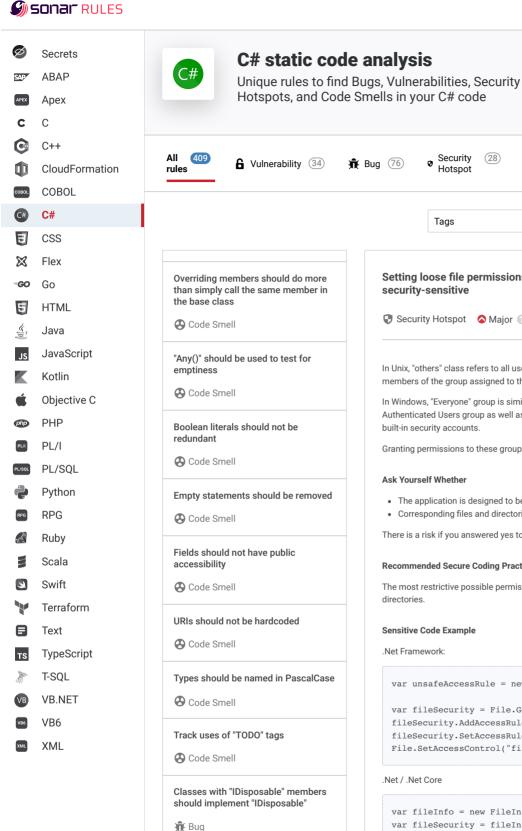
Quick 52 Fix





Calls to "async" methods should not

Child class fields should not shadow

Track lack of copyright and license

.Net Framework

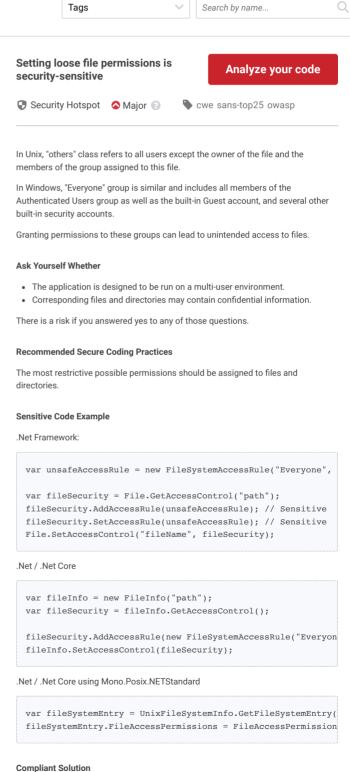
be blocking Code Smell

parent class fields

Code Smell

Code Smell

headers



Security

Hotspot

(28)

Exit methods should not be called

Code Smell

Classes should "Dispose" of members from the classes' own "Dispose" methods

📆 Bug

Reading the Standard Input is security-sensitive

Security Hotspot

Using command line arguments is security-sensitive

Security Hotspot

```
var safeAccessRule = new FileSystemAccessRule("Everyone", Fi

var fileSecurity = File.GetAccessControl("path");
fileSecurity.AddAccessRule(safeAccessRule);
File.SetAccessControl("path", fileSecurity);
```

.Net / .Net Core

```
var safeAccessRule = new FileSystemAccessRule("Everyone", Fi

var fileInfo = new FileInfo("path");
var fileSecurity = fileInfo.GetAccessControl();
fileSecurity.SetAccessRule(safeAccessRule);
fileInfo.SetAccessControl(fileSecurity);
```

.Net / .Net Core using Mono.Posix.NETStandard

```
var fs = UnixFileSystemInfo.GetFileSystemEntry("path");
fs.FileAccessPermissions = FileAccessPermissions.UserExecute
```

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