Search by name.





ABAP

Apex Apex

C C

© C++

CloudFormation

COBOL COBOL

C# C#

CSS

GO Go

HTML

🐇 Java

Js JavaScript

Kotlin

Kubernetes

© Objective C

PHP

PL/I

PL/SQL

Python

RPG RPG

Ruby

Scala

Swift

Terraform

■ Text

Ts TypeScript

T-SQL

VB VB.NET

VB6 VB6

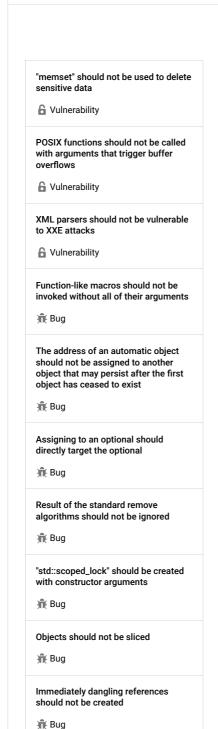
XML XML



C++ static code analysis

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

Tags



"pthread_mutex_t" should be unlocked in the reverse order they were locked

"pthread_mutex_t" should be properly

Bug

```
Functions with "noreturn" attribute
                                                Analyze your code
should not return
👬 Bug \phantom 🕕 Blocker 🕝
                           confusing
The attribute noreturn indicates that a function does not return.
Using this attribute allows the compiler to do some assumptions that can lead to
optimizations. However, if a function with this attribute ever returns, the behavior
becomes undefined.
Noncompliant Code Example
  [[noreturn]] void f () {
    while (1) {
      // ...
      if (/* something*/) {
        return; // Noncompliant, this function should not retur
Compliant Solution
  [[noreturn]] void f() { // Compliant
    while (true) {
      // ...
Or
  void f() {
    while (true) {
      if (/* something*/) {
        return; // Compliant
 Available In:
                                                  Developer
 sonarlint ⊕ | 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

Privacy Policy

initialized and destroyed

in Bug

"pthread_mutex_t" should not be consecutively locked or unlocked twice

in Bug

"std::move" and "std::forward" should not be confused

in Bug

A call to "wait()" on a "std::condition_variable" should have a