



chromium ▾

New issue

Open issues ▾

🔍 Search chromium issue ▾ ⚙️

Sign in

☆ Starred by 3 users

Owner:

kenrb@chromium.org


CC:

rdevl...@chromium.org

nsatr...@chromium.org

kenrb@chromium.org

martinkr@google.com

 agl@chromium.org

Status:

Fixed (*Closed*)

Components:

[Blink>WebAuthentication](#)

Modified:

Jul 28, 2022

Backlog-Rank:

Editors:

EstimatedDays:

NextAction:

OS:

[Linux](#), [Windows](#), [Chrome](#), [Mac](#)

Pri:

2

Type:

[Bug-Security](#)

[Security_Severity-Low](#)

[Security_Impact-Stable](#)

[allpublic](#)

[CVE_description-submitted](#)

[Target-80](#)

[Target-88](#)

[Target-81](#)

[Target-84](#)

[Target-83](#)

[Target-85](#)

[Target-86](#)

[Target-87](#)

[Target-89](#)

[Target-90](#)

[Target-91](#)

[Target-92](#)

[Target-94](#)

[Target-93](#)

[M-98](#)

[Target-96](#)

[Target-98](#)

Issue 1000408: getOriginFromUrl in cryptotoken component extension doesn't use real origin

Reported by [jun.k...@microsoft.com](#) on Tue, Sep 3, 2019, 3:30 PM EDT

Project Member

[↔](#) Code

Description #3 by [jun.k...@microsoft.com](#) (Sep 5, 2019) ▼

Any HTTPS site can send message to cryptotoken extension.

<https://cs.chromium.org/chromium/src/chrome/browser/resources/cryptotoken/manifest.json?l=16>

```
-----
"externally_connectable": {
  "matches": [
    "https:///*/*"
  ],
  "ids": [
    "fjajfjhkeibgmiggdfhjplbhmfkialk"
  ]
}
-----
```

Following `getOriginFromUrl` is used in multiple places.

<https://cs.chromium.org/chromium/src/chrome/browser/resources/cryptotoken/webrequest.js?q=getOriginFromUrl&l=22>

```
-----
function getOriginFromUrl(url) {
  var re = new RegExp("^(https?://)[^/*/?]");
  var originarray = re.exec(url);
  if (originarray == null) {
    return originarray;
  }
  var origin = originarray[0];
  while (origin.charAt(origin.length - 1) == '/') {
    origin = origin.substring(0, origin.length - 1);
  }
  if (origin == 'http:' || origin == 'https:') {
    return null;
  }
  return origin;
}
-----
```

One of them is to get the origin of MessageSender from URL.

```
-----
function createSenderFromMessageSender(messageSender) {
  var origin = getOriginFromUrl(/** @type {string} */ (messageSender.url));
  if (!origin) {
    return null;
  }
  var sender = {origin: origin};
  if (messageSender.tab) {
    sender.tabId = messageSender.tab.id;
  }
  return sender;
}
-----
```

}

I haven't tested the API, but by looking at the code, it seems like this function is problematic in 2 ways.

1. It'll not work on blob: or about: URLs
2. In case the website hosts untrusted content using CSP sandbox, this function would get that origin wrong by taking origin from URL (because CSP sandbox would make origin of document as null).

[Comment 1](#) by [jun.k...@microsoft.com](#) on Tue, Sep 3, 2019, 3:31 PM EDT Project Member

Description was changed.

[Comment 2](#) by [lukasza@chromium.org](#) on Tue, Sep 3, 2019, 3:42 PM EDT Project Member

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

Components: Blink>WebAuthentication

[kenrb@](#), I wonder if you could please help with further triage (as one of //device/fido/OWNERS and as one of security sheriffs)?

//docs/security/origin-vs-url.md does indeed point out that converting a URL into an Origin is wrong (see the "Avoid converting URLs to origins").

I am also surprised that the code in question uses regular expressions to parse a URL - this seems fragile.

[Comment 3](#) by [sheriffbot@chromium.org](#) on Wed, Sep 4, 2019, 11:07 AM EDT Project Member

Status: Assigned (was: Unconfirmed)

[Comment 4](#) by [jun.k...@microsoft.com](#) on Thu, Sep 5, 2019, 4:55 PM EDT Project Member

Description was changed.

[Comment 5](#) by [kenrb@chromium.org](#) on Fri, Sep 6, 2019, 3:57 PM EDT Project Member

Labels: -Restrict-View-SecurityTeam Restrict-View-Google OS-Chrome OS-Linux OS-Mac OS-Windows Pri-2 Type-Bug

After looking through the code and talking with [martinkr@](#) and [agl@](#), this doesn't seem to have any security implications. In particular, all the corner case URLs would cause the function to return null and fail closed. Removing security flags accordingly, but leaving a Restrict-View on for now in case something else comes up later.

I agree this is problematic, and I'd be happy to change it if [issue-987014](#) is addressed and we have an authoritative way to obtain the origin.

[Comment 6](#) by [jun.k...@microsoft.com](#) on Fri, Sep 6, 2019, 4:02 PM EDT Project Member

Re: [Comment 5](#)

>all the corner case URLs would cause the function to return null and fail closed.

Could you tell me how will you handle calls from "<https://www.dropbox.com/>" VS "<https://www.dropbox.com/enterprise/>"?

Those have 2 different origins, and I'm pretty sure this function will see it as same-origin, thus this is a security bug.

[Comment 7](#) by [agl@chromium.org](#) on Fri, Sep 6, 2019, 4:06 PM EDT Project Member

> Those have 2 different origins

In the content of U2F, I believe they have the same origin? (I.e. <https://www.dropbox.com>)

[Comment 8](#) by [jun.k...@microsoft.com](#) on Fri, Sep 6, 2019, 4:10 PM EDT Project Member

You need to check window.origin. "<https://www.dropbox.com/enterprise>" has CSP sandbox. This is what I told in problem 2. >2. In case the website hosts untrusted content using CSP sandbox, this function would get that origin wrong by taking origin from URL (because CSP sandbox would make origin of document as null).

But you guys didn't get it.

[Comment 9](#) by [kenrb@chromium.org](#) on Fri, Sep 6, 2019, 5:10 PM EDT Project Member

Labels: -Restrict-View-Google Restrict-View-SecurityTeam Security_Severity-Low Security_Impact-Stable M-79 Type-Bug-Security

That's true, it does mean the sandboxed page could potentially try to exercise keys that were registered to an unsandboxed page served on the same host, and that does break a security boundary.

[Comment 10](#) by [agl@chromium.org](#) on Fri, Sep 6, 2019, 5:16 PM EDT Project Member

> But you guys didn't get it.

Communication via bug tracker is challenging enough without deliberate obfuscation.

But, as I understand it, you're noting that the CSP sandbox directive is mismatched with the extension framework because it broke the link between the origin and URL of a document. That seems like a reasonable point, but not specific to U2F. (I would imagine that there'll be an nearly endless supply of issues arising from trying to break that link. Good luck to whomever came up with that CSP idea.)

If the extensions team decide to expose an origin, we should update the U2F extension to use it. It looks like this is already blocking on an issue for that. I don't especially feel that this issue needs to be restricted viewing in the mean-time, but have no strong feelings either way.

[Comment 11](#) by [martinkr@google.com](#) on Fri, Sep 6, 2019, 5:35 PM EDT Project Member

Cc: rdevl...@chromium.org

[Comment 12](#) by [sheriffbot@chromium.org](#) on Wed, Feb 5, 2020, 10:48 AM EST Project Member

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

[Comment 13](#) by [sheriffbot](#) on Thu, Apr 9, 2020, 12:29 PM EDT Project Member

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

[Comment 14](#) by [sheriffbot](#) on Wed, May 20, 2020, 1:30 PM EDT Project Member

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

[Comment 15](#) by [jun.k...@microsoft.com](#) on Fri, Jul 10, 2020, 12:52 AM EDT Project Member

Extension API now exposes origin information when receiving messages.

[Comment 16](#) by [sheriffbot](#) on Thu, Jul 16, 2020, 1:33 PM EDT Project Member

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

[Comment 17](#) by [sheriffbot](#) on Wed, Aug 26, 2020, 1:40 PM EDT Project Member

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

[Comment 18](#) by [sheriffbot](#) on Wed, Oct 7, 2020, 1:40 PM EDT Project Member

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

[Comment 19](#) by [sheriffbot](#) on Wed, Nov 18, 2020, 12:25 PM EST Project Member

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

[Comment 20](#) by [sheriffbot](#) on Wed, Jan 20, 2021, 12:24 PM EST Project Member

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

[Comment 21](#) by [sheriffbot](#) on Wed, Mar 3, 2021, 12:24 PM EST Project Member

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

[Comment 22](#) by [sheriffbot](#) on Thu, Apr 15, 2021, 12:25 PM EDT Project Member

Labels: -M-89 M-90 Target-90

[Comment 23](#) by [sheriffbot](#) on Wed, May 26, 2021, 12:25 PM EDT Project Member

Labels: -M-90 M-91 Target-91

[Comment 24](#) by [sheriffbot](#) on Sat, Aug 7, 2021, 12:24 PM EDT Project Member

Labels: -M-91 Target-92 M-92

[Comment 25](#) by [sheriffbot](#) on Sat, Sep 11, 2021, 12:25 PM EDT Project Member

Labels: -M-92 M-93 Target-93

[Comment 26](#) by [sheriffbot](#) on Wed, Sep 22, 2021, 12:25 PM EDT Project Member

Labels: -M-93 Target-94 M-94

[Comment 27](#) by [sheriffbot](#) on Mon, Nov 15, 2021, 12:25 PM EST Project Member

Labels: -M-94 Target-96 M-96

[Comment 28](#) by [sheriffbot](#) on Wed, Feb 2, 2022, 12:25 PM EST Project Member

Labels: -M-96 M-98 Target-98

[Comment 29](#) by [Git Watcher](#) on Wed, Feb 23, 2022, 11:06 PM EST Project Member

Status: Fixed (was: Assigned)

The following revision refers to this bug:

<https://chromium.googlesource.com/chromium/src/+3377c1be19f197323e1f436f4e1380a72ee35eb7>

commit [3377c1be19f197323e1f436f4e1380a72ee35eb7](#)

Author: Ken Buchanan <kenrb@chromium.org>

Date: Thu Feb 24 04:05:50 2022

[U2F] Cryptotoken extension to use origin rather than URL for apId

Currently when a caller invokes U2F through the Cryptotoken extension it validates the origin of the calling page by parsing the URL. That is wrong in cases where the origin differs from the URL.

This change uses the origin field of messageSender directly instead of parsing the URL.

~~Fixed: 1000408~~

Change-Id: Ia45f811b20628ef2d94b498cd97f507b9ad1fd9b

Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+3484604>

Reviewed-by: Martin Kreichgauer <martinkr@google.com>

Commit-Queue: Ken Buchanan <kenrb@chromium.org>

Cr-Commit-Position: refs/heads/main@{#974483}

[add] <https://crrev.com/3377c1be19f197323e1f436f4e1380a72ee35eb7/chrome/test/data/cryptotoken/csp-sandbox.html.mock-http-headers>

[modify]

<https://crrev.com/3377c1be19f197323e1f436f4e1380a72ee35eb7/chrome/browser/resources/cryptotoken/webrequestsender.js>

[modify]

https://crrev.com/3377c1be19f197323e1f436f4e1380a72ee35eb7/chrome/browser/extensions/api/cryptotoken_private/cryptotoken_private_browsertest.cc

[add] <https://crrev.com/3377c1be19f197323e1f436f4e1380a72ee35eb7/chrome/test/data/cryptotoken/csp-sandbox.html>

Comment 30 by [sheriffbot](#) on Thu, Feb 24, 2022, 1:42 PM EST Project Member

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

Comment 31 by amyressler@chromium.org on Mon, Apr 25, 2022, 9:03 PM EDT Project Member

Labels: Release-0-M101

Comment 32 by amyressler@google.com on Tue, Apr 26, 2022, 4:32 PM EDT Project Member

Labels: CVE-2022-1499 CVE_description-missing

Comment 33 by [sheriffbot](#) on Thu, Jun 2, 2022, 1:33 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 34 by amyressler@google.com on Tue, Jul 26, 2022, 5:37 PM EDT Project Member

Labels: CVE_description-submitted -CVE_description-missing

[Comment 35](#) by amyressler@chromium.org on Thu, Jul 28, 2022, 5:47 PM EDT Project Member

Labels: -CVE_description-missing --CVE_description-missing

[About Monorail](#)

[User Guide](#)

[Release Notes](#)

[Feedback on Monorail](#)

[Terms](#)

[Privacy](#)