

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- 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



# Ruby static code analysis

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

All rules 42

Bug 7

Security Hotspot 2

Code Smell 33

Tags ▾

Search by name... 🔍

Using hardcoded IP addresses is security-sensitive
Security Hotspot
Multi-line comments should not be empty
Code Smell
Boolean checks should not be inverted
Code Smell
Unused local variables should be removed
Code Smell
function and block parameter names should comply with a naming convention
Code Smell
Class names should comply with a naming convention
Code Smell
Method names should comply with a naming convention
Code Smell
Track uses of "TODO" tags
Code Smell
Track lack of copyright and license headers
Code Smell
Octal values should not be used
Code Smell
"case" statements should not be nested
Code Smell

## Methods should not have identical implementations

Analyze your code

Code Smell

Major ?

confusing duplicate suspicious

When two methods have the same implementation, either it was a mistake - something else was intended - or the duplication was intentional, but may be confusing to maintainers. In the latter case, one implementation should invoke the other.

### Noncompliant Code Example

```
class Box
  def initialize(length, width, height)
    @length, @width, @height = length, width, height
  end

  def getVolume
    area = @length * @width
    return area * @height
  end

  def getArea
    area = @length * @width
    return area * @height
  end
end
```

### Compliant Solution

```
class Box
  def initialize(length, width, height)
    @length, @width, @height = length, width, height
  end


  def getVolume
    return getArea * @height
  end

  def getArea
    return @length * @width
  end
end
```


### Exceptions

Methods with fewer than 2 statements are ignored.


Control flow statements "if", "for", "while", "until", "case" and "begin...rescue" should not be nested too deeply

 Code Smell


"if ... else if" constructs should end with "else" clauses

 Code Smell

Expressions should not be too complex

 Code Smell

Ruby parser failure

 Code Smell

Functions should not have too many

© 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](#)