☆ Starred by 3 users

Owner:	japhet@chromium.org
CC:	mkwst@chromium.org janag...@google.com pmeuleman@chromium.org arthu...@chromium.org antoniosartori@chromium.org
Status:	Fixed *(Closed)*
Components:	Blink>SecurityFeature>IFrameSandbox
Modified:	23 days ago
Backlog-Rank:	----
Editors:	----
EstimatedDays:	----
NextAction:	----
OS:	Linux, Android, Windows, Chrome, Mac
Pri:	1
Type:	Bug-Security

Hotlist-Merge-Review
reward-3000
Security_Impact-Stable
Security_Severity-Medium
allpublic
reward-inprocess
CVE_description-submitted
Target-88
M-90
Target-87
Target-89
Target-90
Merge-Rejected-90
merge-merged-4240
LTR-Merged-86
LTS-Security-86
external_security_report
merge-merged-4430
merge-merged-90
external_security_bug
LTS-Merged-90
LTS-Security-90
Release-0-M91
CVE-2021-30534

Issue 1151507: Security: Cross-origin iframe can navigate top window to different site via same-site open redirect or XSS redirect
Reported by alesa...@alesandroortiz.com on Fri, Nov 20, 2020, 5:57 PM EST

Code

VULNERABILITY DETAILS
A non-sandboxed cross-origin iframe can navigate the top window to a different origin by abusing an open redirect or XSS vulnerability in a URL which is same-site with the top window's origin.

At first glance, this seems mostly intentional based on discussion in ~~issue 640057~~, https://github.com/WICG/interventions/issues/16 and https://www.chromestatus.com/feature/5851021045661696. There's also issue 624061 which seems related but is private.

The same-origin bypass seems intentional. The same-site bypass might have been implemented to address compatibility issues/concerns.

However, I haven't identified any public discussion about bypasses via open redirects or XSS redirects. If the following scenarios were taken into consideration or discussed somewhere, then feel free to disregard this report.

Scenarios:
1. Same-origin: iframe loaded in https://example.com bypasses via top navigation to https://example.com/redirect?url=https://attacker.com or https://example.com/?xss=location.href%3Dhttps%3A%2F%2Fattacker.com

2. Same-site: iframe loaded in https://example.com bypasses via top navigation to https://subdomain.example.com/redirect?url=https://attacker.com or https://example.com/?xss=location.href%3Dhttps%3A%2F%2Fattacker.com

3. Same-origin/same-site, different scheme: iframe loaded in https://example.com bypasses via top navigation to plaintext http://example.com/attacker-page (note HTTP in destination URL)

The same-site behavior is notable, since same-site bypasses are much more likely due to increased surface area: an open redirect or XSS in *any* subdomain is sufficient to bypass.

The diff-scheme behavior is also notable, since an attacker who has PITM/MITM capabilities can redirect to an HTTP URL on any site without HSTS and then redirect to the attacker URL. No existing open redirect or XSS vulnerability is needed, though the PITM requirement is a significantly higher bar.

This does not affect sandboxed iframes, since they either require user interaction for top navigations (allow-top-navigation-by-user-activation) or intentionally allow all top navigations (allow-top-navigation).

Potential solutions:
1. Add same-scheme or secure-scheme limitation for destination URLs.
2. Allow only same-origin navigations (remove same-site exception).
3. Remove no-interaction top-navigation from iframes unless sandbox="allow-top-navigation" is set.
4. Somehow determine if destination page will redirect to different site. This is probably not feasible, since redirect can be initiated via meta tags or scripts on a delay.

Relevant commit: https://source.chromium.org/chromium/chromium/src/+/3eef8b926bd46f329e372fb674dd6f2d5ad0844d

VERSION
Chrome Version: 87.0.4280.66 (Official Build) (64-bit) (cohort: Stable)
Operating System: Windows 10 OS Version 2004 (Build 19041.630)

REPRODUCTION CASE

Scenario 1 repro:
1. Navigate to https://alesandroortiz.com/security/chromium/nav-top-no-interaction.html

Expected behavior:
iframe cannot navigate top window

Observed behavior:
iframe can navigate top window

iframe URL: https://aogarantiza.com/chromium/nav-top-no-interaction-frame.html

To repro the other scenarios, modify the iframe source code (the other scenarios are commented out).

CREDIT INFORMATION
Reporter credit: Alesandro Ortiz <https://AlesandroOrtiz.com>

nav-top-no-interaction.html
716 bytes View Download

nav-top-no-interaction-frame.html
1.1 KB View Download

Comment 1 by sheriffbot on Fri, Nov 20, 2020, 6:01 PM EST Project Member
Labels: reward-potential

Comment 2 by mbarb...@chromium.org on Mon, Nov 23, 2020, 2:13 PM EST Project Member
Status: Assigned (was: Unconfirmed)
Owner: mkwst@chromium.org
Labels: Security_Severity-Medium Security_Impact-Stable OS-Android OS-Chrome OS-Linux OS-Mac OS-Windows
Components: Blink>SecurityFeature>IFrameSandbox

Comment 3 by sheriffbot on Tue, Nov 24, 2020, 1:04 PM EST Project Member
Labels: M-87 Target-87

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 Tue, Nov 24, 2020, 1:41 PM EST Project Member
Labels: -Pri-3 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 sheriffbot on Sat, Dec 5, 2020, 12:21 PM EST Project Member
mkwst: 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 6 by antoniosartori@google.com on Mon, Dec 7, 2020, 9:20 AM EST Project Member
Status: Untriaged (was: Assigned)
Owner: ----
Labels: Needs-Feedback

I am not sure I understand correctly. A non-sandboxed cross-origin iframe can navigate the main page. For example, this is allowed:

(on https://a.com)
<iframe src="https://b.com"></iframe>

(on https://b.com)
<script>
window.top.location = "https://c.com";
<script>

You don't need to exploit any redirect. Did I miss something here?

Comment 7 by alesa...@alesandroortiz.com on Mon, Dec 7, 2020, 1:47 PM EST
The example in #c6 should not work. If it does work on your device, it's due to another bug (e.g. issue 1085982: extensions may inadvertently activate frames).

Baseline PoC: https://alesandroortiz.com/security/chromium/nav-top-baseline.html (iframe: https://aogarantiza.com/chromium/nav-top-baseline-frame.html)

The redirect will be blocked unless the iframe has had user activation at least once since page load (code [2] checks sticky user activation).

For historical context, the same-origin allowed behavior was added in this commit to implement ~~issue 640057~~:
https://source.chromium.org/chromium/chromium/src/+/3eef8b926bd46f329e372fb674dd6f2d5ad0844d

The current code is here, annotated below:
https://source.chromium.org/chromium/chromium/src/+/master:third_party/blink/renderer/core/frame/local_frame.cc;l=1695;drc=c58b1362e545b075450790b1f3aada2e3952fcea

Scenario 1 PoC at https://alesandroortiz.com/security/chromium/nav-top-no-interaction.html hits [3].
Scenario 2 PoC at https://c2.alesandroortiz.com/security/chromium/nav-top-no-interaction.html hits [6] (same PoC, diff origin)
Scenario 3 would also hit [6]. (No hosted PoC currently available, but I have verified in my environment.)

```
  if (target_frame == Tree().Top()) {                                          <-- [1] True when iframe navigates top frame
    // A frame navigating its top may blocked if the document initiating
    // the navigation has never received a user gesture and the navigation
    // isn't same-origin with the target.
    if (HasStickyUserActivation() ||                                           <-- [2] Another bug + sticky activation check may be causing #c6 behavior
        target_frame.GetSecurityContext()->GetSecurityOrigin()->CanAccess(
            SecurityOrigin::Create(destination_url).get())) {            <-- [3] CanAccess() returns true for Scenario 1 (same-origin scenario, target_frame and destination_url origins
are both https://alesandroortiz.com)
      return true;
    }

    String target_domain = network_utils::GetDomainAndRegistry(
        target_frame.GetSecurityContext()->GetSecurityOrigin()->Domain(),   <-- [4] Domain() returns alesandroortiz.com for origin https://c2.alesandroortiz.com (subdomain)
or http://alesandroortiz.com (HTTP scheme)
        network_utils::kIncludePrivateRegistries);
    String destination_domain = network_utils::GetDomainAndRegistry(
        destination_url.Host(), network_utils::kIncludePrivateRegistries);        <-- [5] GetDomainAndRegistry() returns alesandroortiz.com for origin https://alesandroortiz.com
    if (!target_domain.IsEmpty() && !destination_domain.IsEmpty() &&
        target_domain == destination_domain) {                              <-- [6] target_domain == destination_domain returns true for Scenarios 2 and 3 (same-site scenario or diff-
scheme scenario, target_domain and destination_domain are alesandroortiz.com)
      return true;
    }
    if (auto* settings_client = Client()->GetContentSettingsClient()) {
      if (settings_client->AllowPopupsAndRedirects(false /* default_value*/))
        return true;
    }
    PrintNavigationErrorMessage(
        target_frame,
        "The frame attempting navigation is targeting its top-level window, "
        "but is neither same-origin with its target nor has it received a "
        "user gesture. See "
        "https://www.chromestatus.com/features/5851021045661696.");
    GetLocalFrameHostRemote().DidBlockNavigation(
        destination_url, GetDocument()->Url(),
        mojom::NavigationBlockedReason::kRedirectWithNoUserGesture);             <-- [7] Navigation is blocked (shows omnibox icon on desktop, infobar on Android)
  } else { ... }
```

Comment 8 by alesa...@alesandroortiz.com on Mon, Dec 7, 2020, 3:15 PM EST

Minor corrections for #c7:
[4] should read "GetDomainAndRegistry() returns..."
[7] The navigation is actually blocked by caller when they get `return false` at the end of LocalFrame::CanNavigate() (DidBlockNavigation() [7] triggers the UI).

Comment 9 by wfh@chromium.org on Wed, Dec 9, 2020, 4:26 PM EST Project Member

Labels: -Needs-Feedback

Comment 10 by wfh@chromium.org on Wed, Dec 9, 2020, 4:27 PM EST Project Member

Status: Assigned (was: Untriaged)
Owner: antoniosartori@google.com

based on interaction above, I'm assigning this bug to you, antoniosartori@google.com - can you update with any progress made?

Comment 11 by sheriffbot on Mon, Dec 21, 2020, 12:21 PM EST Project Member

antoniosartori: 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 12 by antoniosartori@google.com on Wed, Dec 23, 2020, 2:08 AM EST Project Member

Owner: antoniosartori@chromium.org

Comment 13 by antoniosartori@chromium.org on Wed, Dec 23, 2020, 2:27 AM EST Project Member

Cc: arthu...@chromium.org mkwst@chromium.org

This does not seem so easy to fix. The code from comment 7 currently checking whether the frame is allowed to navigate or not is in Blink. However, the redirect will be processed in the Browser later in the navigation logic, so we would need to check again there. We might want to move the check in the Browser in order to deduplicate code.

This looks like a bigger task.

Comment 14 by sheriffbot on Wed, Jan 20, 2021, 12:22 PM EST Project Member

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

Comment 15 by adetaylor@google.com on Wed, Jan 20, 2021, 6:57 PM EST Project Member

Labels: -reward-potential external_security_report

Comment 16 by sheriffbot on Mon, Feb 22, 2021, 11:16 AM EST Project Member

This issue hasn't been updated in the last 30 days - please update it or consider lowering its priority.

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 17 by sheriffbot on Wed, Mar 3, 2021, 12:22 PM EST Project Member

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

Comment 18 by antoniosartori@chromium.org on Mon, Mar 8, 2021, 12:05 PM EST Project Member

Status: Available (was: Assigned)
Owner: ----
Cc: antoniosartori@chromium.org

Comment 19 by pmeuleman@chromium.org on Tue, Mar 9, 2021, 6:08 AM EST Project Member

Owner: japhet@chromium.org
Cc: pmeuleman@chromium.org

Confirming the behavior, a cross origin iframe is allowed to navigate its top level frame to a same etld+1 document, without a user gesture. This ignores the scheme. See change's review at https://chromium-review.googlesource.com/c/chromium/src/+/1187326/
I believe this is reasonable from a security perspective, the goal of this intervention being to reduce user annoyance, Embedded iframes must be sandboxed if they are considered untrustworthy. The reflective XSS example is interesting.

Nate: Since you were the author of the change aforementioned, I'd like to have your opinion on that before closing: Was the algorithm made scheme agnostic on purpose?

Do you have references that led to the decision of restricting to eTLD + 1? I suppose that's related to the efforts to mitigate the reports we see in https://github.com/WICG/interventions/issues/16?

Comment 20 by pmeuleman@chromium.org on Tue, Mar 9, 2021, 9:33 AM EST Project Member

Status: Assigned (was: Available)

Comment 21 by japhet@chromium.org on Wed, Mar 10, 2021, 5:50 PM EST Project Member

Yeah, the eTLD+1 restriction was mostly a product of trial and error: we kept having to loosen the intervention because of compatibility requirements for legitimate use cases. I don't know of any specific cases that require scheme-agnosticism, but I'd be surprised if it wouldn't break *something* out there on the web. It might be small enough that it's still worth tightening though?

Comment 22 by sheriffbot on Wed, Mar 10, 2021, 8:04 PM EST Project Member

Labels: reward-potential

Comment 23 by Git Watcher on Wed, Mar 17, 2021, 4:20 PM EDT Project Member

Status: Fixed (was: Assigned)

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/1baf9eba07b806f86a6e60851428c7ab318da093

commit 1baf9eba07b806f86a6e60851428c7ab318da093
Author: Pâris Meuleman <pmeuleman@chromium.org>
Date: Wed Mar 17 20:19:13 2021

Prevent Cross-Origin iframe from navigating top to a different scheme

Cross-origin iframes were prevented to navigate top with [1]. Those
iframes were allowed to navigate top only to same domain (eTLD+1)
following reports of adverse impact. This severely restrains the ability
of said iframe to cause nuisance.
It does not seem necessary however to loosen the constraint to allow
different schemes, especially from https to http. As a result this CL
prevents a cross-origin iframe from navigating top to the same eTLD + 1
with a different schemes if there's no user gesture.

[1] https://github.com/WICG/interventions/issues/16

~~Bug: 1151507~~
~~Fixed: 1151507~~

Change-Id: Ia1568175c044831594154ceea3e3aacb4e2efb2c
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2756509
Commit-Queue: Nate Chapin <japhet@chromium.org>
Auto-Submit: Pâris Meuleman <pmeuleman@chromium.org>
Reviewed-by: Nate Chapin <japhet@chromium.org>
Cr-Commit-Position: refs/heads/master@{#863936}

[modify] https://crrev.com/1baf9eba07b806f86a6e60851428c7ab318da093/third_party/blink/renderer/core/frame/local_frame.cc
[add] https://crrev.com/1baf9eba07b806f86a6e60851428c7ab318da093/third_party/blink/web_tests/http/tests/security/frameNavigation/resources/iframe-that-performs-different-scheme-same-etld-plus-one-top-navigation-without-user-gesture.html
[add] https://crrev.com/1baf9eba07b806f86a6e60851428c7ab318da093/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture-expected.txt
[add] https://crrev.com/1baf9eba07b806f86a6e60851428c7ab318da093/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture.html

Comment 24 by zhangtiff@google.com on Wed, Mar 17, 2021, 7:12 PM EDT Project Member

Labels: -reward-potential external_security_bug

Comment 25 by sheriffbot on Thu, Mar 18, 2021, 12:42 PM EDT Project Member

Labels: reward-topanel

Comment 26 by sheriffbot on Thu, Mar 18, 2021, 1:56 PM EDT Project Member

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

Comment 27 by sheriffbot on Fri, Mar 19, 2021, 2:21 PM EDT Project Member

Labels: Merge-Request-90

Requesting merge to beta M90 because latest trunk commit (863936) appears to be after beta branch point (857950).

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

Comment 28 by sheriffbot on Fri, Mar 19, 2021, 2:23 PM EDT Project Member

Labels: -Merge-Request-90 Merge-Review-90 Hotlist-Merge-Review

This bug requires manual review: M90's targeted beta branch promotion date has already passed, so this requires manual review
Before a merge request will be considered, the following information is required to be added to this bug:

1. Does your merge fit within the Merge Decision Guidelines?
- Chrome: https://chromium.googlesource.com/chromium/src.git/+/master/docs/process/merge_request.md#when-to-request-a-merge
- Chrome OS: https://goto.google.com/cros-release-branch-merge-guidelines
2. Links to the CLs you are requesting to merge.
3. Has the change landed and been verified on ToT?
4. Does this change need to be merged into other active release branches (M-1, M+1)?
5. Why are these changes required in this milestone after branch?
6. Is this a new feature?
7. If it is a new feature, is it behind a flag using finch?

Chrome OS Only:
8. Was the change reviewed and approved by the Eng Prod Representative? See Eng Prod ownership by component: http://go/cros-engprodcomponents

Please contact the milestone owner if you have questions.
Owners: govind@(Android), bindusuvarna@(iOS), cindyb@(ChromeOS), srinivassista@(Desktop)

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

Comment 29 by srinivassista@google.com on Fri, Mar 19, 2021, 6:49 PM EDT Project Member

pls answer comment #28 for review

Comment 30 by japhet@chromium.org on Mon, Mar 22, 2021, 12:30 PM EDT Project Member

Labels: -Merge-Review-90 Merge-Rejected-90

pmeuleman and I just chatted, and we both would recommend against merge here. This issue has been present for quite some time, and the fix has a higher-than-average risk of compatibility issues. Waiting for M91 should be fine.

(Feel free to remove Merge-Rejected-90 label if this is an incorrect usage)

Comment 31 by alesa...@alesandroortiz.com on Mon, Mar 22, 2021, 2:24 PM EDT

To confirm info from #c21 and #c23, the only planned change is to fix Scenario 3 (different scheme)?

Given the strong compatibility concerns and multiple prerequisites for Scenario 1 (same origin), this seems acceptable.

I still have concerns about Scenario 2 (same site/eTLD+1). Large-scale malicious redirect campaigns launched from iframes have been seen in the wild before (e.g. ~~issue 901568~~, forced redirect via sandbox restriction bypass). I understand the prerequisites are difficult and impacts are limited to a single site (eTLD+1), but it's still valuable in high-traffic sites and is easier than Scenario 3.

For Scenario 2, in addition to open redirects or XSS vulnerabilities, subdomain takeovers can also be used to perform top-level navigations. e.g. Attacker takes over subdomain.example.com, and compromised unsandboxed iframe navigates to https://subdomain.example.com which then navigates to a malicious URL (or hosts a malicious page.)

Seems like the decision is that compatibility concerns override security concerns for Scenario 2, but want to triple-confirm this.

Comment 32 by amyressler@google.com on Wed, Mar 24, 2021, 3:57 PM EDT Project Member

Labels: -reward-topanel reward-unpaid reward-3000

*** 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 33 by amyressler@google.com on Wed, Mar 24, 2021, 4:45 PM EDT Project Member

Congratulations, Alesandro! The VRP Panel has awarded you $3000 for this report. Excellent work!

Comment 34 by amyressler@google.com on Mon, Mar 29, 2021, 12:07 PM EDT Project Member

Labels: -reward-unpaid reward-inprocess

Comment 35 by pmeuleman@chromium.org on Mon, Mar 29, 2021, 1:35 PM EDT Project Member

The implementation in c23 covers removes the tolerance for different schemes, i.e. scenario 3:
- navigating top to a same site different scheme (https -> http) url will not be an accepted exception anymore.

I believe we tried to be more restrictive in the past (see pushback on https://github.com/WICG/interventions/issues/16) and settled on this as a best effort.

While scenario 2, i.e. allowing a cross-origin iframe to navigate top same site leads to issues, can be problematic as you highlight I reckon there's quite a few flows, especially around authentication, that break.
IIUC sites that include untrusted iframes must use sandboxes.

japhet@: We could add a behavior (behind a flag) that would forbid scenario 2, and metrics to decide on an activation. But I believe you already did that and the current state was the results of your previous experiments. Do you confirm this and the above?

Comment 36 by Git Watcher on Thu, Apr 8, 2021, 10:45 AM EDT Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/d29062199696de8226720fd8211fe4cf3d36b1de

commit d29062199696de8226720fd8211fe4cf3d36b1de
Author: Pâris MEULEMAN <pmeuleman@chromium.org>
Date: Thu Apr 08 14:44:10 2021

Kill switch for blocking top navigation to different scheme

Add a feature flag acting as a kill switch for the change introduced
in https://chromium-review.googlesource.com/c/chromium/src/+/2756509
This feature flag is enabled by default and can be switched off in
the event the change has impact on legitimate uses.

~~Bug: 1151507~~
Change-Id: Ibe99cff264f9ce3da29e69512e0f4325130a99e5
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2807363
Auto-Submit: Pâris Meuleman <pmeuleman@chromium.org>
Commit-Queue: Pâris Meuleman <pmeuleman@chromium.org>
Reviewed-by: Nate Chapin <japhet@chromium.org>
Cr-Commit-Position: refs/heads/master@{#870512}

[modify] https://crrev.com/d29062199696de8226720fd8211fe4cf3d36b1de/third_party/blink/common/features.cc
[modify] https://crrev.com/d29062199696de8226720fd8211fe4cf3d36b1de/third_party/blink/public/common/features.h
[modify] https://crrev.com/d29062199696de8226720fd8211fe4cf3d36b1de/third_party/blink/renderer/core/frame/local_frame.cc

Comment 37 by alesa...@alesandroortiz.com on Mon, May 3, 2021, 11:18 AM EDT

japhet@: Please see open question re: same-site navigation in #c31 and #c35.

pmeuleman@: Thanks for context in #c35.

If there's data supporting compat issues, then I'm okay leaving as-is. Sandboxing the iframe is currently available as a mitigation for websites.

Comment 38 by japhet@chromium.org on Mon, May 3, 2021, 12:29 PM EDT Project Member

I don't think we explicitly measured the case described in Scenario 2. We tried it and got enough compat breakage reports that we decided it wasn't worth the effort. But that was several years ago, and the landscape might've changed. I'd certainly be open to adding metrics and seeing if we can do something now.

Comment 39 by amyressler@chromium.org on Mon, May 24, 2021, 11:26 AM EDT Project Member

Labels: Release-0-M91

Comment 40 by amyressler@google.com on Mon, May 24, 2021, 2:18 PM EDT Project Member

Labels: CVE-2021-30534 CVE_description-missing

Comment 41 by janag...@google.com on Tue, May 25, 2021, 9:34 AM EDT Project Member

Labels: LTS-Security-86 LTS-Merge-Request-86

Comment 42 by janag...@google.com on Tue, May 25, 2021, 9:35 AM EDT Project Member

Cc: janag...@google.com

Comment 43 by sheriffbot on Tue, May 25, 2021, 12:22 PM EDT Project Member

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

Comment 44 by gianluca@google.com on Wed, May 26, 2021, 11:51 AM EDT Project Member

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

Comment 45 by Git Watcher on Wed, May 26, 2021, 12:07 PM EDT Project Member

Labels: merge-merged-4240

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/dc4d50606409e3cc2cdff252001649ada3697ef4

commit dc4d50606409e3cc2cdff252001649ada3697ef4
Author: Pâris Meuleman <pmeuleman@chromium.org>
Date: Wed May 26 16:06:30 2021

[86-LTS] Prevent Cross-Origin iframe from navigating top to a different scheme

Cross-origin iframes were prevented to navigate top with [1]. Those
iframes were allowed to navigate top only to same domain (eTLD+1)
following reports of adverse impact. This severely restrains the ability
of said iframe to cause nuisance.
It does not seem necessary however to loosen the constraint to allow
different schemes, especially from https to http. As a result this CL
prevents a cross-origin iframe from navigating top to the same eTLD + 1
with a different schemes if there's no user gesture.

[1] https://github.com/WICG/interventions/issues/16

~~Bug: 1151507~~
~~Fixed: 1151507~~

(cherry picked from commit 1baf9eba07b806f86a6e60851428c7ab318da093)

Change-Id: Ia1568175c044831594154ceea3e3aacb4e2efb2c
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2756509
Commit-Queue: Nate Chapin <japhet@chromium.org>
Auto-Submit: Pâris Meuleman <pmeuleman@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#863936}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2917013
Reviewed-by: Victor-Gabriel Savu <vsavu@google.com>
Commit-Queue: Jana Grill <janagrill@google.com>
Owners-Override: Jana Grill <janagrill@google.com>
Cr-Commit-Position: refs/branch-heads/4240@{#1649}
Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify] https://crrev.com/dc4d50606409e3cc2cdff252001649ada3697ef4/third_party/blink/renderer/core/frame/local_frame.cc
[add] https://crrev.com/dc4d50606409e3cc2cdff252001649ada3697ef4/third_party/blink/web_tests/http/tests/security/frameNavigation/resources/iframe-that-performs-different-scheme-same-etld-plus-one-top-navigation-without-user-gesture.html
[add] https://crrev.com/dc4d50606409e3cc2cdff252001649ada3697ef4/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture-expected.txt
[add] https://crrev.com/dc4d50606409e3cc2cdff252001649ada3697ef4/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture.html

Comment 46 by janag...@google.com on Wed, May 26, 2021, 12:23 PM EDT Project Member

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

Comment 47 by amyressler@google.com on Mon, Jun 7, 2021, 3:27 PM EDT Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 48 by vsavu@google.com on Mon, Jun 14, 2021, 12:40 PM EDT Project Member

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

Comment 49 by gianluca@google.com on Tue, Jun 15, 2021, 6:29 AM EDT Project Member

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

Comment 50 by Git Watcher on Wed, Jun 16, 2021, 9:16 AM EDT Project Member

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

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/fa42dbe9b20c062f5bbccc589aee86ae43174cdc

commit fa42dbe9b20c062f5bbccc589aee86ae43174cdc
Author: Pâris Meuleman <pmeuleman@chromium.org>
Date: Wed Jun 16 13:15:07 2021

[M90-LTS] Prevent Cross-Origin iframe from navigating top to a different scheme

Cross-origin iframes were prevented to navigate top with [1]. Those
iframes were allowed to navigate top only to same domain (eTLD+1)
following reports of adverse impact. This severely restrains the ability
of said iframe to cause nuisance.
It does not seem necessary however to loosen the constraint to allow
different schemes, especially from https to http. As a result this CL
prevents a cross-origin iframe from navigating top to the same eTLD + 1
with a different schemes if there's no user gesture.

[1] https://github.com/WICG/interventions/issues/16

~~Bug: 1151507~~
~~Fixed: 1151507~~

(cherry picked from commit 1baf9eba07b806f86a6e60851428c7ab318da093)

Change-Id: Ia1568175c044831594154ceea3e3aacb4e2efb2c
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2756509
Commit-Queue: Nate Chapin <japhet@chromium.org>
Auto-Submit: Pâris Meuleman <pmeuleman@chromium.org>
Reviewed-by: Nate Chapin <japhet@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#863936}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2960870
Reviewed-by: Achuith Bhandarkar <achuith@chromium.org>
Owners-Override: Victor-Gabriel Savu <vsavu@google.com>
Commit-Queue: Victor-Gabriel Savu <vsavu@google.com>
Cr-Commit-Position: refs/branch-heads/4430@{#1528}
Cr-Branched-From: e5ce7dc4f7518237b3d9bb93cccca35d25216cbe-refs/heads/master@{#857950}

[modify] https://crrev.com/fa42dbe9b20c062f5bbccc589aee86ae43174cdc/third_party/blink/renderer/core/frame/local_frame.cc
[add] https://crrev.com/fa42dbe9b20c062f5bbccc589aee86ae43174cdc/third_party/blink/web_tests/http/tests/security/frameNavigation/resources/iframe-that-performs-different-scheme-same-etld-plus-one-top-navigation-without-user-gesture.html
[add] https://crrev.com/fa42dbe9b20c062f5bbccc589aee86ae43174cdc/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture-expected.txt
[add] https://crrev.com/fa42dbe9b20c062f5bbccc589aee86ae43174cdc/third_party/blink/web_tests/http/tests/security/frameNavigation/xss-DENIED-different-scheme-same-etld-plus-1-top-navigation-without-user-gesture.html

Comment 51 by vsavu@google.com on Wed, Jun 16, 2021, 9:24 AM EDT Project Member

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

Comment 52 by alesa...@alesandroortiz.com on Sun, Sep 19, 2021, 2:29 PM EDT

This issue has been fixed for 14 weeks (see #c23 on March 17th). Is there a reason sheriffbot didn't make it public? IIUC should have been automatically disclosed around June 23rd.

Comment 53 by sheriffbot on Wed, Sep 22, 2021, 1:35 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 54 by Git Watcher on Tue, Nov 22, 2022, 2:17 PM EST (23 days ago) Project Member

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/301ecd231061729fff89eb9db3fd628e45d991a8

commit 301ecd231061729fff89eb9db3fd628e45d991a8
Author: 揚帆起航 <uioptt24@gmail.com>
Date: Tue Nov 22 19:16:02 2022

Remove "kBlockCrossOriginTopNavigationToDiffentScheme"

~~Bug: 1151507~~

Change-Id: I1b78eab4207aaa827d0d36a514190b64ed7014a7
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/4048380
Reviewed-by: Philip Rogers <pdr@chromium.org>
Auto-Submit: 揚帆起航 <uioptt24@gmail.com>
Reviewed-by: Peter Kasting <pkasting@chromium.org>
Commit-Queue: Peter Kasting <pkasting@chromium.org>
Cr-Commit-Position: refs/heads/main@{#1074783}

[modify] https://crrev.com/301ecd231061729fff89eb9db3fd628e45d991a8/third_party/blink/common/features.cc
[modify] https://crrev.com/301ecd231061729fff89eb9db3fd628e45d991a8/third_party/blink/public/common/features.h
[modify] https://crrev.com/301ecd231061729fff89eb9db3fd628e45d991a8/third_party/blink/renderer/core/frame/local_frame.cc