




 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



JavaScript static code analysis

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

All rules285

Vulnerability29

Bug62


Security Hotspot43

Code Smell151


Quick Fix41


Tags ▾

Search by name... 🔍


 Bug


Function parameters, caught exceptions and foreach variables' initial values should not be ignored

 Bug


 Security Hotspot


Forwarding client IP address is security-sensitive

 Security Hotspot


 Security Hotspot


Allowing confidential information to be logged is security-sensitive

 Security Hotspot


 Security Hotspot


Allowing browsers to perform DNS prefetching is security-sensitive

 Security Hotspot


 Security Hotspot


Disabling Certificate Transparency monitoring is security-sensitive

 Security Hotspot


 Security Hotspot


Disabling Strict-Transport-Security policy is security-sensitive

 Security Hotspot


 Security Hotspot


Disabling strict HTTP no-referrer policy is security-sensitive

 Security Hotspot


 Security Hotspot


Allowing browsers to sniff MIME types is security-sensitive

 Security Hotspot


 Security Hotspot


Disabling content security policy frame-ancestors directive is security-sensitive

 Security Hotspot

 Security Hotspot



Allowing mixed-content is security-sensitive

 Security Hotspot

 Security Hotspot

Disabling content security policy fetch directives is security-sensitive

Return values from functions without side effects should not be ignored

 Bug  Major ?

When the call to a function doesn't have any side effects, what is the point of making the call if the results are ignored? In such case, either the function call is useless and should be dropped or the source code doesn't behave as expected.

To prevent generating any false-positives, this rule triggers an issues only on a predefined list of known objects & functions.




Noncompliant Code Example

```
'hello'.lastIndexOf('e'); // Noncompliant
```

Compliant Solution

```
let char = 'hello'.lastIndexOf('e');
```





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](#)

https://rules.sonarsource.com/javascript/RSPEC-2201

1/2

 Security Hotspot
Disabling resource integrity features is security-sensitive  Security Hotspot
Disclosing fingerprints from web application technologies is security-sensitive  Security Hotspot
Having a permissive Cross-Origin Resource Sharing policy is security-sensitive  Security Hotspot