




 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... 🔍

⚙️ Code Smell

"String#replace" should be preferred to "String#replaceAll"

⚙️ Code Smell

Derived exceptions should not hide their parents' catch blocks

⚙️ Code Smell

String offset-based methods should be preferred for finding substrings from offsets

⚙️ Code Smell

"default" clauses should be last

⚙️ Code Smell

"equals" method parameters should not be marked "@Nonnull"

⚙️ Code Smell

A conditionally executed single line should be denoted by indentation

⚙️ Code Smell

Conditionals should start on new lines

⚙️ Code Smell

Cognitive Complexity of methods should not be too high

⚙️ Code Smell

Factory method injection should be used in "@Configuration" classes

⚙️ Code Smell

"static" base class members should not be accessed via derived types

⚙️ Code Smell

Instance methods should not write to "static" fields

⚙️ Code Smell

"SecureRandom" seeds should not be predictable

Analyze your code

VulnerabilityCritical🔍cwe cert owasp pitfall

The `java.security.SecureRandom` class provides a strong random number generator (RNG) appropriate for cryptography. However, seeding it with a constant or another predictable value will weaken it significantly. In general, it is much safer to rely on the seed provided by the `SecureRandom` implementation.

This rule raises an issue when `SecureRandom.setSeed()` or `SecureRandom(byte[])` are called with a seed that is either one of:

- a constant
- the system time

Noncompliant Code Example

```
SecureRandom sr = new SecureRandom();
sr.setSeed(123456L); // Noncompliant
int v = sr.next(32);

sr = new SecureRandom("abcdefghijklmnop").getBytes("us-ascii")
v = sr.next(32);
```




Compliant Solution

```
SecureRandom sr = new SecureRandom();
int v = sr.next(32);
```

See

- [OWASP Top 10 2021 Category A2](#) - Cryptographic Failures
- [OWASP Top 10 2017 Category A6](#) - Security Misconfiguration
- [MITRE, CWE-330](#) - Use of Insufficiently Random Values
- [MITRE, CWE-332](#) - Insufficient Entropy in PRNG
- [MITRE, CWE-336](#) - Same Seed in Pseudo-Random Number Generator (PRNG)
- [MITRE, CWE-337](#) - Predictable Seed in Pseudo-Random Number Generator (PRNG)
- [CERT, MSC63J](#) - Ensure that SecureRandom is properly seeded





Available In:

 |  | 

© 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-4347

1/2

<div><div>"indexOf" checks should not be for positive numbers</div><div> Code Smell</div></div>
<div><div>Method overrides should not change contracts</div><div> Code Smell</div></div>
<div><div>Whitespace and control characters in literals should be explicit</div><div> Code Smell</div></div>
<div><div>Null should not be returned from a "Boolean" method</div><div> Code Smell</div></div>