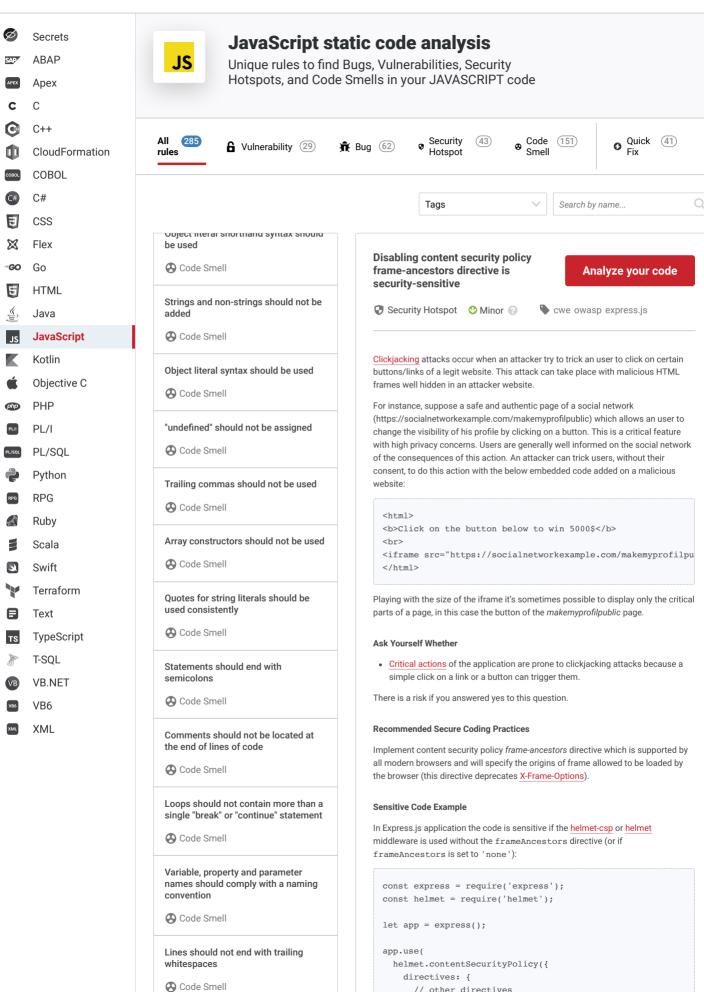


Products ✓



// other directives

frameAncestors: ["'none'"] // Sensitive: frameAncestor

JavaScript static code analysis: Disabling content security policy frame-ancestors directive is security-sensitive

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

```
}
})
);
```

Compliant Solution

In Express.js application a standard way to implement CSP frame-ancestors directive is the helmet-csp or helmet middleware:

```
const express = require('express');
const helmet = require('helmet');

let app = express();

app.use(
  helmet.contentSecurityPolicy({
    directives: {
        // other directives
        frameAncestors: ["'example.com'"] // Compliant
     }
    })
);
```

See

- OWASP Top 10 2021 Category A4 Insecure Design
- OWASP Top 10 2021 Category A5 Security Misconfiguration
- OWASP Top 10 2017 Category A6 Security Misconfiguration
- OWASP Cheat Sheets Clickjacking Defense Cheat Sheet
- developer.mozilla.org Frame-ancestors
- developer.mozilla.org Content Security Policy (CSP)
- MITRE, CWE-451 User Interface (UI) Misrepresentation of Critical Information
- w3.org Content Security Policy Level 3

Available In:

sonarcloud \delta | sonarqube

© 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