In this article

X-XSS-Protection header

> **Non-standard:** This feature is not standardized. We do not recommend using non-standard features in production, as they have limited browser support, and may change or be removed. However, they can be a suitable alternative in specific cases where no standard option exists.

> **Deprecated:** This feature is no longer recommended. Though some browsers might still support it, it may have already been removed from the relevant web standards, may be in the process of being dropped, or may only be kept for compatibility purposes. Avoid using it, and update existing code if possible; see the compatibility table at the bottom of this page to guide your decision. Be aware that this feature may cease to work at any time.

> **Warning:** Even though this feature can protect users of older web browsers that don't support CSP, in some cases, **`X-XSS-Protection` can create XSS vulnerabilities** in otherwise safe websites. See the Security considerations section below for more information.

The HTTP **`X-XSS-Protection`** response header was a feature of Internet Explorer, Chrome and Safari that stopped pages from loading when they detected reflected cross-site scripting (XSS) attacks. These protections are largely unnecessary in modern browsers when sites implement a strong `Content-Security-Policy` that disables the use of inline JavaScript (`'unsafe-inline'`).

It is recommended that you use `Content-Security-Policy` instead of XSS filtering.

Header type	Response header

Syntax

HTTP Copy

```
X-XSS-Protection: 0
X-XSS-Protection: 1
X-XSS-Protection: 1; mode=block
X-XSS-Protection: 1; report=<reporting-uri>
```

Directives

`0`

Disables XSS filtering.

`1`

Enables XSS filtering (usually default in browsers). If a cross-site scripting attack is detected, the browser will sanitize the page (remove the unsafe parts).

`1; mode=block`

Enables XSS filtering. Rather than sanitizing the page, the browser will prevent rendering of the page if an attack is detected.

`1; report=<reporting-URI>` (Chromium only)

Enables XSS filtering. If a cross-site scripting attack is detected, the browser will sanitize the page and report the violation. This uses the functionality of the CSP `report-uri` directive to send a report.

Security considerations

Vulnerabilities caused by XSS filtering

Consider the following excerpt of HTML code for a webpage:

HTML Copy

```
<script>
  var productionMode = true;
</script>
<!-- [...] -->
<script>
  if (!window.productionMode) {
    // Some vulnerable debug code
  }
</script>
```

This code is completely safe if the browser doesn't perform XSS filtering. However, if it does and the search query is `?something=%3Cscript%3Evar%20productionMode%20%3D%20true%3B%3C%2Fscript%3E`, the browser might execute the scripts in the page ignoring `<script>var productionMode = true;</script>` (thinking the server included it in the response because it was in the URI), causing `window.productionMode` to be evaluated to `undefined` and executing the unsafe debug code.

Setting the `X-XSS-Protection` header to either `0` or `1; mode=block` prevents vulnerabilities like the one described above. The former would make the browser run all scripts and the latter would prevent the page from being processed at all (though this approach might be vulnerable to side-channel attacks if the website is embeddable in an `<iframe>`).

Example

Block pages from loading when they detect reflected XSS attacks:

HTTP Copy

```
X-XSS-Protection: 1; mode=block
```

PHP

PHP Copy

```
header("X-XSS-Protection: 1; mode=block");
```

Apache (.htaccess)

APACHECONF Copy

```
<IfModule mod_headers.c>
  Header set X-XSS-Protection "1; mode=block"
</IfModule>
```

Nginx

NGINX Copy

```
add_header "X-XSS-Protection" "1; mode=block";
```

Specifications

Not part of any specifications or drafts.

Browser compatibility

Report problems with this compatibility data • View data on GitHub

	Desktop					Mobile						
	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android	WebView on iOS
`X-XSS-Protection`	4–77	12–16	No	15–64	5–15.3	18–77	No	14–55	4.2–15.3	1–11.2	No	4.2–15.3

Tip: you can click/tap on a cell for more information.

Full support No support

Non-standard. Check cross-browser support before using.

Deprecated. Not for use in new websites. See implementation notes.

See also

- `Content-Security-Policy`
- Controlling the XSS Filter – Microsoft
- Understanding XSS Auditor – Virtue Security
- The misunderstood X-XSS-Protection – blog.innerht.ml

Help improve MDN

Was this page helpful to you?

Yes No

Learn how to contribute

This page was last modified on Nov 21, 2025 by MDN contributors.

View this page on GitHub • Report a problem with this content