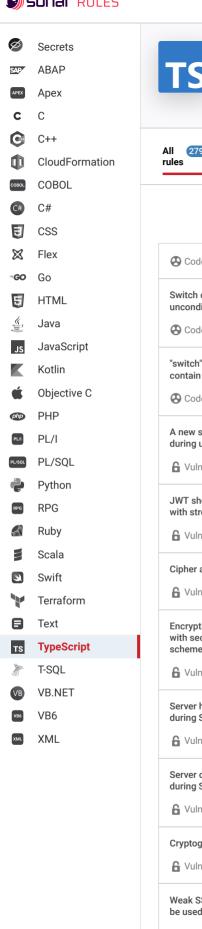
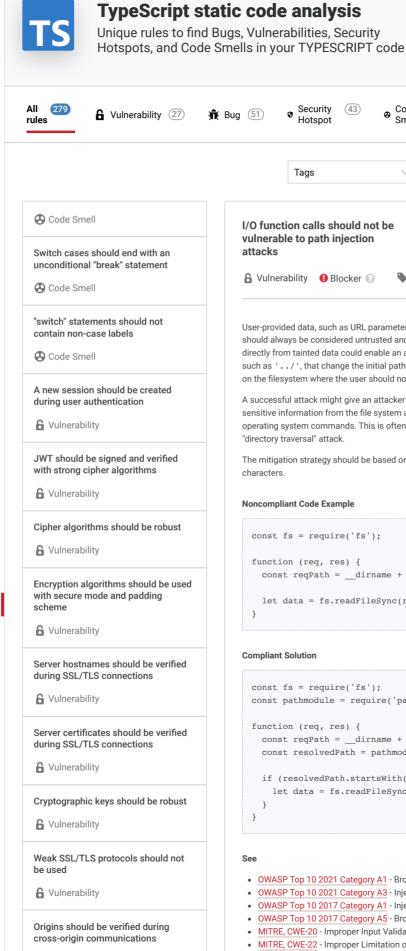
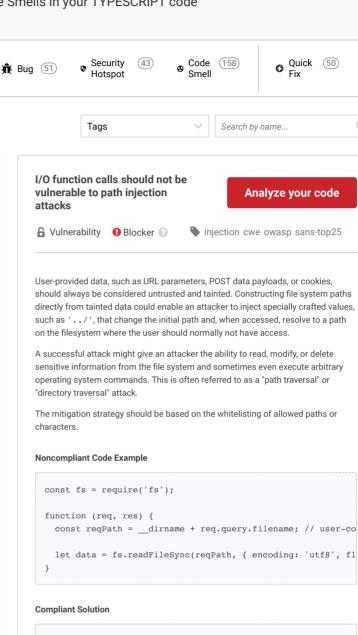


Products ✓







```
const fs = require('fs');
const pathmodule = require('path');
function (req, res) {
 const reqPath = __dirname + req.query.filename; // user-co
  const resolvedPath = pathmodule.resolve(reqPath); // resol
  if (resolvedPath.startsWith(__dirname + '/uploads')) { //
    let data = fs.readFileSync(resolvedPath, { encoding: 'ut
```

- OWASP Top 10 2021 Category A1 Broken Access Control
- OWASP Top 10 2021 Category A3 Injection
- OWASP Top 10 2017 Category A1 Injection
- OWASP Top 10 2017 Category A5 Broken Access Control
- MITRE, CWE-20 Improper Input Validation
- MITRE, CWE-22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')
- MITRE, CWE-99 Improper Control of Resource Identifiers ('Resource Injection')

Vulnerability

TypeScript static code analysis:

Regular expressions should not be vulnerable to Denial of Service attacks

Vulnerability

File uploads should be restricted

Vulnerability

Regular expressions should be syntactically valid

Bug

Types without members, 'any' and 'never' should not be used in type intersections

Bug

MITRE, CWE-641 - Improper Restriction of Names for Files and Other Resources
 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