

Go =GO

5 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



## **Objective C static code analysis**

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

ΑII 315 6 Vulnerability 10 rules

**R** Bug (75)

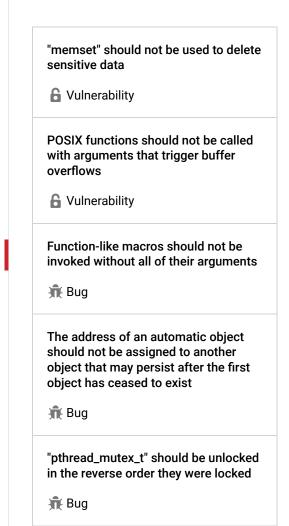
o Security Hotspot

⊗ Code (212)

O Quick 13 Fix

Tags

Search by name...



"pthread\_mutex\_t" should be properly

"pthread\_mutex\_t" should not be

consecutively locked or unlocked

Functions with "noreturn" attribute

"memcmp" should only be called with pointers to trivially copyable types

Stack allocated memory and nonowned memory should not be freed

Closed resources should not be

Dynamically allocated memory should

initialized and destroyed

📆 Bug

twice

📆 Bug

📆 Bug

📆 Bug

📆 Bug

accessed

📆 Bug

be released

📆 Bug

should not return

with no padding

Unused type declarations should Analyze your code be removed based-on-misra unused If a type is declared but not used, then it is unclear to a reviewer if the type is redundant or it has been left unused by mistake. **Noncompliant Code Example** void unusedtype() typedef int local Type; // Noncompliant, unused } See • MISRA C++:2008, 0-1-5 - A project shall not contain unused type declarations. Available In: sonarcloud 🖒 | sonarqube | Develop 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

Freed memory should not be used  Recursion should not be infinite  Bug  Recursion should not be infinite  Bug  Resources should be closed  Bug  Resources should be closed  Code Smell  Switch labels should not be nested inside non-switch blocks  Code Smell  Memory access should be explicitly bounded to prevent buffer overflows  Replication should not lead to unexpected behavior at runtime  Bug  Recursion should not be infinite  Security Bug  Resources should be closed  Code Smell  Switch labels should not be nested inside non-switch blocks  Code Smell	
Memory locations should not be released more than once	Freed memory should not be used
released more than once  ## Bug  Memory access should be explicitly bounded to prevent buffer overflows  ## Bug  Printf-style format strings should not lead to unexpected behavior at runtime  ## Bug  Recursion should not be infinite  ## Bug  Resources should be closed  ## Bug  Hard-coded credentials are security-sensitive  ## Security Hotspot  "goto" should jump to labels declared later in the same function  ## Code Smell  Only standard forms of the "defined" directive should be used  ## Code Smell  Switch labels should not be nested inside non-switch blocks	<b>₩</b> Bug
Memory access should be explicitly bounded to prevent buffer overflows  Bug  Printf-style format strings should not lead to unexpected behavior at runtime  Bug  Recursion should not be infinite  Bug  Resources should be closed  Bug  Hard-coded credentials are security-sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	
Printf-style format strings should not lead to unexpected behavior at runtime	<b>∰</b> Bug
Printf-style format strings should not lead to unexpected behavior at runtime  Bug  Recursion should not be infinite  Bug  Resources should be closed  Bug  Hard-coded credentials are security-sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	
lead to unexpected behavior at runtime  Bug  Recursion should not be infinite  Bug  Resources should be closed  Bug  Hard-coded credentials are security-sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	Rug
Recursion should not be infinite  Resources should be closed  Bug  Hard-coded credentials are security- sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	lead to unexpected behavior at
Resources should be closed  Bug  Hard-coded credentials are security- sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	Rug
Resources should be closed  **Bug  Hard-coded credentials are security- sensitive  **Security Hotspot  "goto" should jump to labels declared later in the same function  **Code Smell  Only standard forms of the "defined" directive should be used  **Code Smell  Switch labels should not be nested inside non-switch blocks	Recursion should not be infinite
Hard-coded credentials are security- sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	<b>∰</b> Bug
Hard-coded credentials are security- sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	Resources should be closed
sensitive  Security Hotspot  "goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	<b>n</b> Bug
"goto" should jump to labels declared later in the same function  Code Smell  Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	
Iater in the same function Code Smell Only standard forms of the "defined" directive should be used Code Smell Switch labels should not be nested inside non-switch blocks	Security Hotspot
Only standard forms of the "defined" directive should be used  Code Smell  Switch labels should not be nested inside non-switch blocks	
directive should be used Code Smell  Switch labels should not be nested inside non-switch blocks	Code Smell
Switch labels should not be nested inside non-switch blocks	
inside non-switch blocks	
Code Smell	