

- 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...

Inheritance tree of classes should not be too deep

Code Smell

Nested blocks of code should not be left empty

Code Smell

Methods should not have too many parameters

Code Smell

Collapsible "if" statements should be merged

Code Smell

OS commands should not be vulnerable to argument injection attacks

Vulnerability

Logging should not be vulnerable to injection attacks

Vulnerability

Empty collections should not be accessed or iterated

Bug

Mutable, non-private fields should not be "readonly"

Bug

"string.ToArray()" should not be called redundantly

Bug

"base.Equals" should not be used to check for reference equality in "Equals" if "base" is not "object"

Bug

Property assignments should not be made for "readonly" fields not constrained to reference types

One-way "OperationContract" methods should have "void" return type

Analyze your code

Bug Major

When declaring a Windows Communication Foundation (WCF) `OperationContract` method one-way, that service method won't return any result, not even an underlying empty confirmation message. These are fire-and-forget methods that are useful in event-like communication. Specifying a return type therefore does not make sense.

Noncompliant Code Example

```
[ServiceContract]
interface IMyService
{
    [OperationContract(IsOneWay = true)]
    int SomethingHappened(int parameter); // Noncompliant
}
```

Compliant Solution





```
[ServiceContract]
interface IMyService
{
    [OperationContract(IsOneWay = true)]
    void SomethingHappened(int parameter);
}
```

Exceptions

The rule doesn't report if `OperationContractAttribute.AsyncPattern` is set to `true`.

Available In:

sonarlint sonarcloud sonarqube

 Bug
<p>Flags enumerations should explicitly initialize all their members</p>  Bug
<p>"GetHashCode" should not reference mutable fields</p>  Bug
<p>Results of integer division should not be assigned to floating point variables</p>  Bug
<p>Integral numbers should not be shifted by zero or more than their number of bits-1</p>