




 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





TypeScript static code analysis


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


All rules 279


 Vulnerability 27


 Bug 51


 Security Hotspot 43

 Code Smell 158


 Quick Fix 50


Tags 

Search by name... 


 Code Smell


Primitive types should be omitted from initialized or defaulted declarations

 Code Smell


 Code Smell


Non-null assertions should not be used

 Code Smell


 Code Smell


"undefined" should not be assigned

 Code Smell


 Code Smell


Trailing commas should not be used

 Code Smell

 Code Smell

Array constructors should not be used

 Code Smell

 Code Smell

Quotes for string literals should be used consistently

 Code Smell

 Code Smell

Statements should end with semicolons

 Code Smell

 Code Smell

Comments should not be located at the end of lines of code

 Code Smell

 Code Smell

Loops should not contain more than a single "break" or "continue" statement

 Code Smell

 Code Smell

Variable, property and parameter names should comply with a naming convention


 Code Smell



 Code Smell


Lines should not end with trailing whitespaces

 Code Smell

Delivering code in production with debug features activated is security-sensitive

 Security Hotspot

 Minor 

 cwe error-handling debug

user-experience express.js owasp

Delivering code in production with debug features activated is security-sensitive. It has led in the past to the following vulnerabilities:

- [CVE-2018-1999007](#)
- [CVE-2015-5306](#)
- [CVE-2013-2006](#)

An application's debug features enable developers to find bugs more easily and thus facilitate also the work of attackers. It often gives access to detailed information on both the system running the application and users.

Ask Yourself Whether

- the code or configuration enabling the application debug features is deployed on production servers or distributed to end users.
- the application runs by default with debug features activated.

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

Recommended Secure Coding Practices

Do not enable debug features on production servers or applications distributed to end users.

Sensitive Code Example

[errorhandler Express.js middleware](#) should not be used in production:

```
const express = require('express');
const errorHandler = require('errorhandler');

let app = express();
app.use(errorHandler()); // Sensitive
```

Compliant Solution

[errorhandler Express.js middleware](#) used only in development mode:

```
const express = require('express');
const errorHandler = require('errorhandler');

let app = express();

if (process.env.NODE_ENV === 'development') { // Compliant
  app.use(errorHandler()); // Compliant
}
```

See

<https://rules.sonarsource.com/typescript/RSPEC-4507>

1/2

Files should contain an empty newline at the end

 Code Smell

An open curly brace should be located at the end of a line

 Code Smell

Tabulation characters should not be used

 Code Smell

Function and method names should comply with a naming convention

 Code Smell

- [OWASP Top 10 2021 Category A5](#) - Security Misconfiguration
- [OWASP Top 10 2017 Category A3](#) - Sensitive Data Exposure
- [MITRE, CWE-489](#) - Active Debug Code
- [MITRE, CWE-215](#) - Information Exposure Through Debug Information

Available In:



© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.
[Privacy Policy](#)