C# static code analysis

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

ASP.NET HTTP request validation feature should not be disabled

Analyze your code

Vulnerability   Major

ASP.Net has a feature to validate HTTP requests to prevent potentially dangerous content to perform a cross-site scripting (XSS) attack. There is no reason to disable this mechanism even if other checks to prevent XXS attacks are in place.

This rule raises an issue if a method with parameters is marked with `System.Web.Mvc.HttpPostAttribute` and not `System.Web.Mvc.ValidateInputAttribute(true)`.

Noncompliant Code Example

```
public class FooBarController : Controller
{
    [HttpPost] // Noncompliant
    [ValidateInput(false)]
    public ActionResult Purchase(string input)
    {
        return Foo(input);
    }

    [HttpPost] // Noncompliant
    public ActionResult PurchaseSomethingElse(string input)
    {
        return Foo(input);
    }
}
```

Compliant Solution

```
public class FooBarController : Controller
{
    [HttpPost]
    [ValidateInput(true)] // Compliant
    public ActionResult Purchase(string input)
    {
        return Foo(input);
    }
}
```

Exceptions

Parameterless methods marked with `System.Web.Mvc.HttpPostAttribute` will not trigger this issue.

See

- OWASP Top 10 2017 Category A7 - Cross-Site Scripting (XSS)
- MITRE, CWE-79 - Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- SANS Top 25 - Insecure Interaction Between Components

A close curly brace should be located at the beginning of a line

Code Smell

Tabulation characters should not be used

Code Smell

Methods and properties should be named in PascalCase

Code Smell

Track uses of in-source issue suppressions

Code Smell

- OWASP ASP.NET Request Validation

Deprecated

This rule is deprecated; use {rule:csharpsquid:S5753} instead.

Available In:

sonarlint | sonarcloud | sonarqube