Code Smell (144)





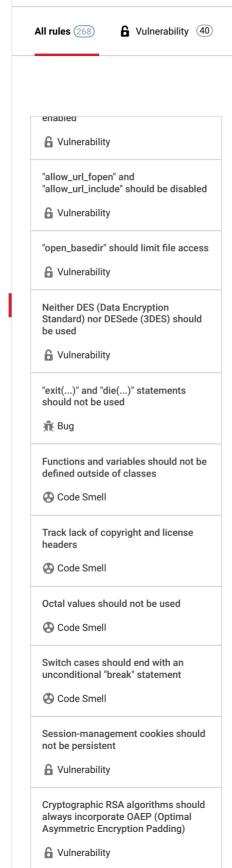


# PHP static code analysis

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

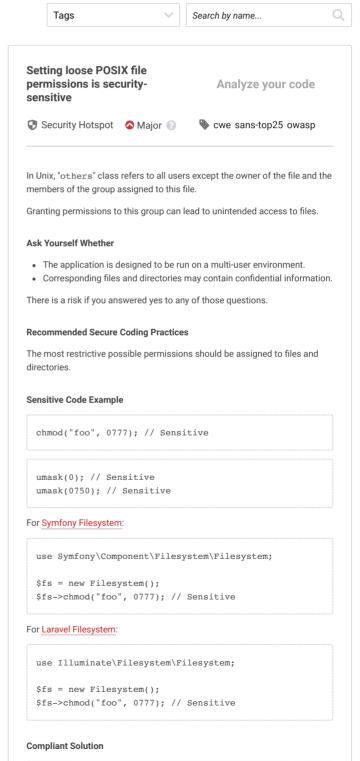
**R** Bug (51)

Security Hotspot 33



SHA-1 and Message-Digest hash

algorithms should not be used in



chmod("foo", 0750); // Compliant

umask(0027); // Compliant

secure contexts

■ Vulnerability

Assertions should not be made at the end of blocks expecting an exception

Rug Bug

Regular expressions should be syntactically valid

Rug Bug

Only one method invocation is expected when testing exceptions

류 Buc

Reading the Standard Input is security-sensitive

## For Symfony Filesystem:

```
use Symfony\Component\Filesystem\Filesystem;

$fs = new Filesystem();

$fs->chmod("foo", 0750); // Compliant
```

### For Laravel Filesystem:

```
use Illuminate\Filesystem\Filesystem;

$fs = new Filesystem();
$fs->chmod("foo", 0750); // Compliant
```

#### See

- OWASP Top 10 2021 Category A1 Broken Access Control
- OWASP Top 10 2021 Category A4 Insecure Design
- OWASP Top 10 2017 Category A5 Broken Access Control
- OWASP File Permission
- MITRE, CWE-732 Incorrect Permission Assignment for Critical Resource
- MITRE, CWE-266 Incorrect Privilege Assignment
- SANS Top 25 Porous Defenses

#### 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