



Go =GO

5

Java

JavaScript

HTML

Kotlin

Kubernetes

Objective C

PHP

PL/I

PL/SQL

Python

RPG

Ruby

Scala

Swift

Terraform

Text

TypeScript

T-SQL

VB.NET

VB6

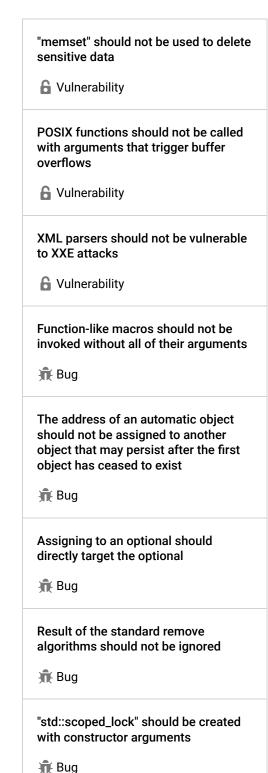
XML

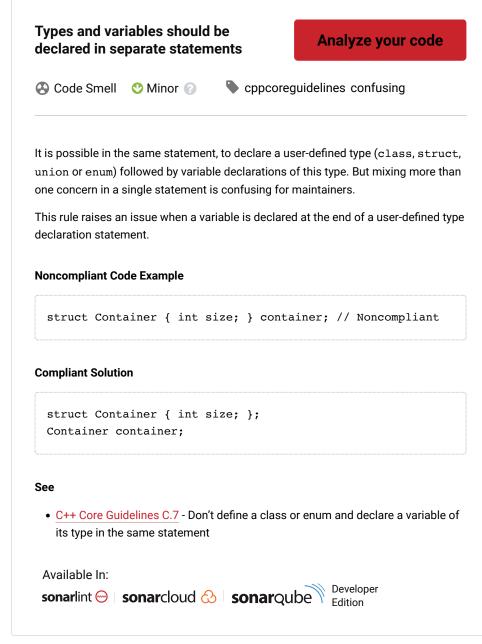


C++ static code analysis

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

o Security **⊗** Code (436) Quick 68 Fix ΑII 578 **R** Bug (111) 6 Vulnerability 13 rules Hotspot Tags Search by name... "memset" should not be used to delete Types and variables should be sensitive data Analyze your code declared in separate statements





© 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**

"pthread_mutex_t" should be unlocked

"pthread_mutex_t" should be properly initialized and destroyed

in the reverse order they were locked

Objects should not be sliced

Immediately dangling references

should not be created

📆 Bug

📆 Bug

📆 Bug

📆 Bug

"pthread_mutex_t" should not be consecutively locked or unlocked

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