☆ Starred by 3 users

Owner:	wolfi@chromium.org
CC:	yangguo@chromium.org rdevl...@chromium.org caseq@chromium.org sigurds@chromium.org solomonkinard@chromium.org tjudkins@chromium.org dsv@google.com
Status:	Fixed *(Closed)*
Components:	Platform>DevTools Platform>Extensions
Modified:	Feb 15, 2022
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Windows, Chrome, Mac
Pri:	2
Type:	Bug-Security

reward-10000
Security_Impact-Stable
Security_Severity-Medium
allpublic
reward-inprocess
CVE_description-submitted
external_security_report
merge-merged-4430
merge-merged-90
LTS-Merged-90
LTS-Security-90
Release-0-M92
LTS-Size-Normal
CVE-2021-30576
LTS-Complexity-Trivial

Issue 1194896: Security: UAF after moving tab associated with undocked devtools instance into another browser window

Reported by derce...@gmail.com on Thu, Apr 1, 2021, 3:57 AM EDT

Code

VULNERABILITY DETAILS
Typically, a devtools instance can be shown within the tab being debugged, or in an undocked browser window. However, by using the chrome.tabs/chrome.windows API, an extension can move the tab associated with an undocked devtools instance into another window. This then leads to a UAF in the browser process when performing an operation that attempts to access the (now destroyed) window that was hosting the devtools.

VERSION
Chrome Version: Tested on 89.0.4389.114 (stable) and 91.0.4464.5 (canary)
Operating System: Windows 10, version 20H2

REPRODUCTION CASE
There are two different ways an extension could trigger a UAF in the way described above:

- An extension with no additional permissions could do so, provided the user opens an undocked devtools instance.
- An extension could trigger the UAF without any further interaction post-install, provided it has the debugger permission. This is the approach demonstrated here.

1. Install the attached extension.
2. Once installed, the extension will open a new popup window, attach the debugger to it and use Input.dispatchKeyEvent to open the devtools. Because the window is a popup window, the devtools will open in an undocked position.
3. The extension will then determine the tab ID assigned to the devtools window using chrome.debugger.getTargets.
4. It will then move the tab to a new window:

```
chrome.windows.create({tabId: devtoolsTabId});
```

5. Finally, it will crash the devtools tab using:

```
chrome.tabs.update(devtoolsTabId, {url: "chrome://checkcrash"});
```

This will trigger a use-after-free in the browser process. You can verify that by installing the extension in an asan build.

CREDIT INFORMATION
Reporter credit: David Erceg

asan_output_864975.txt
16.4 KB View Download

background.js
2.0 KB View Download

manifest.json
201 bytes View Download

Comment 1 by sheriffbot on Thu, Apr 1, 2021, 3:58 AM EDT Project Member

Labels: external_security_report

Comment 2 by derce...@gmail.com on Thu, Apr 1, 2021, 4:02 AM EDT

The demonstration above results in a UAF, since the devtools browser window is destroyed when its only tab (hosting the devtools instance) is moved to a new window, yet the devtools instance continues to refer to it.

The Browser instance is created at:

https://source.chromium.org/chromium/chromium/src/+/master:chrome/browser/devtools/devtools_window.cc;l=1632;drc=a818b8401e99f19526af0d593510fa92fefb83f0

When the renderer process crashes, the DevToolsWindow instance attempts to access the stored Browser instance:

https://source.chromium.org/chromium/chromium/src/+/master:chrome/browser/devtools/devtools_window.cc;l=1538;drc=a818b8401e99f19526af0d593510fa92fefb83f0

However, because the devtools browser window was destroyed when the tab was moved, this results in a use-after-free.

The details of this specific use-after-free aren't too important though, as the main issue appears to be that the chrome.tabs and chrome.windows APIs allow the tab associated with an undocked devtools instance to be moved to another window.

Comment 3 by derce...@gmail.com on Thu, Apr 1, 2021, 4:22 AM EDT

In terms of how an extension with no additional permissions could trigger this use-after-free, the basic procedure would be:

1. The extension could make the following call periodically to determine whether there are any devtools windows open:

chrome.windows.getAll({windowTypes: ["devtools"]}, function (windows) {...});

2. If the user opens an undocked devtools instance, the above call would return a result.

If you set the populate parameter in chrome.windows.getAll to true, information about the devtools tab will be returned, though the tab ID returned will be -1. The tab ID returned by chrome.tabs.getAllInWindow will also be -1.

However, I don't think that's a problem, as the ID of the devtools tab is likely going to be the window ID + 1 (if the devtools is opened in an undocked position).

3. Using the inferred tab ID, the extension could then move it to a new window and crash the devtools renderer, triggering the UAF.

Comment 4 by drubery@chromium.org on Thu, Apr 1, 2021, 8:01 PM EDT Project Member

Status: Assigned (was: Unconfirmed)
Owner: yangguo@chromium.org
Cc: benwells@chromium.org
Labels: Security_Severity-Low Security_Impact-Stable OS-Chrome OS-Linux OS-Mac OS-Windows
Components: Platform>Extensions Platform>DevTools

I'm not sure if the root cause is in DevTools or Extensions code, so adding an owner for each.

Comment 5 by derce...@gmail.com on Fri, Apr 2, 2021, 12:21 AM EDT

Would it be possible to have the severity here re-evaluated? The reasoning would be the same as that given in ~~https://crbug.com/1190550#c8~~.

Also, ~~issue 1188889~~ deals with another browser UAF triggered by an extension with the debugger permission and that was recently marked as high severity (see ~~https://crbug.com/1188889#c16~~).

Comment 6 by yangguo@chromium.org on Fri, Apr 2, 2021, 4:38 AM EDT Project Member

Owner: wolfi@chromium.org
Cc: -benwells@chromium.org yangguo@chromium.org

Wolfgang, could this be similar to the UAF that you recently investigated?

Comment 7 by sheriffbot on Fri, Apr 2, 2021, 1:43 PM EDT Project Member

Labels: -Pri-3 Pri-2

Setting Pri-2 to match security severity Low. 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 8 by wolfi@chromium.org on Tue, Apr 6, 2021, 6:21 AM EDT Project Member

I was able to reproduce this with the provided extension.

Yang, I don't think this is related to my recent UAF bug. The problems there were caused by (off-the-record) profiles, which do not play a role here.

IMO this problem starts with the fact that the extension API is allowed to move a detached DevTools window into another new window (see screenshot). I don't think that this is possible without using the API and it possibly should not be allowed when using the API either.

Screenshot 2021-04-06 at 12.13.00.png
325 KB View Download

Comment 9 by wolfi@chromium.org on Wed, Apr 7, 2021, 8:15 AM EDT Project Member

Cc: sigurds@chromium.org

Comment 10 by sigurds@chromium.org on Wed, Apr 7, 2021, 8:17 AM EDT Project Member

Cc: caseq@chromium.org

Comment 11 by wolfi@chromium.org on Wed, Apr 7, 2021, 10:02 AM EDT Project Member

Cc: rdevl...@chromium.org

Devlin, would it make sense to disallow `chrome.windows.create()` from moving a detached DevTools tab into a new window? If yes, would `ExtensionFunction::ResponseAction WindowsCreateFunction::Run()` (https://source.chromium.org/chromium/chromium/src/+/master:chrome/browser/extensions/api/tabs/tabs_api.cc;l=530?q=tabs_api.cc) be a good place to start looking?

Thanks!

Comment 12 by rdevl...@chromium.org on Tue, Apr 20, 2021, 11:23 AM EDT Project Member

Hey wolfi@! Thanks for looking into this.

Yep, I think we should disallow moving a devtools window into another window - as you mentioned, I don't think this is something that can be done without the API, and don't think it's really supported by the browser. Unfortunately, just updating windows.create() probably isn't sufficient - I think we'll also need to change tabs.update [1] and possible also tabs.group [2]. For each of these, modifying the function implementation (from Run() down the callstack) is the right place to start - you can read up a bit more on how ExtensionFunctions work here [3].

More broadly, I think it might make sense to also add some CHECKs (or gracefully handle) operations that are called on devtools windows - I wouldn't be surprised if there's other cases where callers try to perform operations that are possible on most Browsers, but fail on a devtools window.

[1] https://developer.chrome.com/docs/extensions/reference/tabs/#method-update
[2] https://developer.chrome.com/docs/extensions/reference/tabs/#method-group
[3] https://chromium.googlesource.com/chromium/src/+/HEAD/extensions/docs/api_functions.md

Comment 13 by wolfi@chromium.org on Wed, Apr 21, 2021, 10:16 AM EDT Project Member

Thanks a lot for the pointers! I will take a closer look soon.

Comment 14 by Git Watcher on Tue, May 18, 2021, 7:02 AM EDT Project Member

Status: Fixed (was: Assigned)

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3

commit 4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3
Author: Wolfgang Beyer <wolfi@chromium.org>
Date: Tue May 18 11:01:35 2021

Restrict extensions API from modifying DevTools window

This CL disallows certain methods of the extensions API from
modifying DevTools windows because they would fail or cause unwanted
results when applied to a DevTools window.

The methods covered are `windows.create()`, `tabs.update()`,
`tabs.move()`, `tabs.group()` and `tabs.discard()`.

~~Fixed: 1194896~~
Change-Id: I11d334a6d844bf81e946e5105ea9e2e504017d0b
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2896966
Commit-Queue: Wolfgang Beyer <wolfi@chromium.org>
Reviewed-by: Devlin <rdevlin.cronin@chromium.org>
Cr-Commit-Position: refs/heads/master@{#883881}

[modify] https://crrev.com/4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3/chrome/browser/extensions/api/tabs/tabs_api.cc
[modify] https://crrev.com/4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3/chrome/browser/extensions/api/tabs/tabs_constants.cc
[modify] https://crrev.com/4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3/chrome/browser/extensions/api/tabs/tabs_constants.h
[modify] https://crrev.com/4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3/chrome/browser/extensions/api/tabs/tabs_test.cc

Comment 15 by sheriffbot on Tue, May 18, 2021, 12:43 PM EDT Project Member

Labels: reward-topanel

Comment 16 by sheriffbot on Tue, May 18, 2021, 2:03 PM EDT Project Member

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

Comment 17 by amyressler@google.com on Wed, May 26, 2021, 4:37 PM EDT Project Member

Labels: -Security_Severity-Low Security_Severity-Medium

raising to at least a Medium severity as while a malicious extension and user gesture is required to trigger, it is a UAF in the browser process

Comment 18 by amyressler@google.com on Thu, Jun 10, 2021, 12:32 PM EDT Project Member

Labels: -reward-topanel reward-unpaid reward-10000

*** 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 19 by amyressler@chromium.org on Thu, Jun 10, 2021, 1:02 PM EDT Project Member

And another one. The VRP Panel has decided to award you $10,000 for this report. Nice work, David!

Comment 20 by amyressler@google.com on Mon, Jun 14, 2021, 11:14 AM EDT Project Member

Labels: -reward-unpaid reward-inprocess

Comment 21 by amyressler@chromium.org on Mon, Jul 19, 2021, 4:23 PM EDT Project Member

Labels: Release-0-M92

Comment 22 by amyressler@google.com on Mon, Jul 19, 2021, 7:17 PM EDT Project Member

Labels: CVE-2021-30576 CVE_description-missing

Comment 23 by amyressler@google.com on Tue, Aug 3, 2021, 3:42 PM EDT Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 24 by sheriffbot on Wed, Aug 25, 2021, 1:30 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 25 by janag...@google.com on Wed, Sep 8, 2021, 5:06 AM EDT Project Member

Labels: LTS-Security-90 LTS-Merge-Request-90 LTS-Size-Normal

Comment 26 by janag...@google.com on Wed, Sep 8, 2021, 6:55 AM EDT Project Member

Labels: LTS-Complexity-Trivial

Comment 27 by marinakz@google.com on Wed, Sep 8, 2021, 12:18 PM EDT Project Member

Labels: -LTS-Merge-Request-90 LTS-Merge-Approved-90

Comment 28 by Git Watcher on Wed, Sep 8, 2021, 1:13 PM EDT Project Member

Labels: merge-merged-4430 merge-merged-90

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/b6094535e0a2b5337ece1338151bedb31406031d

commit b6094535e0a2b5337ece1338151bedb31406031d
Author: Wolfgang Beyer <wolfi@chromium.org>
Date: Wed Sep 08 17:12:48 2021

[M90-LTS] Restrict extensions API from modifying DevTools window

This CL disallows certain methods of the extensions API from
modifying DevTools windows because they would fail or cause unwanted
results when applied to a DevTools window.

The methods covered are `windows.create()`, `tabs.update()`,
`tabs.move()`, `tabs.group()` and `tabs.discard()`.

(cherry picked from commit 4ffa92821f0870bdcd79b2ef279bc3b89f9d50a3)

~~Fixed: 1194896~~
Change-Id: I11d334a6d844bf81e946e5105ea9e2e504017d0b
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2896966
Commit-Queue: Wolfgang Beyer <wolfi@chromium.org>
Reviewed-by: Devlin <rdevlin.cronin@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#883881}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/3147335
Reviewed-by: Artem Sumaneev <asumaneev@google.com>
Owners-Override: Artem Sumaneev <asumaneev@google.com>
Commit-Queue: Jana Grill <janagrill@google.com>
Cr-Commit-Position: refs/branch-heads/4430@{#1580}
Cr-Branched-From: e5ce7dc4f7518237b3d9bb93cccca35d25216cbe-refs/heads/master@{#857950}

[modify] https://crrev.com/b6094535e0a2b5337ece1338151bedb31406031d/chrome/browser/extensions/api/tabs/tabs_api.cc
[modify] https://crrev.com/b6094535e0a2b5337ece1338151bedb31406031d/chrome/browser/extensions/api/tabs/tabs_constants.cc
[modify] https://crrev.com/b6094535e0a2b5337ece1338151bedb31406031d/chrome/browser/extensions/api/tabs/tabs_constants.h
[modify] https://crrev.com/b6094535e0a2b5337ece1338151bedb31406031d/chrome/browser/extensions/api/tabs/tabs_test.cc

Comment 29 by janag...@google.com on Wed, Sep 8, 2021, 1:14 PM EDT Project Member

Labels: -LTS-Merge-Approved-90 LTS-Merged-90

Comment 30 by dsv@chromium.org on Tue, Feb 15, 2022, 9:03 AM EST Project Member

Labels: Hotlist-DevTools-BrowserAutomation-Backlog

Comment 31 by dsv@chromium.org on Tue, Feb 15, 2022, 9:12 AM EST Project Member

Labels: -Hotlist-DevTools-BrowserAutomation-Backlog