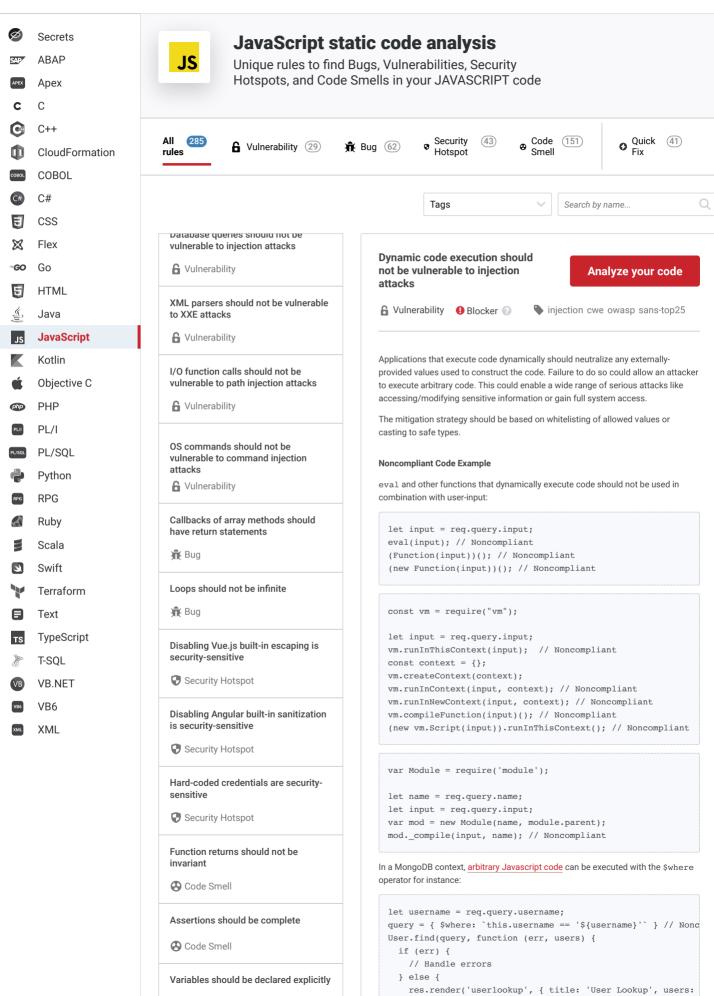


Products ✓



Code Smell

Tests should include assertions Code Smell "future reserved words" should not be used as identifiers Code Smell Octal values should not be used Code Smell Switch cases should end with an unconditional "break" statement Code Smell "switch" statements should not

```
});
Compliant Solution
```

If eval is used to parse a JSON string this should be done instead with JSON.parse:

```
let obj = JSON.parse(req.query.input); // Compliant
```

In a MongoDB context, don't use \$where operator or validate the data:

```
let username = req.query.username;
query = { username: username } // Compliant
User.find(query, function (err, users) {
 if (err) {
    // Handle errors
 } else {
    res.render('userlookup', { title: 'User Lookup', users:
});
```

- OWASP Top 10 2021 Category A3 Injection
- OWASP Top 10 2017 Category A1 Injection
- MITRE, CWE-20 Improper Input Validation
- MITRE, CWE-95 Improper Neutralization of Directives in Dynamically Evaluated Code ('Eval Injection')
- SANS Top 25 Risky Resource Management

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