

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- 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



# PHP static code analysis

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

All rules 268

Vulnerability 40

Bug 51

Security Hotspot 33

Code Smell 144

Tags ▾

Search by name... 🔍

Code Smell
Files should not have too many lines of code
Code Smell
Lines should not be too long
Code Smell
HTTP response headers should not be vulnerable to injection attacks
Vulnerability
"sleep" should not be called
Vulnerability
Static members should be referenced with "static::"
Bug
"require_once" and "include_once" should be used instead of "require" and "include"
Bug
Errors should not be silenced
Bug
Files should not contain characters before "<?php"
Bug
Method visibility should be explicitly declared
Bug
Controlling permissions is security-sensitive
Security Hotspot
Writing cookies is security-sensitive
Security Hotspot

A reason should be provided when skipping a test

Analyze your code

Code Smell

Major ?

tests bad-practice confusing phpunit

When a test fails due, for example, to infrastructure issues, you might want to ignore it temporarily. But without some kind of notation about why the test is being ignored, it may never be reactivated. Such tests are difficult to address without comprehensive knowledge of the project, and end up polluting their projects.

This rule raises an issue on each test that is marked as incomplete or skipped without a message explaining the reasoning behind it.

Noncompliant Code Example

```
protected function setUp() {
    if (!extension_loaded('mysqli')) {
        $this->markTestSkipped(); // Noncompliant
    }
}

public function testSomething()
{
    $this->assertTrue($result->isValid());
    $this->markTestIncomplete(); // Noncompliant
}
```

Compliant Solution

```
protected function setUp() {
    if (!extension_loaded('mysqli')) {
        $this->markTestSkipped( 'The MySQLi extension is no
    }
}

public function testSomething()
{
    $this->assertTrue($result->isValid());
    $this->markTestIncomplete( 'Testing result validation
}
```

Available In:

sonarlint | sonarcloud | sonarqube


Framework-provided functions should be used to test exceptions

 Code Smell

Unicode-aware versions of character classes should be preferred

 Code Smell

Alias functions should not be used

 Code Smell

Perl-style comments should not be used

 Code Smell

Superglobals should not be accessed directly

 Code Smell

Assessments of code quality or in other assessments and copyrights are the property of their respective owners. All rights are expressly reserved.  
[Privacy Policy](#)