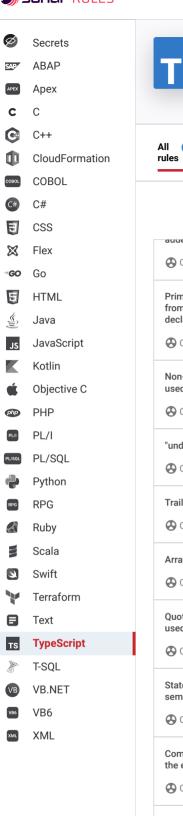
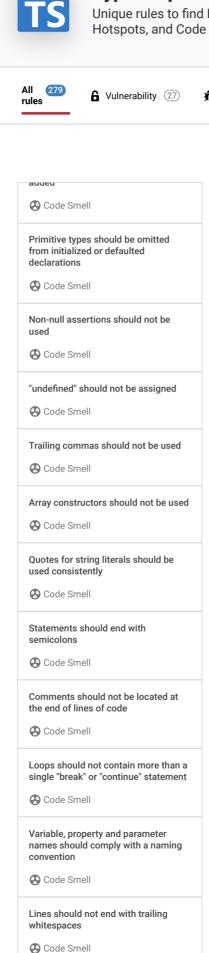
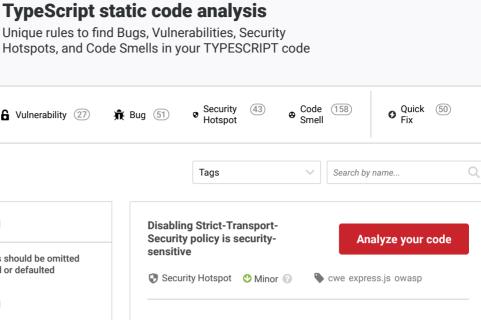


Products ✓







When implementing the HTTPS protocol, the website mostly continue to support the HTTP protocol to redirect users to HTTPS when they request a HTTP version of the website. These redirects are not encrypted and are therefore vulnerable to man in the middle attacks. The Strict-Transport-Security policy header (HSTS) set by an application instructs the web browser to convert any HTTP request to HTTPS.

Web browsers that see the Strict-Transport-Security policy header for the first time record information specified in the header:

- the max-age directive which specify how long the policy should be kept on the web browser.
- the includeSubDomains optional directive which specify if the policy should apply on all sub-domains or not.
- the preload optional directive which is not part of the HSTS specification but supported on all modern web browsers.

With the preload directive the web browser never connects in HTTP to the website and to use this directive, it is required to submit the concerned application to a preload service maintained by Google.

Ask Yourself Whether

• The website is accessible with the unencrypted HTTP protocol

There is a risk if you answered yes to this question.

Recommended Secure Coding Practices

Implement Strict-Transport-Security policy header, it is recommended to apply this policy to all subdomains (includeSubDomains) and for at least 6 months (maxage=15552000) or even better for 1 year (max-age=31536000).

Sensitive Code Example

In Express.js application the code is sensitive if the helmet or hsts middleware are disabled or used without recommended values:

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

let app = express();

app.use(helmet.hsts({
   maxAge: 3153600, // Sensitive, recommended >= 15552000
   includeSubDomains: false // Sensitive, recommended 'true'
}));
```

Compliant Solution

In Express.js application a standard way to implement HSTS is with the <u>helmet</u> or <u>hsts</u> middleware:

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

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

let app = express();

app.use(helmet.hsts({
  maxAge: 31536000,
  includeSubDomains: true
})); // Compliant
```

See

- OWASP Top 10 2021 Category A5 Security Misconfiguration
- OWASP Top 10 2017 Category A3 Sensitive Data Exposure
- developer.mozilla.org Strict Transport Security

Available In:

sonarcloud 🙆 | 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