sonar RULES

Products ⌄

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

JavaScript static code analysis

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

All rules `285` | 🔒 Vulnerability `29` | 🐛 Bug `62` | 🛡 Security Hotspot `43` | ⚙ Code Smell `151` | 🔧 Quick Fix `41`

Tags ⌄   Search by name... 🔍

🐛 Bug

Variables should not be self-assigned

🐛 Bug

Function argument names should be unique

🐛 Bug

Property names should not be duplicated within a class or object literal

🐛 Bug

Bitwise operators should not be used in boolean contexts

🐛 Bug

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

Allowing requests with excessive content length is security-sensitive
🛡 Security Hotspot

Statically serving hidden files is security-sensitive
🛡 Security Hotspot

Using intrusive permissions is security-sensitive
🛡 Security Hotspot

Disabling auto-escaping in template engines is security-sensitive
🛡 Security Hotspot

Using shell interpreter when executing OS commands is security-sensitive
🛡 Security Hotspot

Setting loose POSIX file permissions is security-sensitive
🛡 Security Hotspot

"default" clauses should be last   **Analyze your code**

⚙ Code Smell   ⊘ Critical ❓

`switch` can contain a `default` clause for various reasons: to handle unexpected values, to show that all the cases were properly considered.

For readability purpose, to help a developer to quickly find the default behavior of a `switch` statement, it is recommended to put the `default` clause at the end of the `switch` statement. This rule raises an issue if the `default` clause is not the first or the last one of the `switch`'s cases.

Noncompliant Code Example

```
switch (param) {
  case 0:
    doSomething();
    break;
  default: // default clause should be the first or last one
    error();
    break;
  case 1:
    doSomethingElse();
    break;
}
```

Compliant Solution

```
switch (param) {
  case 0:
    doSomething();
    break;
  case 1:
    doSomethingElse();
    break;
  default:
    error();
    break;
}
```

Available In:

sonarlint 😊 | **sonar**cloud ☁ | **sonar**qube 📶

Formatting SQL queries is security-sensitive

🛡 Security Hotspot

Comma operator should not be used

☢ Code Smell

Regular expressions should not contain empty groups

☢ Code Smell

Regular expressions should not contain multiple spaces

☢ Code Smell