

Integrating SonarCube with Jenkins

1. Make an ec2 instance and install Jenkins
2. Make an ec2 instance and install sonarcube
3. Download sonarcube plugins in Jenkins
 - a. `Maven Integration plugin 3.24`
 - b. `OWASP Dependency-Check Plugin 5.5.1`
 - c. `Sonar Quality Gates Plugin 328.vf4369b_da_d3c2`
 - d. `SonarQube Scanner for Jenkins 2.17.2`
4. Integrate SonarQube with Jenkins Pipeline
 - a. Create a token in SonarQube by logging into SonarQube. Select the Profile and choose MyAccount
 - b. Click on Security
 - c. Give the name of the Token and click on generate
 - d. Copy the token
5. Add SonarQube in Jenkins
 - a. Go to Manage Jenkins > Tools
 - b. Add the SonarQube Scanner installation details:

SonarQube Scanner installations

SonarQube Scanner installations ^ Edited

Add SonarQube Scanner

SonarQube Scanner

Name

sonarqube

☒ Install automatically ?

Install from Maven Central

Version

SonarQube Scanner 6.2.1.4610

Add Installer

c.

d. Add the Maven installation details:

Maven installations

Maven installations ^ Edited

Add Maven

Maven

Name

maven

☒ Install automatically ?

Install from Apache

Version

3.9.9

Add Installer

e.

f. Click on apply and save

g. Create a credential to connect Jenkins and SonarQube server

h. Go to manage jenkins > Credentials > System > Global credentials (unrestricted) > Add Credentials

New credentials

Kind

Secret text

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Secret

.....

ID ?

sonarqube

Description ?

sonarqube

i.

j. Secret == token generated in sonar cube

k. Credentials ID will be used in the Jenkins Pipeline code

6. Establish the Sonar Qube server Connections

a. Go to Manage Jenkins > System

b. Select the check-box of Injecting Environment variables, add the details of the SonarQube Server and also add the server authentication token

SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☒ Environment variables

SonarQube installations

List of SonarQube installations

Name

sonarqube

Server URL

Default is http://localhost:9000

http://34.238.246.210:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

sonarqube

Save Apply

c.

d. Server Url == Your Sonarcube serverurl:9000

e. Click on apply and save

7. Go to Jenkins home page

8. Click on new item

9. Select pipeline and click on ok

10. Go to pipeline

11. Enter the below script:

```
stage('Checkout Code') {  
    steps {  
        git  
        'https://github.com/Tejas-kandregula/addressbook-  
cicd-project' // Use your Git repository URL  
    }  
}
```

```
stage('Build') {  
    steps {  
        pipeline {  
            agent any  
            tools {  
                maven 'maven'  
            }  
        }  
    }  
}
```

```
stages {  
    stage('Compile') {  
        sh 'mvn compile'  
    }  
}
```

```
stage('Test') {
```

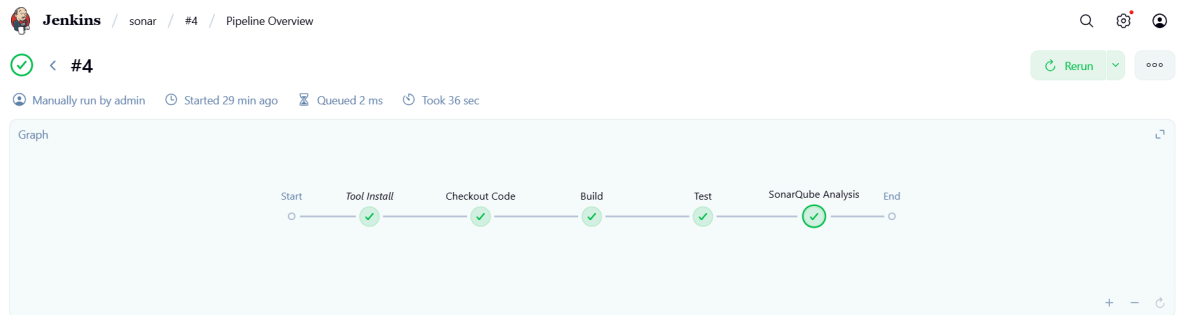
```

        steps {
            sh 'mvn test'
        }
    }

    stage('SonarQube Analysis') {
        environment {
            SONAR_HOST_URL =
'http://3.95.138.160:9000' // Replace with your
SonarQube URL
            SONAR_AUTH_TOKEN =
credentials('sonarcube') // Store your token in
Jenkins credentials
        }
        steps {
            sh 'mvn sonar:sonar
-Dsonar.projectKey=sample_project
-Dsonar.host.url=$SONAR_HOST_URL
-Dsonar.login=$SONAR_AUTH_TOKEN'
        }
    }
}

```

12. Click on save and apply



14. Once the analysis is completed, the SonarQube Project analysis report looks like below.

