

RPG

Ruby

Scala

Swift

Text

T-SQL

VB.NET

VB6

XML

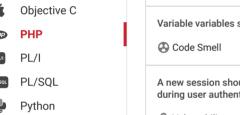
Terraform

TypeScript

J.

月

тѕ



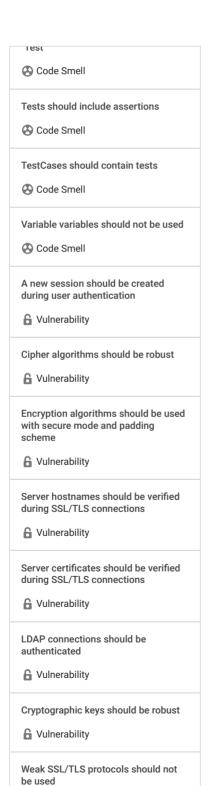


All rules (268)

PHP static code analysis

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





Vulnerability

```
Tags
                                    Search by name...
XPath expressions should
not be vulnerable to injection
                                         Analyze your code
attacks
6 Vulnerability ● Blocker ②
                                  injection cwe owasp
User-provided data, such as URL parameters, should always be considered
untrusted and tainted. Constructing XPath expressions directly from tainted
data enables attackers to inject specially crafted values that changes the
initial meaning of the expression itself. Successful XPath injection attacks
can read sensitive information from XML documents.
Noncompliant Code Example
  $user = $_GET["user"];
  $pass = $_GET["pass"];
  $doc = new DOMDocument();
  $doc->load("test.xml");
  $xpath = new DOMXPath($doc);
  $expression = "/users/user[@name='" . $user . "' and @p
  $xpath->evaluate($expression); // Noncompliant
Compliant Solution
  $user = $_GET["user"];
  $pass = $_GET["pass"];
```

```
$user = $_GET["user"];
$pass = $_GET["pass"];

$doc = new DOMDocument();
$doc->load("test.xml");
$xpath = new DOMXPath($doc);

$user = str_replace("'", "'", $user);
$pass = str_replace("'", "'", $pass);

$expression = "/users/user[@name='" . $user . "' and @p $xpath->evaluate($expression); // Compliant
```

See

- OWASP Top 10 2021 Category A3 Injection
- OWASP Top 10 2017 Category A1 Injection
- MITRE, CWE-20 Improper Input Validation
- <u>MITRE, CWE-643</u> Improper Neutralization of Data within XPath Expressions

Available In:

sonarcloud 🙆 | sonarqube | Develor Edition

Regular expressions should not be vulnerable to Denial of Service attacks

Vulnerability

Hashes should include an unpredictable salt

Vulnerability

Regular expressions should have valid delimiters

Bug

Regex lookahead assertions should not be contradictory

Bug

Back references in regular expressions should only refer to

© 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