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

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...

Related "if/else if" statements should not have the same condition
Bug

Objects should not be created to be dropped immediately without being used
Bug

Identical expressions should not be used on both sides of a binary operator
Bug

All code should be reachable
Bug

Loops with at most one iteration should be refactored
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

Equality operators should not be used in "for" loop termination conditions

Analyze your code

Code Smell Critical cwe suspicious

Testing `for` loop termination using an equality operator (`==` and `!=`) is dangerous, because it could set up an infinite loop. Using a broader relational operator instead casts a wider net, and makes it harder (but not impossible) to accidentally write an infinite loop.

Noncompliant Code Example

```
for (var i = 1; i != 10; i += 2)  // Noncompliant. Infinite;
{
  //...
}
```

Compliant Solution

```
for (var i = 1; i <= 10; i += 2)  // Compliant
{
  //...
}
```

Exceptions

Equality operators are ignored if the loop counter is not modified within the body of the loop and either:

- starts below the ending value and is incremented by 1 on each iteration.
- starts above the ending value and is decremented by 1 on each iteration.

Equality operators are also ignored when the test is against `null`.

```
for (var i = 0; arr[i] != null; i++) {
  // ...
}

for (var i = 0; (item = arr[i]) != null; i++) {
  // ...
}
```

See

- MITRE, CWE-835 - Loop with Unreachable Exit Condition ('Infinite Loop')

Available In:
sonarlint sonarcloud sonarqube

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