

New issue

[Jump to bottom](#)

Strip referrer and origin headers in xorigin requests from a .onion #10760

 **Merged** fmarier merged 3 commits into [master](#) from [referrer-onion-18071](#) 📄 on Nov 18, 2021

Conversation 20 Commits 3 Checks 4 Files changed 17



fmarier commented on Oct 28, 2021 • edited ▼

Member

Resolves [brave/brave-browser#18071](#)

Security report: <https://hackerone.com/reports/1337624>

Submitter Checklist:

- ☒ I confirm that no security/privacy review [is needed](#), or that I have [requested](#) one
- ☒ There is a [ticket](#) for my issue
- ☒ Used Github [auto-closing keywords](#) in the PR description above
- ☒ Wrote a good [PR/commit description](#)
- ☒ Added appropriate labels (QA/Yes OR QA/No ; release-notes/include OR release-notes/exclude ; OS/...) to the associated issue
- ☐ Checked the PR locally: `npm run test -- brave_browser_tests`, `npm run test -- brave_unit_tests`, `npm run lint`, `npm run gn_check`, `npm run tslint`
- ☐ Ran `git rebase master` (if needed)

Reviewer Checklist:

- ☐ A security review [is not needed](#), or a link to one is included in the PR description
- ☐ New files have MPL-2.0 license header
- ☐ Adequate test coverage exists to prevent regressions
- ☐ Major classes, functions and non-trivial code blocks are well-commented
- ☐ Changes in component dependencies are properly reflected in `gn`

- ☐ Code follows the [style guide](#)
- ☐ Test plan is specified in PR before merging

After-merge Checklist:

- ☒ The associated issue milestone is set to the smallest version that the changes has landed on
- ☒ All relevant documentation has been updated, for instance:
 - ☒ [https://github.com/brave/brave-browser/wiki/Deviations-from-Chromium-\(features-we-disable-or-remove\)](https://github.com/brave/brave-browser/wiki/Deviations-from-Chromium-(features-we-disable-or-remove))
 - ☐ <https://github.com/brave/brave-browser/wiki/Proxy-redirected-URLs>
 - ☐ <https://github.com/brave/brave-browser/wiki/Fingerprinting-Protections>
 - ☐ <https://github.com/brave/brave-browser/wiki/Brave%E2%80%99s-Use-of-Referral-Codes>
 - ☐ <https://github.com/brave/brave-browser/wiki/Custom-Headers>
 - ☐ <https://github.com/brave/brave-browser/wiki/Web-Compatibility-Exceptions-in-Brave>
 - ☐ <https://github.com/brave/brave-browser/wiki/QA-Guide>
 - ☐ <https://github.com/brave/brave-browser/wiki/P3A>

Test Plan:

1. Open <http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion/referrer/onion.html> in a Tor window
2. Open the *Network* tab of the devtools.
3. Follow the instructions on that test page.

  **fmarier** requested review from **mariospr** and **iefremov** 13 months ago

  **fmarier** self-assigned this on Oct 28, 2021

  **fmarier** requested review from **a team** as code owners 13 months ago

fmarier commented on Oct 28, 2021

Member



Author

The extra patches are unfortunate, but I could not find a way to get our existing hooks to cover all of the test cases in my test page.

Note that I am planning to upstream (at least) the referrer patch into the [Referrer Policy spec](#) and then Chromium, given that the `.onion` scheme has been formally described in [RFC 7686](#).



1


  **fmarier** changed the title ~~Strip referrer and origin in xorigin requests from a .onion~~ Strip referrer and origin headers in xorigin requests from a .onion on Oct 28, 2021

 **goodov** reviewed on Oct 29, 2021

[View changes](#)

patches/services-network-cors-cors_url_loader.cc.patch Outdated  Show resolved

patches/net-url_request-url_request_job.cc.patch Outdated  Show resolved

chromium_src/services/network/cors/cors_url_loader.cc Outdated  Show resolved

chromium_src/net/url_request/url_request_job.cc Outdated  Show resolved

 **iefremov** reviewed on Oct 29, 2021

[View changes](#)

chromium_src/net/url_request/url_request_job.cc Outdated

```
3 | + * License, v. 2.0. If a copy of the MPL was not distributed with this file,  
4 | + * You can obtain one at https://mozilla.org/MPL/2.0/. */  
5 | +  
6 | + #define BRAVE_COMPUTE_REFERRER_FOR_POLICY \
```



iefremov on Oct 29, 2021

Contributor

why don't we do this in MaybeChangeReferrer in brave_shields_util.cc ?



fmarier on Oct 29, 2021

Member

Author

Because it doesn't cover all cases,sadly. That was my first attempt, but none of our existing hooks were even called in the case of a redirected CORS request for example (test case [#6](#) on my test page).

Changing this function also takes care of the origin header in most cases (though not the CORS ones).



iefremov on Nov 3, 2021

Contributor

if it doesn't cover all cases, do we have a problem with the referrer capping that we do in MaybeChangeReferrer?..



fmarier on Nov 4, 2021

Member

Author

We might, I haven't gotten around to testing that yet.



fmarier force-pushed the `referrer-onion-18071` branch from `4ef37e0` to `e04ea51`

13 months ago

[Compare](#)



fmarier requested a review from **a team** as a code owner 13 months ago



fmarier force-pushed the `referrer-onion-18071` branch from `e04ea51` to `3253526`

13 months ago

[Compare](#)



fmarier commented on Oct 29, 2021

[View changes](#)

`chromium_src/services/network/cors/cors_url_loader.cc`

```
12 +         url::Origin::Create(request_.url)) {           \
13 +     request_.headers.SetHeader(net::HttpRequestHeaders::kOrigin, \
14 +                                url::Origin().Serialize()); \
15 + } else /* NOLINT */
```



fmarier on Oct 29, 2021

Member

Author

Needed, otherwise the linter complains about a missing brace.



fmarier force-pushed the `referrer-onion-18071` branch from `3253526` to `088cbb9`

13 months ago

[Compare](#)



goodov reviewed on Nov 1, 2021

[View changes](#)

`chromium_src/net/url_request/url_request_job.cc`



Outdated



Show resolved

goodov approved these changes on Nov 1, 2021

[View changes](#)

  **fmarier** force-pushed the `referrer-onion-18071` branch from `088cbb9` to `ffa2c9b`
13 months ago

[Compare](#)

  **fmarier** requested a review from **iefremov** 13 months ago

 **iefremov** reviewed on Nov 3, 2021

[View changes](#)



`chromium_src/net/url_request/url_request_job.cc`

 Show resolved

iefremov commented on Nov 3, 2021

Contributor

If the fix is not super urgent we should invest into a browsertest, the url loader patch looks pretty fragile

  **fmarier** force-pushed the `referrer-onion-18071` branch 2 times, most recently from `15bf75c` to `fe3b682`
13 months ago



[Compare](#)

  **fmarier** mentioned this pull request on Nov 10, 2021

Send "null" Origin header on cross-origin .onion requests [whatwg/fetch#1351](#)

 Draft

 3 tasks

  **fmarier** force-pushed the `referrer-onion-18071` branch from `fe3b682` to `0f3e7cd`
13 months ago

[Compare](#)

  **fmarier** requested a review from **iefremov** 13 months ago

fmarier commented on Nov 16, 2021

Member

Author












@iefremov I have added a browsertest which covers all of the test cases I have manually tested. Could you please review again?

 **iefremov** reviewed on Nov 18, 2021

[View changes](#)

iefremov approved these changes on Nov 18, 2021

[View changes](#)

-   **fmarier** force-pushed the `referrer-onion-18071` branch from `0f3e7cd` to `030bf22` 12 months ago [Compare](#)
-   Strip referrer header in xorigin requests from `.onion` (fixes [brave/br...](#) ...) `5f931c8`
-  **fmarier** added 2 commits 12 months ago
-   Nullify Origin header in xorigin CORS requests from `.onion` (fixes [bra...](#) ...) `1784f7b`
-   Add browsertest for cross-origin `.onion` requests. ✓ `5e415ed`
-   **fmarier** force-pushed the `referrer-onion-18071` branch from `030bf22` to `5e415ed` 12 months ago [Compare](#)
-  **fmarier** merged commit `38b8fc1` into `master` on Nov 18, 2021 [View details](#)
10 checks passed

  **fmarier** deleted the `referrer-onion-18071` branch 12 months ago

  **fmarier** added this to the **1.34.x - Nightly** milestone on Nov 18, 2021

 **brave-builds** pushed a commit that referenced this pull request on Nov 19, 2021

 Uplift of [#10760](#) (squashed) to beta ✓ `a035f37`

  **brave-builds** mentioned this pull request on Nov 19, 2021

Strip referrer and origin headers in xorigin requests from a `.onion` (uplift to 1.33.x) [#11186](#)

 Merged

 7 tasks

Verification **PASSED** on Win 11 x64 using the following build:

```
Brave | 1.34.38 Chromium: 96.0.4664.45 (Official Build) nightly (64-bit)
-- | --
Revision | 76e4c1bb2ab4671b8beba3444e61c0f17584b2fc-refs/branch-heads/4664@{#947}
OS | Windows 11 Version 21H2 (Build 22000.348)
```

Sub-resources

Same-origin

Test Case #1

onion16_1.png - was loaded with the full Referer header and the origin of this page in the Origin header.

- Origin: <http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion>
- Referer: <http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion/referer/onion.html>

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.

Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- was loaded with the full *Referer* header and the origin of this page in the *Origin* header.
- was loaded with a full *Referer* header and without an *Origin* header.
- was loaded with a full *Referer* header and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- was loaded without a *Referer* header and with a value of *null* in the *Origin* header.
- was loaded without *Referer* or *Origin* headers.
- was loaded without a *Referer* header and a value of *null* in the *Origin* header.

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For *referrer* tests, look at the *JS referrer* displayed on the page, as well as the *request header* in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- [Check referrer](#) after a same-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *Origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- [Check referrer](#) after a same-origin POST navigation.
- [Check referrer](#) after a POST navigation ending up in a redirect.

Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- [Check referrer](#) after a cross-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- [Check referrer](#) after a cross-origin POST navigation.
- [Check referrer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin POST navigation ending up in a same-origin redirect.

The screenshot shows the Network tab in a browser's developer tools. A request to `onion16_1.png` is selected. The 'Headers' pane is open, showing the 'Request Headers' section. The 'Referer' header is present and contains the full URL of the page. The 'Origin' header is also present and contains the same origin as the page. The 'Response Headers' section is also visible, showing various headers like 'Access-Control-Allow-Origin' and 'Cache-Control'.

Test Case #2

onion16_2.png - was loaded with a full Referer header and without an Origin header.

- Referer: `http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion/referrer/onion.html`

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.

Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🍌 was loaded with the **full** *Referer* header and the origin of this page in the *Origin* header.
- 🍌 was loaded **with a full *Referer* header and without an *Origin* header**
- 🍌 was loaded **with a full *Referer* header** and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🍌 was loaded **without** a *Referer* header and with a value of *null* in the *Origin* header.
- 🍌 was loaded **without** *Referer* or *Origin* headers.
- 🍌 was loaded **without** a *Referer* header and a value of *null* in the *Origin* header.

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For *referrer* tests, look at the **JS referrer** displayed on the page, as well as the **request header** in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- 🍌 [Check referrer](#) after a same-origin GET navigation.
- 🍌 [Check referrer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *Origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- 🍌 [Check referrer](#) after a same-origin POST navigation.
- 🍌 [Check referrer](#) after a POST navigation ending up in a redirect.

Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- 🍌 [Check referrer](#) after a cross-origin GET navigation.
- 🍌 [Check referrer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- 🍌 [Check referrer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- 🍌 [Check referrer](#) after a cross-origin POST navigation.
- 🍌 [Check referrer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- 🍌 [Check referrer](#) after a cross-origin POST navigation ending up in a same-origin redirect.

11 requests | 13.9 kB transferred

Test Case #3

onion16_3.png - was loaded with a full *Referer* header and the origin of this page in the *Origin* header.




- Origin: `http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion`
- Referer: `http://ixrdj3iwwhkuau5tby5jh3a536a2rdhpbdbu6ldhng43r47kim7a3lid.onion/referrer/onion.html`

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.




Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

-  was loaded with the **full** *Referer* header and the origin of this page in the *Origin* header.
-  was loaded **with a full** *Referer* header and **without** an *Origin* header.
-  was loaded **with a full** *Referer* header and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

-  was loaded **without** a *Referer* header and with a value of *null* in the *Origin* header.
-  was loaded **without** *Referer* or *Origin* headers.
-  was loaded **without** a *Referer* header and a value of *null* in the *Origin* header.

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For *referrer* tests, look at the **JS referrer** displayed on the page, as well as the **request header** in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- [Check referrer](#) after a same-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *Origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- [Check referrer](#) after a same-origin POST navigation.
- [Check referrer](#) after a POST navigation ending up in a redirect.

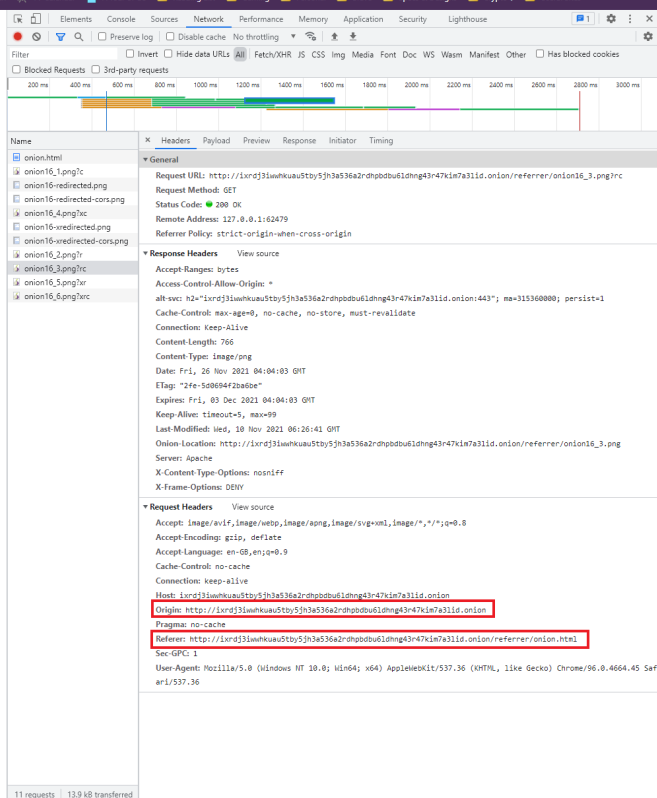
Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- [Check referrer](#) after a cross-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- [Check referrer](#) after a cross-origin POST navigation.
- [Check referrer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin POST navigation ending up in a same-origin redirect.



The screenshot shows the Chrome DevTools Network tab with the 'Headers' panel expanded for the request 'onion16_3.png'. The 'Request Headers' section shows the following details:

- Request URL:** http://localhost:3000/onion16_3.png
- Request Method:** GET
- Status Code:** 200 OK
- Remote Address:** 127.0.0.1:62479
- Referer Policy:** strict-origin-when-cross-origin

The 'Response Headers' section shows:

- Accept-Ranges:** bytes
- Access-Control-Allow-Origin:** *
- Cache-Control:** max-age=0, no-cache, no-store, must-revalidate
- Content-Length:** 766
- Content-Type:** image/png
- Date:** Fri, 26 Nov 2021 04:04:03 GMT
- Etag:** "2fe-508694f2ba6be"
- Expires:** Fri, 03 Dec 2021 04:04:03 GMT
- Keep-Alive:** timeout=5, max=99
- Last-Modified:** Wed, 10 Nov 2021 06:26:41 GMT
- Onion Location:** http://localhost:3000/onion16_3.png
- Server:** Apache
- X-Content-Type-Options:** nosniff
- X-Frame-Options:** DENY

The 'Request Headers' section shows:

- Host:** localhost:3000
- Origin:** http://localhost:3000
- Pragma:** no-cache
- Referer:** http://localhost:3000/onion16_1.png
- Sec-GPC:** 1
- User-Agent:** Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36

Cross-origin

Test Case #1

onion16_4.png - was loaded without a *Referer* header and with a value of *null* in the *Origin* header.

- origin: null

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.

Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🔗 was loaded with the **full** *Referer* header and the origin of this page in the *Origin* header.
- 🔗 was loaded **with a full** *Referer* header and **without** an *Origin* header.
- 🔗 was loaded **with a full** *Referer* header and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🔗 was loaded **without a** *Referer* header and with a value of **null** in the *Origin* header.
- 🔗 was loaded **without** *Referer* or *Origin* headers.
- 🔗 was loaded **without** a *Referer* header and a value of **null** in the *Origin* header.

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For *referrer* tests, look at the **JS referrer** displayed on the page, as well as the **request header** in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- 🔗 [Check referrer](#) after a same-origin GET navigation.
- 🔗 [Check referrer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- 🔗 [Check referrer](#) after a same-origin POST navigation.
- 🔗 [Check referrer](#) after a POST navigation ending up in a redirect.

Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- 🔗 [Check referrer](#) after a cross-origin GET navigation.
- 🔗 [Check referrer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- 🔗 [Check referrer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- 🔗 [Check referrer](#) after a cross-origin POST navigation.
- 🔗 [Check referrer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- 🔗 [Check referrer](#) after a cross-origin POST navigation ending up in a same-origin redirect.

11 requests | 13.9 kB transferred

onion.html

onion16_1.png?c

onion16_redirected.png

onion16_4.png?c

onion16_redirected.png

onion16_redirected.png

onion16_2.png?c

onion16_3.png?c

onion16_5.png?c

onion16_8.png?c

General

Request URL: https://fearier.com/referrer/onion16_4.png?c

Request Method: GET

Status Code: 200

Remote Address: 127.0.0.1:62479

Referer Policy: strict-origin-when-cross-origin

Response Headers

accept-ranges: bytes

access-control-allow-origins: *

alt-svc: h2="127.0.0.1:62479"; ma=315360000; persist=1

cache-control: max-age=0, no-cache, no-store, must-revalidate

content-length: 766

content-type: image/png

date: Fri, 26 Nov 2021 04:04:03 GMT

etag: "2fa-508694f2b60a"

expires: Fri, 03 Dec 2021 04:04:03 GMT

last-modified: Wed, 10 Nov 2021 06:26:41 GMT

onion-location: http://127.0.0.1:62479/referrer/onion16_4.png

server: Apache

x-content-type-options: nosniff

x-frame-options: DENY

Request Headers

authority: fearier.com

method: GET

path: /referrer/onion16_4.png?c

scheme: https

accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8

accept-encoding: gzip, deflate, br

accept-language: en-GB,en;q=0.9

cache-control: no-cache

origin: null

pragma: no-cache

sec-fetch-dest: image

sec-fetch-mode: cors

sec-fetch-site: cross-site

sec-gpc: 1

user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36

Test Case #2

onion16_5.png - was loaded without *Referer* or *Origin* headers.

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.

Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- was loaded with the **full** *Referer* header and the origin of this page in the *Origin* header.
- was loaded **with a full** *Referer* header and **without** an *Origin* header.
- was loaded **with a full** *Referer* header and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- was loaded **without** a *Referer* header and with a value of *null* in the *Origin* header.
- was loaded **without** *Referer* or *Origin* headers
- was loaded **without** a *Referer* header and a value of *null* in the *Origin* header.

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For *referrer* tests, look at the **JS referrer** displayed on the page, as well as the **request header** in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- [Check referrer](#) after a same-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *Origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- [Check referrer](#) after a same-origin POST navigation.
- [Check referrer](#) after a POST navigation ending up in a redirect.

Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- [Check referrer](#) after a cross-origin GET navigation.
- [Check referrer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- [Check referrer](#) after a cross-origin POST navigation.
- [Check referrer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- [Check referrer](#) after a cross-origin POST navigation ending up in a same-origin redirect.

11 requests | 13.9 kB transferred

onion.html

onion16_1.png?c

onion16_redirected.png

onion16_redirected-cors.png

onion16_4.png?c

onion16_redirected.png

onion16_redirected-cors.png

onion16_2.png?c

onion16_3.png?c

onion16_5.png?c

onion16_6.png?c

General

Request URL: https://fmarier.com/referrer/onion16_5.png?c

Request Method: GET

Status Code: 200

Remote Address: 127.0.0.1:62479

Referer Policy: strict-origin-when-cross-origin

Response Headers

accept-ranges: bytes

alt-svc: h2="1xrdj3lwa8kuaustby5jh3a536a2rdhpbdu6ldmg43r47kln7a3l1d.onion:443"; ma=315360000; persist=1

cache-control: max-age=0, no-cache, no-store, must-revalidate

content-length: 766

content-type: image/png

date: Fri, 26 Nov 2021 04:04:04 GMT

etag: "2fe-598694f2ba0a"

expires: Fri, 03 Dec 2021 04:04:04 GMT

last-modified: Wed, 10 Nov 2021 06:26:41 GMT

onion-location: http://1xrdj3lwa8kuaustby5jh3a536a2rdhpbdu6ldmg43r47kln7a3l1d.onion/referrer/onion16_5.png

server: Apache

x-content-type-options: nosniff

x-frame-options: DENY

Request Headers

authority: fmarier.com

method: GET

path: /referrer/onion16_5.png?c

scheme: https

accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8

accept-encoding: gzip, deflate, br

accept-language: en-GB,en;q=0.9

cache-control: no-cache

pragma: no-cache

sec-fetch-dest: image

sec-fetch-mode: no-cors

sec-fetch-site: cross-site

sec-gpc: 1

user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36

Test Case #3

onion16_6.png - was loaded without a Referer header and a value of null in the Origin header.

Sub-resources

Test this first since the navigation tests will cause these sub-resources to be loaded from cache.

Same-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🔗 was loaded with the **full** *Referer* header and the origin of this page in the *Origin* header.
- 🔗 was loaded **with a full** *Referer* header and **without** an *Origin* header.
- 🔗 was loaded **with a full** *Referer* header and the origin of this page in the *Origin* header.

Cross-origin

Check the *Request Headers* in the *Network* tab of the devtools to ensure that:

- 🔗 was loaded **without** a *Referer* header and with a value of `null` in the *Origin* header.
- 🔗 was loaded **without** *Referer* or *Origin* headers
- 🔗 was loaded **without a** *Referer* header **and** a value of `null` in the *Origin* header

Navigations

You'll have to use the *Back* button to come back to this page after each test.

For referer tests, look at the **JS referer** displayed on the page, as well as the **request header** in the devtools.

Because of caching issues, if you want to repeat any of these tests, it's best to close the browser and restart it first.

Same-origin

The *Referer* header should be **present** (full URL of this page) in this example:

- 🔗 [Check referer](#) after a same-origin GET navigation.
- 🔗 [Check referer](#) after a same-origin GET navigation ending up in a redirect.

The *Referer* and *Origin* headers should be **present** (full URL, and same hostname as this page, respectively) in all of these examples:

- 🔗 [Check referer](#) after a same-origin POST navigation.
- 🔗 [Check referer](#) after a POST navigation ending up in a redirect.

Cross-origin

Neither the *Referer* nor the *Origin* header should be present in these examples:

- 🔗 [Check referer](#) after a cross-origin GET navigation.
- 🔗 [Check referer](#) after a same-origin GET navigation ending up in a cross-origin redirect.
- 🔗 [Check referer](#) after a cross-origin GET navigation ending up in a same-origin redirect.

The *Referer* header should **not be present** and the *Origin* header should be **null** in all of these examples:

- 🔗 [Check referer](#) after a cross-origin POST navigation.
- 🔗 [Check referer](#) after a same-origin POST navigation ending up in a cross-origin redirect.
- 🔗 [Check referer](#) after a cross-origin POST navigation ending up in a same-origin redirect.

```
Request Headers
Request URL: https://fearier.com/referer/onion16_6.png?c
Request Method: GET
Status Code: 200
Remote Address: 127.0.0.1:62479
Referer Policy: strict-origin-when-cross-origin

Response Headers
access-ranges: bytes
access-control-allow-origin: *
alt-svc: h2="1";url="https://fearier.com/referer/onion16_6.png?c";ma=315360000;persist=1
cache-control: max-age=0, no-cache, no-store, must-revalidate
content-length: 766
content-type: image/png
date: Fri, 26 Nov 2021 04:04:03 GMT
etag: "2fe-508694f20a6be"
expires: Fri, 03 Dec 2021 04:04:03 GMT
last-modified: Wed, 10 Nov 2021 06:26:41 GMT
onion-location: http://127.0.0.1:62479/referer/onion16_6.png
server: Apache
x-content-type-options: nosniff
x-frame-options: DENY

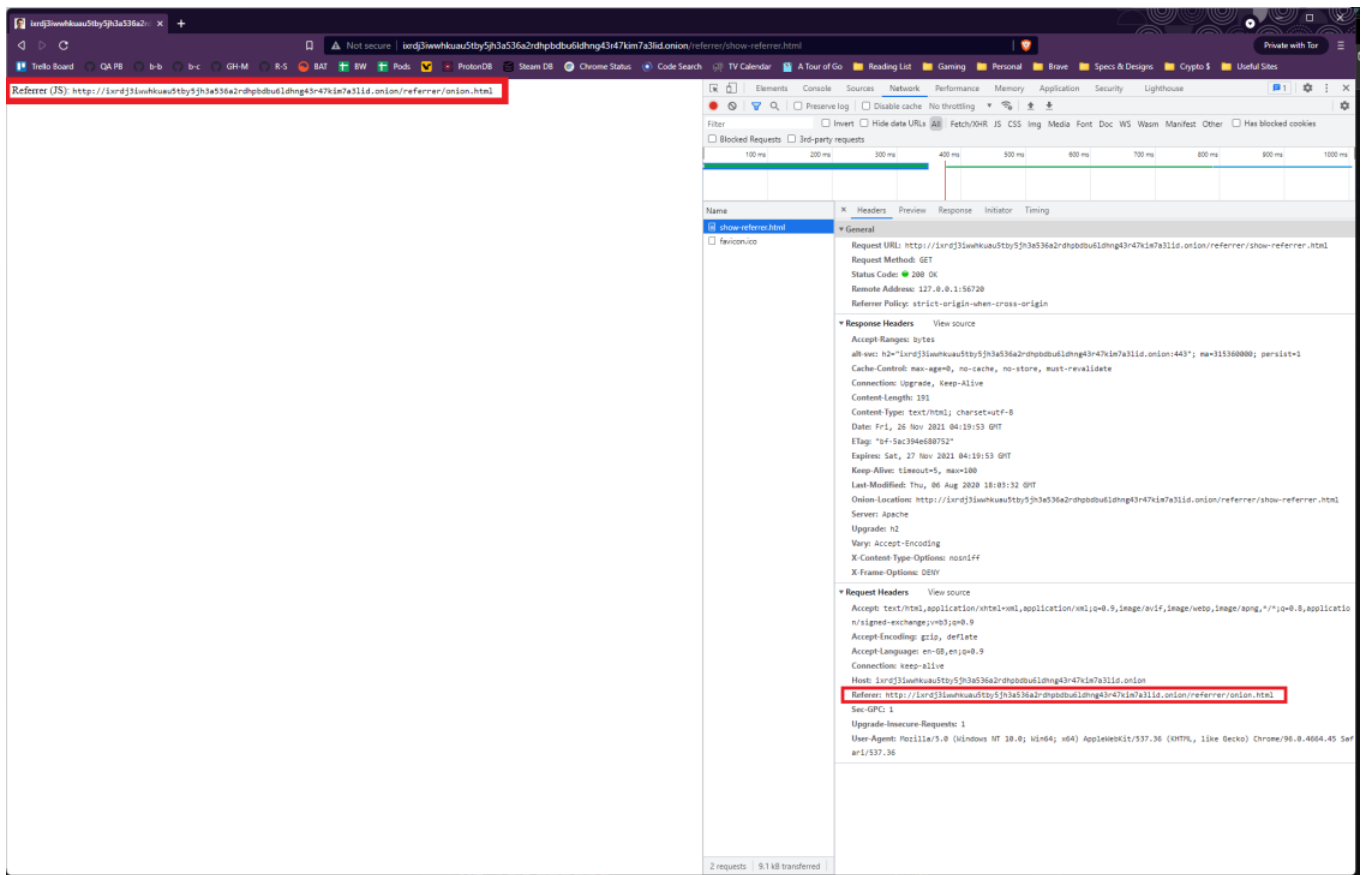
Request Headers
authority: fearier.com
method: GET
path: /referer/onion16_6.png?c
scheme: https
accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
accept-encoding: gzip, deflate, br
accept-language: en-US,en;q=0.9
cache-control: no-cache
origin: null
pragma: no-cache
sec-fetch-dest: image
sec-fetch-mode: cors
sec-fetch-site: cross-site
sec-gpc: 1
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
```

Navigations

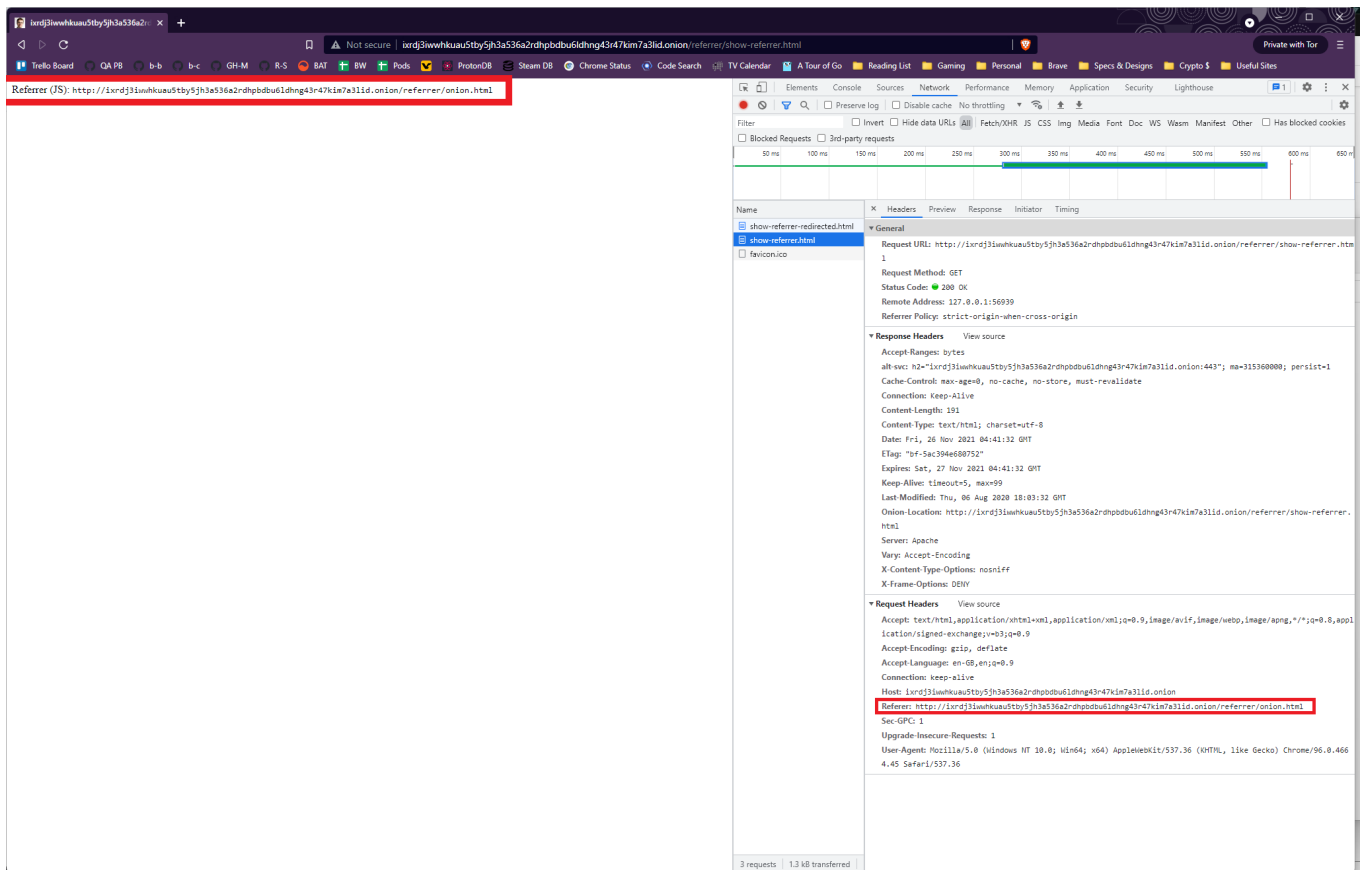
Same-origin

The Referer header should be present (full URL of this page) in this example:

Test Case #1 - after a same-origin GET navigation



Test Case #2 - after a same-origin GET navigation ending up in a redirect



The Referer and Origin headers should be present (full URL, and same hostname as this page, respectively) in all of these examples:

Test Case #1 - after a same-origin POST navigation

The screenshot shows a web browser with the address bar displaying the URL: `http://lrdj3lswkxuu5tby5j3a536a2rphobu1dng43r47x1a7a3l1d.onion/referrer/show-referrer.html`. The browser's developer tools are open, showing the Network tab. A request to `show-referrer.html` is selected. The request method is POST, and the status is 200 OK. The response headers show the `Referer` header as `http://lrdj3lswkxuu5tby5j3a536a2rphobu1dng43r47x1a7a3l1d.onion/referrer/onion.html`, which is highlighted with a red box. The request headers show the `Origin` header as `http://lrdj3lswkxuu5tby5j3a536a2rphobu1dng43r47x1a7a3l1d.onion`, also highlighted with a red box.

Test Case #2 - after a POST navigation ending up in a redirect

The screenshot shows a web browser with a network request in the DevTools console. The browser address bar shows a URL from lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion. The console shows a 'Referer (JS)' entry with a URL. The network tab shows a request to 'show-referrer.html' with various headers and a response.

Referer (JS): <http://lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion/referer/onion.html>

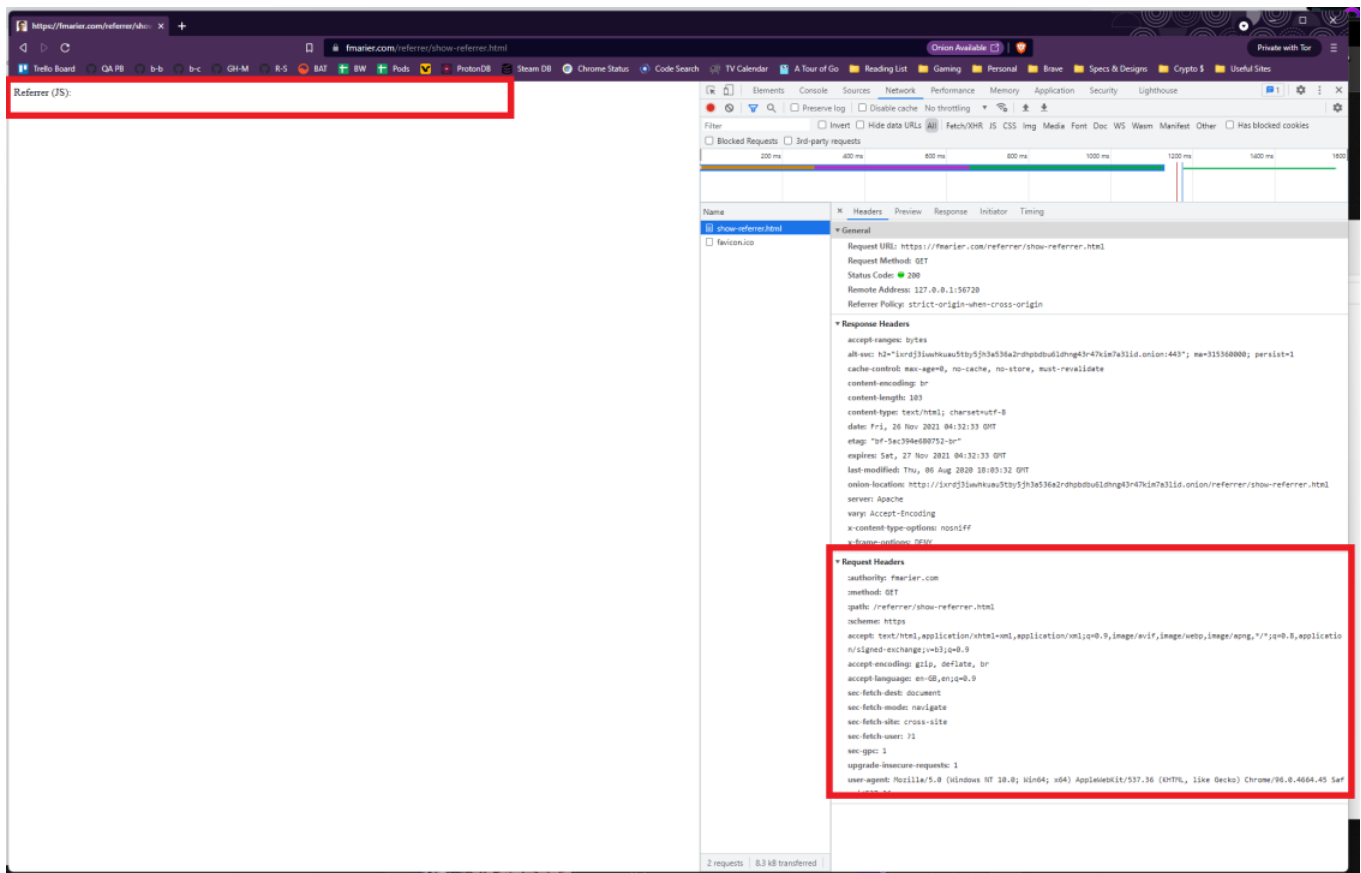
Network request details:

- Name: show-referrer.html
- Request URL: <http://lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion/referer/show-referrer.html>
- Request Method: POST
- Status Code: 200 OK
- Remote Address: 127.0.0.1:56939
- Referer Policy: strict-origin-when-cross-origin
- Response Headers:
 - Accept-Ranges: bytes
 - alt-svc: h2="lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion:443"; ma=315360000; persist=1
 - Cache-Control: max-age=0, no-cache, no-store, must-revalidate
 - Connection: Keep-Alive
 - Content-Length: 191
 - Content-Type: text/html; charset=utf-8
 - Date: Fri, 26 Nov 2021 04:43:11 GMT
 - ETag: "bf-5ac3946680752"
 - Expires: Sat, 27 Nov 2021 04:43:11 GMT
 - Keep-Alive: timeout=5, max=98
 - Last-Modified: Thu, 06 Aug 2020 18:03:32 GMT
 - Onion-Location: <http://lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion/referer/show-referrer.html>
 - Server: Apache
 - Vary: Accept-Encoding
 - X-Content-Type-Options: nosniff
 - X-Frame-Options: DENY
- Request Headers:
 - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml;q=0.8,application/signed-exchange;v=b3;q=0.9
 - Accept-Encoding: gzip, deflate
 - Accept-Language: en-GB,en;q=0.9
 - Cache-Control: max-age=0
 - Connection: keep-alive
 - Content-Length: 0
 - Content-Type: application/x-www-form-urlencoded
 - Host: lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion
 - Origin: <http://lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion>
 - Referer: <http://lxr.d3l3wkhkua5tby5jh3a536a2rdhpbdu6idmg43r47kim7a3lid.onion/referer/onion.html>
 - Sec-GPC: 1
 - Upgrade-Insecure-Requests: 1
 - User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36

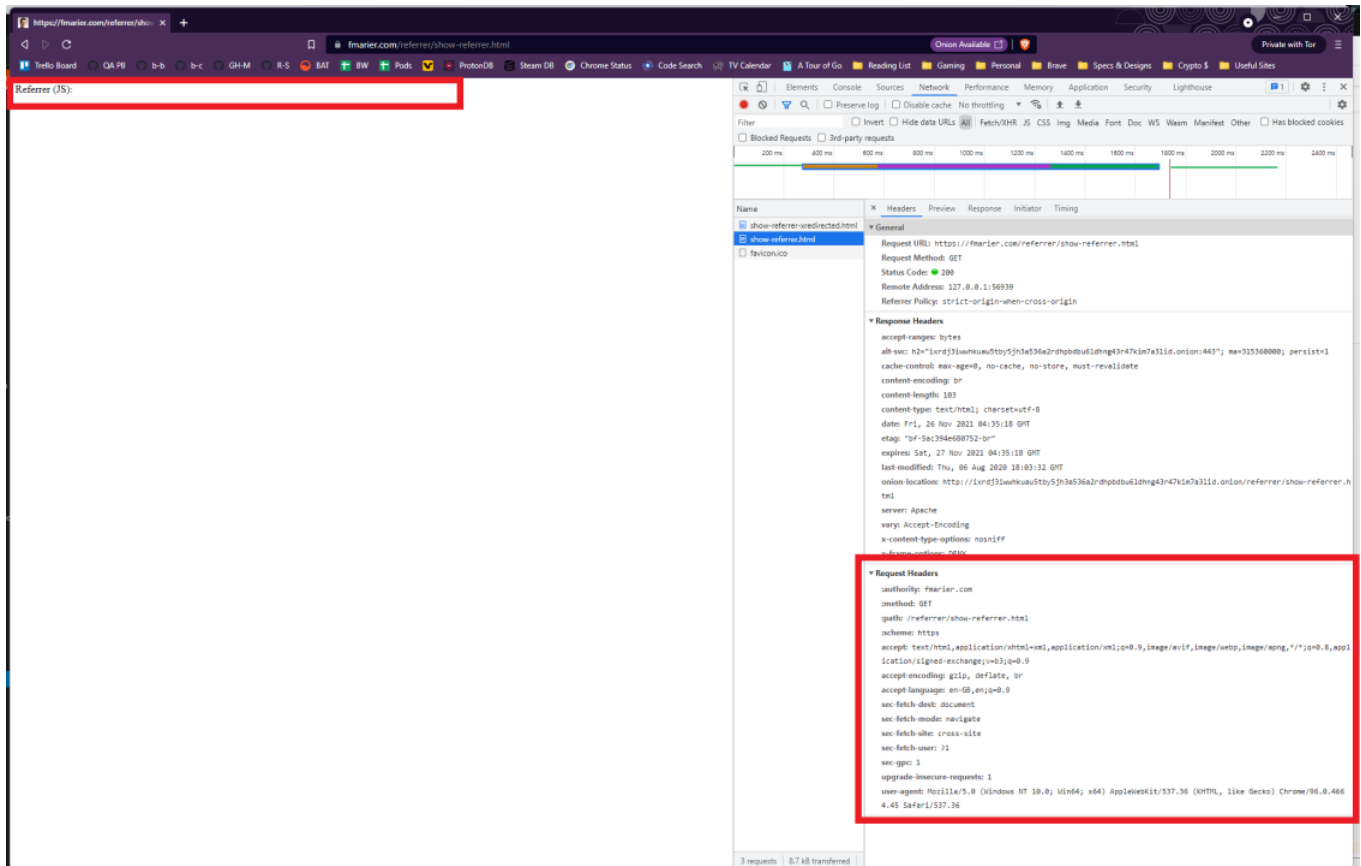
Cross-origin

Neither the Referer not the Origin header should be present in these examples:

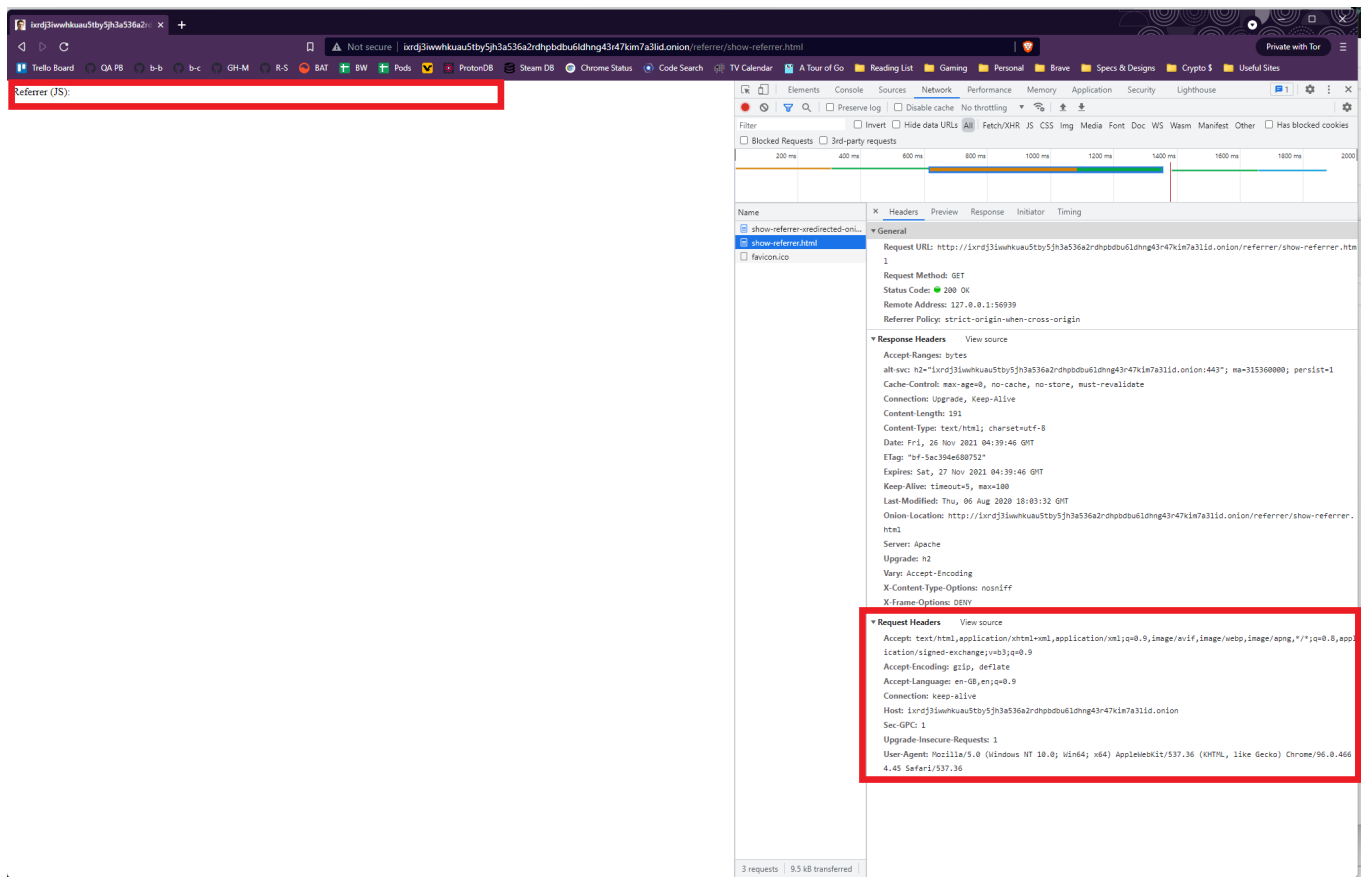
Test Case #1 - after a cross-origin GET navigation



Test Case #2 - after a same-origin GET navigation ending up in a cross-origin redirect



Test Case #3 - after a cross-origin GET navigation ending up in a same-origin redirect



The Referrer header should not be present and the Origin header should be null in all of these examples:

Test Case #1 - after a cross-origin POST navigation

lrdj3lwkhkua5tby5jh3a536a2r

Not secure | lrdj3lwkhkua5tby5jh3a536a2r.dhpbdu6idhng43r47k7a31id.onion/referrer/show-referrer.html

Trello Board QA PB b-b b-c GH-M R-S BAT BW Pods ProtonDB Steam DB Chrome Status Code Search TV Calendar A Tour of Go Reading List Gaming Personal Brave Specs & Designs Crypto \$ Useful Sites

Referrer (JS):

Elements Console Sources Network Performance Memory Application Security Lighthouse

Filter Blocked Requests 2nd-party requests

Name

show-referrer-redirected-oni...

show-referrer.html

favicon.ico

General

Request URL: http://lrdj3lwkhkua5tby5jh3a536a2r.dhpbdu6idhng43r47k7a31id.onion/referrer/show-referrer.html

Request Method: POST

Status Code: 200 OK

Remote Address: 127.0.0.1:56939

Referrer Policy: strict-origin-when-cross-origin

Response Headers

Accept Ranges: bytes

alt-svc: h2="lrdj3lwkhkua5tby5jh3a536a2r.dhpbdu6idhng43r47k7a31id.onion:443"; max=315360000; persist=1

Cache-Control: max-age=0, no-cache, no-store, must-revalidate

Connection Upgrade, keep-alive

Content-Length: 191

Content-Type: text/html; charset=utf-8

Date: Fri, 26 Nov 2021 04:48:00 GMT

Etag: "bf5ac394e680752"

Expires: Sat, 27 Nov 2021 04:48:00 GMT

Keep-Alive: timeout=5, max=100

Last-Modified: Thu, 06 Aug 2020 18:03:32 GMT

Onion-Location: http://lrdj3lwkhkua5tby5jh3a536a2r.dhpbdu6idhng43r47k7a31id.onion/referrer/show-referrer.html

Server: Apache

Upgrade: h2

Vary: Accept-Encoding

X-Content-Type-Options: nosniff

X-Frame-Options: DENY

Request Headers

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9

Accept-Encoding: gzip, deflate

Accept-Language: en-gb,en;q=0.9

Cache-Control: max-age=0

Connection: keep-alive

Content-Length: 0

Content-Type: application/x-www-form-urlencoded

Host: lrdj3lwkhkua5tby5jh3a536a2r.dhpbdu6idhng43r47k7a31id.onion

Origin: null

Sec-EPIC: 1

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36

3 requests | 1.3 kB transferred

Reviewers

iefremov

goodov

mariospr

Assignees

fmarier

Labels

None yet

Projects

None yet

Milestone

1.34.x - Release

4 participants

