Spring AU '21 – Java Code Quality – Morning Session

Name: Sheik Abudhahir K

Date: 27/01/2021

<u>Secure coding standards</u>

<u>1.CWE – Common Weakness Enumeration</u>

- enumerates design and architectural weaknesses, as well as low-level coding and design errors.
- enables better understanding and management of software weaknesses related to architecture and design
- find weaknesses in source code and operational systems
- category system for software weaknesses and vulnerabilities

eg: CWE category 121 is for stack-based buffer overflows

<u>2.OWASP Top 10 – Open web application security project</u>

- provides a checklist for secure coding practices

Some of top OWASP practices to help you protect against vulnerabilities are

- Security by Design
- Password Management
- Access Control
- Error Handling and Logging
- System Configuration
- Threat Modeling
- Cryptographic Practices
- Input Validation and Output Encoding

<u>3.CERT</u>

- Secure coding standards commonly used for C, C++ and Java
- Guidelines for avoiding coding and implementation errors.
- Encourage programmers to follow a uniform set of rules and guidelines

Risk assessment methodology using failure mode, effects and criticality analysis

Eg: Severity – How serious are the consequences of the rule being ignored

3 levels – (1-Low) (2-medium) (3-high)

Low – eg: denial of service attack, abnormal termination

Medium – eg: data integrity violation, unintentional information disclosure

High – eg: run arbitrary code.

4. SANS 25

- Most dangerous software error – links with CWE
- list of the most widespread and critical errors that can lead to serious vulnerabilities in softwares

Top errors:

CWE	Name
89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
120	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
306	Missing Authentication for Critical Function

SonarQube Server

Starting SonarQube server

It is running in localhost:9000

Login page:

Main page:

Project creation:

Token generation:

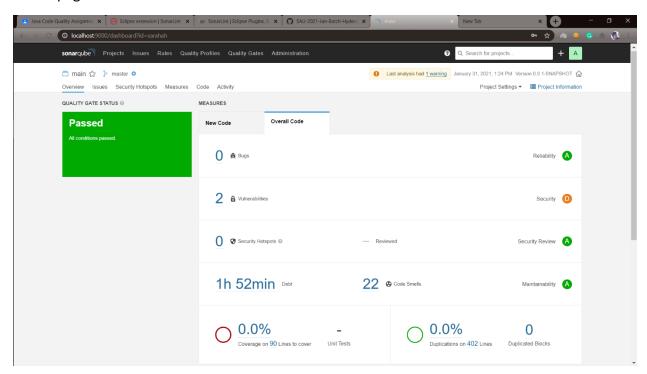Token generated with the given name
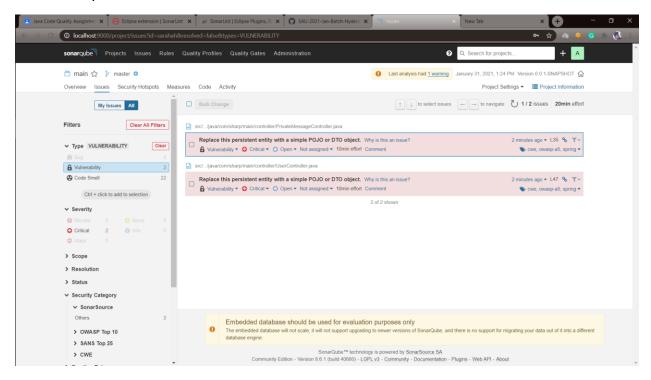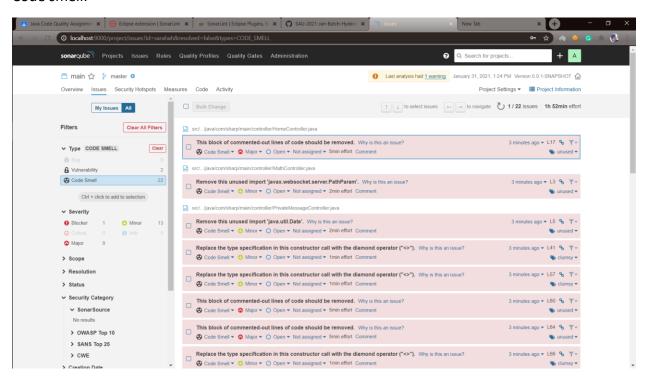
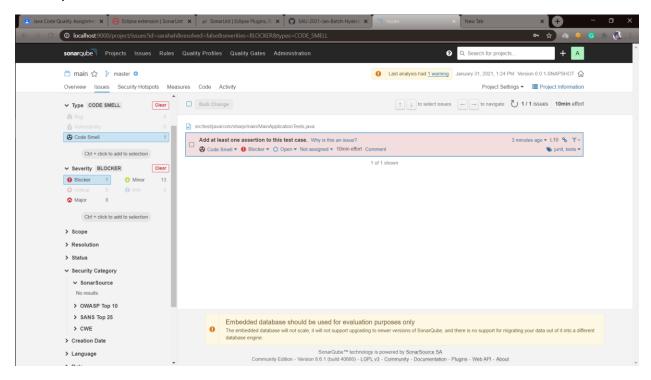# Run analysis on project

Command Running

Result page:

Vulnerability:

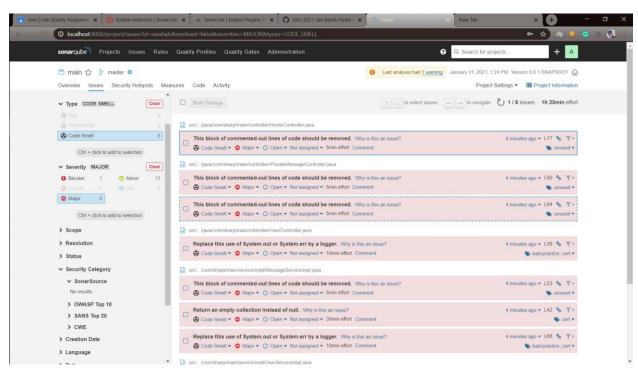## Code smell:
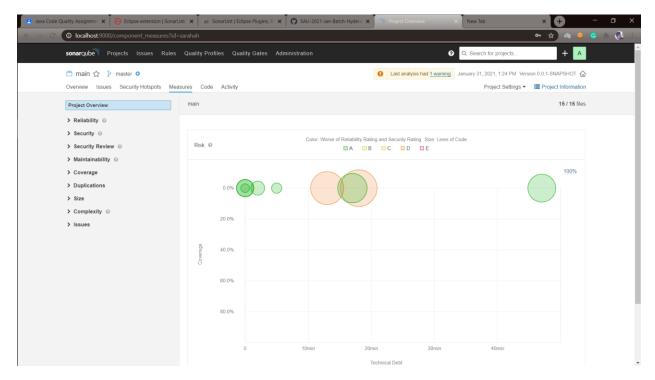
Severity – Blocker

Major

Measures:

I tried to install code analyzer plugins in eclipse, but I got some errors during installing, I changed my jre version to java 11, still the problem not resolved.