



ABAP

- _{АРЕХ} Арех
- **c** C
- C++
- CloudFormation
- COBOL COBOL
- C# C#
- CSS
- X Flex
- **GO** Go
- 5 HTML
- 👙 Java
- Js JavaScript
- Kotlin
- Kubernetes
- **Objective C**
- PP PHP
- PL/I
- PL/SQL PL/SQL
- Python
- RPG RPG
- Ruby
- Scala
- Swift
- **Terraform**
- **Text**
- Ts TypeScript
- T-SQL
- VB VB.NET
- VB6 VB6
- XML XML

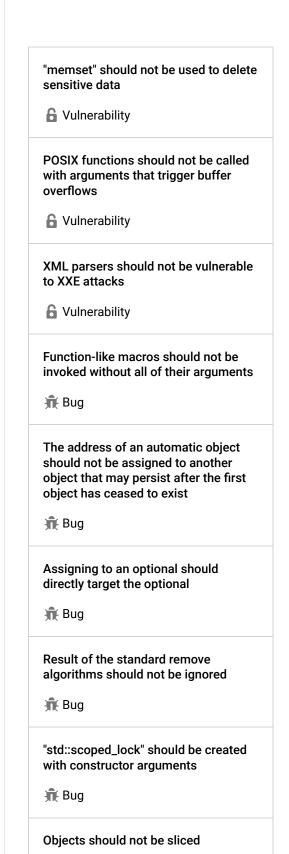


C++ static code analysis

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

All 578 rules	6 Vulnerability 13	∰ Bug 1111	Security 18 Hotspot	Code Smell 436	Quick 68 Fix

Tags



📆 Bug

📆 Bug

📆 Bug

📆 Bug

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

```
Mathematical constants should not
                                           Analyze your code
be hardcoded
since-c++20 clumsy
Starting from C++20, mathematical constants are defined in the header <numbers>.
You should prefer using them instead of hardcoding your own constants.
Noncompliant Code Example
 const double pi = 3.14159265358979323846; // Noncompliant
 auto computeCirclePerimeter(double radius) {
      return 2 * pi * radius;
Compliant Solution
 #include <numbers>
 auto computeCirclePerimeter(double radius) {
      return 2 * std::numbers::pi * radius;
 }
 Available In:
 sonarlint in sonarcloud on sonarqube Developer Edition
```

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
∰ 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