




 Secrets


 ABAP


 Apex


 C


 C++


 CloudFormation


 COBOL


 C#


 CSS


 Flex


 Go


 HTML


 **Java**


 JavaScript


 Kotlin


 Objective C


 PHP


 PL/I


 PL/SQL


 Python


 RPG


 Ruby


 Scala


 Swift


 Terraform


 Text


 TypeScript

 T-SQL

 VB.NET

 VB6

 XML



Java static code analysis

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

All rules632

Vulnerability53

Bug154


Security Hotspot36

Code Smell389


Quick Fix42


Tags ▾

Search by name... 🔍


 Bug


Methods should not be named "toString", "hashCode" or "equal"




 Bug


"Thread.run()" should not be called directly




 Bug


"equals" method overrides should accept "Object" parameters




 Bug


The Object.finalize() method should not be called




 Security Hotspot


Enabling file access for WebViews is security-sensitive




 Security Hotspot


Enabling JavaScript support for WebViews is security-sensitive




 Security Hotspot


Constructing arguments of system commands from user input is security-sensitive




 Security Hotspot


Using unencrypted files in mobile applications is security-sensitive




 Security Hotspot


Using biometric authentication without a cryptographic solution is security-sensitive




 Security Hotspot

Using unencrypted databases in mobile applications is security-sensitive





 Security Hotspot


Authorizing non-authenticated users to use keys in the Android KeyStore is security-sensitive



Execution of the Garbage Collector should be triggered only by the JVM

 Code Smell

 Critical

 unpredictable bad-practice




Calling `System.gc()` or `Runtime.getRuntime().gc()` is a bad idea for a simple reason: there is no way to know exactly what will be done under the hood by the JVM because the behavior will depend on its vendor, version and options:

- Will the whole application be frozen during the call?
- Is the `-XX:DisableExplicitGC` option activated?
- Will the JVM simply ignore the call?
- ...

Like for `System.gc()`, there is no reason to manually call `runFinalization()` to force the call of finalization methods of any objects pending finalization.

An application relying on these unpredictable methods is also unpredictable and therefore broken. The task of running the garbage collector and calling `finalize()` methods should be left exclusively to the JVM.

Available In:





 **sonarlint** |  **sonarcloud** |  **sonarqube**

© 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](#)

https://rules.sonarsource.com/java/RSPEC-1215

1/2

security-sensitive  Security Hotspot
Allowing user enumeration is security-sensitive  Security Hotspot
Allowing requests with excessive content length is security-sensitive  Security Hotspot
Disabling auto-escaping in template engines is security-sensitive  Security Hotspot
Allowing deserialization of LDAP