SonarQube

Overview

SonarQube is an automatic code review tool to detect bugs, vulnerabilities and code smells in your code. It can integrate with your existing workflow to enable continuous code inspection across your project branches and pull requests.

The SonarQube Platform is made of 4 components:

1.One SonarQube Server starting 3 main processes:

•Web Server for developers, managers to browse quality snapshots and configure the SonarQube instance

•Search Server based on Elasticsearch to back searches from the UI

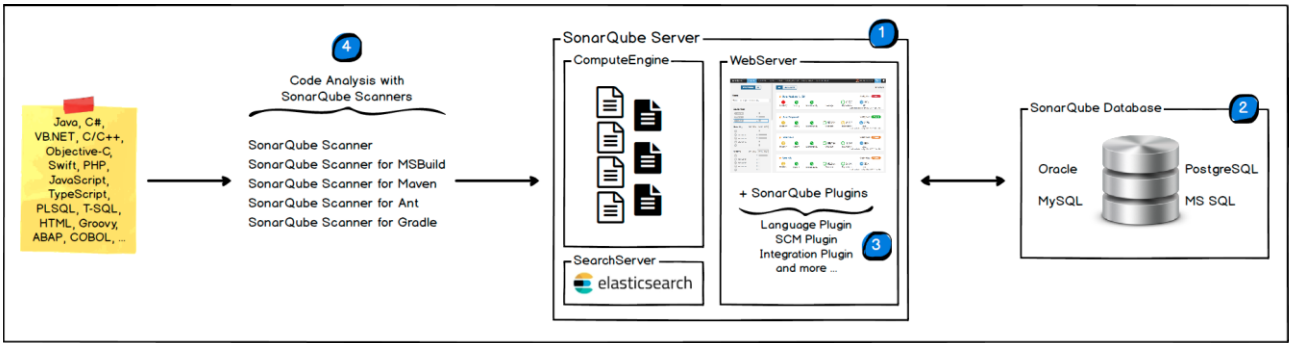
•Compute Engine Server in charge of processing code analysis reports and saving them in the SonarQube Database

2.One SonarQube Database to store:

•the configuration of the SonarQube instance (security, plugins settings, etc.)

•the quality snapshots of projects, views, etc.

3.Multiple SonarQube Plugins installed on the server, possibly including language, SCM, integration, authentication, and governance plugins



4.One or more SonarScanners running on your Build / Continuous Integration Servers to analyze projects

SonarQube SetUp for Crowsoft

Installation

Follow the below commands to install sonarqube on the given OS (ubuntu 16.04).

Step 1: Perform a system update

Before installing any packages on the Ubuntu server instance, it is recommended to update the system. Log in using the sudo user and run the following commands to update the system.

```
sudo apt-get update

sudo apt-get -y upgrade
```

Step 2: Install JDK

Add the Oracle Java repository on the server by running.

```
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
```

Install Oracle JDK by typing:

```
sudo apt install oracle-java8-installer
```

You can now check the version of Java by typing:

```
java -version
```

Step 3: Install and configure PostgreSQL

Install the PostgreSQL repository.

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" >>
/etc/apt/sources.list.d/pgdg.list'

wget -q https://www.postgresql.org/media/keys/ACCC4CF8.asc -O - | sudo apt-key add -
```

Install the PostgreSQL database server by running:

```
sudo apt-get -y install postgresql postgresql-contrib
```

Start PostgreSQL server and enable it to start automatically at boot time by running:

```
sudo systemctl start postgresql

sudo systemctl enable postgresql
```

Change the password for the default PostgreSQL user.

```
sudo passwd postgres
```

Switch to the postgres user.

```
su - postgres
```

Enter password for the user postgres (used password : csoftsonar)

Create a new user by typing:

```
createuser sonar
```

password for sonar used: sonarcsoft

Switch to the PostgreSQL shell.

```
psql
```

Set a password for the newly created user for SonarQube database.

```
ALTER USER sonar WITH ENCRYPTED password 'sonarcsoft';
```

Create a new database for PostgreSQL database by running:

```
CREATE DATABASE sonar OWNER sonar;
```

Exit from the psql shell:

```
\q
```

Switch back to the sudo user by running the exit command.

Step 4: Download and configure SonarQube

Download the SonarQube installer files archive.

```
wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-7.7.zip
```

You can always look for the link to the latest version of the application on the SonarQube download page.

Install unzip by running:

```
apt-get -y install unzip
```

Unzip the archive using the following command.

```
sudo unzip sonarqube-7.7.zip -d /opt
```

Rename the directory:

```
sudo mv /opt/sonarqube-7.7 /opt/sonarqube
```

Open the SonarQube configuration file using your favorite text editor.

```
sudo nano /opt/sonarqube/conf/sonar.properties
```

Find the following lines.

```
#sonar.jdbc.username=
#sonar.jdbc.password=
```

Uncomment and provide the PostgreSQL username and password of the database that we have created earlier. It should look like:

```
sonar.jdbc.username=sonar

sonar.jdbc.password= sonarcsoft
```

Next, find:

```
#sonar.jdbc.url=jdbc:postgresql://localhost/sonar
```

Uncomment the line, save the file and exit from the editor.

Step 5: Configure Systemd service

SonarQube can be started directly using the startup script provided in the installer package. As a matter of convenience, you should setup a Systemd unit file for SonarQube.

```
nano /etc/systemd/system/sonar.service
```

Populate the file with:

```
[Unit]

Description=SonarQube service

After=syslog.target network.target



[Service]

Type=forking



ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start

ExecStop=/opt/sonarqube/bin/linux-x86-64/sonar.sh stop



User=sonar
```

```
Group=sonar

Restart=always



[Install]

WantedBy=multi-user.target
```

Start the application by running:

```
sudo systemctl start sonar
```

Enable the SonarQube service to automatically start at boot time.

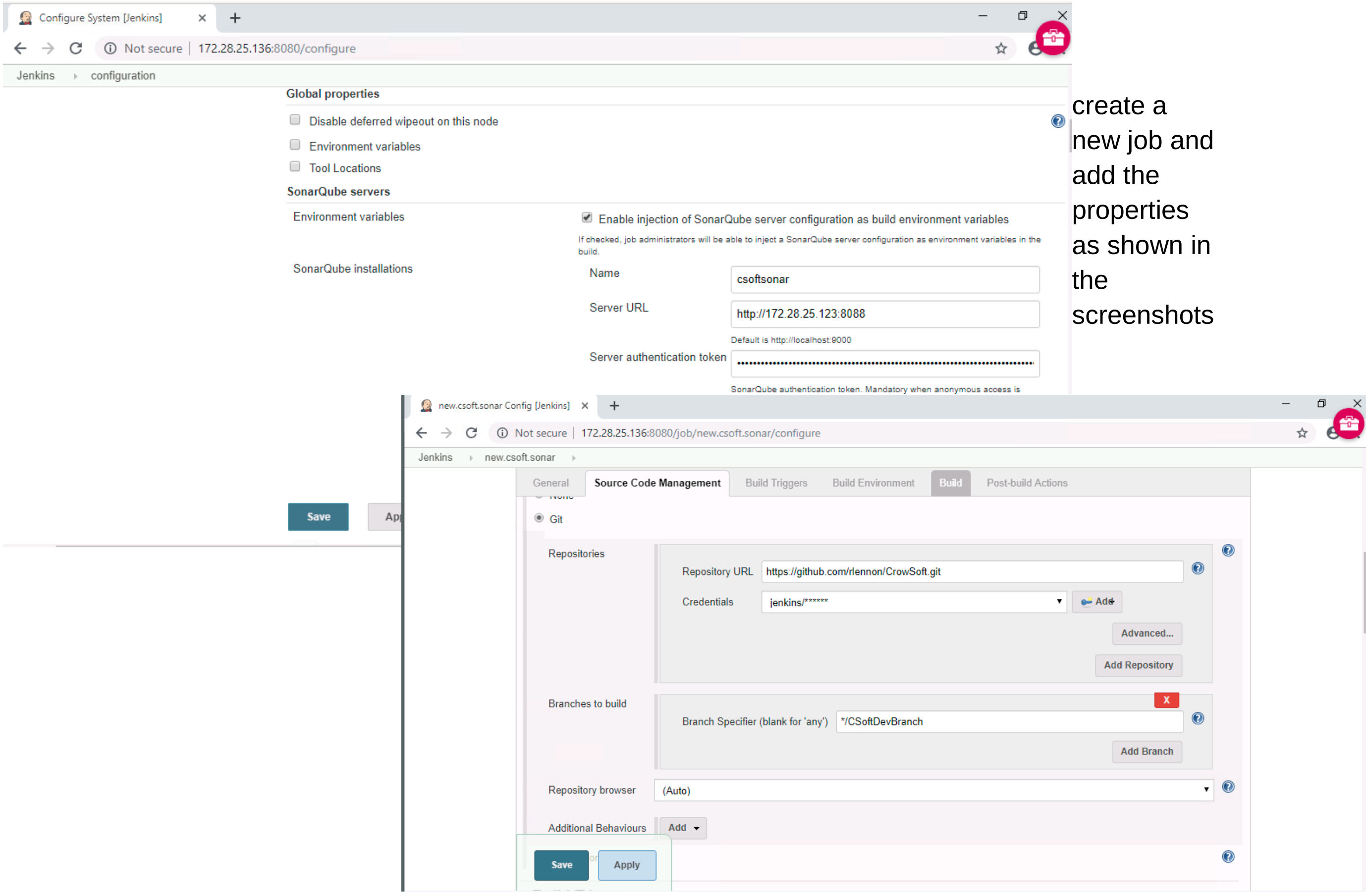
```
sudo systemctl enable sonar
```

To check if the service is running, run:

```
sudo systemctl status sonar
```

SonarQube with Jenkins Integration

Under Configure System add sonar server details as shown in the screenshot



Configure System [Jenkins]

Not secure | 172.28.25.136:8080/configure

Jenkins › configuration

Global properties

□ Disable deferred wipeout on this node

□ Environment variables

□ Tool Locations

SonarQube servers

Environment variables: ☑ Enable injection of SonarQube server configuration as build environment variables

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

SonarQube installations

Name: csoftsonar

Server URL: http://172.28.25.123:8088

Default is http://localhost:9000

Server authentication token: ••

SonarQube authentication token. Mandatory when anonymous access is

create a new job and add the properties as shown in the screenshots

new.csoft.sonar Config [Jenkins]

Not secure | 172.28.25.136:8080/job/new.csoft.sonar/configure

Jenkins › new.csoft.sonar ›

● Git

Repositories

Repository URL: https://github.com/rlennon/CrowSoft.git

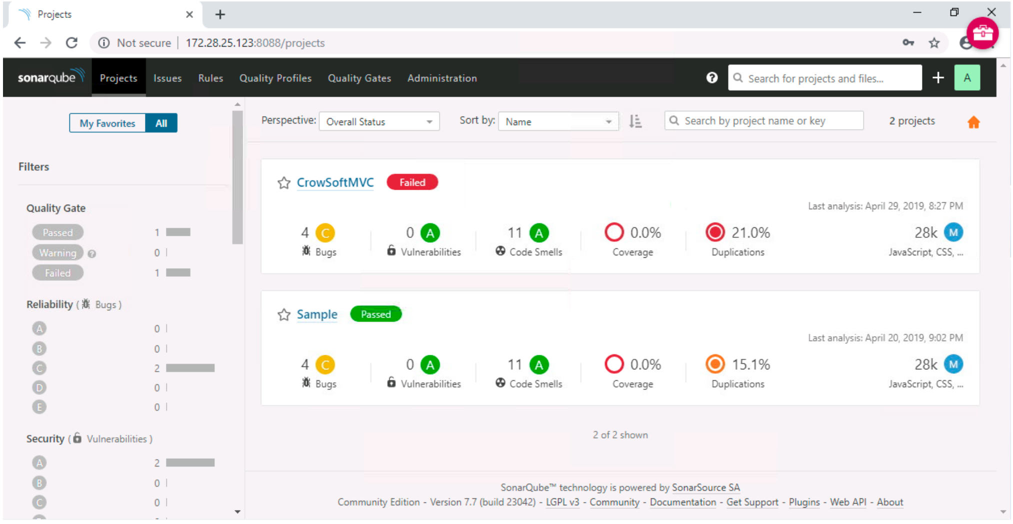
Credentials: jenkins/******

Branches to build

Branch Specifier (blank for 'any'): */CSoftDevBranch

Repository browser: (Auto)

Additional Behaviours



CrowSoftMVC Failed

Last analysis: April 29, 2019, 8:27 PM

4 C	0 A	11 A	0.0%	21.0%	28k M
Bugs	Vulnerabilities	Code Smells	Coverage	Duplications	JavaScript, CSS, ...

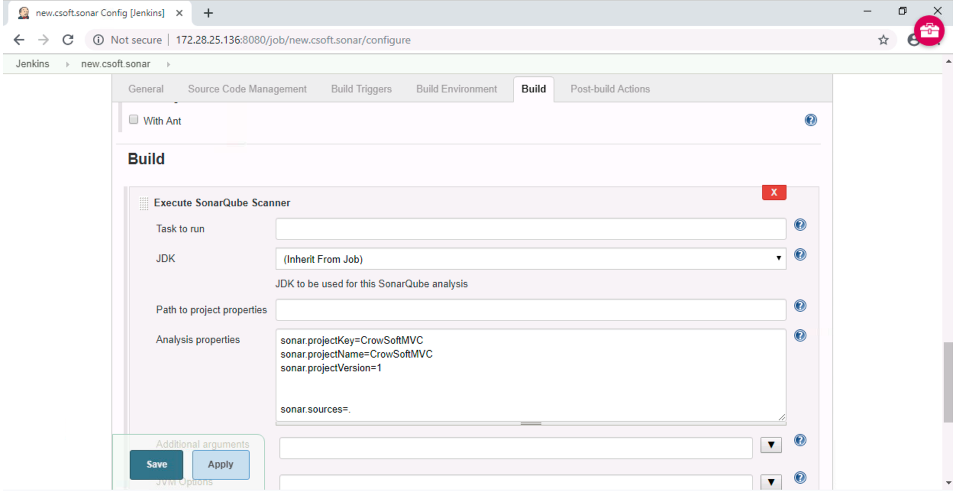
Sample Passed

Last analysis: April 20, 2019, 9:02 PM

4 C	0 A	11 A	0.0%	15.1%	28k M
Bugs	Vulnerabilities	Code Smells	Coverage	Duplications	JavaScript, CSS, ...

2 of 2 shown

SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 7.7 (build 23042) - LGPL v3 - Community - Documentation - Get Support - Plugins - Web API - About



□ With Ant

Build

Execute SonarQube Scanner

Task to run:

JDK: (Inherit From Job)

JDK to be used for this SonarQube analysis

Path to project properties:

Analysis properties:

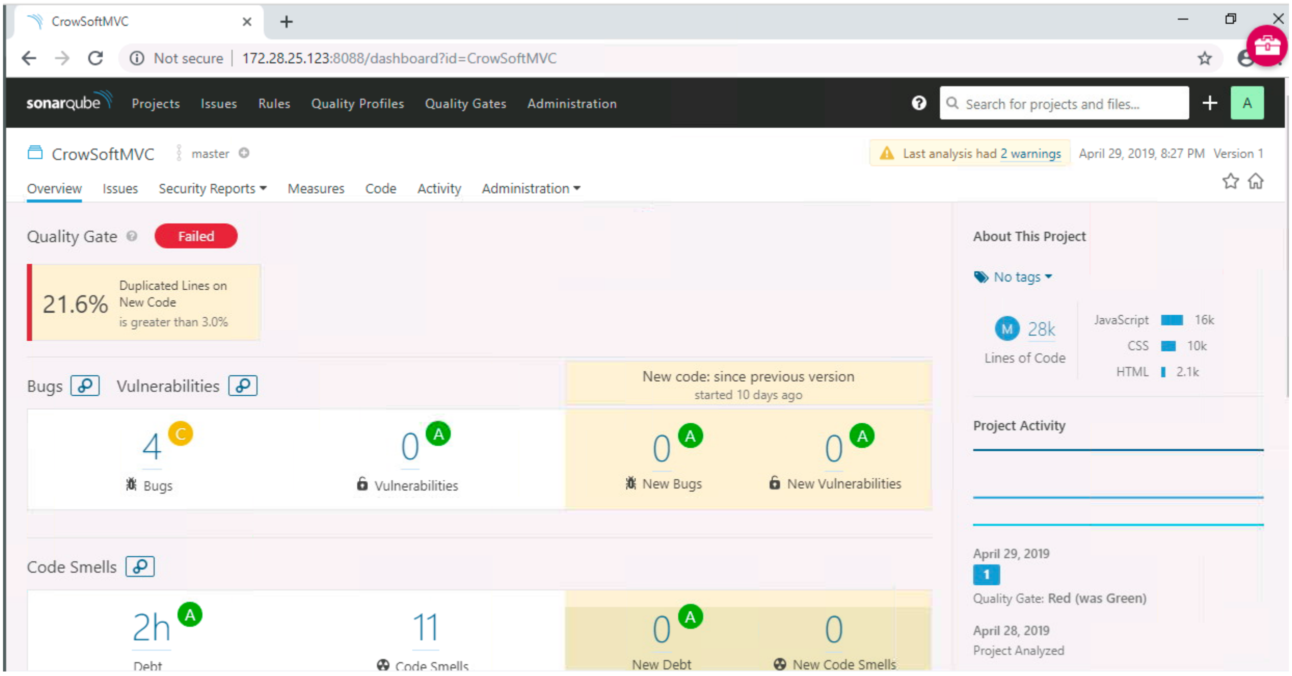
```
sonar.projectKey=CrowSoftMVC
sonar.projectName=CrowSoftMVC
sonar.projectVersion=1


sonar.sources=.
```

Additional arguments:

JVM Options:

After doing build now from the job , we can see the analysis results at sonarqube as below



CrowSoftMVC master

Last analysis had 2 warnings April 29, 2019, 8:27 PM Version 1

Quality Gate Failed

21.6% Duplicated Lines on New Code is greater than 3.0%

Bugs Vulnerabilities

New code: since previous version
started 10 days ago

4 C Bugs

0 A Vulnerabilities

0 A New Bugs

0 A New Vulnerabilities

Code Smells

2h A Debt

11 Code Smells

0 A New Debt

0 New Code Smells

About This Project

No tags

M 28k Lines of Code

JavaScript	16k
CSS	10k
HTML	2.1k

Project Activity

April 29, 2019
1
Quality Gate: Red (was Green)

April 28, 2019
Project Analyzed

-- End --