Working with SonarQube

Ref: <https://github.com/chaitanyagaajula/DevOpsDemos>

**SonarQube Installation:**

SonarQube provides the capability to not only show health of an application but also to highlight issues newly introduced. With a Quality Gate in place, you can fix the leak and therefore improve code quality systematically.

**Prerequisites:**

* EC2 instance with Java installed
* MySQL Database Server or MyQL RDS instance. Get help here
* Installation
* Install MySQL client version

# yum install mysql

Download stable SonarQube version from below website.

Website: https://www.sonarqube.org/downloads/

Note: This Article written for SonarQube6.0

Download & unzip SonarQube 6.0

# cd /opt

# wget https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-6.0.zip

# unzip sonarqube-6.0.zip

# mv /opt/sonarqube-6.0 /opt/sonar

Allow RDS instance security group to access SonarQube server

Connect to RDS instance with database credentials

mysql -h <RDS\_Instance\_endpoint>:3306 -u <DB\_USER\_NAME> -p <DB\_PASSWORD>

**Create a new sonar database :**

CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8\_general\_ci;

Create a local and a remote user

CREATE USER sonar@localhost IDENTIFIED BY 'sonar';

CREATE USER sonar@'%' IDENTIFIED BY 'sonar';

Grant database access permissions to users

GRANT ALL ON sonar.\* TO sonar@localhost;

GRANT ALL ON sonar.\* TO sonar@'%';

check users and databases

use mysql

show databases;

SELECT User FROM mysql.user;

FLUSH PRIVILEGES;

QUIT

So for you have configured required database information on RDS. Let’s Jump back to your EC2 instance and enable SonarQube properties file to connect his Database.

**ON EC2 Instance**

Edit sonar properties file to uncomment and provide required information for below properties.

File Name: /opt/sonar/conf/sonar.properties

sonar.jdbc.username=sonar

sonar.jdbc.password=sonar

sonar.jdbc.url=jdbc:mysql://<RDS\_DATABAE\_ENDPOINT>:3306/sonar?useUnicode=true&characterEncoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance&useSSL=false

sonar.web.host=0.0.0.0

sonar.web.context=/sonar

Start SonarQube service

# cd /opt/sonar/bin/linux-x86-64/

# ./sonar.sh start

Run SonarQube as a default service

Implement SonarQube server as a service

Copy sonar.sh to etc/init.d/sonar and modify it according to your platform.

# sudo cp /opt/sonar/bin/linux-x86-64/sonar.sh /etc/init.d/sonar

# sudo vi /etc/init.d/sonar

Add below values to your /etc/init.d/sonar file

Insert/modify below values

SONAR\_HOME=/opt/sonar

PLATFORM=linux-x86-64

WRAPPER\_CMD="${SONAR\_HOME}/bin/${PLATFORM}/wrapper"

WRAPPER\_CONF="${SONAR\_HOME}/conf/wrapper.conf"

PIDDIR="/var/run"

Start SonarQube server

# service sonar start

SonarQube application uses port 9000. access SonarQube from browser

**http://<EC2\_PUBLIC\_IP>:9000/sonar**

**SonarQube Integration with Jenkins**

### Prerequisites

1. SonarQube Server
2. Jenkins Server

### Implementation

Login to Jenkins server and install sonarqube scanner.

* SonarQube scanner URL : <https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner>
* Package : <https://sonarsource.bintray.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-3.2.0.1227-linux.zip>

# wget https://sonarsource.bintray.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-3.2.0.1227-linux.zip

# unzip sonar-scanner-cli-3.2.0.1227-linux.zip

# mv sonar-scanner-3.2.0.1227-linux /opt/sonar\_scanner

Set SonarQube server details in sonar-scanner property file

* Sonar properties file: /opt/sonar\_scanner/conf/sonar-scanner.properties
  + sonar.host.url=http://<SONAR\_SERVER\_IP>:9000

Login to Jenkins GUI console and install " SonarQube scanner" plugin

* Manage Jenkins > Manage Plugins > Avalable > SonarQube scanner

Configure SonarQube scanner home path

* Manage Jenkins > Global Tool Configuration > SonarQube Scanner
  + Name : sonar\_scanner
  + SONAR\_RUNNER\_HOME : /opt/sonar\_scanner

Configure SonarQube server name and authentication token

* Manage Jenkins > Configure Systems > SonarQube Servers
  + Name : SonarQube
  + ServerURL : http://<Sonarqube\_server>:9000/sonar
  + Server authentication token To Get Authentication code follow below steps. Login to SonarQube server as a admin My Account > Security > Generate Token

Create a job to test SonarQube. Provide below sonar properties details in the job under build

* Build:
  + Execute SonarQube Scanner > Analysis properties (it is mandatary).
    - sonar.projectKey=Valaxy
    - sonar.projectName=ValaxyDemo
    - sonar.projectVersion=1.0
    - sonar.sources=/var/lib/jenkins/workspace/$JOB\_NAME/<PROJECT\_NAME>/src

Execute job to get analysis report.

