

- 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...



A "while" loop should be used instead of a "for" loop

Code Smell

"Equals" and the comparison operators should be overridden when implementing "IComparable"

Code Smell

Nested code blocks should not be used

Code Smell

Overriding members should do more than simply call the same member in the base class

Code Smell

"Any()" should be used to test for emptiness

Code Smell

Boolean literals should not be redundant

Code Smell

Empty statements should be removed

Code Smell

Fields should not have public accessibility

Code Smell

URIs should not be hardcoded

Code Smell

Types should be named in PascalCase

Code Smell

Track uses of "TODO" tags

Code Smell

Classes with "IDisposable" members should implement "IDisposable"

Bug

### Allowing requests with excessive content length is security-sensitive

Analyze your code

Security Hotspot Major cwe owasp

Rejecting requests with significant content length is a good practice to control the network traffic intensity and thus resource consumption in order to prevent DoS attacks.

#### Ask Yourself Whether

- size limits are not defined for the different resources of the web application.
- the web application is not protected by **rate limiting** features.
- the web application infrastructure has limited resources.

There is a risk if you answered yes to any of those questions.

#### Recommended Secure Coding Practices

- For most of the features of an application, it is recommended to limit the size of requests to:
  - lower or equal to 8mb for file uploads.
  - lower or equal to 2mb for other requests.

It is recommended to customize the rule with the limit values that correspond to the web application.

#### Sensitive Code Example

```
using Microsoft.AspNetCore.Mvc;

public class MyController : Controller
{
    [HttpPost]
    [DisableRequestSizeLimit] // Sensitive: No size limit
    [RequestSizeLimit(10000000)] // Sensitive: 10MB is more
    public IActionResult PostRequest(Model model)
    {
        // ...
    }

    [HttpPost]
    [RequestFormLimits(MultipartBodyLengthLimit = 8000000)]
    public IActionResult MultipartFormRequest(Model model)
    {
        // ...
    }
}
```


In Web.config:

```
<configuration>
  <system.web>
    <httpRuntime maxRequestLength="81920" executionTimeo
    <!-- Sensitive: maxRequestLength is exprimed in KB,
```


Calls to "async" methods should not be blocking

 Code Smell

Child class fields should not shadow parent class fields

 Code Smell

Track lack of copyright and license headers

 Code Smell

Exit methods should not be called

 Code Smell

```
</system.web>
<system.webServer>
  <security>
    <requestFiltering>
      <requestLimits maxAllowedContentLength="8388
        <!-- Sensitive: maxAllowedContentLength is e
    </requestFiltering>
  </security>
</system.webServer>
</configuration>
```

#### Compliant Solution

```
using Microsoft.AspNetCore.Mvc;

public class MyController : Controller
{
    [HttpPost]
    [RequestSizeLimit(8000000)] // Compliant: 8MB
    public IActionResult PostRequest(Model model)
    {
        // ...
    }

    [HttpPost]
    [RequestFormLimits(MultipartBodyLengthLimit = 8000000)]
    public IActionResult MultipartFormRequest(Model model)
    {
        // ...
    }
}
```

In Web.config:

```
<configuration>
  <system.web>
    <httpRuntime maxRequestLength="8192" executionTimeou
    <!-- Compliant: maxRequestLength is exprimed in KB,
  </system.web>
  <system.webServer>
    <security>
      <requestFiltering>
        <requestLimits maxAllowedContentLength="8388
        <!-- Comliant: maxAllowedContentLength is ex
      </requestFiltering>
    </security>
  </system.webServer>
</configuration>
```

#### See

- [OWASP Top 10 2021 Category A5](#) - Security Misconfiguration
- [Owasp Cheat Sheet](#) - Owasp Denial of Service Cheat Sheet
- [OWASP Top 10 2017 Category A6](#) - Security Misconfiguration
- [MITRE, CWE-770](#) - Allocation of Resources Without Limits or Throttling
- [MITRE, CWE-400](#) - Uncontrolled Resource Consumption
- [Web.config](#) - XML-formatted config file for IIS applications

Available In:

sonarcloud  | sonarqube 