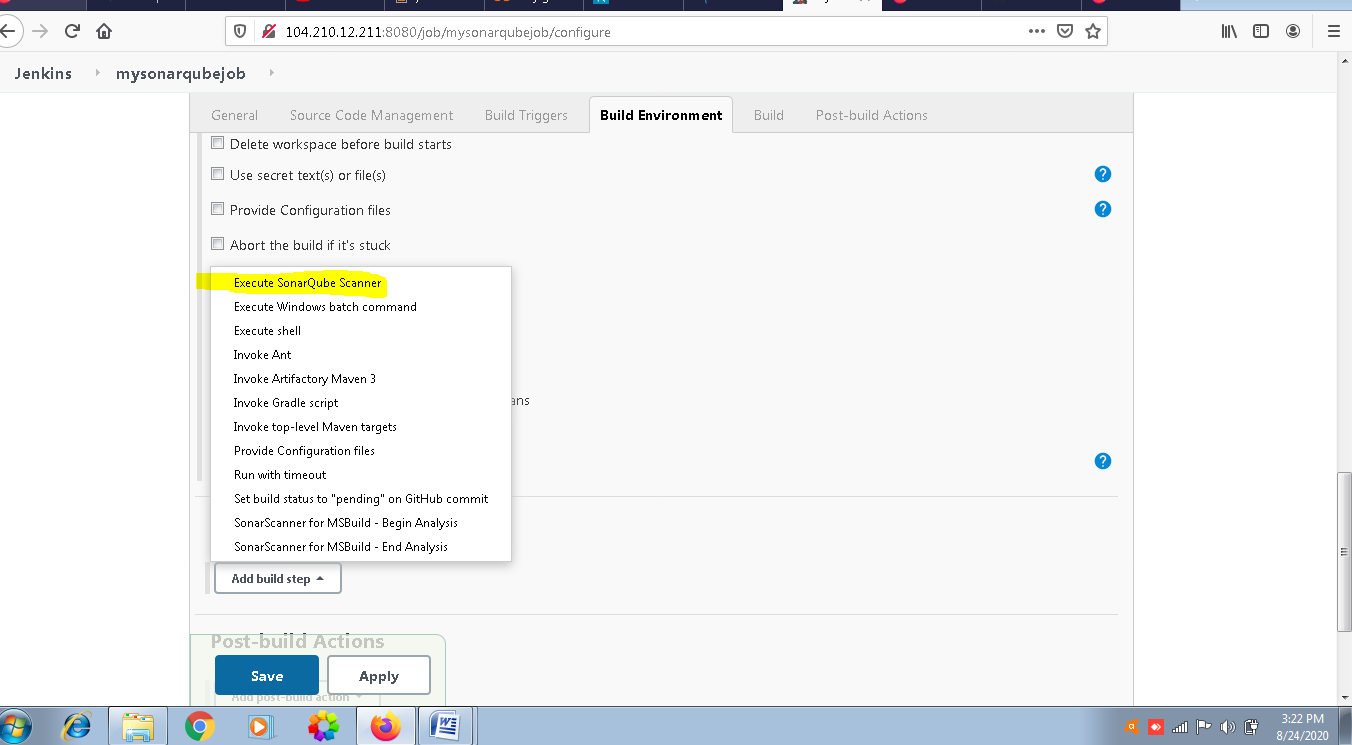


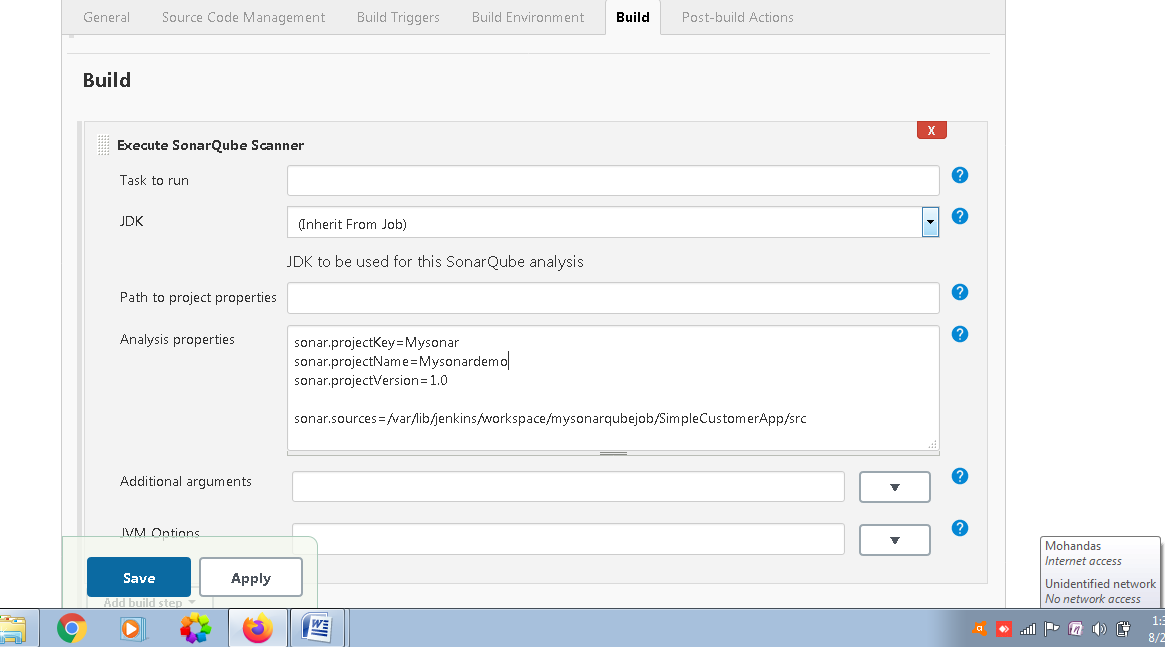
Now I will be creating a Jenkins job for testing it





Need to mention in build tab





Once done you will build the project