

- Secrets
- ABAP
- Apex
- C**
- C++
- CloudFormation
- COBOL
- C#
- CSS
- Flex
- Go
- 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



C static code analysis

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

All rules 311

Vulnerability 13

Bug 74

Security Hotspot 18

Code Smell 206

Quick Fix 14

Tags

Search by name...



"memset" should not be used to delete sensitive data

Vulnerability

POSIX functions should not be called with arguments that trigger buffer overflows

Vulnerability

XML parsers should not be vulnerable to XXE attacks

Vulnerability

Function-like macros should not be invoked without all of their arguments

Bug

The address of an automatic object should not be assigned to another object that may persist after the first object has ceased to exist

Bug

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

Bug

"pthread_mutex_t" should be properly initialized and destroyed

Bug

"pthread_mutex_t" should not be consecutively locked or unlocked twice

Bug

Functions with "noreturn" attribute should not return

Bug

"memcpy" should only be called with pointers to trivially copyable types with no padding

Bug

Pointers should not be cast to integral types

Analyze your code

Bug Critical based-on-misra cert

The size of integer required to hold a memory address is implementation-dependent. Therefore, casting a pointer (i.e. a memory address) to any integral data type may result in data loss because the integral type is too small to hold the full address value.

When treating a memory address as integer type is absolutely required, you should be sure to use a large enough type to hold all the data.

Noncompliant Code Example

```
int *p;
int addr = ( int ) &p;
```

See

- MISRA C:2004, 11.3 - A cast should not be performed between a pointer type and an integral type.
- MISRA C++:2008, 5-2-9 - A cast should not convert a pointer type to an integral type.
- MISRA C:2012, 11.4 - A conversion should not be performed between a pointer to object and an integer type
- CERT, INT36-C.** - Converting a pointer to integer or integer to pointer

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](#)

Stack allocated memory and non-owned memory should not be freed

 Bug

Closed resources should not be accessed

 Bug

Dynamically allocated memory should be released

 Bug

Freed memory should not be used