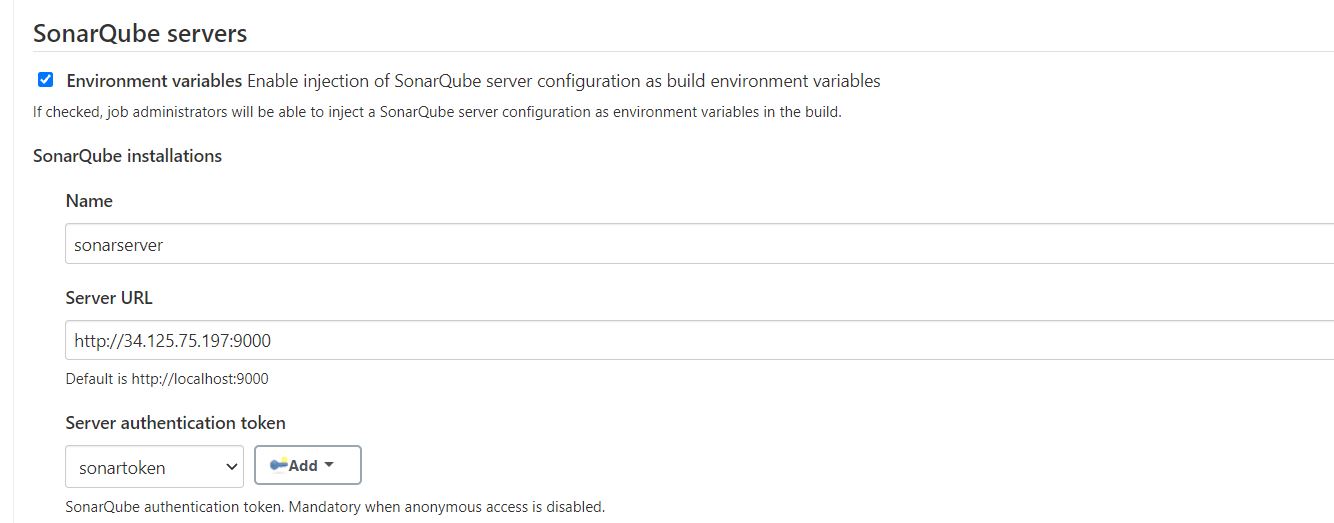
# Integrating Sonarqube with Jenkins

## **Pre-requisite**

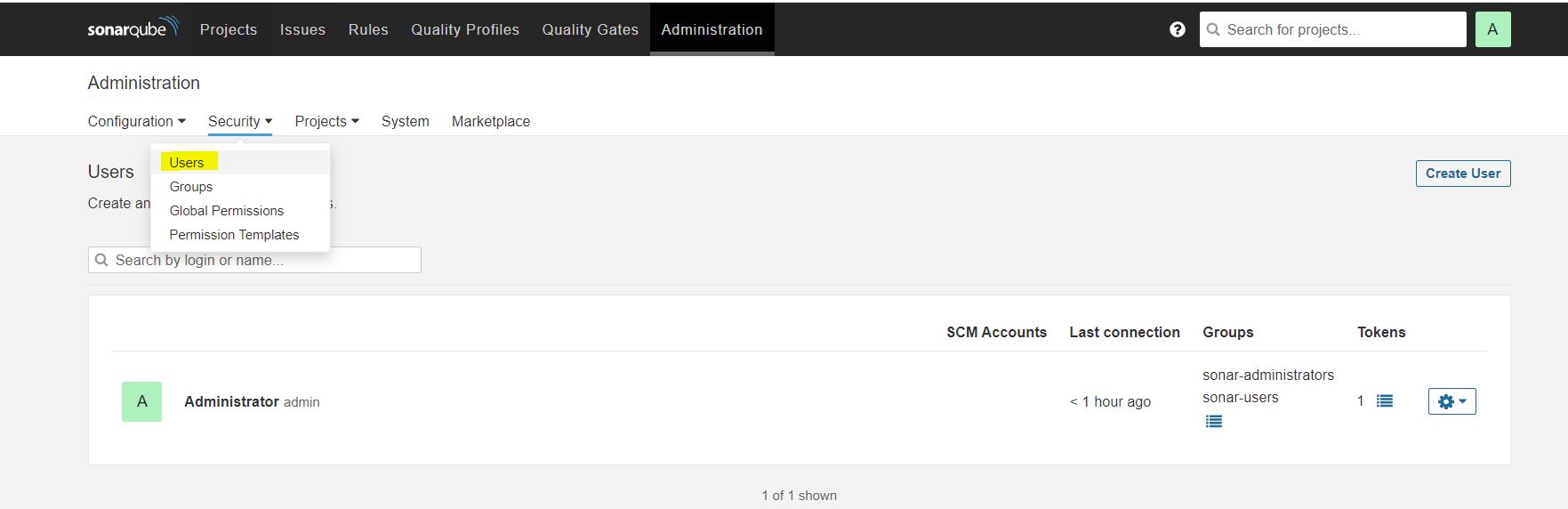
* We need to have Jenkins and SonarQube server up and running (by default Jenkins runs on 8080 and SonarQube at 9000)

## **Initial setup**

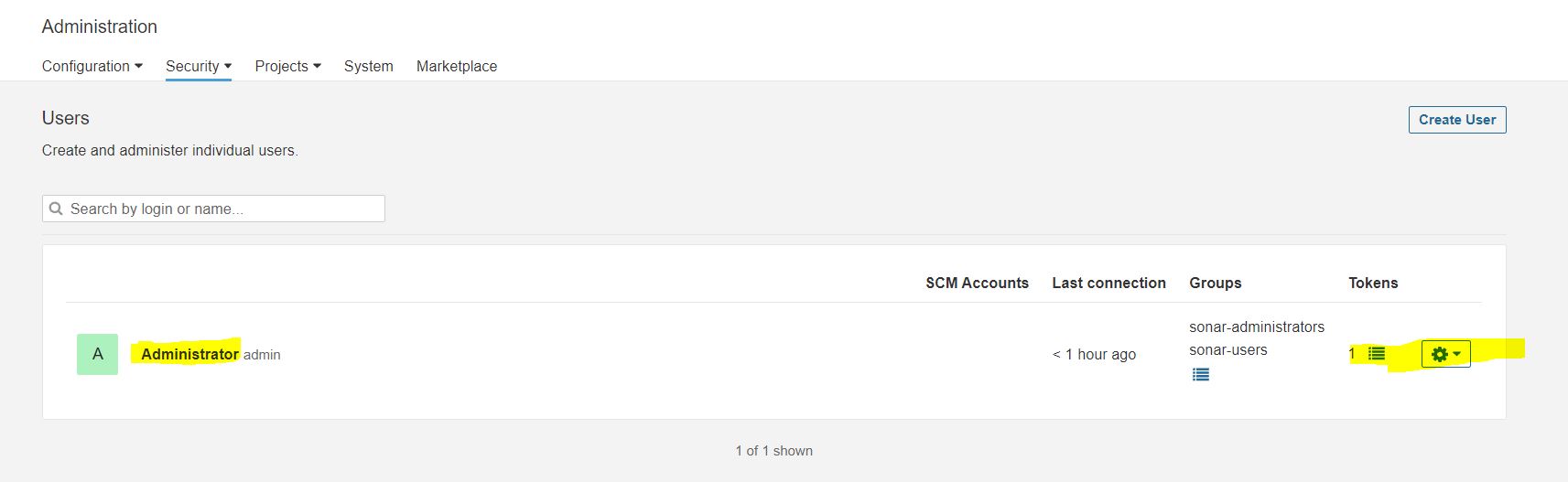
In Jenkins install plugins like Docker, SonarQube and Sonar quality gate plugin (manage Jenkins --> manage plugins --> in available tab search for respective plugins) After sonar plugin installation in configure system enter sonar information as shown in below picture



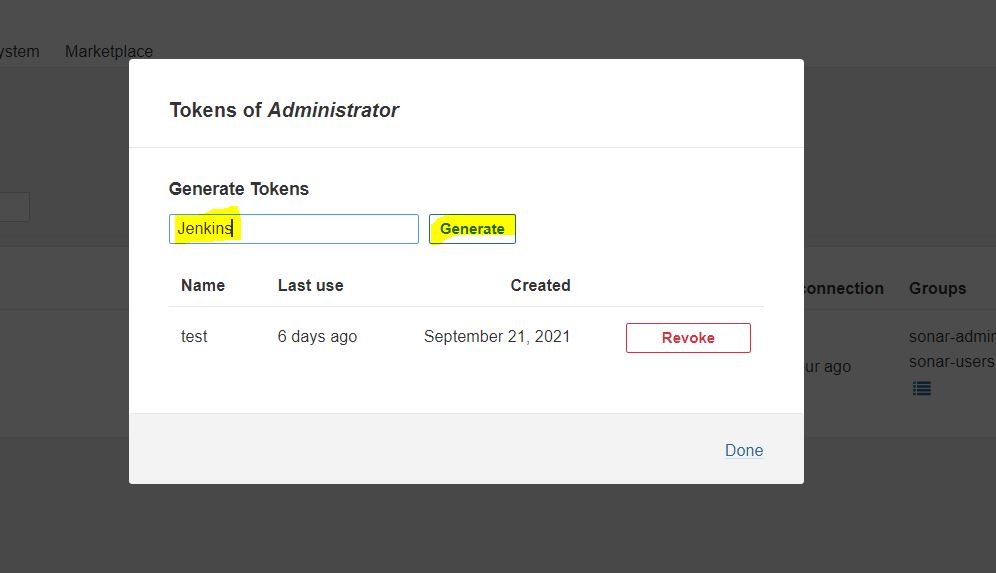
* Name: any meaningful name, but this will be referred in pipeline while executing sonar steps
* Server URL: sonarqube url
* Server authentication token: this token has to be created in sonarqube To create that token in sonarqube navigate to administration --> Security --> Users ( after which you will see list of users)



Click on token button, you will prompted to create token as shown in below pictures

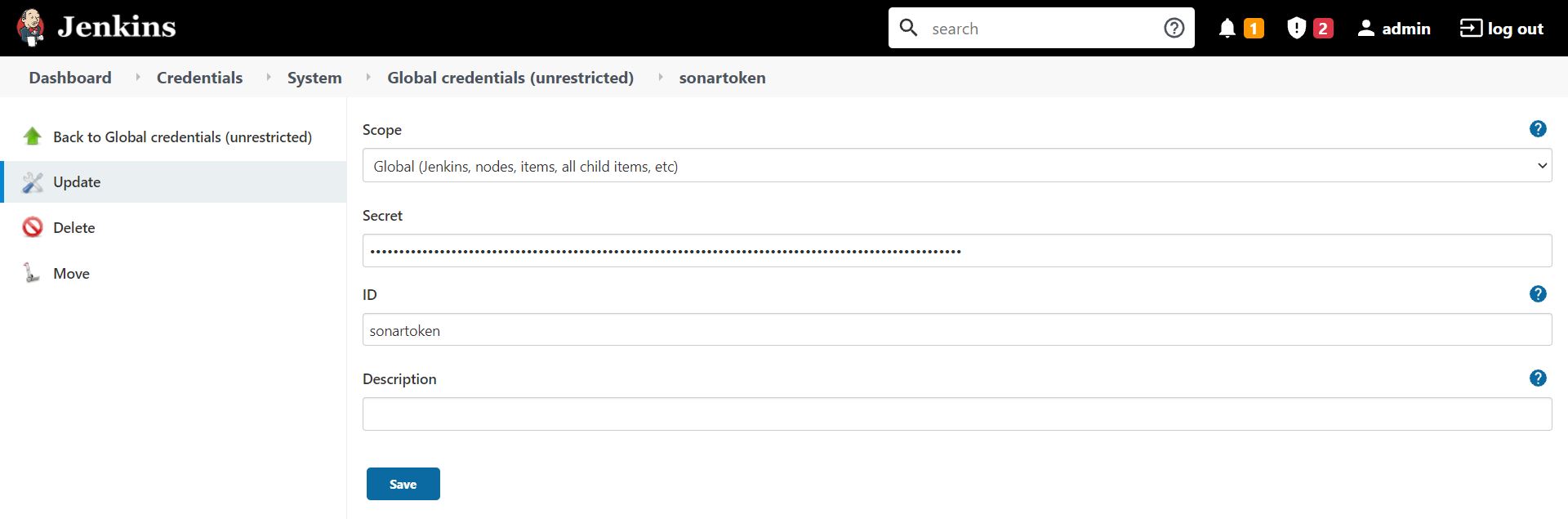


Give meaningful name and click on generate:

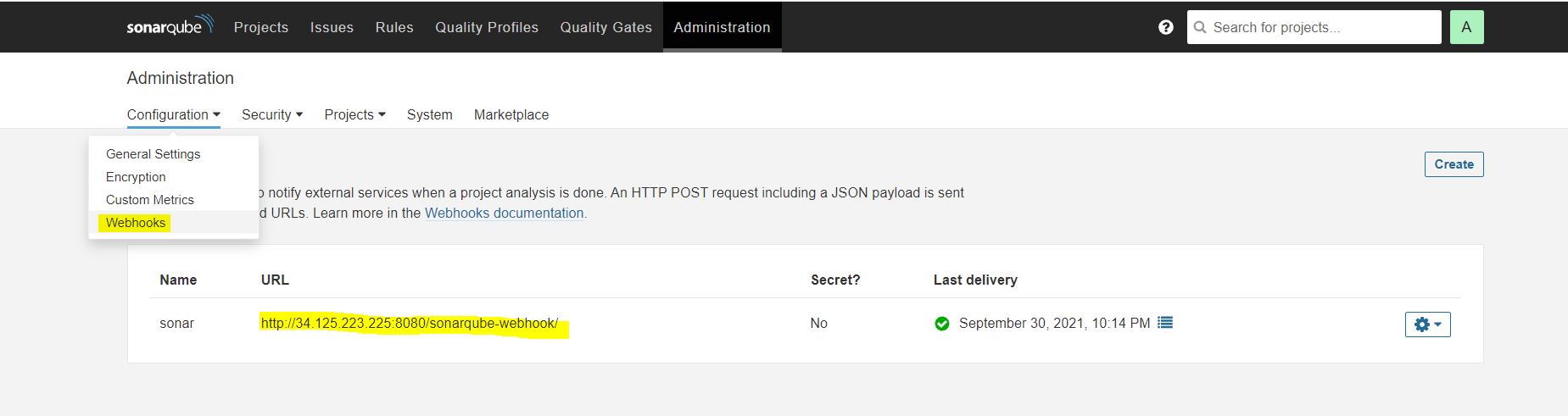


Copy the token to create secret text in jenkins

In Jenkins navigate manage Jenkins --> manage credentials --> add credentials --> select kind as secret text --> give value in secret ( will show up in sonar authentication token dropdown)



Also need to create webhook to have communication between jenkins and sonarqube To create webhook navigate to administration --> configuration --> webhooks --> then provide name and url of your jenkins suffixed with /sonarqube-webhook/



timeout(time: 1, unit: 'HOURS') {

def qg = waitForQualityGate()

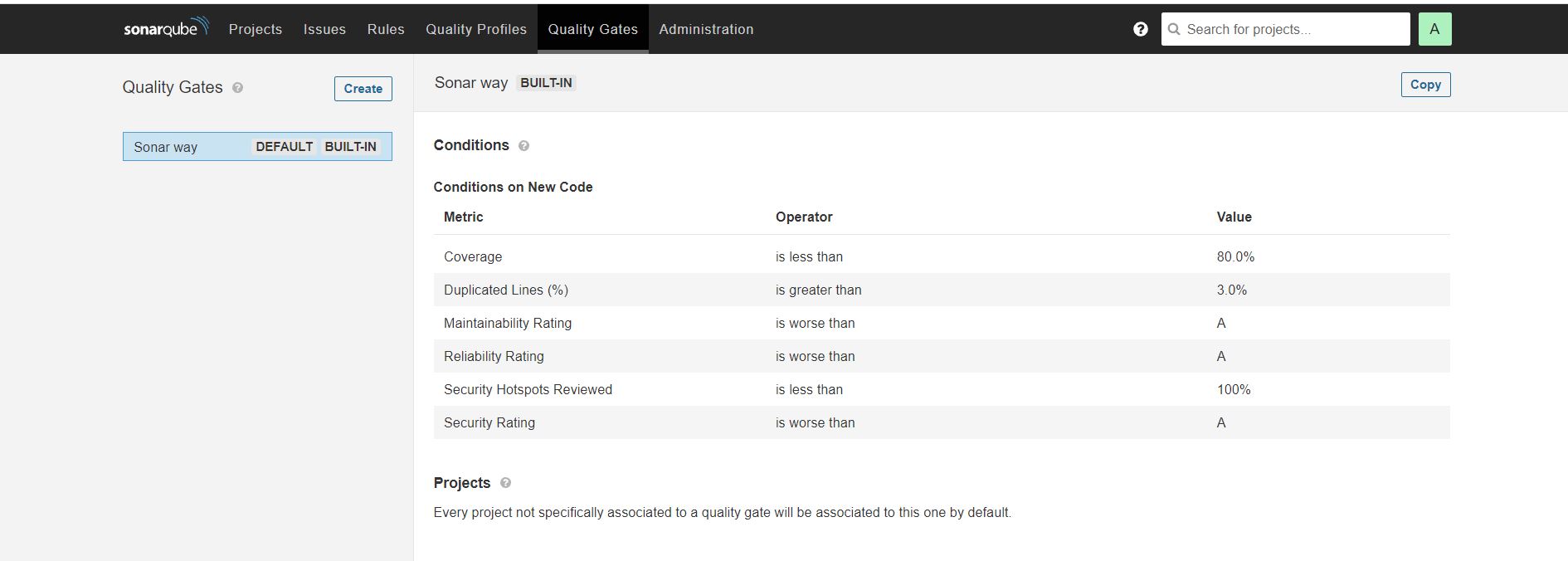
if (qg.status != 'OK') {

error "Pipeline aborted due to quality gate failure: ${qg.status}"

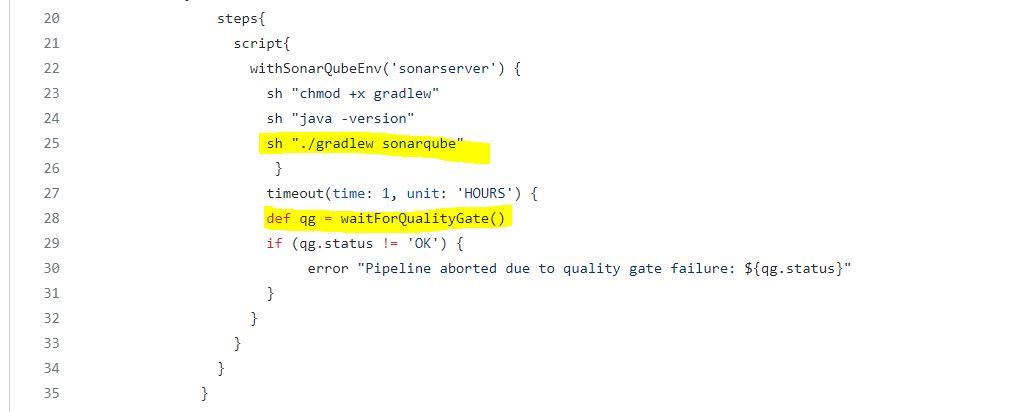
}

}

Then in sonarqube configure quality gates, below is default if want we can change it



Jenkinsfile would look like this, as you can see we are verifying our code against sonarqube quality gate, it code satisfies with quality gate then it will procced further else it will stop build in that stage



stage("build & SonarQube analysis") {

steps {

withSonarQubeEnv(credentialsId: 'Sonnar-token') {

\\sh 'mvn clean package sonar:sonar'

sh "mvn sonar:sonar \

-Dsonar.projectKey=maven-jenkins-pipeline \

-Dsonar.host.url=http://13.126.108.209:9000/"

}

}

}

stage("Quality Gate") {

steps {

timeout(time: 1, unit: 'HOURS') {

waitForQualityGate abortPipeline: true

//waitForQualityGate abortPipeline: true, credentialsId: 'Sonnar-token'

}

}

}

If you are quality gate satisfices then you will see passed in jenkins page

