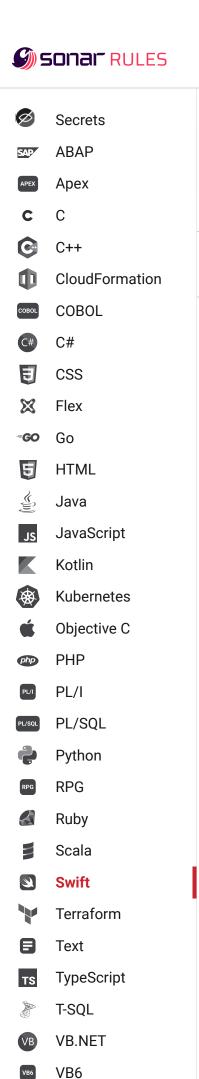
Code Smell 99



XML



## **Swift static code analysis**

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

**♣** Bug 14

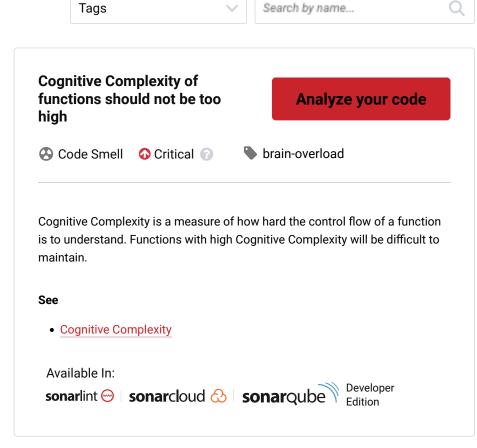
Hard-cor	ded credent	ials are security	<i>-</i> -
sensitive		iais are security	,
Secu	rity Hotspot	:	
	ame or diffe	ames should no er only by	ot
Code	Smell		
Cipher al	lgorithms sl	hould be robust	
<b>읍</b> Vulne	erability		
	eak hashing sensitive	algorithms is	
Security	rity Hotspot	:	
•	e Complexit ot be too hi	ty of functions gh	
Code	Smell		
"try!" sho	ould not be ι	used	
Code	Smell		
String lit	erals should	d not be duplica	ited
Code	Smell		
Function empty	s and closu	ires should not	be
Code	Smell		
	on elements unconditio	should not be	
📆 Bug			
Collectio make se		nparisons shou	ld
🕀 Bug			
	ot have exa	nditional structo	ıre
🕕 Bug			

update their left operands

Precedence and associativity of standard operators should not be

Rug Bug

changed



Security Hotspot (3)

© 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

<b>∰</b> Bug
Return values from functions without side effects should not be ignored
<b>∰</b> Bug
Related "if/else if" statements and "cases" in a "switch" should not have the same condition
Rug
Identical expressions should not be used on both sides of a binary operator
🖟 Bug
All code should be reachable
Rug
Loops with at most one iteration should be refactored
🖟 Bug
"IBInspectable" should be used correctly
Functions should not have identical implementations
Ternary operators should not be nested
Closure expressions should not be nested too deeply
Code Smell
Backticks should not be used around