Content Security Policy

Ian Oxley, OWASP Newcastle, 29 September 2015
What is Content Security Policy?

Policy Directives, CSP Level 1, and CSP Level 2

Reporting
Security is hard
XSS ranked in the top 3 vulnerabilities on the OWASP Top 10 since forever *

* 2007, 2010, and 2013
Content Security Policy:

Gives the browser a whitelist of trusted sources where content can be loaded or executed from.
HTTP Header

Content-Security-Policy: script-src 'self';

One or more directives
Refused to load the script 'http://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js' because it violates the following Content Security Policy directive: "script-src 'self'".
Content Security Policy 1.0

Mitigate cross-site scripting attacks by whitelisting allowed sources of script, style, and other resources.

<table>
<thead>
<tr>
<th></th>
<th>IE</th>
<th>Edge</th>
<th>Firefox</th>
<th>Chrome</th>
<th>Safari</th>
<th>Opera</th>
<th>iOS Safari</th>
<th>Opera Mini</th>
<th>Android Browser</th>
<th>Chrome for Android</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>8</td>
<td>39</td>
<td>43</td>
<td>44</td>
<td>7.1</td>
<td>7.1</td>
<td>8</td>
<td>32</td>
<td>8</td>
<td>4.1</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>40</td>
<td>44</td>
<td>7.1</td>
<td>8.4</td>
<td>4.3</td>
<td>8</td>
<td>9</td>
<td>4.4</td>
<td>4.4</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>41</td>
<td>45</td>
<td>8</td>
<td>32</td>
<td>44</td>
<td>9</td>
<td>8</td>
<td>44</td>
<td>44</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>42</td>
<td>46</td>
<td>9</td>
<td>33</td>
<td>34</td>
<td>9</td>
<td>34</td>
<td>48</td>
<td></td>
</tr>
</tbody>
</table>

Notes: Known issues (3) Resources (5) Feedback

The standard HTTP header is `Content-Security-Policy` which is used unless otherwise noted.

1. Supported through the `X-Content-Security-Policy` header
2. Supported through the `X-WebKit-CSP` header

http://caniuse.com/#feat=contentsecuritypolicy
‘self’
none
URL

* wildcard support
data:
https://cdn.example.com
*:///*.*.example.com:*
```
<script>
document.documentElement.className = 'js';
</script>

```

```
<style>
  .container {
    margin-top: 2rem;
    ...
  }
</style>

```

```
<section style="margin-top: 1rem;">
  ...
</section>

```

```
<a href="javascript:link();">...
</a>

```

```
<img onclick="loadPreview()">
```javascript
var foo = new Function('foo', 'bar', 'return foo + bar);
```

```javascript
eval('console.log("foo")');
```

```javascript
setTimeout('console.log(foo);', 5000);
setInterval('console.log(foo);', 5000);
```
var foo = new Function('foo', 'bar', 'return foo + bar');

eval('console.log("foo")');

setTimeout('console.log(foo);', 5000);

setInterval('console.log(foo);', 5000);
Content-Security-Policy:
default-src 'self';
Content-Security-Policy:
default-src 'self' https::;
Content-Security-Policy:
default-src ‘self’ https://
script-src https://cdn.example.com
Content-Security-Policy:

default-src ‘self’ https:;

script-src ‘self’ https: https://cdn.example.com
Content-Security-Policy:

default-src 'self' https:;

script-src 'self' https: https://cdn.example.com https://ajax.googleapis.com;

style-src 'self' https: https://cdn.example.com;
Content-Security-Policy:
default-src 'self' https::;
script-src 'self' https: https://cdn.example.com https://ajax.googleapis.com;
style-src 'self' https: https://cdn.example.com;
object-src 'none';
Level 2
## Content Security Policy Level 2

Mitigate cross-site scripting attacks by whitelisting allowed sources of script, style, and other resources. CSP 2 adds hash-source, nonce-source, and five new directives.

<table>
<thead>
<tr>
<th></th>
<th>IE</th>
<th>Edge</th>
<th>Firefox</th>
<th>Chrome</th>
<th>Safari</th>
<th>Opera</th>
<th>iOS Safari</th>
<th>Opera Mini</th>
<th>Android Browser</th>
<th>Chrome for Android</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Notes
1. Firefox 31-34 is missing the plugin-types, child-src, frame-ancestors, base-uri, and form-action directives.
2. Firefox 35 is missing the plugin-types, child-src, frame-ancestors, and form-action directives.
3. Firefox 36+ is missing the plugin-types and child-src directives.
4. Chrome 36-38 & Opera 23-25 are missing the plugin-types, child-src, frame-ancestors, base-uri, and form-action directives.
5. Chrome 39 and Opera 26 are missing the plugin-types, child-src, base-uri, and form-action directives.
6. Firefox 38 on Android is missing the child-src directive.

[http://caniuse.com/#feat=contentsecuritypolicy2](http://caniuse.com/#feat=contentsecuritypolicy2)
<meta http-equiv="Content-Security-Policy" content="default-src 'self' https:;"/>
Using a nonce

Content-Security-Policy:

default-src ‘self’;

script-src ‘self’ https://example.com ‘nonce-X87di93dkeff’;
console.log("No nonce attribute - won’t execute");
console.log("Nonce mismatch - won’t execute");
<script nonce="X87di93dkeff">
    console.log("Nonce matches - script executes");
</script>
// Valid nonce - script executes
<script nonce="X87di93dkeff">
  src="//url.com/not-on-whitelist"
</script>
Using a hash

Content-Security-Policy:
default-src ‘self’;
script-src ‘self’
‘sha256-2eXTeAxXc4NfEdtTitmpuNQV41/dtCeCYiAwxZCvkGo=’;
// Script executes - computed hash
// matches the one in the header
// matches the one in the header
<script>alert(‘Hello, OWASP.’);</script>
// Neither of these execute - whitespace
// and newlines cause a different hash
// to be computed
<script> alert('Hello, OWASP.'); </script>
<script>
    alert('Hello, OWASP.');
</script>
If you really, absolutely must have inline script and style, you can enable it by adding 'unsafe-inline' as an allowed source in a script-src or style-src directive. You can also use a nonce or a hash (see below). But please don’t. Banning inline script is the biggest security win CSP provides, and banning inline style likewise hardens your application.

Mike West, An Introduction to Content Security Policy
Reporting
Browser consoles don’t work so well in production.
report-uri directive takes a URL as its value. JSON sent via HTTP POST for each policy violation.
Content-Security-Policy:

default-src 'self';

script-src 'self' http://cdn.example.com;

style-src 'self' http://cdn.example.com;

report-uri /csp-report;
Content-Security-Policy-Report-Only:
default-src ‘self’ https://; https://cdn.example.com
report-uri /csp-report;
Content-Security-Policy:

default-src ‘self’;
script-src ‘self’ http://cdn.example.com;
style-src ‘self’ http://cdn.example.com;
report-uri /csp-report;

Content-Security-Policy-Report-Only:

default-src ‘self’ https::; https://cdn.example.com
report-uri /csp-report;
CSP whitelists trusted origins using policy directives

Although you can enable them, inline styles and scripts are off by default for a reason

report-uri can monitor CSP in production
any questions?

@ianoxley
thanks

@ianoxley