




 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

used misleadingly
Code Smell
Collection and array contents should be used
Code Smell
Literals should not be thrown
Code Smell
Array indexes should be numeric
Code Smell
Assertion arguments should be passed in the correct order
Code Smell
Ternary operators should not be nested
Code Smell
"delete" should not be used on arrays
Code Smell
Variables and functions should not be redeclared
Code Smell
"indexOf" checks should not be for positive numbers
Code Smell
"arguments.caller" and "arguments.callee" should not be used
Code Smell
Multiline blocks should be enclosed in curly braces
Code Smell
Boolean expressions should not be gratuitous
Code Smell
Variables should be used in the blocks

Tags ▾

Search by name... 🔍

Mocha timeout should be disabled by setting it to "0".

Analyze your code

Bug

Major

tests

Starting from Mocha v3.0.0, calling `this.timeout(x)` with `x` greater than the **maximum delay value** (2,147,483,647 ms) **will cause the timeout to be disabled**. This might not be what the developer intended. If the goal is really to disable the timeout, `this.timeout(0)` should be used instead.

Noncompliant Code Example

```
describe("testing this.timeout", function() {
  it("unexpectedly disables the timeout", function(done) {
    this.timeout(2147483648); // Noncompliant
  });
});
```

Compliant Solution

```
describe("testing this.timeout", function() {
  it("doesn't disable the timeout", function(done) {
    this.timeout(1000);
  });
});
```

Or if you meant to disable the timeout





```
describe("testing this.timeout", function() {
  it("disables the timeout as expected", function(done) {
    this.timeout(0);
  });
});
```

See

- Mocha documentation

Available In:

sonarlint | sonarcloud | sonarqube

<div>variables should be used in the blocks where they are declared</div> <div> Code Smell</div>
<div>Parameters should be passed in the correct order</div> <div> Code Smell</div>
<div>Two branches in a conditional structure should not have exactly the same implementation</div> <div> Code Smell</div>
<div>Unused assignments should be removed</div> <div> Code Smell</div>