Starred by 1 user

Owner:	kbr@chromium.org
CC:	kbr@chromium.org amineer@chromium.org vmi...@chromium.org pbomm...@chromium.org ligim...@chromium.org
Status:	Fixed *(Closed)*
Components:	Blink>WebGL
Modified:	Nov 18, 2017
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Android
Pri:	1
Type:	Bug-Security

Hotlist-Merge-Review
reward-2000
Security_Impact-Stable
Security_Severity-Medium
allpublic
reward-inprocess
M-60

Issue 675658: Security: Malicious WebGL page can capture and upload contents of other tabs

Reported by pault...@gmail.com on Mon, Dec 19, 2016, 6:23 PM UTC

Code

VULNERABILITY DETAILS

Using WebGL, a malicious web page can capture the contents of other tabs
as images and upload them to a server. Reproducible on a Samsung Galaxy S6.
Probably MEDIUM severity.

VERSION

Chrome Version: [55.0.2883.91] + [stable]
Operating System: [Android 6.0.1; SM-G920F Build/MMB29K]
Samsung Galaxy S6 UK model.

REPRODUCTION CASE

Note that since the issue might only be reproducible on the Samsung Galaxy S6,
you may prefer to just read `README.md` in the zip, which includes the reproduction instructions.
I attach `index.html`, a self-contained HTML file demonstrating the issue.
I also attach a zip archive with more details, the HTML file,
and a self-contained Python 3 script that will serve the HTML file
and will save the uploaded PNG image files captured by the HTML file.
It also includes some sample images to show what can be captured;
these are also described in the README.md.
See README.md for instructions and more details.

index.html
9.9 KB View Download

webgl-bug.zip
3.5 MB Download

Comment 1 by mbarb...@chromium.org on Mon, Dec 19, 2016, 7:28 PM UTC Project Member

Status: Assigned (was: Unconfirmed)

Owner: kbr@chromium.org
Labels: Security_Severity-Medium Security_Impact-Stable OS-Android Pri-1
Components: Internals>GPU>WebGL

I unfortunately don't have access to a Galaxy S6 to test this, but this looks like an interesting bug.

Ken, any ideas here? I'm wondering if there's any additional validation we could do.

Comment 2 by kbr@chromium.org on Mon, Dec 19, 2016, 11:17 PM UTC Project Member

Cc: pbomm...@chromium.org vmi...@chromium.org
Labels: -Pri-1 -Security_Severity-Medium Security_Severity-Low Pri-2
Components: Blink>WebGL

I tested this on a Google Pixel and can't reproduce the capture of corrupted VRAM.

I'd like to understand better whether the browser thinks the WebGL context's been lost at the time toDataURL is called. If it is, we should return a blank image rather than attempting to actually perform the readback.

The problem appears to be related to the injection of this long-running no-op loop in the fragment shader:

```
 for(
    int GLF_live3i = 0;
    GLF_live3i < 12244;
    GLF_live3i ++
 )
 {
 }
```

The first time I tried this test case on my MacBook Pro it caused a GPU hang and reset, but it wasn't reproducible afterward. I didn't have the server running (would be better if it was written in Python 2.x for macOS compatibility) but did see a flash of what looked like garbage VRAM.

Reducing to low security severity and P2 because evidence is that this affects only specific devices. We'd need one to diagnose this in more detail. Prudhvi, do we have an S6 in house that we can use for testing?

Comment 3 by pbomm...@chromium.org on Mon, Dec 19, 2016, 11:19 PM UTC Project Member

Cc: amineer@chromium.org

Comment 4 by pault...@gmail.com on Tue, Dec 20, 2016, 12:02 AM UTC

I am fairly certain the WebGL context is not lost on the Samsung Galaxy S6. There is no popup and no refreshing occurred.

The included fragment shader indeed probably only works for ARM Mali GPUs as in the Samsung Galaxy S6. I think some Sony Xperia phones may also have Mali GPUs, if that helps.

A different fragment shader might be able to cause a similar issue on other GPUs. Sometimes adding a "discard" or early "return" statement (possibly conditional on a uniform) is enough to cause garbage. But in this case we indeed believe it is the long running loop.

Changing the loop bound can also have an effect. E.g. a lower loop bound might not hang certain GPUs. Note that the loop bound could be provided in a uniform parameter, so a straightforward loop bound static analysis is unlikely to be enough to detect this.

I attach an updated zip where `server.py` works with Python 2 and 3.

webgl-bug2.zip
3.5 MB Download

Comment 5 by kbr@chromium.org on Tue, Dec 20, 2016, 12:49 AM UTC Project Member

Owner: ligim...@chromium.org
Cc: ligim...@chromium.org

Thanks. No garbage observed with your server on macOS 10.11.6 with Chrome 57.0.2956.0 (Official Build) canary (64-bit).

Ligi, Prudhvi, do we have a Samsung Galaxy S6 on which I could run a test? Ligi, assigning to you so this doesn't get dropped.

Comment 6 by amineer@chromium.org on Tue, Dec 20, 2016, 12:51 AM UTC Project Member

Cc: kbr@chromium.org

Comment 7 by kbr@chromium.org on Tue, Dec 20, 2016, 1:38 AM UTC Project Member

Status: ExternalDependency (was: Assigned)
Owner: kbr@chromium.org
Labels: -Pri-2 -Security_Severity-Low Security_Severity-Medium Pri-1

Got the S6 (thanks Prudhvi), and confirm that garbage is observed when running the test case.

Upgrading this bug to Security_Severity-Medium and P1. I think the information leak is significant

Upgrading this back to Security_Severity-Medium and P1. I think the information leak is significant.

Filed internal bug b/33757419 about this, in the bug category for the Mali GPU. I'll also contact a couple of colleagues at ARM and Samsung who may be able to help triage this.

My best guess is that the Mali GPU is aborting a long-running command buffer without signaling that the context is lost via the GL_KHR_robustness extension, although Chrome is allocating its contexts with that reset strategy turned on (visible in about:gpu). Chrome needs that reset notification in order to lose the WebGL context and prevent information leakage such as this. There is already code in place for this purpose.

I'm not sure what we can do about this yet. Marking ExternalDependency to indicate that we don't have any solutions yet from the Chrome side, and need feedback from ARM.

Comment 8 by sheriffbot@chromium.org on Tue, Dec 20, 2016, 2:02 PM UTC Project Member

Labels: M-56

Comment 9 by pault...@gmail.com on Wed, Jan 25, 2017, 11:48 AM UTC

Could I check the status of this?

1. Has the issue been reported to ARM? To whom? Has there been a response?
2. Has the issue been reported to Samsung? To whom? Has there been a response?
3. Can we get cc'ed in these conversations? And/or is there an id or reference number (from ARM/Samsung) we can use if we want to refer to these bug reports?
4. Do ARM and Samsung know that the shader was generated by fuzzing (using GLFuzz)?

Thanks

Comment 10 by amineer@chromium.org on Thu, Jan 26, 2017, 4:40 PM UTC Project Member

I've asked those questions on our internal bug. What I can share now is that Samsung is definitely aware of this; hopefully we can answer the rest of your questions in the near future.

kbr@, can you circle back once we have more details?

Comment 11 by kbr@chromium.org on Fri, Jan 27, 2017, 7:55 PM UTC Project Member

Yes, I'll update this bug when either ARM or Samsung gets back to us.

Comment 12 by awhalley@chromium.org on Mon, Feb 13, 2017, 8:13 PM UTC Project Member

Labels: reward-topanel

Comment 13 by sheriffbot@chromium.org on Fri, Mar 10, 2017, 2:00 PM UTC Project Member

Labels: -M-56 M-57

Comment 14 by sheriffbot@chromium.org on Thu, Apr 20, 2017, 1:00 PM UTC Project Member

Labels: -M-57 M-58

Comment 15 by sheriffbot@chromium.org on Tue, Jun 6, 2017, 1:01 PM UTC Project Member

Labels: -M-58 M-59

Comment 16 by lafo...@chromium.org on Tue, Jun 20, 2017, 7:32 PM UTC Project Member

Components: -Internals>GPU>WebGL

Comment 17 by pault...@gmail.com on Tue, Jul 18, 2017, 8:45 PM UTC

Could I check on the status of this?

Comment 18 by kbr@chromium.org on Tue, Jul 18, 2017, 8:50 PM UTC Project Member

Asked ARM/Samsung for a status update on the internal bug b/33757419 .

Comment 19 by pault...@gmail.com on Wed, Jul 19, 2017, 2:12 PM UTC

I can no longer reproduce this after updating my Samsung S6 (Europe).
Android 7.0; SM-G920F Build/NRD90M

The rendered image is transparent and, sometimes, the context is lost (I see "Rats! WebGL hit a snag").

Comment 20 by kbr@chromium.org on Wed, Jul 19, 2017, 3:33 PM UTC Project Member

Thanks. That's the expected behavior; KHR_robustness is supposed to lose the context in this situation. I've asked for confirmation on b/33757419 but most likely we'll have to close this as WontFix (no longer reproducible). Also, we never got a bug ID from ARM or Samsung.

Comment 21 by sheriffbot@chromium.org on Wed, Jul 26, 2017, 1:01 PM UTC Project Member

Labels: -M-59 M-60

Comment 22 by pault...@gmail.com on Thu, Aug 3, 2017, 3:07 PM UTC

Thanks. We have confirmation from Samsung that they saw the bug report and confirmation from ARM that they saw the report and fixed the bug thanks to our report.

Comment 23 by kbr@chromium.org on Tue, Aug 8, 2017, 3:47 AM UTC Project Member

That's great news.

Comment 24 by kbr@chromium.org on Tue, Aug 8, 2017, 4:19 AM UTC Project Member

Status: WontFix (was: ExternalDependency)

Closed the internal bug as Fixed, but closing this one as WontFix since we didn't make any code changes to Chromium in response, and since it's no longer reproducible after a rollout of Samsung's newest driver.

Comment 25 by sheriffbot@chromium.org on Tue, Aug 8, 2017, 1:02 PM UTC Project Member

Labels: -reward-topanel reward-ineligible

Comment 26 by pault...@gmail.com on Thu, Aug 10, 2017, 6:07 AM UTC

Is it possible to manually check if we are still eligible for a reward? My understanding of the program is that we might still be eligible. But I assume we are automatically marked as ineligible when the status is changed to WontFix.

"Bugs may be eligible even if they are part of the base operating system and can manifest through Chrome."

"...bugs may be eligible even if they are caused by components of the operating system or standard libraries. We're interested in rewarding any information that enables us to better protect our users. In the event of bugs in an external component, we are happy to take care of responsibly notifying other affected parties."

Comment 27 by kbr@chromium.org on Thu, Aug 10, 2017, 11:07 PM UTC Project Member

Status: ExternalDependency (was: WontFix)
Labels: -reward-ineligible reward-topanel

Oops. Sorry, didn't mean to break your eligibility for a reward, especially since the capture of content occurred through the browser.

Marking as ExternalDependency again and resetting the reward label.

Comment 28 by pault...@gmail.com on Thu, Aug 10, 2017, 11:37 PM UTC

Thanks so much. Hopefully it is not necessary for the bug to be marked as Fixed/Verified to trigger the next stage of the reward process. If it is necessary, perhaps this bug

can be manually flagged for review, as we believe it is fixed (and the fix is deployed) but this may not be clear.

Comment 29 by awhalley@google.com on Fri, Aug 11, 2017, 4:30 PM UTC Project Member

Status: Fixed (was: ExternalDependency)

Thanks! Going to mark this as Fixed since the security issue is indeed fixed (this is in line with how we track this in similar situations, like security bugs in Flash). The VRP panel will look at this soon. Thanks!

Comment 30 by sheriffbot@chromium.org on Sat, Aug 12, 2017, 1:04 PM UTC Project Member

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

Comment 31 by sheriffbot@chromium.org on Mon, Aug 14, 2017, 1:04 PM UTC Project Member

Labels: Merge-Request-61

Comment 32 by sheriffbot@chromium.org on Mon, Aug 14, 2017, 1:05 PM UTC Project Member

Labels: -Merge-Request-61 Merge-Review-61 Hotlist-Merge-Review

This bug requires manual review: M61 has already been promoted to the beta branch, so this requires manual review
Please contact the milestone owner if you have questions.
Owners: amineer@(Android), cmasso@(iOS), ketakid@(ChromeOS), govind@(Desktop)

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

Comment 33 by amineer@chromium.org on Mon, Aug 14, 2017, 3:28 PM UTC Project Member

Labels: -Merge-Review-61

Per c#24 there is nothing to merge here, but correct me if I'm wrong...

Comment 34 by pault...@gmail.com on Fri, Aug 18, 2017, 11:11 AM UTC

That's right! I think the aim is to mark this as fixed, merged, etc. (whatever is appropriate), despite the fact that there are no changes to Chromium, so that the rewards panel will review this. I am not sure if some additional labels are needed.

Comment 35 by awhalley@chromium.org on Mon, Aug 28, 2017, 4:09 PM UTC Project Member

Labels: -reward-topanel reward-unpaid reward-2000

*** 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.
****

Comment 36 by awhalley@chromium.org on Mon, Aug 28, 2017, 4:24 PM UTC Project Member

Congratulations! The Chrome VRP panel decided to award $2,000 for this report. A member of our finance team will be in touch to arrange payment.

Comment 37 by awhalley@chromium.org on Tue, Aug 29, 2017, 2:21 PM UTC Project Member

Labels: -reward-unpaid reward-inprocess

Comment 38 by pault...@gmail.com on Sat, Nov 4, 2017, 11:16 PM UTC

Thanks! We received the reward! Since the fix was actually released some time ago (I think probably around April 2017), is it possible to make this Chromium issue/report public so we can refer to it?

Comment 39 by sheriffbot@chromium.org on Sat, Nov 18, 2017, 2:05 PM UTC 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