

☆ Starred by 8 users

Owner:

CC:

Status:

Components:

Modified:

Backlog-Rank:

Editors:


EstimatedDays:








NextAction:

OS:

Pri:

Type:

 chlily@chromium.org
No longer actively working on Chromium

 mkwst@chromium.org
 yhirano@chromium.org
miketaylr@chromium.org
 mef@chromium.org
bingler@chromium.org
morlovich@chromium.org
lukasza@chromium.org
 mmenke@chromium.org
annev...@gmail.com
 fortenforge@google.com
 chlily@chromium.org
amarc...@mozilla.com
 pelizzi@google.com
sporeba@google.com
stefanoduo@google.com

Fixed (Closed)

Internals>Network>Cookies
Blink>SecurityFeature

Jul 21, 2021

Mac

2

Bug-Security

Security_Severity-Low
reward-3000
Security_Impact-Stable
allpublic
reward-inprocess
Via-Wizard-Security
Target-69
Target-70
CVE_description-submitted
M-89
Target-71
Target-72

Issue 830101: SameSite cookie bypass via redirect
Reported by s.h.h...@gmail.com on Fri, Apr 6, 2018, 7:44 PM EDT

 Code

UserAgent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/65.0.3325.181 Safari/537.36

Steps to reproduce the problem:
1. Go to <https://shnhjk.azurewebsites.net/SameSite.php> (Sets SameSite cookie)
2. Copy <https://test.shnhjk.com/location.php?url=https://shnhjk.azurewebsites.net/SameSite.php> and paste it to new tab

What is the expected behavior?
SameSite cookie not sent

What went wrong?
Redirect from attacker page to victim site doesn't prevent SameSite cookie to be sent.

Did this work before? N/A

Chrome version: 65.0.3325.181 Channel: stable
OS Version: OS X 10.13.4
Flash Version:

Comment 1 by jialiul@chromium.org on Fri, Apr 6, 2018, 8:29 PM EDT Project Member
Components: Internals>Network>Cookies

Comment 2 by jialiul@chromium.org on Sat, Apr 7, 2018, 2:00 PM EDT Project Member
Components: Blink>SecurityFeature

Comment 3 by elawrence@chromium.org on Mon, Apr 9, 2018, 1:06 PM EDT Project Member
Neat!

Do you have a way to reproduce this that does NOT involve the user entering something in the omnibox?
(It's already possible to circumvent same-site with non-default user interactions. For instance, middle-clicking on a link to open in a new tab sends samesite=strict cookies).

Comment 4 by s.h.h...@gmail.com on Mon, Apr 9, 2018, 1:12 PM EDT
>Do you have a way to reproduce this that does NOT
>involve the user entering something in the omnibox?
I don't know anyway to do it with server side redirect. But I have 2 other variations that I haven't reported yet. I was thinking to report it after checking the fix because
noopener bug might kill them all. But if you want, I can report them all.

Comment 5 by elawrence@chromium.org on Mon, Apr 9, 2018, 1:28 PM EDT Project Member
Yes, please do report them all... playing whack-a-mole with fixes is less efficient and leaves users more at risk.

Comment 6 by s.h.h...@gmail.com on Mon, Apr 9, 2018, 1:41 PM EDT

Filed the most interesting one (issue 830799) and least interesting one (~~issue 830800~~).
I have some more idea which I will test it later.

Comment 7 by s.h.h...@gmail.com on Thu, Apr 12, 2018, 6:54 PM EDT

>(It's already possible to circumvent same-site with non-default user interactions. For
>instance, middle-clicking on a link to open in a new tab sends samesite=strict cookies)
Interestingly, Firefox Nightly doesn't send SameSite Strict cookie even if you Ctrl + Click the link or Right click + open in a new Tab.

I would love to know what's the correct behavior per spec on this.

Comment 8 by carlosil@chromium.org on Fri, Apr 13, 2018, 1:20 PM EDT Project Member

Status: Available (was: Unconfirmed)
Cc: mkwst@chromium.org
Labels: Security_Severity-Low

Comment 9 by s.h.h...@gmail.com on Mon, Apr 23, 2018, 6:33 AM EDT

Re: #c3
>Do you have a way to reproduce this that does NOT involve the user entering something in the omnibox?

Redirect within iframe also works.

<https://shnhjk.azurewebsites.net/iframe.php?url=https://test.shnhjk.com/location.php?url=https://shnhjk.azurewebsites.net/SameSite.php>

Comment 10 by s.h.h...@gmail.com on Fri, Apr 27, 2018, 6:18 AM EDT

Step 3 of Document-based requests (<https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site-00#section-2.1.1>) algorithm says:
'Let "documents" be a list containing "document" and each of "document"'s ancestor browsing contexts' active documents.'

But redirect response doesn't have an active document.

Comment 11 by tsepez@chromium.org on Thu, May 3, 2018, 1:44 PM EDT Project Member

Status: Assigned (was: Available)
Owner: mkwst@chromium.org
Labels: M-68

Mike, are you the person handling these same-site cookie issues? Thanks.

Comment 12 by sheriffbot@chromium.org on Fri, May 4, 2018, 9:00 AM EDT Project Member

Labels: Security_Impact-Head

Comment 13 by sheriffbot@chromium.org on Wed, May 30, 2018, 9:00 AM EDT Project Member

Labels: -Security_Impact-Head Security_Impact-Beta

Comment 14 by och...@chromium.org on Mon, Jun 25, 2018, 3:08 AM EDT Project Member

~~Issue 855360~~ has been merged into this issue.

Comment 15 by sheriffbot@chromium.org on Wed, Jul 25, 2018, 9:00 AM EDT Project Member

Labels: -Security_Impact-Beta Security_Impact-Stable

Comment 16 by sheriffbot@chromium.org on Wed, Sep 5, 2018, 9:01 AM EDT Project Member

Labels: -M-68 M-69 Target-69

Comment 17 by rsesek@chromium.org on Thu, Sep 13, 2018, 12:50 PM EDT Project Member

~~Issue 883664~~ has been merged into this issue.

Comment 18 by mkwst@chromium.org on Thu, Oct 4, 2018, 8:56 AM EDT Project Member

Status: Untriaged (was: Assigned)
Owner: a_deleted_user
Cc: chliily@chromium.org mef@chromium.org mmenke@chromium.org morlovich@chromium.org
Labels: Hotlist-Cookies

(Unassigning myself, marking untriaged in preparation to retriage with folks who will do a better job taking care of cookies than I've been able to)

Comment 19 by sheriffbot@chromium.org on Wed, Oct 17, 2018, 9:01 AM EDT Project Member

Labels: -M-69 Target-70 M-70

Comment 20 by sheriffbot@chromium.org on Wed, Dec 5, 2018, 9:02 AM EST Project Member

Labels: -M-70 Target-71 M-71

Comment 21 by sheriffbot@chromium.org on Wed, Jan 30, 2019, 9:02 AM EST Project Member

Labels: -M-71 Target-72 M-72

Comment 22 by mkwst@chromium.org on Tue, Feb 12, 2019, 4:35 AM EST Project Member

Status: Available (was: Untriaged)
Cc: lukasza@chromium.org yhirano@chromium.org

CCing some folks who might have bandwidth.

Comment 23 by rsesek@chromium.org on Fri, Mar 1, 2019, 7:52 PM EST Project Member

~~Issue 937274~~ has been merged into this issue.

Comment 24 by mmenke@chromium.org on Mon, Mar 4, 2019, 6:11 PM EST Project Member

Isn't this working as expected? Lax cookies should be sent for top-level redirects (And I think that's the whole point of the Lax vs Strict distinction?). Not sure about the iframe case.

Comment 25 by s.h.h...@gmail.com on Wed, Mar 6, 2019, 3:56 AM EST

Strict cookie is also being sent, which is the issue in this bug.

Comment 26 by sheriffbot@chromium.org on Wed, Mar 13, 2019, 9:02 AM EDT Project Member

Labels: -M-72 Target-73 M-73

Comment 27 by sheriffbot@chromium.org on Wed, Apr 24, 2019, 9:03 AM EDT Project Member

Labels: -M-73 Target-74 M-74

[Comment 28](#) by [sheriffbot@chromium.org](#) on Thu, Jun 6, 2019, 9:07 AM EDT Project Member

Labels: -M-74 M-75 Target-75

[Comment 29](#) by [ochang@google.com](#) on Wed, Jul 24, 2019, 9:39 PM EDT Project Member

Issue 987388 has been merged into this issue.

[Comment 30](#) by [ochang@google.com](#) on Wed, Jul 24, 2019, 9:40 PM EDT Project Member

Any updates on this bug? We just got another report in bug 987388

[Comment 31](#) by [ochang@google.com](#) on Thu, Jul 25, 2019, 8:13 PM EDT Project Member

Cc: [fortenforge@google.com](#) [pelizzi@google.com](#)

[Comment 32](#) by [mkwst@chromium.org](#) on Wed, Jul 31, 2019, 2:27 AM EDT Project Member

Status: Assigned (was: Available)

Owner: [chilly@chromium.org](#)

+chilly@: Can you triage this, please?

[Comment 33](#) by [sheriffbot@chromium.org](#) on Wed, Jul 31, 2019, 9:04 AM EDT Project Member

Labels: -M-75 M-76 Target-76

[Comment 34](#) by [chilly@chromium.org](#) on Wed, Jul 31, 2019, 12:29 PM EDT Project Member

The SameSite context is being computed incorrectly because we update the `site_for_cookies` on the redirect, but the initiator here isn't being set properly (it's still null on the redirect). The initiator check is the only difference between returning strict/lax, so we can't distinguish between strict and lax correctly if the initiator isn't set.

[Comment 35](#) by [chilly@chromium.org](#) on Thu, Aug 1, 2019, 2:36 PM EDT Project Member

mkwst: What is the desired behavior for (main frame) cross-site redirects? I'm having a bit of trouble understanding what the spec says about them.

I have a CL in the works that:

- * does not send Strict cookies to b for a->b redirects,

- * does not send Strict cookies to a (the second time around) for a->b->a redirects,

- * does not send Strict cookies on any redirect that has had a cross-origin redirect anywhere along the chain.

Does that match the requirements of the spec?

[Comment 36](#) by [chilly@chromium.org](#) on Thu, Aug 1, 2019, 4:02 PM EDT Project Member

On second thought, `s/origin/eTLD+1/`. So an `a.com->sub.a.com` redirect would send Strict cookies.

[Comment 37](#) by [chilly@chromium.org](#) on Thu, Aug 1, 2019, 9:35 PM EDT Project Member

Huh. Apparently there are WPTs that disagree with me.

https://cs.chromium.org/chromium/src/third_party/blink/web_tests/external/wpt/cookies/samesite/fetch.html?i=31
`create_test(ORIGIN, redirectTo(CROSS_SITE_ORIGIN, ORIGIN), SameSiteStatus.STRICT, "Cross-site redirecting to same-host fetches are strictly same-site");`

Mike, I would appreciate it if you could clarify what is supposed to happen here?

[Comment 38](#) by [chilly@chromium.org](#) on Thu, Aug 8, 2019, 8:08 PM EDT Project Member

Status: WontFix (was: Assigned)

Ok, after talking with Mike, it seems this is working as intended.

Because the first navigation is initiated from the omnibox, we can consider this a browser-initiated navigation so Strict cookies should be sent.

Thanks for the bug report though! Please let us know if you find any other possible SameSite bugs.

[Comment 39](#) by [s.h.h...@gmail.com](#) on Thu, Aug 8, 2019, 8:10 PM EDT

Have you also looked into example of [Comment 9](#)?

[Comment 40](#) by [chilly@chromium.org](#) on Thu, Aug 8, 2019, 8:14 PM EDT Project Member

I believe the example in [comment 9](#) is the same case as reported in <https://bug.com/937239> which is also working as intended, apparently.

[Comment 41](#) by [s.h.h...@gmail.com](#) on Thu, Aug 8, 2019, 8:20 PM EDT

Redirect inside iframe sends Strict cookie. So AFAICT, you are saying that any website that embed other site has potential SameSite Strict cookie bypass? That's ... great :)

[Comment 42](#) by [mnenke@chromium.org](#) on Thu, Aug 8, 2019, 8:35 PM EDT Project Member

Suppose you have more control over who you embed than who tries to embed you, but yea, seems surprising to me, too.

[Comment 43](#) by [lukasza@chromium.org](#) on Mon, Aug 12, 2019, 2:38 PM EDT Project Member

Status: Assigned (was: WontFix)

I am not sure how much can be done for browser-initiated navigations (with `request_initiator=none`). Some aspects of this are being discussed in [issue 946505](#) and [issue 946501](#) (in context of Sec-Fetch-Site, but I think the outcome should also apply to SameSite cookies by virtue of it also being based on `request_initiator`).

OTOH, it seems to me that we "can" fix things for website-initiated navigations/requests (including subframe navigations like in [#c9](#) here). I note that Sec-Fetch-Site algorithm specifically considers redirects (e.g. see <https://github.com/w3c/webappsec-fetch-metadata/issues/28>) to protect itself against this kind of attacks - maybe SameSite-cookie algorithm should also do something similar (although this probably would have to be treated as a breaking change...). FWIW, I don't see any hits for "redirect" on <https://tools.ietf.org/html/draft-west-first-party-cookies-07>.

[Comment 44](#) by [mmoroz@chromium.org](#) on Mon, Aug 19, 2019, 1:38 PM EDT Project Member

[Issue 906442](#) has been merged into this issue.

[Comment 45](#) by [sheriffbot@chromium.org](#) on Wed, Sep 11, 2019, 9:05 AM EDT Project Member

Labels: -M-76 M-77 Target-77

[Comment 46](#) by [ajgo@google.com](#) on Thu, Oct 10, 2019, 2:31 PM EDT Project Member

[Issue 1012042](#) has been merged into this issue.

[Comment 47](#) by [sheriffbot@chromium.org](#) on Wed, Oct 23, 2019, 9:15 AM EDT Project Member

Labels: -M-77 Target-78 M-78

[Comment 48](#) by [sheriffbot@chromium.org](#) on Wed, Dec 11, 2019, 9:16 AM EST Project Member

Labels: -M-78 Target-79 M-79

Comment 49 by sheriffbot@chromium.org on Wed, Feb 5, 2020, 10:51 AM EST Project Member

Labels: -M-79 M-80 Target-80

Comment 50 by mkwst@chromium.org on Thu, Feb 20, 2020, 4:57 AM EST Project Member

Cc: annev...@gmail.com

Elsewhere, Anne noted:

===
We fixed an equivalent bug in https://bugzilla.mozilla.org/show_bug.cgi?id=1453814 but the cookie drafts have changed quite a bit since then and Chrome doesn't follow the old draft at least. And now we're getting compat issues.
===

Comment 51 by chilly@chromium.org on Mon, Feb 24, 2020, 7:50 PM EST Project Member

mkwst: I'm happy to make the changes if you tell me what the correct behavior should be.

I believe Chrome's current implementation is correct for this scenario. The spec currently bases the same-site/cross-site calculation (<https://tools.ietf.org/html/draft-ietf-httpbis-rtc6265bis-05#section-5.2>) on the request's client, and as far as I can tell, the request's client is not updated on a redirect (<https://fetch.spec.whatwg.org/#http-redirect-fetch>). Please let me know if that's wrong.

IRC, the last time we spoke about this, the concern was that dropping these cookies would break stuff like companies' internal shortlinks.

Comment 52 by annev...@gmail.com on Tue, Feb 25, 2020, 5:58 AM EST Project Member

I think there's two separate problems as also noted upthread:

1. User-initiated top-level navigation that redirects. I'm inclined to agree that sending cookies there is probably fine.
2. Framed navigation that redirects: <https://shhnhk.azurewebsites.net/framer.php?url=https://test.shhnhk.com/location.php?url=https://shhnhk.azurewebsites.net/SameSite.php>. Chrome's cookie sharing here seems rather problematic.

Comment 53 by chilly@chromium.org on Tue, Feb 25, 2020, 10:52 AM EST Project Member

For 2, that was addressed separately here (<https://bug.com/937334>) and the conclusion at the time was that A framing [B->A redirect] should send SameSite cookies on the redirected request to A, as that is an A frame inside of A, which is same-site. I'm still having trouble reading the spec in a way that suggests anything should change on redirects.

My first thought at the time was that it seemed intuitively incorrect to allow SameSite cookies in the framed redirect case, but I was then convinced that it was correct. Willing to be convinced back again... I agree that if we're having compat problems across browsers, we should align on something and update the spec if it's unclear.

Comment 54 by annev...@gmail.com on Tue, Feb 25, 2020, 11:35 AM EST Project Member

Thanks chilly, that's helpful!

My unease with the "Chrome model" is that elsewhere we do try to be "strict" about these kind of redirects as they can theoretically influence the target page in unexpected ways that compromise its security. See also <https://github.com/whatwg/fetch/issues/737>.

I also have a hard time pinpointing where the specification deals with the source of a request (rather than ancestor documents), but it does claim to care about CSRF and I don't think this adequately defends against CSRF. If instead of doing redirects, B would navigate to A, would the cookies then also be included? The explanatory section suggests that for strict cookies that ought not to work, but the normative algorithm is rather unclear on this to me. And I'm not sure how to distinguish a redirect from B to A from B navigating to A threat-model-wise. Both seem equally bad.

Hope that helps.

Comment 55 by chilly@chromium.org on Tue, Feb 25, 2020, 5:02 PM EST Project Member

I think it would make sense to treat redirects as if they were navigations. I'm not sure how to work that into the spec in a reasonable way. I'd guess we'd have to define it based on something other than the request's client, which isn't updated on redirects. Maybe it makes sense to also incorporate the request's url list, mirroring the sec-fetch-site definition of "same-site"?

For more SameSite vs redirect fun, see also:

<https://github.com/httpwg/http-extensions/issues/593>
<https://github.com/httpwg/http-extensions/issues/889>
<https://github.com/httpwg/http-extensions/issues/773>

Comment 56 by lukasza@chromium.org on Tue, Feb 25, 2020, 5:36 PM EST Project Member

RE: #c55:

I think I agree that it is desirable for strict SameSite cookies to behave consistently with Sec-Fetch-Site (wrt handling of redirects).

OTOH, right now Sec-Fetch-Site does track redirect hops, but the redirecting servers do not replace initiators - instead the initiator is compared with each intermediate target and the safest minimum (cross-site < cross-origin < same-origin) is used [1,2]. AFAIU this is different from what #c54 and #c55 want - I think they want the intermediate servers to replace the initiator, not the target.

- [1] See the std::max usage in https://chromium.googlesource.com/chromium/src/+1309ed3cd6027d82c3780aae0c42de7704825833/services/network/sec_header_helpers.cc#105
[2] <https://www.w3.org/TR/fetch-metadata/#sec-fetch-site-header>

Comment 57 by annev...@gmail.com on Wed, Feb 26, 2020, 4:28 AM EST Project Member

Cc: amarc...@mozilla.com

Comment 58 by chilly@chromium.org on Wed, Feb 26, 2020, 1:01 PM EST Project Member

I think I had tried modifying the initiator on redirect hops at one point, but it broke some other stuff that depends on initiator origin. Maybe we just need a separate SameSite-cookie-specific initiator field? It's possible that we could just look at the URLRequest's redirect chain, since it seems that document.cookie doesn't look at the initiator anyway.

Comment 59 by mmenke@chromium.org on Wed, Feb 26, 2020, 1:03 PM EST Project Member

Note that URLRequest doesn't currently have the full redirect chain - extensions (and maybe ServiceWorker - think other things, too) can intercept the request and redirect it, and we lose the entire pre-existing chain when that happens.

Comment 60 by sheriffbot on Thu, Apr 9, 2020, 12:32 PM EDT Project Member

Labels: -M-80 Target-81 M-81

Comment 61 by ajgo@google.com on Mon, Apr 20, 2020, 4:05 PM EDT Project Member

~~Issue-806483~~ has been merged into this issue.

Comment 62 by ajgo@google.com on Mon, Apr 20, 2020, 4:06 PM EDT Project Member

Marshal ping: any progress on this samesite redirection problem?

Comment 63 by chilly@chromium.org on Mon, Apr 20, 2020, 4:14 PM EDT Project Member

I don't think anyone is actively working on this, so no.

Comment 64 by sheriffbot on Wed, May 20, 2020, 1:33 PM EDT Project Member

Labels: -M-81 M-83 Target-83

Comment 65 by sheriffbot on Thu, Jul 16, 2020, 1:37 PM EDT Project Member

Labels: -M-83 Target-84 M-84

Comment 66 by sheriffbot on Wed, Aug 26, 2020, 1:43 PM EDT Project Member

Labels: -M-84 Target-85 M-85

Comment 67 by sheriffbot on Wed, Oct 7, 2020, 1:43 PM EDT Project Member

Labels: -M-85 M-86 Target-86

Comment 68 by sheriffbot on Fri, Oct 30, 2020, 6:49 PM EDT Project Member

Labels: reward-potential

Comment 69 by sheriffbot on Wed, Nov 18, 2020, 12:28 PM EST Project Member

Labels: -M-86 M-87 Target-87

Comment 70 by chilly@chromium.org on Tue, Dec 29, 2020, 5:13 PM EST Project Member

There is a spec change that addresses this issue: <https://github.com/httpwg/http-extensions/pull/1348>

I'm now working on incorporating the redirect chain into the Strict context calculation to align with the spec change.

Comment 71 by chilly@chromium.org on Wed, Jan 13, 2021, 1:02 PM EST Project Member

Cc: miketaylor@chromium.org

Comment 72 by sheriffbot on Wed, Jan 20, 2021, 12:27 PM EST Project Member

Labels: -M-87 Target-88 M-88

Comment 73 by adetaylor@google.com on Wed, Jan 20, 2021, 6:55 PM EST Project Member

Labels: -reward-potential external_security_report

Comment 74 by sheriffbot on Wed, Mar 3, 2021, 12:26 PM EST Project Member

Labels: -M-88 Target-89 M-89

Comment 75 by Git Watcher on Mon, Mar 8, 2021, 5:32 PM EST Project Member

The following revision refers to this bug:

<https://chromium.googlesource.com/chromium/src/+306b8fba167a809c5389a58d65bee438ca3bd15d>

commit 306b8fba167a809c5389a58d65bee438ca3bd15d

Author: Lily Chen <chilly@chromium.org>

Date: Mon Mar 08 22:31:23 2021

SameSite cookies: Consider redirect chain for same-site requests

The cookie spec is being amended in

<https://github.com/httpwg/http-extensions/pull/1348>

to consider the redirect chain when computing whether a request is considered same-site.

This aligns with the new specification by considering a request cross-site if any URL in the redirect chain was cross-site from the current request URL.

~~Bug-820404~~

Change-Id: I060026647ccea2a97267e865c8292ac64915e87b

Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2605504>

Commit-Queue: Lily Chen <chilly@chromium.org>

Reviewed-by: Maksim Orlovich <morlovich@chromium.org>

Reviewed-by: Min Qin <qinmin@chromium.org>

Reviewed-by: Andrey Kosyakov <cseq@chromium.org>

Cr-Commit-Position: refs/heads/master@{#60890}

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/content/browser/devtools/devtools_url_loader_interceptor.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/content/browser/download/download_browsertest.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/content/browser/net/http_cookie_browsertest.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/net/cookies/cookie_util.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/net/cookies/cookie_util.h

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/net/cookies/cookie_util_unittest.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/net/url_request/url_request_http_job.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/net/url_request/url_request_unittest.cc

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/external/wpt/cookies/samesite/fetch.https.html

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/external/wpt/cookies/samesite/form-get-blank.https.html

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/external/wpt/cookies/samesite/form-post-blank.https.html

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/external/wpt/cookies/samesite/img.https.html

[modify] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/external/wpt/cookies/samesite/multiple-samesite-attributes.https.html

[add] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/http/tests/inspector-protocol/fetch/fetch-samesite-cookies-expected.txt

[add] https://crrev.com/306b8fba167a809c5389a58d65bee438ca3bd15d/third_party/blink/web_tests/http/tests/inspector-protocol/fetch/fetch-samesite-cookies.js

Comment 76 by adetaylor@google.com on Wed, Mar 10, 2021, 4:39 PM EST Project Member

chilly@ thanks for working on this really long-standing security bug. Yay! Do you consider [#c75](#) to be a complete fix? If so please mark the bug as fixed so we can start the processes to credit and/or reward the reporter properly.

Comment 77 by chilly@chromium.org on Wed, Mar 10, 2021, 4:59 PM EST Project Member

It's like 95% of a fix (the remainder is that, if the page is reloaded by the browser following a cross-site redirect, we're suppose to treat it as cross-site, but since we lose the redirect chain at that point there's no easy way to do that. I don't have any plans to tackle this issue at this time, since it seems edge-casey.)

I would also be cautious about this causing lots of web breakage, especially around logins and payments. We may end up having to revert it and/or roll out slowly/add a finch kill-switch due to the potential for web incompatibility.

Tl;dr: I would keep this open at this time.

(Feel free to reward the reporter though. I don't really know how that process works.)

Comment 78 by sheriffbot on Wed, Mar 10, 2021, 8:07 PM EST Project Member

Labels: reward-potential

Comment 79 by [zhangtiff@google.com](#) on Wed, Mar 17, 2021, 7:09 PM EDT Project Member

Labels: -reward-potential external_security_bug

Comment 80 by [chilly@chromium.org](#) on Wed, Mar 17, 2021, 7:26 PM EDT Project Member

Cc: [bingler@chromium.org](#)

Comment 81 by [adetaylor@google.com](#) on Tue, Apr 6, 2021, 12:32 PM EDT Project Member

chilly@ thanks. It's fine with me to keep this open, though we'll only send it to the VRP panel when the bug is finally marked fixed. The alternative is to mark this as Fixed and raise a new crbug for the edge-case.

Comment 82 by [chilly@chromium.org](#) on Tue, Apr 13, 2021, 1:29 PM EDT Project Member

Status: Fixed (was: Assigned)

Comment 83 by [chilly@chromium.org](#) on Tue, Apr 13, 2021, 1:45 PM EDT Project Member

I filed <https://crbug.com/1198620> for the refresh case.

Comment 84 by [sheriffbot](#) on Tue, Apr 13, 2021, 1:56 PM EDT Project Member

Labels: -Restrict-View-SecurityTeam Restrict-View-SecurityNotify

Comment 85 by [sheriffbot](#) on Wed, Apr 14, 2021, 12:41 PM EDT Project Member

Labels: reward-topanel

Comment 86 by [amyressler@google.com](#) on Wed, Apr 28, 2021, 7:24 PM EDT Project Member

Labels: -reward-topanel reward-unpaid reward-3000

*** Boilerplate reminders! ***

Please do NOT publicly disclose details until a fix has been released to all our users. Early public disclosure may cancel the provisional reward. Also, please be considerate about disclosure when the bug affects a core library that may be used by other products. Please do NOT share this information with third parties who are not directly involved in fixing the bug. Doing so may cancel the provisional reward. Please be honest if you have already disclosed anything publicly or to third parties. Lastly, we understand that some of you are not interested in money. We offer the option to donate your reward to an eligible charity. If you prefer this option, let us know and we will also match your donation - subject to our discretion. Any rewards that are unclaimed after 12 months will be donated to a charity of our choosing.

Please contact [security-vrp@chromium.org](#) with any questions.

Comment 87 by [amyressler@chromium.org](#) on Wed, Apr 28, 2021, 8:05 PM EDT Project Member

Congratulations! The VRP Panel has decided to award you \$3000 for this report. Nice job!

Comment 88 by [s.h.h...@gmail.com](#) on Wed, Apr 28, 2021, 8:20 PM EDT

Thanks!

Comment 89 by [amyressler@google.com](#) on Fri, Apr 30, 2021, 2:00 PM EDT Project Member

Labels: -reward-unpaid reward-inprocess

Comment 90 by [amyressler@chromium.org](#) on Mon, May 24, 2021, 11:44 AM EDT Project Member

Labels: Release-0-M91

Comment 91 by [amyressler@google.com](#) on Mon, May 24, 2021, 2:19 PM EDT Project Member

Labels: CVE-2021-30537 CVE_description-missing

Comment 92 by [amyressler@google.com](#) on Mon, Jun 7, 2021, 3:27 PM EDT Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 93 by [sheriffbot](#) on Wed, Jul 21, 2021, 1:32 PM EDT Project Member

Labels: -Restrict-View-SecurityNotify allpublic

This bug has been closed for more than 14 weeks. Removing security view restrictions.

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot