☆ Starred by 4 users

| | |
|---|---|
| **Owner:** | mek@chromium.org |
| **CC:** | mkwst@chromium.org<br>adetaylor@chromium.org<br>jsb...@chromium.org<br>c...@chromium.org<br>mek@chromium.org<br>pwnall@chromium.org |
| **Status:** | Fixed *(Closed)* |
| **Components:** | Blink>Storage>AppCache |
| **Modified:** | Jun 16, 2021 |
| **Backlog-Rank:** | ---- |
| **Editors:** | ---- |
| **EstimatedDays:** | ---- |
| **NextAction:** | ---- |
| **OS:** | Linux, Android, Windows, Chrome, Mac, Fuchsia |
| **Pri:** | 1 |
| **Type:** | Bug-Security |

Hotlist-Merge-Review
reward-5000
Security_Impact-Stable
Security_Severity-Medium
allpublic
reward-inprocess
CVE_description-submitted
M-89
Target-87
Target-89
Merge-Rejected-88
merge-merged-4240
merge-merged-86
LTR-Merged-86
LTS-Security-86
Release-0-M89
external_security_report
CVE-2021-21168

### Issue 1152226: Leaking the URL of any cross-origin redirect through AppCache's network section

Reported by herre...@gmail.com on Mon, Nov 23, 2020, 9:02 PM EST

Code

Today I was preparing a few PoCs for an upcoming talk about XSLeaks and I ended up playing with the PoC for ~~bug 1030860~~ and realized there is a variation of the original attack that still works.

When a manifest has a NETWORK section, all requests initiated on the page are blocked and only URLs contained in the section are allowed to go through.

The problem is that the functionality behind blocking or allowing a request checks whether the URLs added to the Network Section are contained in the URL of requests that were initiated on the page, instead of making a comparison between the entire URLs.

By abusing this behavior it is possible to leak the full URL of any cross-origin redirect (similarly to ~~bug 1030860~~). This is a serious vulnerability because there are a lot of endpoints on the web that redirect to session tokens / CSRF tokens / sensitive information.

I will be using the https://www.facebook.com/me endpoint to describe how the attack works.

When a logged user accesses it, they are redirected to their profile.
In this example, let's assume that when https://www.facebook.com/me is accessed, the victim is redirected to https://www.facebook.com/victim

1. The attacker's page creates a manifest in the following way:

```
CACHE MANIFEST

NETWORK:
https://www.facebook.com/me
https://www.facebook.com/v
```

2. After the manifest is installed, the attacker's page does a fetch to https://www.facebook.com/me:

```
fetch("https://www.facebook.com/me", {
    mode: "no-cors",
    credentials: "include"
}).then(() => {
    console.log("Pattern matched");
}).catch(() => {
    console.log("Pattern didn't match");
});
```

3. The fetch will be successful because https://www.facebook.com/me redirects to https://www.facebook.com/victim and the URL on the Network Section (https://www.facebook.com/v) is contained in it. This allows the attacker to infer that the first character of the victim's account name starts with "v". This can be extended to all characters of the URL.

The PoC I provided is a bit more complex than the described example because it does a binary search to leak the information faster, but the underlying concept is the same. I have also attached the server used in case you want to reproduce it locally.

Here's an unlisted video demonstrating the issue:
https://youtu.be/dHfXJGASChg

**VERSION**
Version 87.0.4280.66 (Official Build) (64-bit)

**REPRODUCTION CASE**
1. Make sure you are logged into a Facebook account.
2. Go to https://lbherrera.me/appcache-redirect/index.php
3. Check the DevTools console - your account name should start to be brute-forced.

If you download the files and try to reproduce locally, you will need to replace both origin-trial keys in /cache.php and /manifest.php with your own (given AppCache is behind a reverse origin-trial). Also, if the PoC stops working after the first repro, please clear the application cache to make it work again.

**CREDIT INFORMATION**
Reporter credit: Luan Herrera (@lbherrera_)

**appcache-redirect.zip**
2.3 KB Download

Comment 1 by sheriffbot on Mon, Nov 23, 2020, 9:07 PM EST

**Labels:** reward-potential

Comment 2 by mbarb...@chromium.org on Tue, Nov 24, 2020, 2:52 AM EST

**Status:** Assigned (was: Unconfirmed)
**Owner:** c...@chromium.org
**Cc:** jsb...@chromium.org mek@chromium.org
**Labels:** Security_Severity-Medium Security_Impact-Stable OS-Android OS-Chrome OS-Fuchsia OS-Linux OS-Mac OS-Windows
**Components:** Blink>Storage>AppCache

Comment 3 by sheriffbot on Tue, Nov 24, 2020, 1:04 PM EST

**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

**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 mek@chromium.org on Tue, Nov 24, 2020, 10:45 PM EST

**Cc:** pwnall@chromium.org

Huh, fun... It seems the simplest fix here might be to treat every manifest as if * was present in the NETWORK section. It won't be spec compliant, but it seems the least likely thing to break websites since it will only change fetches that were previously blocked to now not be blocked.

Although if websites were relying on this ability of appcache to block requests to arbitrary URLs as some kind of security boundary doing so would break that. I'm not entirely sure what using appcache that way would actually protect against. CSP seems a better mechanism to restrict what subresources can be fetches...

Unfortunately I don't think we really have metrics around usage for this. Another option would be to only block access to same-origin resources if there is no "*" in the NETWORK section, but always fall back to the network for cross-origin resources. Not sure if that is any better than just never blocking.

Comment 6 by mkwst@google.com on Wed, Nov 25, 2020, 2:25 AM EST

**Cc:** mkwst@chromium.org

> Unfortunately I don't think we really have metrics around usage for this

Would digging into manifests present in HTTP Archive be helpful? From an API owner's perspective, I wouldn't set a terribly high bar for this kind of change with direct security impact if you can hand-wave at some numbers that set an upper limit for "breakage".

(Apropos of nothing, what's our current timeline for getting rid of AppCache? :) )

Comment 7 by mek@chromium.org on Wed, Nov 25, 2020, 4:32 PM EST

Re timeline, currently AppCache is in a reverse origin trial, and our plan/hope is still to fully remove it by M90/April 2021. But that depends on some large partners finishing their migration away from AppCache.

That does mean that we have a full list of the ~1000 origins that are still using AppCache/have contact information for all of them.

If http archive does indeed record manifests that might be indeed be a way to get some idea how many manifests use NETWORK without something other than *.

Comment 8 by mkwst@chromium.org on Mon, Nov 30, 2020, 6:59 AM EST

If we only have 1000 origins that are using AppCache, then we probably don't need HTTP Archive, and can just grab their manifests directly? Give me a list of origins, I'll do some legwork?

Comment 9 by c...@chromium.org on Tue, Dec 1, 2020, 2:27 PM EST

**Owner:** mek@chromium.org
**Cc:** c...@chromium.org

Hey Marijn, I chatted with Victor and he said you're the right owner for this, so updating to reflect.

Comment 10 by mek@chromium.org on Tue, Dec 1, 2020, 4:37 PM EST

I asked the origin trials team to share the list of origins with mkwst@. The list I got is locked down such that I can't share it myself.

Comment 11 by mkwst@chromium.org on Mon, Dec 14, 2020, 5:30 AM EST

I have the list, but I haven't been able to go through all of it. That said, spot-checking ~70 of the ~1000, I didn't find any other than Docs that were actually _using_ the origin trial. It might well be the case that appcache is only used on pages that require login, or some internal pages beyond the homepage?

My understanding is that treating Docs' `NETWORK` section as though it contained `*` wouldn't cause any issues (beyond potentially some performance impact). If that's the case, perhaps we could simply try shipping that behavior and crossing our fingers?

Comment 12 by mek@chromium.org on Tue, Dec 15, 2020, 5:30 PM EST

Just trying to ship sounds good to me. I imagine we'd want to/have to go through the intents process, although I'm never sure what the best order is to do things in/how much to share publicly with web exposed security issues like this (this one being particularly odd in that the relevant spec has already been deleted entirely).

Comment 13 by pwnall@chromium.org on Tue, Dec 15, 2020, 7:22 PM EST

We only posted FYIs to blink-dev@ for security-driven AppCache behavior changes in the last couple of years. Concrete examples: padding in quota calculations, manifest scope restrictions.

I think we should do the same here.

Comment 14 by mek@chromium.org on Wed, Dec 16, 2020, 1:57 PM EST

I'm fine with just doing an FYI. There was some pushback from API owners on a different FYI for a security issue [1] recently, but perhaps this situation is different enough that here just an FYI would be accepted.

[1] https://groups.google.com/a/chromium.org/g/blink-dev/c/0d4L4zZWcH4/m/vkm5xhu-AgAJ

Comment 15 by mek@chromium.org on Wed, Dec 16, 2020, 2:22 PM EST

https://chromium-review.googlesource.com/c/chromium/src/+/2594235 is one possible way to implement this

Comment 16 by mkwst@chromium.org on Thu, Dec 17, 2020, 3:20 AM EST

I think that changes to deprecated APIs behind a reverse origin trial are different in kind to changing APIs that are widely available. PSAs for security issues in the former seem reasonable. PSAs for the latter are harder (for me) to justify.

AppCache falls squarely into the former category, and I'd be comfy with a PSA after the fact.

Comment 17 by sheriffbot on Thu, Dec 31, 2020, 12:21 PM EST

mek: 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 18 by herre...@gmail.com on Tue, Jan 5, 2021, 4:57 PM EST

Hey, friendly ping!

Comment 19 Deleted

Comment 20 by bugdroid on Thu, Jan 14, 2021, 12:49 PM EST

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/b590da564f47bf5b6094cd8db45e9dd9c47a0d06

commit b590da564f47bf5b6094cd8db45e9dd9c47a0d06
Author: Marijn Kruisselbrink <mek@chromium.org>
Date: Thu Jan 14 17:48:07 2021

[AppCache] Add feature to always fallback requests to the network.

This adds a (default enabled) feature AppCacheAlwaysFallbackToNetwork,
when enabled chrome will behave as if every manifest file included a
NETWORK: * line, indicating that all requests should fall back to the
network.

~~Bug: 1152226~~
Change-Id: I02f126a91d7061183274a66ad759cde58da533cb
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2594235
Reviewed-by: Marijn Kruisselbrink <mek@chromium.org>
Reviewed-by: Victor Costan <pwnall@chromium.org>
Commit-Queue: Marijn Kruisselbrink <mek@chromium.org>
Cr-Commit-Position: refs/heads/master@{#843607}

[modify] https://crrev.com/b590da564f47bf5b6094cd8db45e9dd9c47a0d06/content/browser/appcache/appcache_request_handler.h
[modify] https://crrev.com/b590da564f47bf5b6094cd8db45e9dd9c47a0d06/content/browser/appcache/appcache_request_handler_unittest.cc
[modify] https://crrev.com/b590da564f47bf5b6094cd8db45e9dd9c47a0d06/third_party/blink/web_tests/VirtualTestSuites
[modify] https://crrev.com/b590da564f47bf5b6094cd8db45e9dd9c47a0d06/content/browser/appcache/appcache_request_handler.cc

Comment 21 by mek@chromium.org on Thu, Jan 14, 2021, 12:57 PM EST

**Status:** Fixed (was: Assigned)

Comment 22 by sheriffbot on Thu, Jan 14, 2021, 1:57 PM EST

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

Comment 23 by sheriffbot on Thu, Jan 14, 2021, 2:22 PM EST

**Labels:** Merge-Request-88

Requesting merge to beta M88 because latest trunk commit (843607) appears to be after beta branch point (827102).

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

Comment 24 by sheriffbot on Thu, Jan 14, 2021, 2:24 PM EST

**Labels:** -Merge-Request-88 Merge-Review-88 Hotlist-Merge-Review

This bug requires manual review: We are only 4 days from stable.
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), marinakz@(ChromeOS), srinivassista @(Desktop)

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

Comment 25 by srinivassista@google.com on Thu, Jan 14, 2021, 2:53 PM EST
**Cc:** adetaylor@chromium.org

+adetaylor@

Comment 26 by adetaylor@chromium.org on Thu, Jan 14, 2021, 3:02 PM EST
**Labels:** -Merge-Review-88 Merge-Rejected-88

It's too late to get this into the initial M88 release (technically we _could_ but it's not sufficiently severe to merit the risk and disruption). Because it could conceivably have compatibility implications, I'm disinclined to release this in an M88 security refresh. As such I think it's best if this releases in M89.

Comment 27 by mek@chromium.org on Tue, Jan 19, 2021, 1:59 PM EST
I wrote a draft blink-dev PSA at https://docs.google.com/document/d/1MfDz7ZwYIoMmchzlOXRCtMfscam1drYO6we_gwQWUrs/edit?usp=sharing&resourcekey=0-00TIFVnXXBj8kITNTa4K3Q, plan to send something out like that later this week.

Comment 28 by adetaylor@google.com on Wed, Jan 20, 2021, 6:57 PM EST
**Labels:** -reward-potential external_security_report

Comment 29 by sheriffbot on Thu, Jan 21, 2021, 12:43 PM EST
**Labels:** reward-topanel

Comment 30 by amyressler@google.com on Wed, Jan 27, 2021, 6:17 PM EST
**Labels:** -reward-topanel reward-unpaid reward-5000

*** 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 31 by amyressler@google.com on Wed, Jan 27, 2021, 7:16 PM EST
Congratulations, Luan! The VRP Panel has decided to award you $5,000 for this report. Nice job!

Comment 32 by amyressler@google.com on Thu, Jan 28, 2021, 3:13 PM EST
**Labels:** -reward-unpaid reward-inprocess

Comment 33 by adetaylor@google.com on Fri, Feb 26, 2021, 1:08 PM EST
**Labels:** Release-0-M89

Comment 34 by adetaylor@google.com on Mon, Mar 1, 2021, 7:27 PM EST
**Labels:** CVE-2021-21168 CVE_description-missing

Comment 35 by vsavu@google.com on Wed, Mar 3, 2021, 5:48 AM EST
**Labels:** LTS-Merge-Request-86

Comment 36 by vsavu@google.com on Wed, Mar 3, 2021, 6:01 AM EST
**Labels:** LTS-Security-86

Comment 37 by gianluca@google.com on Wed, Mar 3, 2021, 10:37 AM EST
**Labels:** LTS-Merge-Approved-86

Comment 38 by sheriffbot on Wed, Mar 3, 2021, 12:22 PM EST
**Labels:** -M-87 Target-89 M-89

Comment 39 by amyressler@google.com on Tue, Mar 9, 2021, 12:58 PM EST
**Labels:** -CVE_description-missing CVE_description-submitted

Comment 40 by Git Watcher on Wed, Mar 10, 2021, 4:36 AM EST
**Labels:** merge-merged-4240 merge-merged-86

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/1e4ddb75b0c36d7bb987256f74a5fec4d842e53b

commit 1e4ddb75b0c36d7bb987256f74a5fec4d842e53b
Author: Marijn Kruisselbrink <mek@chromium.org>
Date: Wed Mar 10 09:35:48 2021

[AppCache] Add feature to always fallback requests to the network.

This adds a (default enabled) feature AppCacheAlwaysFallbackToNetwork,
when enabled chrome will behave as if every manifest file included a
NETWORK: * line, indicating that all requests should fall back to the
network.

[M86 Merge]: Added missing include in appcache_request_handler.h.

(cherry picked from commit b590da564f47bf5b6094cd8db45e9dd9c47a0d06)

~~Bug: 1152226~~
Change-Id: I02f126a91d7061183274a66ad759cde58da533cb
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2594235
Reviewed-by: Marijn Kruisselbrink <mek@chromium.org>
Reviewed-by: Victor Costan <pwnall@chromium.org>
Commit-Queue: Marijn Kruisselbrink <mek@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#843607}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2731808
Reviewed-by: Achuith Bhandarkar <achuith@chromium.org>
Commit-Queue: Victor-Gabriel Savu <vsavu@google.com>
Cr-Commit-Position: refs/branch-heads/4240@{#1570}
Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify] https://crrev.com/1e4ddb75b0c36d7bb987256f74a5fec4d842e53b/content/browser/appcache/appcache_request_handler.cc
[modify] https://crrev.com/1e4ddb75b0c36d7bb987256f74a5fec4d842e53b/content/browser/appcache/appcache_request_handler.h

[modify] https://crrev.com/1e4ddb75b0c36d7bb987256f74a5fec4d842e53b/content/browser/appcache/appcache_request_handler_unittest.cc
[modify] https://crrev.com/1e4ddb75b0c36d7bb987256f74a5fec4d842e53b/third_party/blink/web_tests/VirtualTestSuites

Comment 41 by asumaneev@google.com on Thu, Mar 25, 2021, 12:04 PM EDT

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

Comment 42 by sheriffbot on Wed, Jun 16, 2021, 1:52 PM EDT

**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

About Monorail User Guide Release Notes Feedback on Monorail Terms Privacy