

Ruby
Scala
Swift

Terraform

■ Text

Ts TypeScript

→ T-SQL

VB VB.NET

xmL XML

C static code analysis

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

All 311 rules	6 Vulnerability 13	∰ Bug 74	Security 18 Hotspot	Ode Smell	06) Quick 14) Fix	
			Tags	V S	serch by name	Q
"memset" s	should not be used to delete ata	Track I	ack of copyright and	d license	Analyze vour code	

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 "pthread_mutex_t" should be properly initialized and destroyed # Bua "pthread_mutex_t" should not be consecutively locked or unlocked

Bug

Bug

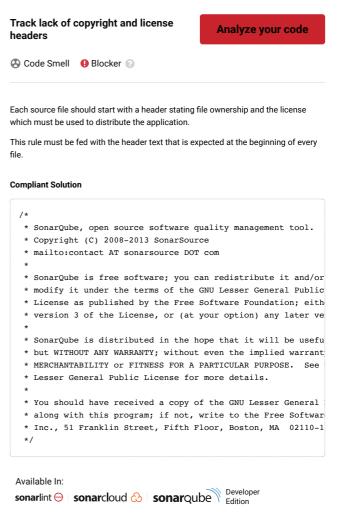
🖷 Bug

should not return

with no padding

Functions with "noreturn" attribute

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



© 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 nonowned memory should not be freed

R
Bug

Closed resources should not be
accessed
Bug

Dynamically allocated memory should
be released
Bug

Freed memory should not be used