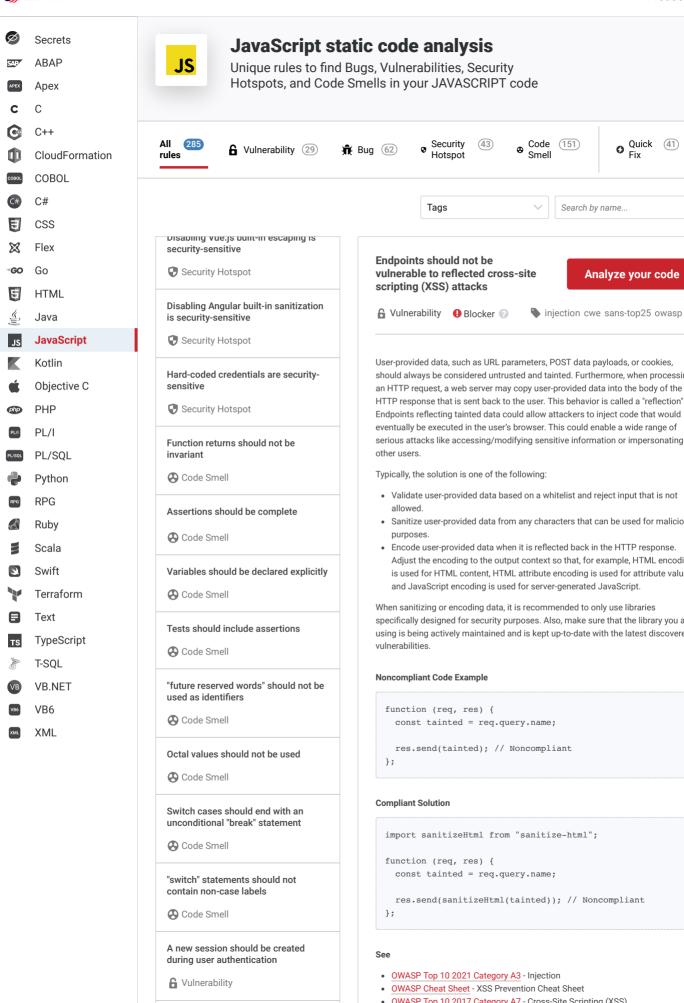


Products >



Analyze your code

O Quick 41 Fix

Q

User-provided data, such as URL parameters, POST data payloads, or cookies, should always be considered untrusted and tainted. Furthermore, when processing an HTTP request, a web server may copy user-provided data into the body of the HTTP response that is sent back to the user. This behavior is called a "reflection". Endpoints reflecting tainted data could allow attackers to inject code that would eventually be executed in the user's browser. This could enable a wide range of serious attacks like accessing/modifying sensitive information or impersonating

- · Validate user-provided data based on a whitelist and reject input that is not
- Sanitize user-provided data from any characters that can be used for malicious
- Adjust the encoding to the output context so that, for example, HTML encoding is used for HTML content, HTML attribute encoding is used for attribute values, and JavaScript encoding is used for server-generated JavaScript.

When sanitizing or encoding data, it is recommended to only use libraries specifically designed for security purposes. Also, make sure that the library you are using is being actively maintained and is kept up-to-date with the latest discovered

```
res.send(sanitizeHtml(tainted)); // Noncompliant
```

OWASP Top 10 2017 Category A7 - Cross-Site Scripting (XSS)

JWT should be signed and verified

JavaScript static code analysis: Endpoints should not be vulnerable to reflected cross-site scripting (XSS) attacks

JavaScript static code analysis: Endpoi
with strong cipher algorithms
6 Vulnerability
Cipher algorithms should be robust
★ Vulnerability
Encryption algorithms should be used with secure mode and padding scheme
6 Vulnerability
Server hostnames should be verified during SSL/TLS connections
☆ Vulnerability
Server certificates should be verified

MITRE, CWE-79 - Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

SANS Top 25 - Insecure Interaction Between Components

Available In:

Sonarcloud Sonarqube Developer Edition

© 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