Referrer-Policy

The **Referrer-Policy** HTTP header governs which referrer information, sent in the `Referer` header, should be included with requests made.

Header type	Response header
Forbidden header name	no

Syntax

Note that `Referer` is actually a misspelling of the word "referrer". The `Referrer-Policy` header does not share this misspelling.

```
Referrer-Policy: no-referrer
Referrer-Policy: no-referrer-when-downgrade
Referrer-Policy: origin
Referrer-Policy: origin-when-cross-origin
Referrer-Policy: same-origin
Referrer-Policy: strict-origin
```

```
Referrer-Policy: strict-origin-when-cross-origin
Referrer-Policy: unsafe-url
```

Directives

no-referrer

The `Referer` header will be omitted entirely. No referrer information is sent along with requests.

no-referrer-when-downgrade (default)

This is the user agent's default behavior if no policy is specified. The origin is sent as a referrer when the protocol security level stays the same (HTTPS->HTTPS), but isn't sent to a less secure destination (HTTPS->HTTP).

origin

Only send the origin of the document as the referrer in all cases.
The document `https://example.com/page.html` will send the referrer `https://example.com/`.

origin-when-cross-origin

Send a full URL when performing a same-origin request, but only send the origin of the document for other cases.

same-origin

A referrer will be sent for same-site origins, but cross-origin requests will contain no referrer information.

strict-origin

Only send the origin of the document as the referrer to a-priori as-much-secure destination (HTTPS->HTTPS), but don't send it to a less secure destination (HTTPS->HTTP).

strict-origin-when-cross-origin

Send a full URL when performing a same-origin request, only send the origin of the document to a-priori as-much-secure destination (HTTPS->HTTPS), and send no header to a less secure destination (HTTPS->HTTP).

unsafe-url

Send a full URL when performing a same-origin or cross-origin request.

> 🗒 This policy will leak origins and paths from TLS-protected resources to insecure origins. Carefully consider the impact of this setting.

Examples

Policy	Document	Navigation to	Referrer
no-referrer	https://example.com/page.html	any domain or path	no referre
no-referrer-when-downgrade	https://example.com/page.html	https://example.com/otherpage.html	https://exa
no-referrer-when-downgrade	https://example.com/page.html	https://mozilla.org	https://exa
no-referrer-when-downgrade	https://example.com/page.html	**http**://example.org	no referre
origin	https://example.com/page.html	any domain or path	https://exa
origin-when-cross-origin	https://example.com/page.html	https://example.com/otherpage.html	https://exa

Policy	Document	Navigation to	Referrer
`origin-when-cross-origin`	https://example.com/page.html	https://mozilla.org	https://exa
`origin-when-cross-origin`	https://example.com/page.html	**http**://example.com/page.html	https://exa
`same-origin`	https://example.com/page.html	https://example.com/otherpage.html	https://exa
`same-origin`	https://example.com/page.html	https://mozilla.org	no referre
`strict-origin`	https://example.com/page.html	https://mozilla.org	https://exa
`strict-origin`	https://example.com/page.html	**http**://example.org	no referre
`strict-origin`	**http**://example.com/page.html	any domain or path	http://exa
`strict-origin-when-cross-origin`	https://example.com/page.html	https://example.com/otherpage.html	https://exa
`strict-origin-when-cross-origin`	https://example.com/page.html	https://mozilla.org	https://exa

Policy	Document	Navigation to	Referrer
`strict-origin-when-cross-origin`	https://example.com/page.html	**http**://example.org	no referre
`unsafe-url`	https://example.com/page.html?q=123	any domain or path	https://exaq=123

Specifications

Specification	Status
↗ Referrer Policy	Editor's draft

Browser compatibility

New compatibility tables are in beta ▼

Basic support

🖥 ⊙	56	
🖥 e	No	
🖥 ❂	50	
🖥 □	No	
🖥 O	No	
🖥 ⊘	No	
🤖 🤖	56	
🤖 ⊙	56	

📱 e	No	
🤖 🦊	50	
🤖 O	No	
📱 🧭	No	
	?	

same-origin

🖥 🌐	61	
🖥 e	No	
🖥 🦊	52	
🖥 ⬜	No	
🖥 O	48	
🖥 🧭	No	
🤖 🤖	61	
🤖 🌐	61	
📱 e	No	
🤖 🦊	52	
🤖 O	48	
📱 🧭	No	
	?	

strict-origin

🖥 🌐	61	
🖥 e	No	
🖥 🦊	52	
🖥 ⬜	No	
🖥 O	48	
🖥 🧭	No	
🤖 🤖	61	

📱🟢	61	
📱ℯ	No	
📱🦊	52	
📱⭕	48	
📱🧭	No	
	?	

strict-origin-when-cross-origin

🖥️🟢	61	
🖥️ℯ	No	
🖥️🦊	52	
🖥️⬜	No	
🖥️⭕	48	
🖥️🧭	No	
📱📱	61	
📱🟢	61	
📱ℯ	No	
📱🦊	52	
📱⭕	48	
📱🧭	No	
	?	

▫️	Full support
▫️	No support
▫️	Compatibility unknown

📝 **Note:**

- From version 53 onwards, Gecko has a pref available in `about:config` to allow users to set their default `Referrer-Policy` — `network.http.referer.userControlPolicy`.
- From version 59 onwards (See ⧉ #587523), this has been replaced by `network.http.referer.defaultPolicy` and `network.http.referer.defaultPolicy.pbmode`.

Possible values are:

- `0` — `no-referrer`
- `1` — `same-origin`
- `2` — `strict-origin-when-cross-origin`
- `3` — `no-referrer-when-downgrade` (the default)

See also

- ⧉ HTTP referer on Wikipedia
- Other ways to set a referrer policy:
  - A `<meta>` element with a name of referrer.
  - A referrerpolicy attribute on an `<a>`, `<area>`, `<img>`, `<iframe>`, or `<link>` element.
  - The `noreferrer` link relation on an a, area, or link element (`rel="noreferrer"`).
  - When using Fetch: `Request.referrerPolicy`
- Same-origin policy
- ⧉ Tighter Control Over Your Referrers – Mozilla Security Blog