☆ Starred by 9 users

Owner:	davi...@chromium.org
CC:	falken@chromium.org davi...@chromium.org jochen@chromium.org mkwst@chromium.org yhirano@chromium.org adetaylor@chromium.org dom@chromium.org domenic@chromium.org kaustubhag@chromium.org futhark@chromium.org lukasza@chromium.org bashi@chromium.org
Status:	Fixed *(Closed)*
Components:	Blink>SecurityFeature>Referrer
Modified:	Jun 11, 2021
Backlog-Rank:	----
Editors:	----
EstimatedDays:	1
NextAction:	----
OS:	Linux, Android, Windows, Chrome, Mac
Pri:	1
Type:	Bug-Security

Hotlist-Merge-Review
reward-500
Needs-Feedback
Security_Impact-Stable
Security_Severity-Medium
allpublic
reward-inprocess
CVE_description-submitted
M-89
Target-88
Target-89
Merge-Rejected-88
merge-merged-4240
merge-merged-86
LTR-Merged-86
LTS-Security-86
Release-0-M89

Issue 1158010: Security: Referrer Header Spoofing Vulnerability via <base> tags

Reported by techn...@gmail.com on Fri, Dec 11, 2020, 5:53 PM EST

Code

VULNERABILITY DETAILS

While doing some analysis it was observed that when a <base> tag is set to a page and a css request is being made to a page, Chrome browser makes use of the url in base tag as a referrer header value , this is only valid in case a base header is set , in case the header is absent a plain requirest is made to the url for css call.

This was only observed with chrome and other browsers set the referrer to URI requesting the resource.

For example in below code, chrome doesnt set any referrer header automatically when making request to x.burpcollaborator.net .

```
<html><head>
      <title>Test</title>
<style>
   @import 'https://x.burpcollaborator.net/x';<!--change this to any address of your choice and verify the referrer header!-->
</style>
   </head>
   <body>
test
<body></html>
```

However in below code a referrer header is set to google.com in case base header is set , it can be set to any value

```
<html><head>
      <title>Test/title>
   <base href="https://www.gmail.com/">
<style>
   @import 'https://x.burpcollaborator.net/x';<!--change this to any address of your choice and verify the referrer header!-->
</style>
   </head>
   <body>
test
<body></html>
```

VERSION
Chrome Version: Latest (83.0.4103.61)
Operating System: Windows, Mac , Android

REPRODUCTION CASE
Please use below code and save it as html and open via chrome

CREDIT INFORMATION
Externally reported security bugs may appear in Chrome release notes. If

this bug is included, how would you like to be credited?
Reporter credit: Ashish Gautam Kamble

Comment 1 by sheriffbot on Fri, Dec 11, 2020, 5:55 PM EST Project Member

Labels: reward-potential

Comment 2 by wfh@chromium.org on Mon, Dec 14, 2020, 8:14 PM EST Project Member

Cc: davi...@chromium.org kaustubhag@chromium.org mkwst@chromium.org
Labels: Needs-Feedback OS-Android OS-Chrome OS-Linux OS-Mac OS-Windows Pri-2
Components: Blink>SecurityFeature>Referrer

Thanks for your report. This might be a spec issue rather than a security issue. I'm unsure how an attacker could use this to endanger users, since they would already have to control the website in order to set the base tag.

Can you elaborate on how this might be leveraged into an attack? In the meantime, I will add some additional developers on this issue for their comment.

Comment 3 by davi...@chromium.org on Mon, Dec 14, 2020, 8:55 PM EST Project Member

Blockedon: 1158645

Comment 4 by davi...@chromium.org on Mon, Dec 14, 2020, 9:47 PM EST Project Member

Cc: lukasza@chromium.org

Hi, thanks: I was able to verify

```
<base href="https://www.google.com">
<style>
   @import 'https://www.flickr.com/picture.jpg';
</style>
```

hosted on a non-google domain resulted in a google.com referrer.

Without the <base> tag, the request had no referrer (rather than a referrer of the initiating origin, which was what I expected).

This behavior seems to come from the following code:
- CSSParserContext's constructors look for a base URL (for instance from Document::BaseURL) and then set CSSParserContext::referrer_ from this value. [1]
- StyleRuleImport then uses this value when populating fetching arguments. [2]

[1]:
https://source.chromium.org/chromium/chromium/src/+/master:third_party/blink/renderer/core/css/parser/css_parser_context.cc;l=125;drc=ec0e29ae0cc41412b3c0594f1d94ceaa3d9e3870;bpv=1;bpt=1
[2]:
https://source.chromium.org/chromium/chromium/src/+/master:third_party/blink/renderer/core/css/style_rule_import.cc;l=152;drc=ec0e29ae0cc41412b3c0594f1d94ceaa3d9e3870;bpv=1;bpt=1?q=cssparsercontext::getreferrer&ss=chromium%2Fchromium%2Fsrc

I think this is a one-line fix and have written up crrev.com/c/2592447.

Mike is better equipped than me to assess whether there's a security impact.

Comment 5 by davi...@chromium.org on Mon, Dec 14, 2020, 9:48 PM EST Project Member

(Lukasz, I initially added you because I was curious if this could lead to initiatior mismatches, but I think the answer is probably not: we just use this information to set the referrer, not the initiator.)

Comment 6 by wfh@chromium.org on Mon, Dec 14, 2020, 9:50 PM EST Project Member

Labels: Security_Severity-Low Security_Impact-Stable

Comment 7 by wfh@chromium.org on Mon, Dec 14, 2020, 9:51 PM EST Project Member

Status: Assigned (was: Unconfirmed)
Owner: davi...@chromium.org
Cc: -davi...@chromium.org
Labels: -Security_Severity-Low Security_Severity-Medium

assigning as per comment #4. I've given it Medium severity for now.

Comment 8 by davi...@chromium.org on Mon, Dec 14, 2020, 11:48 PM EST Project Member

Owner: falken@chromium.org
Cc: falken@chromium.org

-> falken

Matt, I noticed you've submitted a change or two dealing with CSS fetching and particularly the base URL.

Could you please take a quick look at this report and help confirm whether the behavior described in the report is WAI, or whether it needs a fix (e.g. crrev.com/c/2592447)?

Thanks!

Comment 9 by falken@chromium.org on Tue, Dec 15, 2020, 1:17 AM EST Project Member

Cc: domenic@chromium.org yhirano@chromium.org

Looping in also yhirano@ who has more referrer experience, and domenic@ who has commented on the github issue.

I don't have a very quick answer. The two signals I'd look at are how other browsers behaves and what the spec says.

As for what the spec says, this is a bit hard to chase down. From what I see, it looks like the base URL should not affect Referrer.

1) I don't actually see where the CSS spec says to fetch @import URLs:
https://drafts.csswg.org/css-cascade-4/#at-ruledef-import

2) I assume it should do something similar to <link rel="stylesheet">. This seems to go here:
https://html.spec.whatwg.org/multipage/semantics.html#default-fetch-and-process-the-linked-resource

although there is a TODO and issue to use CSSOM's fetching steps instead:
https://github.com/whatwg/html/issues/968
https://drafts.csswg.org/cssom/#fetching-css-style-sheets

3) Looking at the Fetch spec, it appears to use the document URL rather than the document base URL for the referrer:
"If request's referrer is not "no-referrer", set request's referrer to the result of invoking determine request's referrer. [REFERRER]" at https://fetch.spec.whatwg.org/#main-fetch
https://w3c.github.io/webappsec-referrer-policy/#determine-requests-referrer

I don't see anything that says to use base URL as the referrer anywhere.

So I think changing to use the document URL is indeed correct. It'd be good to see if other browsers do this too. It'd be especially interesting to see when the referrer code was written, if it came from WebKit days or was intentionally added later.

Comment 10 by falken@chromium.org on Tue, Dec 15, 2020, 1:29 AM EST Project Member

Cc: bashi@chromium.org jochen@chromium.org

+bashi@ who seems to have added the original change (but is likely ooo, +jochen as the reviewer):

https://codereview.chromium.org/314893003 for https://bugs.chromium.org/p/chromium/issues/detail?id=380457

This says "For css resources specified in stylesheets, their referrer should be set to the stylesheet's URL, not the document's URL." so this was intentional, but maybe base URL and stylesheet URL are different?

Comment 11 by mkwst@chromium.org on Tue, Dec 15, 2020, 2:00 AM EST Project Member

Cc: dom@chromium.org

1. Unless something has changed in the last year or so, fetching for CSS is not yet well-defined. https://w3c.github.io/webappsec-referrer-policy/#integration-with-css hand-waves at the problem in ways that we considered good-enough while writing it, but it would be excellent for someone (not it!) to actually hammer out the integration. dom@ might be interested.

2. Regardless of spec text, `<base>` should not influence the referrer. The text above points to what that change was supposed to achieve: the stylesheet's URL should be used as the referrer, not the document's URL. In this case, though, the stylesheet is inline in the document, so we do want to use the document's URL.

This does have security implications, insofar as some sites use `referer` as an access-control measure: it shouldn't be possible for a site to forge cross-origin `referer` headers. I'm not sure whether it's possible to send CORS-enabled requests from stylesheets (perhaps fonts?), but it would be interesting to verify that the `Origin` header and the `Sec-Fetch-Site` headers are behaving correctly as well, as I have vague recollections of those being wrapped up in the same data source as the referrer. TL;DR: Medium severity feels right.

Comment 12 by jochen@chromium.org on Tue, Dec 15, 2020, 10:21 AM EST Project Member

yeah, the intention is that if it's an inline stylesheet, the document's referrer should be used, but if it's an standalone stylesheet, that stylesheet's base URL (well, whatever it's called, it's not defined in CSS, but it should be a thing that can't be overwritten) should be used

Comment 13 by lukasza@chromium.org on Tue, Dec 15, 2020, 11:37 AM EST Project Member

RE: #c11: mkwst@: interesting to verify that the `Origin` header and the `Sec-Fetch-Site` headers are behaving correctly

Both `Origin` and `Sec-Fetch-Site` are based on network::ResourceRequest::request_initiator (soon-to-be-verified everywhere against `request_initiator_site_lock` and so unspoofable/protected against compromised renderers).

OTOH, (based on #c12) it seems the `Referer` header can mismatch `request_initiator_site_lock` in the following scenario:

```
    a.com/main.html: <link rel="stylesheet" href="https://b.com/style.css">
    b.com/style.css: @import url("https://c.com/nested.css");
```

Here, the fetch for c.com/nested.css will use `Sec-Fetch-Site` based on request_initiator=a.com (all requests within the frame use the same URLLoaderFactory), but (based on #c12) `Referer` should be based on b.com/style.css (IIUC) - the latter is incompatible with `request_initiator_origin_lock`.

So, if I understood the above correctly, then we get the following conclusion: `Referer` should not be used as an access-control measure / security mechanism (because it is not secure against compromised renderers).

Comment 14 by mkwst@chromium.org on Tue, Dec 15, 2020, 11:50 AM EST Project Member

> Both `Origin` and `Sec-Fetch-Site` are ... unspoofable

Excellent, thanks Lukasz!

> it seems the `Referer` header can mismatch `request_initiator_site_lock` in the following scenario: ... `Referer` should not be used as an access-control measure

Correct. It shouldn't be used; other mechanisms are more robust. "Should", however, has little impact on "does". :) It would be ideal for us to limit the scenarios in which this mechanism has unexpected values to those that require actively corrupting the renderer. (So, I could live with "Low" severity instead of "Medium", but it's still a bug we ought to fix.)

Comment 15 by sheriffbot on Tue, Dec 15, 2020, 1:03 PM EST Project Member

Labels: Target-88 M-88

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 16 by sheriffbot on Tue, Dec 15, 2020, 1:39 PM EST Project Member

Labels: -Pri-2 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 17 by falken@chromium.org on Tue, Dec 15, 2020, 9:04 PM EST Project Member

Cc: davi...@chromium.org

Comment 18 by mkwst@google.com on Wed, Dec 16, 2020, 4:30 AM EST Project Member

~~Issue 1152999~~ has been merged into this issue.

Comment 19 by est...@chromium.org on Wed, Dec 16, 2020, 11:59 AM EST Project Member

(I left a comment on ~~issue 1152999~~ about why I initially removed security labels, since that seems to have caused some confusion.)

Comment 20 by falken@chromium.org on Wed, Dec 16, 2020, 12:01 PM EST Project Member

Owner: davi...@chromium.org

davidvc: would you like to own this again?

Comment 21 by davi...@chromium.org on Wed, Dec 16, 2020, 12:05 PM EST Project Member

Status: Started (was: Assigned)
EstimatedDays: 1

I'll take a look at the failing tests and try to get this in today.

Comment 22 by techn...@gmail.com on Wed, Dec 16, 2020, 12:33 PM EST

will a CVE be assigned for this ?

Comment 23 by davi...@chromium.org on Wed, Dec 16, 2020, 10:31 PM EST Project Member

CL in review (crrev.com/c/2592447)

Comment 24 by falken@chromium.org on Thu, Dec 17, 2020, 12:50 AM EST Project Member

Cc: futhark@chromium.org

adding futhark@ for review context

Comment 25 by techn...@gmail.com on Mon, Dec 21, 2020, 4:10 PM EST

will a CVE be assigned for this ?

Comment 26 by techn...@gmail.com on Mon, Dec 21, 2020, 4:11 PM EST

or i will have to apply for a cve

Comment 27 by adetaylor@google.com on Wed, Dec 23, 2020, 11:48 AM EST Project Member

A CVE will be assigned when we release the fix.

Comment 28 by bugdroid on Fri, Jan 8, 2021, 11:07 AM EST Project Member

Status: Fixed (was: Started)

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/0b1539fcb923056624d4adc84b88140d367d92da

commit 0b1539fcb923056624d4adc84b88140d367d92da
Author: David Van Cleve <davidvc@chromium.org>
Date: Fri Jan 08 16:06:24 2021

css: Make fetches from inline CSS use the document's URL as referrer

Right now, fetches from inline CSS use the inline CSS's base URL instead of the URL from the context that embeds the inline CSS: for instance, loading a source-site.com page with the following code

```
  <base href="https://other-site.com">
  <style type=text/css> @import('best-sheet.com') </style>
```

should lead to the best-sheet.com sheet getting fetched with a source-site.com referrer, but it will currently provide an other-site.com referrer. However, if the imported sheet from best-sheet.com makes more nested fetches, those nested requests should use best-sheet.com as the basis for their referrers (as they do currently).

This CL updates CSSParserContext's referrer setting logic to roughly do the following:
- inline CSS: use the embedding document's URL as the referrer, or, for srcdoc iframes, walk up the frame tree until hitting a non-srcdoc frame
- requests from fetched stylesheets: just as currently, use the fetched sheet's URL as the basis for constructing the referrer

This seemed like it required refactoring CSSParserContext slightly because there are constructors that take both a Document and a base URL, and it's not obvious from the constructor signature whether the Document or the base URL should be the one that provides the referrer. To resolve this ambiguity, the refactor updates these CSSParserContext constructors to take caller-provided Referrer objects.

Change-Id: If5a99d8057dff5e771e821d0e1f605566e28ff1d
~~Fixed: 1158645~~, ~~1158010~~
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2592447
Reviewed-by: Rune Lillesveen <futhark@chromium.org>
Reviewed-by: Matt Falkenhagen <falken@chromium.org>
Commit-Queue: David Van Cleve <davidvc@chromium.org>
Cr-Commit-Position: refs/heads/master@{#841509}

[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/parser/css_parser_context.h
[add] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/web_tests/external/wpt/referrer-policy/css-integration/image/inline-style-with-differentorigin-base-tag.tentative.html
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/css_style_sheet.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/html/track/vtt/vtt_parser.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/selector_query.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/html/link_style.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/dom/processing_instruction.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/style_rule_import.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/web_tests/TestExpectations
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/web_tests/http/tests/css/resources/referrer-check.php
[add] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/web_tests/external/wpt/referrer-policy/css-integration/svg/inline-style-with-differentorigin-base-tag.tentative.html
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/selector_query_test.cc
[modify] https://crrev.com/0b1539fcb923056624d4adc84b88140d367d92da/third_party/blink/renderer/core/css/parser/css_parser_context.cc

Comment 29 by sheriffbot on Fri, Jan 8, 2021, 12:43 PM EST Project Member

Labels: reward-topanel

Comment 30 by sheriffbot on Fri, Jan 8, 2021, 1:59 PM EST Project Member

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

Comment 31 by sheriffbot on Fri, Jan 8, 2021, 2:25 PM EST Project Member

Labels: Merge-Request-88

This is sufficiently serious that it should be merged to beta. But I can't see a Chromium repo commit here, so you will need to investigate what - if anything - needs to be merged to M88. Is there a fix in some other repo which should be merged? Or, perhaps this ticket is a duplicate of some other ticket which has the real fix: please track that down and ensure it is merged appropriately.

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

Comment 32 by sheriffbot on Fri, Jan 8, 2021, 2:26 PM EST Project Member

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

This bug requires manual review: We are only 10 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 33 by davi...@chromium.org on Fri, Jan 8, 2021, 4:54 PM EST Project Member

Cc: adetaylor@chromium.org

+adetaylor: Hi Adrian! Is a medium-severity bug worth merging? The fix is relatively simple, but not totally trivial, and it's in some finicky, old code I'm not too familiar with.

Comment 34 by adetaylor@chromium.org on Fri, Jan 8, 2021, 6:50 PM EST Project Member

Labels: -Merge-Review-88 Merge-Rejected-88

Per policy yes, for externally reported medium severity bugs we do merge them back to both beta and stable. But if there's any doubt at all about stability risks, we don't, so I'll adjust labels thusly.

Comment 35 by techn...@gmail.com on Tue, Jan 12, 2021, 11:30 AM EST

Hi @adetaylor@chromium.org . Will this be patched in upcoming chrome release

Comment 36 by adetaylor@chromium.org on Tue, Jan 12, 2021, 12:13 PM EST Project Member

This will be released in Chrome 89 - due for release in March - https://chromiumdash.appspot.com/schedule

Comment 37 by sheriffbot on Thu, Jan 14, 2021, 4:22 PM EST Project Member

Labels: external_security_report

Comment 38 by adetaylor@google.com on Wed, Jan 20, 2021, 7:01 PM EST Project Member

Labels: -reward-potential

Comment 39 by amyressler@google.com on Wed, Jan 20, 2021, 7:10 PM EST Project Member

Labels: -reward-topanel reward-unpaid reward-500

*** 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 40 by amyressler@google.com on Wed, Jan 20, 2021, 7:46 PM EST Project Member

Congratulation, Ashish - the VRP Panel has decided to award you $500 for this report. A member of our finance team will be in touch with you soon to arrange payment. Thank you for your work and nice job!

Comment 41 by amyressler@google.com on Thu, Jan 21, 2021, 1:50 PM EST Project Member

Labels: -reward-unpaid reward-inprocess

Comment 42 by vakh@chromium.org on Mon, Jan 25, 2021, 6:12 PM EST Project Member

~~Issue 1170235~~ has been merged into this issue.

Comment 43 by adetaylor@google.com on Fri, Feb 26, 2021, 1:08 PM EST Project Member

Labels: Release-0-M89

Comment 44 by adetaylor@google.com on Mon, Mar 1, 2021, 7:27 PM EST Project Member

Labels: CVE-2021-21174 CVE_description-missing

Comment 45 by vsavu@google.com on Wed, Mar 3, 2021, 5:40 AM EST Project Member

Labels: LTS-Merge-Request-86

Comment 46 by vsavu@google.com on Wed, Mar 3, 2021, 6:01 AM EST Project Member

Labels: LTS-Security-86

Comment 47 by gianluca@google.com on Wed, Mar 3, 2021, 10:36 AM EST Project Member

Labels: LTS-Merge-Approved-86

Comment 48 by sheriffbot on Wed, Mar 3, 2021, 12:22 PM EST Project Member

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

Comment 49 by achuith@chromium.org on Thu, Mar 4, 2021, 10:42 AM EST Project Member

Labels: -LTS-Merge-Request-86

Comment 50 by Git Watcher on Thu, Mar 4, 2021, 11:51 AM EST Project Member

Labels: merge-merged-4240 merge-merged-86

The following revision refers to this bug:
https://chromium.googlesource.com/chromium/src/+/e1505713dc313c6666b65b073bc7da9cfa1bf765

commit e1505713dc313c6666b65b073bc7da9cfa1bf765
Author: David Van Cleve <davidvc@chromium.org>
Date: Thu Mar 04 16:50:46 2021

css: Make fetches from inline CSS use the document's URL as referrer

Right now, fetches from inline CSS use the inline CSS's base URL
instead of the URL from the context that embeds the inline CSS: for

instance, loading a source-site.com page with the following code
  <base href="https://other-site.com">
  <style type=text/css> @import('best-sheet.com') </style>
should lead to the best-sheet.com sheet getting fetched with a
source-site.com referrer, but it will currently provide an
other-site.com referrer. However, if the imported sheet from
best-sheet.com makes more nested fetches, those nested requests should
use best-sheet.com as the basis for their referrers (as they do
currently).

This CL updates CSSParserContext's referrer setting logic to roughly do
the following:
- inline CSS: use the embedding document's URL as the referrer, or, for
srcdoc iframes, walk up the frame tree until hitting a non-srcdoc frame
- requests from fetched stylesheets: just as currently, use the fetched
sheet's URL as the basis for constructing the referrer

This seemed like it required refactoring CSSParserContext slightly
because there are constructors that take both a Document and a base URL,
and it's not obvious from the constructor signature whether the
Document or the base URL should be the one that provides the referrer.
To resolve this ambiguity, the refactor updates these CSSParserContext
constructors to take caller-provided Referrer objects.

(cherry picked from commit 0b1539fcb923056624d4adc84b88140d367d92da)

Change-Id: If5a99d8057dff5e771e821d0e1f605566e28ff1d
~~Fixed: 1158645~~, ~~1158010~~
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2592447
Reviewed-by: Rune Lillesveen <futhark@chromium.org>
Reviewed-by: Matt Falkenhagen <falken@chromium.org>
Commit-Queue: David Van Cleve <davidvc@chromium.org>
Cr-Original-Commit-Position: refs/heads/master@{#841509}
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/+/2731576
Reviewed-by: Achuith Bhandarkar <achuith@chromium.org>
Commit-Queue: Victor-Gabriel Savu <vsavu@google.com>
Cr-Commit-Position: refs/branch-heads/4240@{#1558}
Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/css_style_sheet.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/parser/css_parser_context.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/parser/css_parser_context.h
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/selector_query.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/selector_query_test.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/css/style_rule_import.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/dom/processing_instruction.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/html/link_style.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/renderer/core/html/track/vtt/vtt_parser.cc
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/web_tests/TestExpectations
[add] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/web_tests/external/wpt/referrer-policy/css-integration/image/inline-style-with-differentorigin-base-tag.tentative.html
[add] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/web_tests/external/wpt/referrer-policy/css-integration/svg/inline-style-with-differentorigin-base-tag.tentative.html
[modify] https://crrev.com/e1505713dc313c6666b65b073bc7da9cfa1bf765/third_party/blink/web_tests/http/tests/css/resources/referrer-check.php

Comment 51 by vsavu@google.com on Mon, Mar 8, 2021, 11:15 AM EST Project Member

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

Comment 52 by amyressler@google.com on Tue, Mar 9, 2021, 12:58 PM EST Project Member

Labels: -CVE_description-missing CVE_description-submitted

Comment 53 by sheriffbot on Fri, Jun 11, 2021, 1:52 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