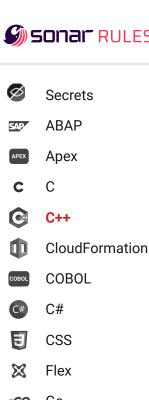
Quick 68 Fix





Go =GO

5 HTML

Java JavaScript

Kotlin

Kubernetes

Objective C

PHP

PL/I

PL/SQL

Python

**RPG** 

Ruby

Scala

Swift

Terraform

Text

**TypeScript** 

T-SQL

**VB.NET** 

VB6

XML



All

578

## C++ static code analysis

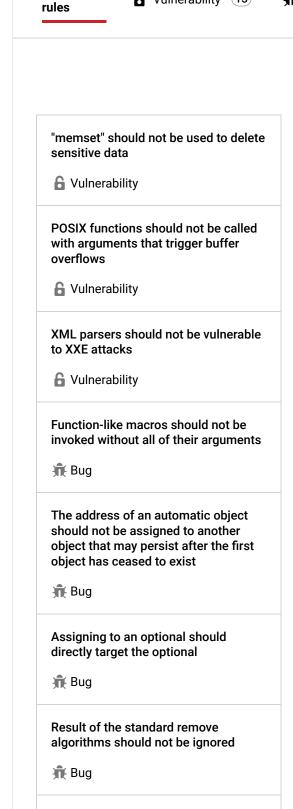
6 Vulnerability (13)

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

**R** Bug (111)

Security Hotspot

Tags



"std::scoped\_lock" should be created

with constructor arguments

Objects should not be sliced

Immediately dangling references

"pthread\_mutex\_t" should be unlocked in the reverse order they were locked

"pthread\_mutex\_t" should be properly

"pthread\_mutex\_t" should not be consecutively locked or unlocked

initialized and destroyed

should not be created

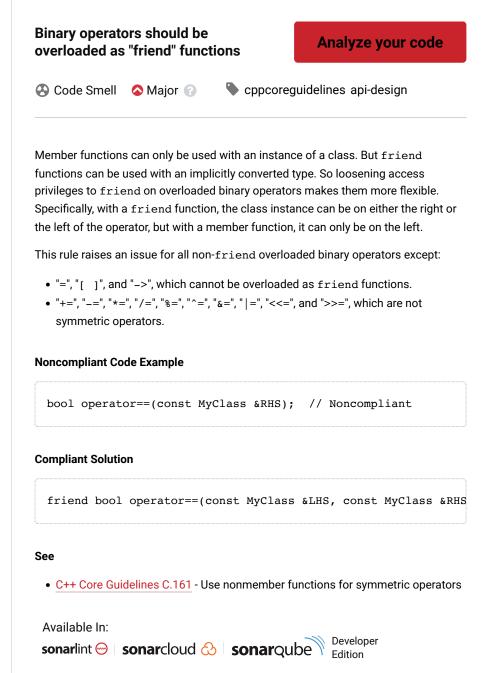
🖷 Bug

📆 Bug

📆 Bug

📆 Bug

📆 Bug



**⇔** Code (436)

Search by name...

© 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** 

| I   |
|---|
| 🖟 Bug   |
| "std::move" and "std::forward" should not be confused   |
| <b>∰</b> Bug  |
| A call to "wait()" on a  "std::condition_variable" should have a  condition   |
| n Bug   |
| A pointer to a virtual base class shall<br>only be cast to a pointer to a derived<br>class by means of dynamic_cast |
| <b>ਜ਼ਿ</b> Bug  |
| Functions with "noreturn" attribute should not return   |
| 👬 Bug   |
| RAII objects should not be temporary  |
| ्रे Bug   |
| "memcmp" should only be called with<br>pointers to trivially copyable types<br>with no padding                      |
| 🙃 Bug   |
| "memcpy", "memmove", and "memset"<br>should only be called with pointers to<br>trivially copyable types             |
| 🙃 Bug   |
| "std::auto_ptr" should not be used  |
| <b>n</b> Bug  |
| Destructors should be "noexcept"  |
| 🖟 Bug   |