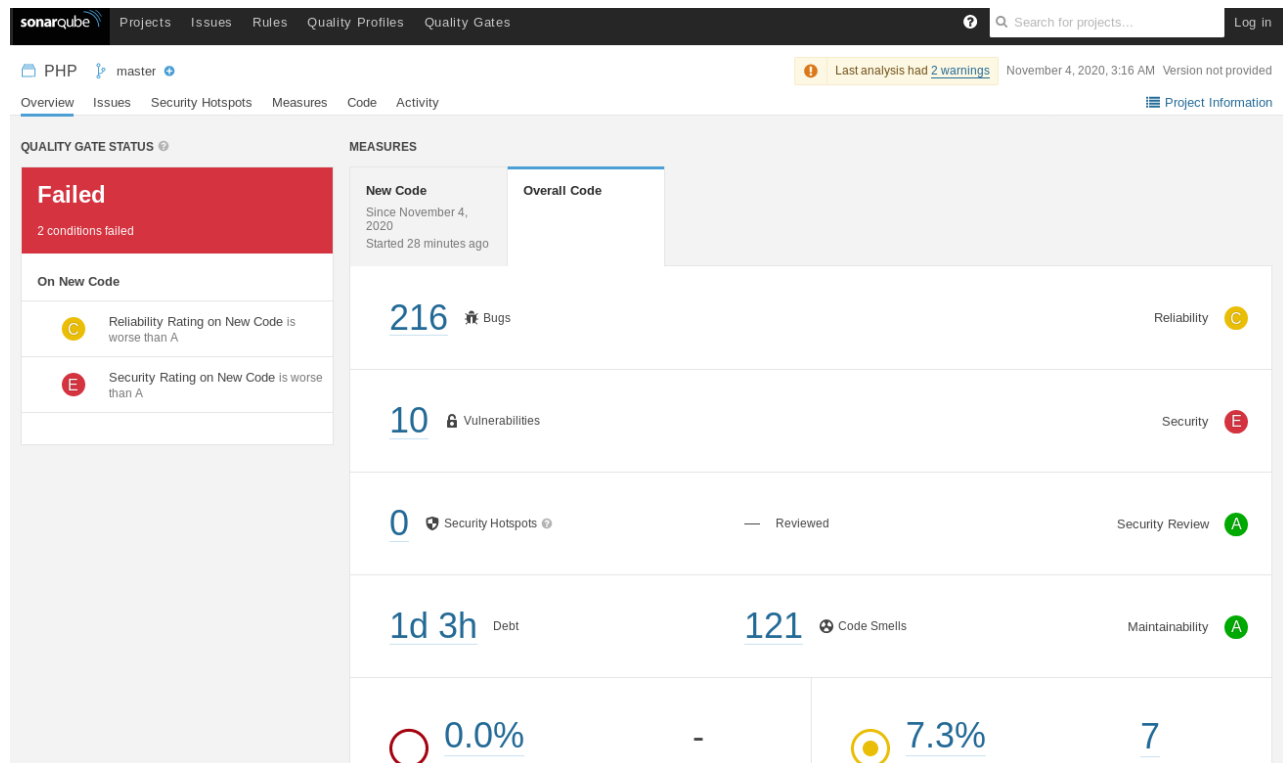


Static analysis using SonarQube

Overview



In this lab, you are going to learn how to use SonarQube to generate static code analysis report.

SonarQube includes support for the programming languages Java (including Android), C#, PHP, JavaScript, TypeScript, C/C++, Ruby, Kotlin, Go, COBOL, PL/SQL, PL/I, ABAP, VB.NET, VB6, Python, RPG, Flex, Objective-C, Swift, CSS, HTML, and XML.

Outcomes

Upon completion of this session, you should be able to

- Use SonarQube to analysis source code
- Incorporating Jenkins Pipeline with SonarQube into your team project
- Analysis the findings generated by SonarQube

1: Installation

This lab is based on the instruction <https://docs.sonarqube.org/latest/setup/get-started-2-minutes/> and the OWASP - Vulnerable Web Application <https://github.com/OWASP/Vulnerable-Web-Application>, but it also requires different docker SAST image / software to be installed before you can incorporate Jenkins Pipeline.

1. Install the **SonarQube Scanner** under the Plugin Manager

Jenkins > Plugin Manager

Back to Dashboard
Manage Jenkins

Search: son

Updates Available Installed Advanced

Install +	Name	Version	Released
<input checked="" type="checkbox"/>	SonarQube Scanner External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube , the open source platform for Continuous Inspection of code quality.	2.12	2 mo 1 day ago
<input type="checkbox"/>	Generic Webhook Trigger bitbucket bitbucket-server github gitlab jira notification Build Parameters Build Triggers webhook Can receive any HTTP request, extract any values from JSON or XML and trigger a job with those values available as variables. Works with GitHub, GitLab, Bitbucket, Jira and many more.	1.71	6 days 14 hr ago
<input type="checkbox"/>	Cucumber reports Build Reports Provides pretty html reports for Cucumber. Can be used anywhere a json report is generated (Java, Ruby, JavaScript and other implementations). Now with support for percentage thresholds.	5.3.1	2 mo 24 days ago
<input type="checkbox"/>	global-build-stats Build Reports	1.5	3 yr 0 mo ago


2. Restart Jenkins

3. Install the **SonarQube docker** by

```
$ docker pull sonarqube
```


```
$ docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
```

4. Log in to <http://localhost:9000> with System Administrator credentials (login=admin, password=admin).

 Projects Issues Rules Quality Profiles Quality Gates Administration


How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform. First, you need to set up a DevOps platform configuration.




From Azure DevOps

Set up global configuration




From Bitbucket

Set up global configuration



From GitHub


Set up global configuration



From GitLab

Set up global configuration

Are you just testing or have an advanced use-case? Create a project manually.




Manually

5. Press the “Manually” icon.


Create a project

All fields marked with * are required

Project display name *

 
Up to 255 characters. Some scanners might override the value you provide.

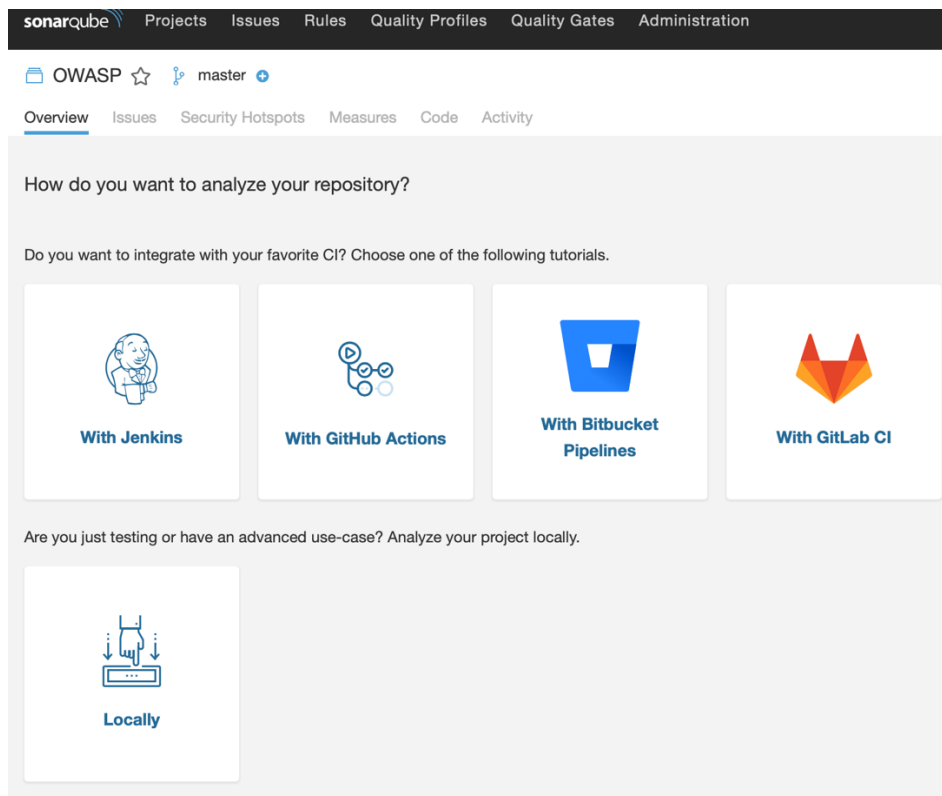
Project key *

 
The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.

[Set Up](#)

6. Input the project key and project name.

7. Press “Locally”.



7. Generate a token for your project, copy the token and you will be using the token in the Jenkinsfile.

OWASP ☆ master

Overview Issues Security Hotspots Measures Code Activity

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

- 1 Provide a token**
 - ☒ Generate a token

Generate

 - ☐ Use existing token

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your [user account](#).
- 2 Run analysis on your project**

OWASP ☆ master

Overview Issues Security Hotspots Measures Code Activity

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

- 1 Provide a token**

textingtoken: **abefbf23a49092ae4d7b834338eceb4b2f441896** 🗑️

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your [user account](#).

Continue
- 2 Run analysis on your project**

Lab 9

8. Select the language and OS you are using, copy the Scanner information and you will be using it in the Jenkinsfile.

1 Provide a token

textingtoken:abefbf23a49092ae4d7b834338eceb4b2f441896

2 Run analysis on your project

What is your project's main language?

Java

C# or VB.NET

Other (JS, TS, Go, Python, PHP, ...)

What is your OS?

Linux

Windows

macOS

Download and unzip the Scanner for Linux

And add the `bin` directory to the `PATH` environment variable

Download

Execute the Scanner from your computer

Running a SonarQube analysis is straightforward. You just need to execute the following commands in your project's folder.

```
sonar -scanner \
-Dsonar.projectKey=OWASP \
-Dsonar.sources=. \
-Dsonar.host.url=http://192.168.16.128:9000 \
-Dsonar.login=abefbf23a49092ae4d7b834338eceb4b2f441896
```

Copy

Please visit the [official documentation of the Scanner](#) for more details.

9. Back to Jenkin, go to Configure System to enable SonarQube Scanner.

To Configure your SonarQube server(s):

1. Log into Jenkins as an administrator and go to **Manage Jenkins > Configure System**.
2. Scroll down to the SonarQube configuration section, click **Add SonarQube**, and add the values you're prompted for.
3. The server authentication token should be created as a 'Secret Text' credential.

Dashboard > 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

SonarQube

Server URL

http://10.0.0.6:9000/

Default is http://localhost:9000

Server authentication token

Secret text

Add

SonarQube authentication token. Mandatory when anonymous access is disabled.

Advanced...

Delete SonarQube

Add SonarQube

List of SonarQube installations

10. Back to Jenkin, go to Global Tool Configuration to define the SonarQube Scanner Tool.

Global Tool Configuration

SonarQube Scanner

SonarQube Scanner installations

Add SonarQube Scanner



SonarQube Scanner

Name

SonarQube

☒ Install automatically



Install from Maven Central

Version

SonarQube Scanner 4.6.2.2472

Add Installer ▾

2: Configuration

Use the following Jenkinsfile in your pipeline:

```
pipeline {
  agent any
  stages {
    stage ('Checkout') {
      steps {
        git branch:'master', url: 'https://github.com/OWASP/Vulnerable-Web-Application.git'
      }
    }

    stage('Code Quality Check via SonarQube') {
      steps {
        script {
          def scannerHome = tool 'SonarQube';
          withSonarQubeEnv('SonarQube') {
            sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=OWASP -Dsonar.sources=."
          }
        }
      }
    }
  }
  post {
    always {
      recordIssues enabledForFailure: true, tool: sonarQube()
    }
  }
}
```

Note that:

“SonarQube” is the name you used in System configuration.

“OWASP” is the SonarQube project name

Jenkins PHPVuln

Back to Dashboard

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

SonarQube

Open Blue Ocean

Rename

SonarQube Warnings

Pipeline Syntax

Build History trend ^

Pipeline PHPVuln

Recent Changes

Stage View

	Checkout	Code Quality Check via SonarQube	Declarative: Post Actions
Average stage times: (Average full run time: ~25s)	2s	18s	272ms
#6 Nov 04 03:16 No Changes	1s	17s	356ms
#5 Nov 04 03:13 No Changes	1s	18s	72ms
#4			

Press “Build Now” and then “SonarQube”

PHP master

Last analysis had 2 warnings November 4, 2020, 3:16 AM Version not provided

Overview Issues Security Hotspots Measures Code Activity

Project Settings Project Information

1 / 10 issues 4h 34min effort

My Issues All

Clear All Filters

Filters

Type VULNERABILITY Clear

Bug 216

Vulnerability 10

Code Smell 121

Ctrl + click to add to selection

Severity

Blocker 10 Minor 0

Critical 0 Info 0

Major 0

Scope

Resolution

Status

Security Category

SonarSource

Authentication 6

Others 4

OWASP Top 10

SQL/sql1.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L30 cwe, owasp-a3

SQL/sql2.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L29 cwe, owasp-a3

SQL/sql3.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L30 cwe, owasp-a3

SQL/sql4.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L34 cwe, owasp-a3

SQL/sql5.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L32 cwe, owasp-a3

SQL/sql6.php

Add password protection to this database. Why is this an issue?

Vulnerability Blocker Open Not assigned 45min effort Comment

2 years ago L26 cwe, owasp-a3

You may start analyzing the result!

3: Standalone SonarQube Scanner

Alternatively, you can run the standalone SonarQube Scanner in your project folder to scan the source code. Especially when you are the 3rd party code auditor.

```
docker run --rm -e SONAR_HOST_URL=http://192.168.16.128:9000 -e  
SONAR_LOGIN=2742fdcd3a1fe63a0912d32ebd77a1c74a4e212d -it -v "$(pwd):/usr/src"  
sonarsource/sonar-scanner-cli -Dsonar.projectKey=OWASP
```

4: Docker memory issue

You may face the out-of-memory issue on the Jenkins / SonarQube docker. You may use the following command to increase the memory from 2GB to 3GB:

```
docker run --memory=3g the-remaining-command
```

5: Reference

<https://docs.sonarqube.org/latest/>

<https://docs.sonarqube.org/latest/analysis/scan/sonarscanner-for-jenkins/>

END OF DOCUMENT