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

C# static code analysis

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

All rules 409 | Vulnerability 34 | Bug 76 | Security Hotspot 28 | Code Smell 271 | Quick Fix 52

Tags

Search by name...

algorithms is security-sensitive

Security Hotspot

Using pseudorandom number generators (PRNGs) is security-sensitive

Security Hotspot

Parameter names should match base declaration and other partial definitions

Code Smell

"ValueTask" should be consumed correctly

Code Smell

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

Code Smell

"default" clauses should be first or last

Code Smell

Unread "private" fields should be removed

Code Smell

Base class methods should not be hidden

Code Smell

Inherited member visibility should not be decreased

Code Smell

Threads should not lock on objects with weak identity

Code Smell

A conditionally executed single line should be denoted by indentation

Code Smell

Conditionals should start on new lines

"async" and "await" should not be used as identifiers

Analyze your code

Code Smell   Blocker   pitfall

Since C# 5.0, `async` and `await` are contextual keywords. Contextual keywords do have a particular meaning in some contexts, but can still be used as variable names. Keywords, on the other hand, are always reserved, and therefore are not valid variable names. To avoid any confusion though, it is best to not use `async` and `await` as identifiers.

Noncompliant Code Example

```
int await = 42; // Noncompliant
```

Compliant Solution

```
int someOtherName = 42;
```

Available In:

sonarlint sonarcloud sonarqube

Code Smell

Assemblies should have version information

Code Smell

Exception types should be "public"

Code Smell

Cognitive Complexity of methods should not be too high

Code Smell

"params" should not be introduced on overrides

Code Smell