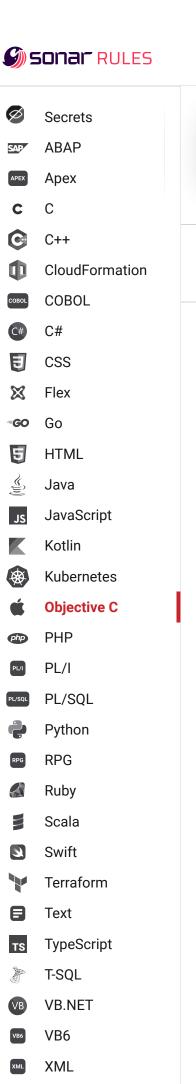
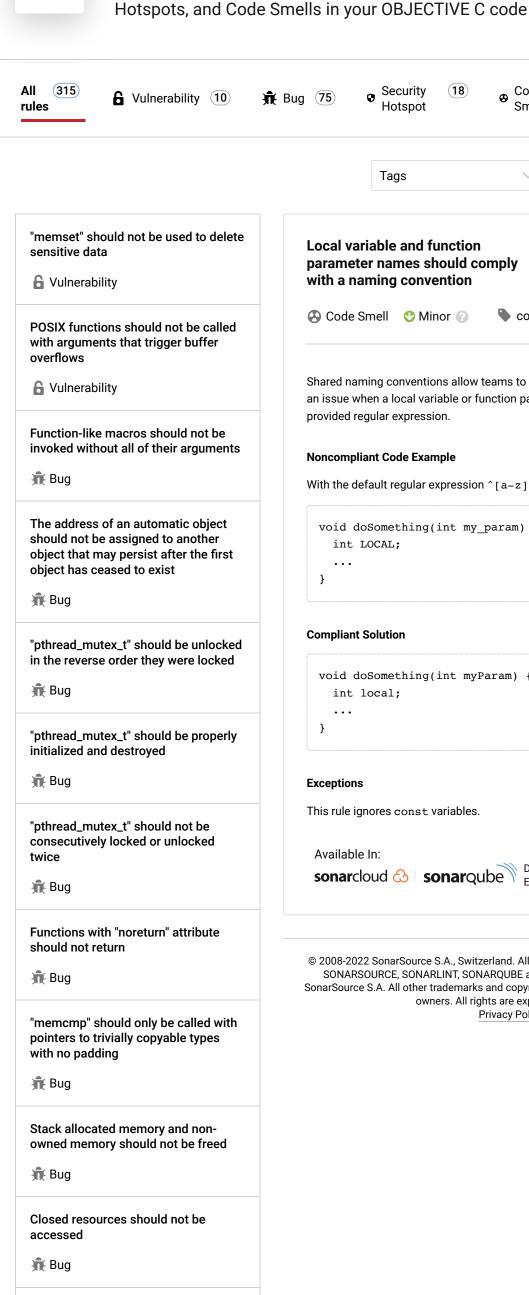
O Quick 13
Fix





Dynamically allocated memory should

📆 Bug

Objective C static code analysis

Unique rules to find Bugs, Vulnerabilities, Security

o Security

Hotspot

```
Tags
                                        Search by name...
Local variable and function
                                              Analyze your code
parameter names should comply
with a naming convention
convention
Shared naming conventions allow teams to collaborate effectively. This rule raises
an issue when a local variable or function parameter name does not match the
provided regular expression.
Noncompliant Code Example
With the default regular expression [a-z][a-zA-z0-9]*:
 void doSomething(int my_param) {
    int LOCAL;
Compliant Solution
 void doSomething(int myParam) {
    int local;
Exceptions
This rule ignores const variables.
 Available In:
sonarcloud 🚳 | sonarqube | Developer
```

⊗ Code (212)

© 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.

| 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 | ₩ 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 | ∰ 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 | ∰ 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 | n 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 | |
| | |