

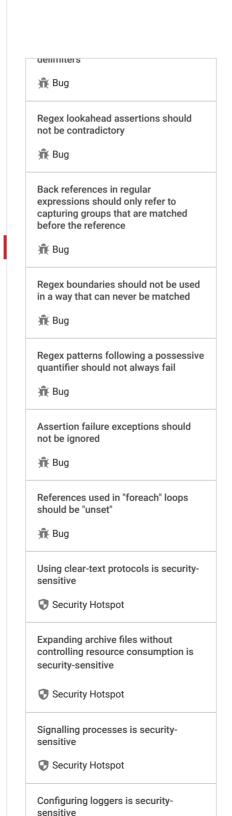




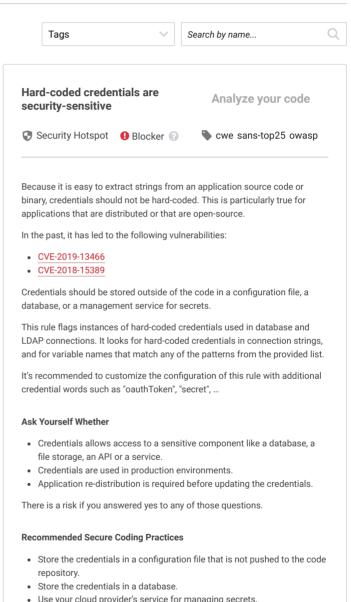
PHP static code analysis

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





Security Hotspot



• If a password has been disclosed through the source code: change it.

\$httpUrl = "https://example.domain?user=user&password=6 \$sshUrl = "ssh://user:65DBGgwe4uazdWQA@example.domain"

\$password = "65DBGgwe4uazdWQA"; // Sensitive

\$password = getPassword(); // Compliant

Sensitive Code Example

Compliant Solution

\$user = getUser();

Using weak hashing algorithms is security-sensitive

Security Hotspot

Disabling CSRF protections is security-sensitive

Security Hotspot

Using pseudorandom number generators (PRNGs) is security-sensitive

Security Hotspot

Dynamically executing code is security-sensitive

Security Hotspot

\$httpUrl = "https://example.domain?user=\$user&password= \$sshUrl = "ssh://\$user:\$password@example.domain" // Com

See

- OWASP Top 10 2021 Category A7 Identification and Authentication Failures
- OWASP Top 10 2017 Category A2 Broken Authentication
- MITRE, CWE-798 Use of Hard-coded Credentials
- MITRE, CWE-259 Use of Hard-coded Password
- SANS Top 25 Porous Defenses
- Derived from FindSecBugs rule Hard Coded Password

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