Starred by 2 users

Owner:	antoniosartori@chromium.org
CC:	mkwst@chromium.org rakina@chromium.org szuend@chromium.org dbarcos@chromium.org creis@chromium.org lukasza@chromium.org arthu...@chromium.org sigurds@chromium.org antoniosartori@chromium.org wfh@chromium.org ajgo@chromium.org
Status:	Fixed *(Closed)*
Components:	Internals>Sandbox>SiteIsolation
Modified:	Jun 15, 2021
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Android, Windows, Chrome, Mac
Pri:	1
Type:	Bug-Security

Security_Impact-Stable
Security_Severity-Medium
allpublic
CVE_description-submitted
merge-merged-4240
merge-merged-86
LTR-Merged-86
LTS-Security-86
Release-0-M89
CVE-2021-21175

Blocking: Issue 1170242

Issue 1146651: X-Frame-Options console error leaks cross-origin redirect information to a cross-site renderer process

Reported by jun.k...@microsoft.com on Fri, Nov 6, 2020, 8:44 PM EST Project Member

Code

VULNERABILITY DETAILS

When X-Frame-Options (XFO) is set to cross-origin redirect destination, XFO error is sent to the parent frame.
https://source.chromium.org/chromium/chromium/src/+/master:content/browser/renderer_host/ancestor_throttle.cc;l=307-308;drc=a94154612c4dab6de019ce58df17924b9c016f77

Therefore, cross-origin redirect information can be obtained by reading memory (e.g. a Spectre attack).

This attack would allow e.g. reading Facebook username of the user by iframing https://www.facebook.com/me.

Adding XFO header in redirect can't mitigate this issue because it's ignored for historical reason (~~issue 835465~~). So if this can't be fixed by the browser, I think it's better to enforce CORP on navigations triggered by frames (i.e. https://github.com/whatwg/fetch/issues/1113).

VERSION

Chrome Version: 86.0.4240.183 stable
Operating System: Windows 10

REPRODUCTION CASE

1. Open goog_read.html
2. Attach WinDbg
3. Click Go button
4. Click break in WinDbg and run `!address /f:Heap /c:"s -a %1 %2 \"https://www.google.com/?\"""`
5. Observe that `https://www.google.com/?secret` is in the memory.

goog_read.html
110 bytes View Download

redirect_to_secret.php
96 bytes View Download

Comment 1 by creis@chromium.org on Fri, Nov 6, 2020, 8:54 PM EST Project Member

Cc: lukasza@chromium.org creis@chromium.org

lukasza@: Would this be related to the work in issue 973885? Not sure if XFO was considered there, or if there's a separate way to address DevTools console messages with sensitive information like this.

Comment 2 by kenrb@chromium.org on Fri, Nov 6, 2020, 9:53 PM EST Project Member

Labels: Security_Impact-Stable Security_Severity-Low OS-Android OS-Chrome OS-Linux OS-Mac OS-Windows Pri-2

Setting this as low severity since it only leaks redirect URLs. Someone on the CSA team can bump it if they think it is more serious than that.

Comment 3 by mkwst@chromium.org on Mon, Nov 9, 2020, 2:10 AM EST Project Member

Cc: arthu...@chromium.org antoniosartori@chromium.org

My recollection is that this is safe because devtools' renderer is process-isolated from the web-facing renderer. Is that not the case? We've made the assumption that it's safe-enough to send things to devtools that we wouldn't send to the web, and if that assumption doesn't hold, it seems like something we should make true rather than reducing the context we provide to developers. :)

+Arthur and Antonio who worked in this area for both XFO and `frame-ancestors` (which likely has the same information in its console message).

Comment 4 by jun.k...@microsoft.com on Mon, Nov 9, 2020, 2:56 AM EST Project Member

While I'm not too familiar with Devtools, this particular console message (and likely others) is just sent to renderer process. This kinda makes sense because at this point, devtools might not be opened, so devtools process might not exist. So from the code, it looks like the message is just sent to the renderer. Which is fine in many cases but not this one because this message is sent to parent frame which can be a cross-site renderer.

Comment 5 by mkwst@chromium.org on Mon, Nov 9, 2020, 3:10 AM EST Project Member

Cc: sigurds@chromium.org dbarcos@chromium.org

Interesting. That's not the model I had in mind, and I agree that it's unfortunate to leak this data to a web renderer (though I agree with Ken's suggestion of severity=low).

+Sigurd and Daniel for insight from the devtools side.

Comment 6 by antoniosartori@chromium.org on Mon, Nov 9, 2020, 3:17 AM EST Project Member

For the CSP frame-ancestors directive, we are doing some extra work to strip sensitive information from the url we report in case of a cross-origin parent iframe.

Should we just do the same also for X-Frame-Options violations? I think using the original url before redirects should be fine there.

Comment 7 by sigurds@chromium.org on Mon, Nov 9, 2020, 3:29 AM EST Project Member

We could file an inspector issue on the browser-side storage, and remove the critical information from the console message / remove the console message altogether. The issue storage in the browser is (obviously) not shared with the renderer. We'd need to take a look whether we can log to the current RFH, or to the parent, to make sure the issue is available.

Comment 8 by arthu...@chromium.org on Mon, Nov 9, 2020, 4:30 AM EST Project Member

Yes, the problem is we need route the message via a cross-origin renderer process to reach the devtool process (e.g. issue 721329)
[comment 6] is the temporary solution I would have suggested, but [comment 7] looks promising. sigurds@ would you have more information about this? What functions from the browser process can be used for this?

Comment 9 by arthu...@chromium.org on Mon, Nov 9, 2020, 7:33 AM EST Project Member

Cc: szuend@chromium.org

Comment 10 by sigurds@chromium.org on Mon, Nov 9, 2020, 7:35 AM EST Project Member

One could use

https://source.chromium.org/chromium/chromium/src/+/master:content/browser/devtools/devtools_instrumentation.h;l=209;drc=863a8984e2cc361ca2c000d42660b3c2801fb1aa

to add an issue. There would be small front-end changes required (to add the issue text and link the resources), and there would be no console message anymore. Would that be acceptable?

Comment 11 by sigurds@chromium.org on Mon, Nov 9, 2020, 7:41 AM EST Project Member

(Alternatively a console message without the sensitive URL could be issued on top of the issue).

Comment 12 by arthu...@chromium.org on Mon, Nov 9, 2020, 7:47 AM EST Project Member

That sounds great! That would be acceptable.

This can't be attributed to an existing document (=RenderFrameHostImpl), but I see the implementation of this function doesn't really care anyway.

Comment 13 by sigurds@chromium.org on Mon, Nov 9, 2020, 7:50 AM EST Project Member

> This can't be attributed to an existing document (=RenderFrameHostImpl), but I see the implementation of this function doesn't really care anyway.
It should care, as noted on the other bug 721329, we are using render document associated storage. Maybe filing the issue with the parent RFH is good enough?

Comment 14 by arthu...@chromium.org on Mon, Nov 9, 2020, 7:57 AM EST Project Member

> Maybe filing the issue with the parent RFH is good enough?

I think it's good, because the parent tries to embedded a resource (the children) that doesn't want to be embedded. I would attribute the error to the parent. As long as we do not sent the data to the parent's renderer process, it's good.

Comment 15 by antoniosartori@chromium.org on Mon, Nov 9, 2020, 8:19 AM EST Project Member

This sounds very interesting! It would be helpful for many other security violation messages we have. I think we should discuss it more in depth and plan it properly.

Until then, however, I would still solve this bug by just stripping sensitive data from the url.

Comment 16 by ellyj...@chromium.org on Wed, Nov 11, 2020, 6:40 PM EST Project Member

Status: Untriaged (was: Unconfirmed)

Mac triage: am I correct that this bug is confirmed at this point and awaiting triage?

Comment 17 by sigurds@chromium.org on Thu, Nov 12, 2020, 2:42 AM EST Project Member

I didn't reproduce personally, but I think to fix this two independent steps would be good:
- (immediately, asap) Remove the critical information from the console message
- for M89: Add an inspector issue with the critical information to the browser-side storage and show it in the issues tab in the DT front-end

Comment 18 by antoniosartori@chromium.org on Thu, Nov 12, 2020, 3:40 AM EST Project Member

Status: Assigned (was: Untriaged)
Owner: antoniosartori@chromium.org

Agree. I think I can do the immediate fix.

Comment 19 by bugdroid on Wed, Nov 18, 2020, 4:34 AM EST Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src.git/+/93ce5606cd9a9597993ba70670b4092ab6722281

commit 93ce5606cd9a9597993ba70670b4092ab6722281
Author: Antonio Sartori <antoniosartori@chromium.org>
Date: Wed Nov 18 09:33:55 2020

Strip url to origin in X-Frame-Options violation messages

X-Frame-Options violations are logged via a console message in the
parent frame. To avoid leaking sensitive data to the parent frame,
let's report as "blocked url" just the origin of the blocked frame's

url, as we are already doing for the frame-ancestors CSP directive.

~~Bug: 1146654~~
Change-Id: If5e5ac62f7e44e714b109e6adc389f11999e0f8b
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2534851
Commit-Queue: Antonio Sartori <antoniosartori@chromium.org>
Reviewed-by: Charlie Reis <creis@chromium.org>
Reviewed-by: Arthur Sonzogni <arthursonzogni@chromium.org>
Cr-Commit-Position: refs/heads/master@{#828651}

[modify]
https://crrev.com/93ce5606cd9a9597993ba70670b4092ab6722281/android_webview/javatests/src/org/chromium/android_webview/test/ConsoleMessagesForBlockedLoadsTest.java
[modify] https://crrev.com/93ce5606cd9a9597993ba70670b4092ab6722281/content/browser/renderer_host/ancestor_throttle.cc
[modify] https://crrev.com/93ce5606cd9a9597993ba70670b4092ab6722281/content/browser/site_per_process_browsertest.cc
[modify] https://crrev.com/93ce5606cd9a9597993ba70670b4092ab6722281/third_party/blink/web_tests/http/tests/security/XFrameOptions/x-frame-options-deny-delete-frame-in-load-event-expected.txt

Comment 20 by jun.k...@microsoft.com on Wed, Nov 25, 2020, 1:18 AM EST Project Member

Hi, could someone explain why this is a low severity bug? Historically, leaking cross-origin redirect information was treated as medium severity bug (e.g. ~~bug 799847~~).

Comment 21 by antoniosartori@chromium.org on Wed, Nov 25, 2020, 2:34 AM EST Project Member

I am not sure about the priority, but notice the difference with ~~bug 799847~~: while in that case the leaked information was directly accessible via javascript, here we are leaking only to the renderer process memory, but the information is not exposed to javascript. You still need some attack to read arbitrary memory in the renderer process in order to actually access the information.

Comment 22 by jun.k...@microsoft.com on Wed, Nov 25, 2020, 12:33 PM EST Project Member

Sure, but with Spectre exploit, you can read memory inside a renderer process[1] (which is not a bug, but by design at this point). So I'm not sure that should make a difference in severity.

[1] https://v8.dev/blog/spectre#:~:text=read%20a%20process%E2%80%99s%20entire%20address%20space

Comment 23 by nasko@chromium.org on Wed, Nov 25, 2020, 6:19 PM EST Project Member

Given the change that landed in #19, we will only be sending the origin to the renderer process, right? Doesn't this make it Low severity at this point? We should no longer be sending the full URL, so knowing the origin to which the redirect was is still a leak, but I wouldn't expect it to be divulging a lot of information. Do you know of cases where attacker can discern private information of the user based on the origin to which a navigation is redirected?

Comment 24 by jun.k...@microsoft.com on Wed, Nov 25, 2020, 7:25 PM EST Project Member

Yes, after the patch, it'll be low severity or not a security bug at all. However I was arguing about initial severity assessment against original report before the patch :)

Comment 25 by antoniosartori@chromium.org on Thu, Nov 26, 2020, 3:50 AM EST Project Member

Status: Available (was: Assigned)
Owner: ----

Unassigning myself since the immediate fix is done. We should still avoid leaking the origin, but notice this problem is shared by all messages reported by the AncestorsThrottle (i.e. also for CSP 'frame-ancestors' and CSPEE violations). Not sure if we should open additional bugs for them.

Comment 26 by arthursonzogni@google.com on Thu, Nov 26, 2020, 5:36 AM EST Project Member

> this problem is shared by all messages reported by the AncestorsThrottle (i.e. also for CSP 'frame-ancestors' and CSPEE violations

From the beginning, I added CSPContext::SanitizeDataForUseInCspViolation for this purpose. Do you think this isn't enough?

Comment 27 by antoniosartori@chromium.org on Thu, Nov 26, 2020, 5:48 AM EST Project Member

CSPContext::SanitizeDataForUseInCspViolation does leak the origin of the blocked frame, doesn't it?

Comment 28 by arthursonzogni@google.com on Thu, Nov 26, 2020, 6:36 AM EST Project Member

Cross-origin URL are stripped. The origin remains. At this time, we judged this was okay, but we might want to reconsider this. Could you open a bug?

Comment 29 by creis@chromium.org on Fri, Jan 22, 2021, 6:21 PM EST Project Member

Status: Fixed (was: Available)
Owner: antoniosartori@chromium.org
Labels: -Security_Severity-Low Security_Severity-Medium

I think there might be some confusion in the last few comments: we do not treat leaks of origins as a Site Isolation violation, so I don't think there's more to do on the security side. However, leaks of full URLs to a cross-site renderer are generally/aspirationally considered a Site Isolation violation, though we still have some that need to be eliminated in other places. As comment 22 notes, this matters even if you need Spectre or a compromised renderer to get the data.

In that sense, it sounds like this was a real vulnerability, and the full URL was removed in Antonio's r828651, which fixed the security issue. I agree with Jun in comment 20 that this should probably be Medium severity to be consistent with ~~issue 799847~~ (while perhaps not meriting High severity because redirect URLs are a limited type of cross-site data).

sigurds@: If there is additional DevTools work to do from comment 17, can you file a separate bug for that? I'll close this issue since (AIUI, and per comment 24) the security issue is resolved.

Comment 30 by creis@chromium.org on Fri, Jan 22, 2021, 6:21 PM EST Project Member

Cc: rakina@chromium.org

Comment 31 by sheriffbot on Sun, Jan 24, 2021, 1:38 PM EST Project Member

Labels: -Pri-2 Pri-1

Setting Pri-1 to match security severity Medium. If this is incorrect, please reset the priority. Sheriffbot won't make this change again.

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

Comment 32 by sheriffbot on Sun, Jan 24, 2021, 1:56 PM EST Project Member

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

Comment 33 by sigurds@chromium.org on Mon, Jan 25, 2021, 2:29 AM EST Project Member

Blocking: 1170242

Comment 34 by sigurds@chromium.org on Mon, Jan 25, 2021, 2:30 AM EST Project Member

Re comment 29: Filed bug 1170242 as a follow up.

Comment 35 by adetaylor@google.com on Fri, Feb 26, 2021, 1:08 PM EST Project Member

Labels: Release-0-M89

Comment 36 by adetaylor@google.com on Mon, Mar 1, 2021, 7:27 PM EST Project Member

Labels: CVE-2021-21175 CVE_description-missing

Comment 37 by vsavu@google.com on Wed, Mar 3, 2021, 5:55 AM EST Project Member

Labels: LTS-Merge-Request-86

Comment 38 by vsavu@google.com on Wed, Mar 3, 2021, 6:02 AM EST Project Member

Labels: LTS-Security-86

Comment 39 by gianluca@google.com on Wed, Mar 3, 2021, 10:38 AM EST Project Member

Labels: LTS-Merge-Approved-86

Comment 40 by Git Watcher on Mon, Mar 8, 2021, 5:29 AM EST Project Member

Labels: merge-merged-4240 merge-merged-86

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/37210e5ab0062b95e60c3169f6d07f765cadb6b8

commit 37210e5ab0062b95e60c3169f6d07f765cadb6b8
Author: Antonio Sartori <antoniosartori@chromium.org>
Date: Mon Mar 08 10:28:40 2021

Strip url to origin in X-Frame-Options violation messages

X-Frame-Options violations are logged via a console message in the
parent frame. To avoid leaking sensitive data to the parent frame,
let's report as "blocked url" just the origin of the blocked frame's
url, as we are already doing for the frame-ancestors CSP directive.

[M86 Merge]: ancestor_throttle.cc was moved.

(cherry picked from commit 93ce5606cd9a9597993ba70670b4092ab6722281)

~~Bug: 1146651~~
Change-Id: If5e5ac62f7e44e714b109e6adc389f11999e0f8b
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2534851
Commit-Queue: Antonio Sartori <antoniosartori@chromium.org>
Reviewed-by: Charlie Reis <creis@chromium.org>
Reviewed-by: Arthur Sonzogni <arthursonzogni@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#828651}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2731577
Reviewed-by: Achuith Bhandarkar <achuith@chromium.org>
Commit-Queue: Victor-Gabriel Savu <vsavu@google.com>
Cr-Commit-Position: refs/branch-heads/4240@{#1563}
Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify]
https://crrev.com/37210e5ab0062b95e60c3169f6d07f765cadb6b8/android_webview/javatests/src/org/chromium/android_webview/test/ConsoleMessagesForBlockedLoadsTest.java
[modify] https://crrev.com/37210e5ab0062b95e60c3169f6d07f765cadb6b8/content/browser/frame_host/ancestor_throttle.cc
[modify] https://crrev.com/37210e5ab0062b95e60c3169f6d07f765cadb6b8/content/browser/site_per_process_browsertest.cc
[modify] https://crrev.com/37210e5ab0062b95e60c3169f6d07f765cadb6b8/third_party/blink/web_tests/http/tests/security/XFrameOptions/x-frame-options-deny-delete-frame-in-load-event-expected.txt

Comment 41 by vsavu@google.com on Mon, Mar 8, 2021, 11:18 AM EST Project Member

Labels: -LTS-Merge-Approved-86 -LTS-Merge-Request-86 LTR-Merged-86

Comment 42 by amyressler@google.com on Tue, Mar 9, 2021, 12:58 PM EST Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 43 by sheriffbot on Tue, Jun 15, 2021, 1:52 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