Code Smell (144)





Scala

Swift

Text

T-SQL

**VB.NET** 

VB6

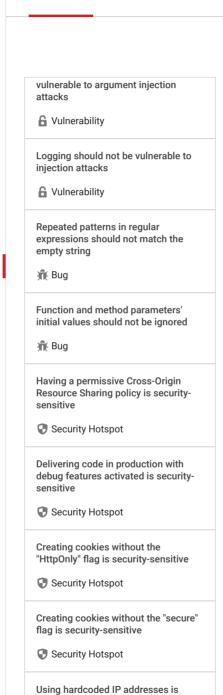
XML

Terraform

**TypeScript** 

D

тѕ



security-sensitive Security Hotspot

concisely

Code Smell

Regular expression quantifiers and character classes should be used

Character classes should be preferred over reluctant quantifiers in regular

All rules (268)

PHP static code analysis

6 Vulnerability (40)

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

**R** Bug (51)

Tags Search by name... Objects should not be created to be dropped Analyze your code immediately without being used 📆 Bug 🔷 Major 🕝 There is no good reason to create a new object to not do anything with it. Most of the time, this is due to a missing piece of code and so could lead to an unexpected behavior in production. If it was done on purpose because the constructor has side-effects, then that side-effect code should be moved into a separate, static method and called directly. **Noncompliant Code Example** if (\$x < 0) { new foo; // Noncompliant **Compliant Solution** Svar = NULL: if (\$x < 0) { \$var = new foo; Available In: sonarlint ⊕ | sonarcloud ♦ | sonarqube

Security Hotspot 33

© 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

expressions
☼ Code Smell
A subclass should not be in the same "catch" clause as a parent class  Code Smell
Jump statements should not be redundant  Code Smell
"catch" clauses should do more than rethrow  Code Smell
"&&" and "  " should be used