sonar RULES

Products ⌄

Secrets
ABAP
Apex
C
C++
CloudFormation
COBOL
C#
CSS
Flex
Go
HTML
Java
JavaScript
Kotlin
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

Java static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your JAVA code

All rules 632 | Vulnerability 53 | Bug 154 | Security Hotspot 36 | Code Smell 389 | Quick Fix 42

Tags ⌄ | Search by name...

Broadcasting intents is security-sensitive
Security Hotspot

Expanding archive files without controlling resource consumption is security-sensitive
Security Hotspot

Configuring loggers is security-sensitive
Security Hotspot

Using weak hashing algorithms is security-sensitive
Security Hotspot

Using unsafe Jackson deserialization configuration is security-sensitive
Security Hotspot

Setting JavaBean properties is security-sensitive
Security Hotspot

Disabling CSRF protections is security-sensitive
Security Hotspot

Using non-standard cryptographic algorithms is security-sensitive
Security Hotspot

Using pseudorandom number generators (PRNGs) is security-sensitive
Security Hotspot

Mocking all non-private methods of a class should be avoided
Code Smell

Empty lines should not be tested with regex MULTILINE flag
Code Smell

Server certificates should be verified during SSL/TLS connections

Analyze your code

Vulnerability | Critical | cwe privacy cert owasp ssl

Validation of X.509 certificates is essential to create secure SSL/TLS sessions not vulnerable to man-in-the-middle attacks.

The certificate chain validation includes these steps:

- The certificate is issued by its parent Certificate Authority or the root CA trusted by the system.
- Each CA is allowed to issue certificates.
- Each certificate in the chain is not expired.

This rule raises an issue when an implementation of X509TrustManager is not controlling the validity of the certificate (ie: no exception is raised). Empty implementations of the `X509TrustManager` interface are often created to disable certificate validation. The correct solution is to provide an appropriate trust store.

Noncompliant Code Example

```
class TrustAllManager implements X509TrustManager {

    @Override
    public void checkClientTrusted(X509Certificate[] chain,
    }

    @Override
    public void checkServerTrusted(X509Certificate[] chain,
        LOG.log(Level.SEVERE, ERROR_MESSAGE);
    }

    @Override
    public X509Certificate[] getAcceptedIssuers() {
        return null;
    }
}
```

See

- OWASP Top 10 2021 Category A2 - Cryptographic Failures
- OWASP Top 10 2021 Category A5 - Security Misconfiguration
- OWASP Top 10 2021 Category A7 - Identification and Authentication Failures
- OWASP Top 10 2017 Category A6 - Security Misconfiguration
- MITRE, CWE-295 - Improper Certificate Validation
- CERT, MSC61-J. - Do not use insecure or weak cryptographic algorithms

Available In:

sonarlint | sonarcloud | sonarqube

Methods setUp() and tearDown() should be correctly annotated starting with JUnit4

Code Smell

Class members annotated with "@VisibleForTesting" should not be accessed from production code

Code Smell

"String#replace" should be preferred to "String#replaceAll"

Code Smell

Derived exceptions should not hide their parents' catch blocks

Privacy Policy