

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



6 Vulnerability (13)

rules

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

⊗ Code ΑII 311 Security 18 206 ₩ Bug (74)

> "bool" expressions should not be used as operands to built-in operators other than =, &&, ||, !, ==,

Hotspot

Tags

Analyze your code

Search by name.

based-on-misra suspicious

O Quick 14

!=, unary &, and the conditional operator

Smell

The use of bool operands with other operators is unlikely to be meaningful (or intended). Best case it will be confusing to maintainers, worst case it will not have the intended effect. Either way, it is highly recommended to stick to boolean operators when dealing with bool operands.

This rule allows the detection of such uses, which often occur because the logical operators (&&, | | and !) can be easily confused with the bitwise operators (&, | and ~).

Noncompliant Code Example

```
bool b1 = true;
bool b2 = false;
int8 t s8a:
if ( b1 & b2 ) // Noncompliant
if ( ~b1 ) // Noncompliant
if ( b1 < b2 ) // Noncompliant
if ( b1 ^ b2 ) // Noncompliant
```

Compliant Solution

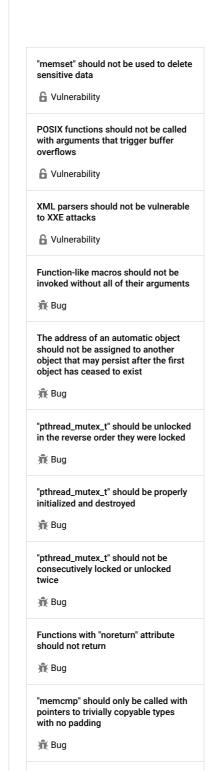
```
if ( b1 && b2 )
if ( !b1 )
if ( b1 == false )
if ( b1 == b2 )
if ( b1 != b2 )
s8a = b1 ? 3 : 7:
```

Exceptions

Operators | = and &= are ignored when used with bool operands. Operator ++ is also ignored with a bool operand because it is covered by rule {rule:cpp:S2668}.

```
void test(bool b1, bool b2, int i1) {
 b1 |= b2; // ignored
 b1++; // ignored here, handled by S2668
 b1 &= b2: // ignored
 b1 &= i1; // Noncompliant; right operand is not a bool
```

• MISRA C++:2008, 4-5-1 - Expressions with type bool shall not be used as operands to built-in operators other than the assignment operator =, the logical operators &&, ||, !, the equality operators == and !=, the unary & operator, and the conditional operator.



Stack allocated memory and nonowned memory should not be freed

Bug

Closed resources should not be accessed

Dynamically allocated memory should be released

👬 Bug

📆 Bug

Freed memory should not be used

Available In:

sonarlint ⊖ | sonarcloud ♂ | sonarqube Developer Edition

© 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