

Ruby

Scala Swift

Terraform

Text

TypeScript

T-SQL

VB.NET

VB6

XML

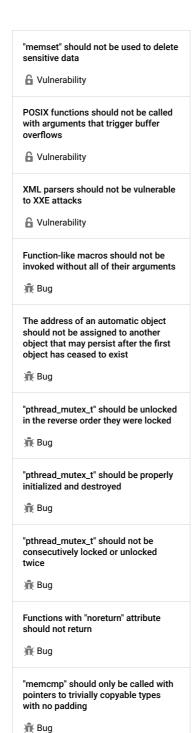


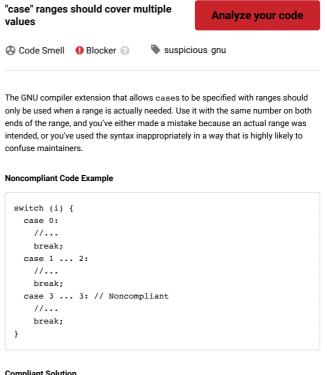
C static code analysis

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

Security ⊗ Code 206 O Quick 14 ΑII 311 18 6 Vulnerability (13) **₩** Bug (74) rules Hotspot Smell

Tags





Search by name.

Compliant Solution

```
switch (i) {
 case 0:
    //...
    break;
  case 1 ... 2:
    //...
    break;
  case 3:
    //...
    break;
}
```

```
switch (i) {
  case 0:
    //...
    break:
  case 1 ... 2:
    //...
    break;
  case 3 ... 5:
    //...
    break;
}
```

Stack allocated memory and nonowned 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

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