

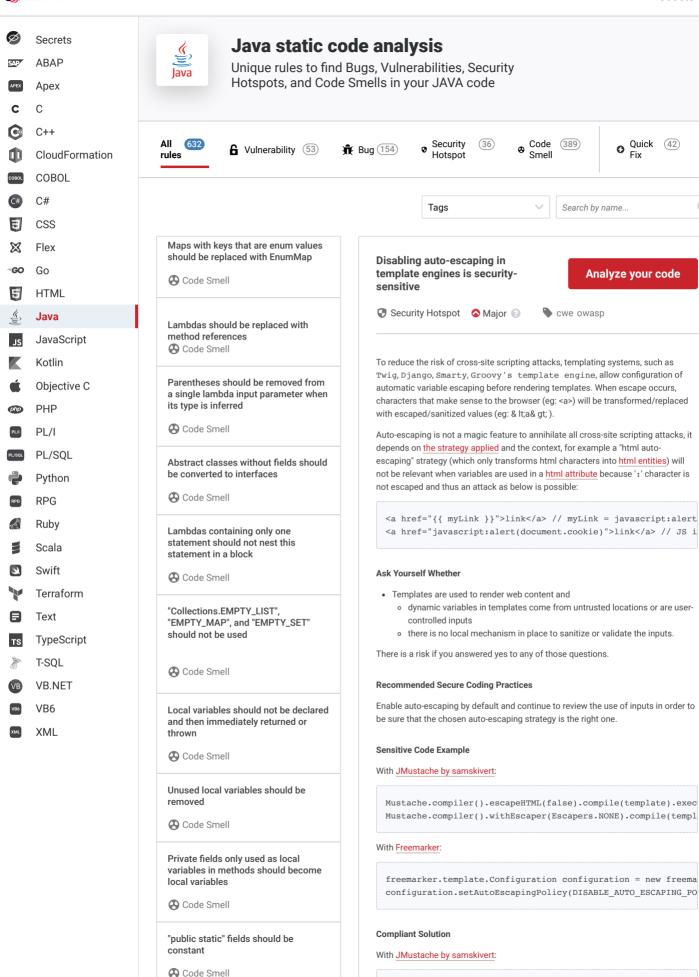
Products ✓

O Quick 42 Fix

(389)

Search by name.

Smell



Loops should not contain more than a single "break" or "continue" statement

Analyze your code cwe owasp To reduce the risk of cross-site scripting attacks, templating systems, such as Twig, Diango, Smarty, Groovy's template engine, allow configuration of automatic variable escaping before rendering templates. When escape occurs, characters that make sense to the browser (eg: <a>) will be transformed/replaced with escaped/sanitized values (eg: & lt;a& gt;). Auto-escaping is not a magic feature to annihilate all cross-site scripting attacks, it depends on the strategy applied and the context, for example a "html autoescaping" strategy (which only transforms html characters into html entities) will not be relevant when variables are used in a httml attribute because ': 'character is not escaped and thus an attack as below is possible: link // myLink = javascript:alert link // JS i • Templates are used to render web content and o dynamic variables in templates come from untrusted locations or are usero there is no local mechanism in place to sanitize or validate the inputs.

Mustache.compiler().compile(template).execute(context); // C Mustache.compiler().escapeHTML(true).compile(template).execu Declarations should use Java collection interfaces such as "List" rather than specific implementation classes such as "LinkedList"

Code Smell

"switch" statements should have at least 3 "case" clauses

Code Smell

A "while" loop should be used instead of a "for" loop

Code Smell

With Freemarker. See <u>"setAutoEscapingPolicy" documentation</u> for more details.

freemarker.template.Configuration configuration = new freema
configuration.setAutoEscapingPolicy(ENABLE_IF_DEFAULT_AUTO_E

See

- OWASP Top 10 2021 Category A3 Injection
- OWASP Cheat Sheet XSS Prevention Cheat Sheet
- OWASP Top 10 2017 Category A7 Cross-Site Scripting (XSS)
- MITRE, CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

Available In:

sonarcloud 🙆 | sonarqube

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.

Privacy Policy