Starred by 3 users

Owner:	csharrison@chromium.org
CC:	szager@chromium.org bokan@chromium.org
Status:	Fixed *(Closed)*
Components:	UI Blink>Input
Modified:	Nov 2, 2020
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Windows
Pri:	1
Type:	Bug-Security

Security_Impact-Stable
Security_Severity-Medium
allpublic
CVE_description-submitted
M-84
Target-84
Release-0-M86
CVE-2020-15985

Issue 1099276: Security: Cursor hijacking mitigation bypass

Reported by abalq...@microsoft.com on Thu, Jun 25, 2020, 10:43 AM EDT

Code

VULNERABILITY DETAILS
Please provide a brief explanation of the security issue.

VERSION
Chrome Version: 83.0.4103.116 (Official Build) (64-bit) (and tested latest canary)
Operating System: Windows 10 pro

REPRODUCTION CASE
Host the attached HTML files in one directory and open 'p.html' (from HTTP not FILE) to see a simulated attack using this.

Minimum repro (notice how the custom cursor covers more of the UI):
data:text/html,<iframe src="http://cr.kungfoo.net/style/cursor/abusive-cursor.html" style="width:700px;height:1000px;position:absolute;top:-100px;left:-100px;">

Background:

"[Deprecation] Custom cursors with size greater than 32x32 DIP intersecting native UI is deprecated and will be removed in M75, around June 2019. See https://www.chromestatus.com/features/5825971391299584 for more details."

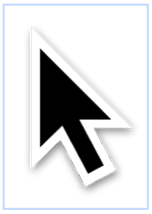
https://bugs.chromium.org/p/chromium/issues/detail?id=640227
https://bugs.chromium.org/p/chromium/issues/detail?id=880863

https://benjaminbenben.com/cursory-hack/
https://jameshfisher.github.io/cursory-hack/
http://cr.kungfoo.net/style/cursor/abusive-cursor.html

cursor-jacking-old.html
2.5 KB View Download

p.html
819 bytes View Download

cursor.png
9.8 KB View Download



popout.png
28.2 KB View Download

padlock.png
3.9 KB View Download

Comment 1 by bdea@chromium.org on Thu, Jun 25, 2020, 3:34 PM EDT Project Member

Owner: csharrison@chromium.org
Labels: Security_Impact-Stable Security_Severity-Medium
Components: UI

@csharrison can you take a look at this? I was able to reproduce it. It doesn't prevent me from closing the tab however.

Comment 2 by sheriffbot on Thu, Jun 25, 2020, 3:37 PM EDT Project Member

Status: Assigned (was: Unconfirmed)

Comment 3 by sheriffbot on Fri, Jun 26, 2020, 2:09 PM EDT Project Member

Labels: Target-84 M-84

Setting milestone and target because of Security_Impact=Stable and medium severity.

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

Comment 4 by sheriffbot on Fri, Jun 26, 2020, 2:46 PM EDT Project Member

Labels: 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 5 by csharrison@chromium.org on Wed, Jul 1, 2020, 6:51 PM EDT Project Member

Owner: ----
Cc: csharrison@chromium.org
Components: Blink>Input

Ah yeah this is an issue with the code. We are looking at intersections in two separate coordinate spaces, one for the OOPIF-iframe and one for the root page. I think that getting page-offsets in the OOPIF may not be possible.

One way to fix this would be to resolve the TODO here:
https://source.chromium.org/chromium/chromium/src/+/master:third_party/blink/renderer/core/input/event_handler.cc;drc=e3c225a83f88829c19e8120a9d9d3dcaf531cbfc;l=592

```
// TODO(csharrison): Consider sending a fallback cursor in the IPC to the
// browser process so we can do that calculation there instead.
```

This way the browser process can do the coordinate transformation, and we can also make this protection robust to malicious renderers as well. I don't have much expertise in this code (or cycles, since I have moved to another project), so adding Blink>Input for feedback.

Comment 6 by csharrison@chromium.org on Wed, Jul 1, 2020, 6:51 PM EDT Project Member

Status: Available (was: Assigned)

Comment 7 by csharrison@chromium.org on Thu, Jul 2, 2020, 9:53 AM EDT Project Member

I did a little digging and I think computing the offset from the root is doable without going through the browser proc:
https://chromium-review.googlesource.com/c/chromium/src/+/2278597

Let me see what a layout expert thinks.

Comment 8 by csharrison@chromium.org on Sun, Jul 5, 2020, 12:51 PM EDT Project Member

Status: Assigned (was: Available)
Owner: csharrison@chromium.org
Cc: -csharrison@chromium.org szager@chromium.org bokan@chromium.org

+dbokan, szager, moving CL discussion to the bug. szager for context the linked CL above attempts to drop large cursors (in the renderer) that are not fully contained within the visual viewport. One thing that came up in an offline chat w/ kenrb is that we don't have CSS transform information in the subframe. This brings up a few possible solutions:

1. Drop the large cursors if they are not fully contained within the enclosing frame itself (drop the visual viewport restriction). This entails changing the intervention scoping so might require a blink-dev PSA or amendment to the intent

2. Keep the existing patch as a partial mitigation that breaks down w/ transforms (or is possibly overly restrictive in some circumstances)
3. Something more sophisticated on the browser side

interested in any thoughts here

Comment 9 by szager@chromium.org on Mon, Jul 6, 2020, 12:17 PM EDT Project Member

There is currently a CL under review that would propagate a full transformation matrix from the parent frame to the child frame:

https://chromium-review.googlesource.com/c/chromium/src/+/2206924

You might be able to leverage that for a full solution here.

Comment 10 by bokan@chromium.org on Mon, Jul 6, 2020, 12:56 PM EDT Project Member

IMHO #1 is the way to go but, I think the way it's calculated, today if you pinch zoom in and mouse over the tab strip we'd still consider the cursor as over the OOPIF (conceptually the zoom puts the OOPIF behind the tab strip, consider a fullscreen OOPIF and zoom into the bottom of it). I think the same issue occurs in the main frame and is why it was compared to the visual viewport...

Not sure if this edge case is worth worrying about (since pages can't change zoom themselves) but could we intersect with the visual viewport as well as the OOPIF rect?

Comment 11 by csharrison@chromium.org on Mon, Jul 6, 2020, 2:34 PM EDT Project Member

bokan WDYT about using the transormation matrix szager mentioned in #9? I agree it doesn't have the conceptual simplicity of #1 but it probably is better for compat and is a fix that can be applied without any blink process. The main downside I could see is potential perf impact of applying the coordinate translation but I don't expect that to be too bad.

Comment 12 by bokan@chromium.org on Mon, Jul 6, 2020, 3:50 PM EDT Project Member

I missed his comment before sending mine. If we have the full transform then, yeah, that seems like it'd just fix the existing code which sgtm

Comment 13 by csharrison@chromium.org on Mon, Jul 6, 2020, 11:03 PM EDT Project Member

SGTM. I can try to code up a fix next week when I am back in office.

Comment 14 by bugdroid on Mon, Jul 20, 2020, 5:36 PM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src.git/+/827f22ba010c3304d0360227a64378ec79645278

commit 827f22ba010c3304d0360227a64378ec79645278
Author: Charlie Harrison <csharrison@chromium.org>
Date: Mon Jul 20 21:33:59 2020

Large cursor fallback: ensure correct coordinate space for OOPIFs

A previous change ensured that large custom cursors > 32x32
would be dropped if they are not fully contained with the visual
viewport. However, the computation did not account properly for OOPIFs,
where cursor coordinates were not adjusted to the viewport offset.
This CL further adjusts the cursor rect by translating it to the root
view's coordinate space via LocalToAncestorPoint, before
checking for containment within the visual viewport.

~~Bug: 1099276~~
Change-Id: I0a03e7cc249cd785f9e76f931cfc7931b127d56b
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2278597
Auto-Submit: Charlie Harrison <csharrison@chromium.org>
Reviewed-by: Ken Buchanan <kenrb@chromium.org>
Reviewed-by: Stefan Zager <szager@chromium.org>
Reviewed-by: David Bokan <bokan@chromium.org>
Commit-Queue: Charlie Harrison <csharrison@chromium.org>
Cr-Commit-Position: refs/heads/master@{#790122}

[modify] https://crrev.com/827f22ba010c3304d0360227a64378ec79645278/content/browser/site_per_process_hit_test_browsertest.cc
[add] https://crrev.com/827f22ba010c3304d0360227a64378ec79645278/content/test/data/large-cursor.html
[modify] https://crrev.com/827f22ba010c3304d0360227a64378ec79645278/third_party/blink/renderer/core/input/event_handler.cc

Comment 15 by sheriffbot on Tue, Jul 21, 2020, 1:32 PM EDT Project Member

csharrison: Uh oh! This issue still open and hasn't been updated in the last 14 days. This is a serious vulnerability, and we want to ensure that there's progress. Could you please leave an update with the current status and any potential blockers?

If you're not the right owner for this issue, could you please remove yourself as soon as possible or help us find the right one?

If the issue is fixed or you can't reproduce it, please close the bug. If you've started working on a fix, please set the status to Started.

Thanks for your time! To disable nags, add the Disable-Nags label.

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

Comment 16 by csharrison@chromium.org on Fri, Jul 24, 2020, 10:46 AM EDT Project Member

Status: Fixed (was: Assigned)

Comment 17 by sheriffbot on Fri, Jul 24, 2020, 3:12 PM EDT Project Member

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

Comment 18 by abalq...@microsoft.com on Tue, Aug 11, 2020, 12:32 PM EDT

CREDIT INFORMATION

Reporter credit: Abdulrahman Alqabandi, Microsoft Browser Vulnerability Research.

Comment 19 by adetaylor@google.com on Thu, Oct 1, 2020, 3:47 PM EDT Project Member

Labels: Release-0-M86

Comment 20 by adetaylor@google.com on Mon, Oct 5, 2020, 1:00 AM EDT Project Member

Labels: CVE-2020-15985 CVE_description-missing

Comment 21 by adetaylor@google.com on Mon, Oct 5, 2020, 11:20 AM EDT Project Member

Labels: OS-Windows

Comment 22 by adetaylor@google.com on Mon, Oct 5, 2020, 11:21 AM EDT Project Member

Setting OS=Windows to keep scripts happy. This may well affect many other platforms.

Comment 23 by sheriffbot on Fri, Oct 30, 2020, 1:49 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

Comment 24 by adetaylor@google.com on Mon, Nov 2, 2020, 9:15 PM EST Project Member

Labels: -CVE_description-missing CVE_description-submitted