

✔ User content can redirect the logout button to different URL (CVE-2020-10959)

✔ Closed, ResolvedPublic

Actions

Assigned To

sbassett

Authored By

Fomafix
2019-09-14 19:43:59 (UTC+0)

Tags

MediaWiki-Interface (Backlog)

Security

Security-Team (In Progress)

MW-1.35-notes (1.35.0-wmf.26; 2020-03-31)

MW-1.34-notes

Referenced Files

F31613044: T232932-rev3.patch
2020-02-12 18:00:19 (UTC+0)

F31611278: T232932-rev2.patch
2020-02-12 15:39:44 (UTC+0)

F31608866: T232932.patch
2020-02-11 23:11:12 (UTC+0)

Subscribers

Aklapper

Fomafix

Jdlrobson

Krenair

Krinkle

Ladsgroup

Reedy

View All 9 Subscribers

Description

Steps to reproduce

- Put the following wiki text into a wiki page:

```
<div id="pt-logout">[https://www.example.com/ click]</div>
```

- Log in the wiki.
- Load the page with the rendered wiki text.
- Click on the logout button.

-> The user get logged out.

-> The browser redirects to <https://www.example.com/>

Expected behavior

It should not possible to change the target of an interface button by user content.

Mitigation

<https://gerit.wikimedia.org/r/536725> ([rMWd4a552e65bdf](#)) by [@Krinkle](#) mitigates this issue.

The button in the user content still logs out. Mitigation for this is to add `data-mw="interface"` as HTML attribute to the logout button and add `[data-mw="interface"]` to the jQuery selector for selecting the button.

Details









Project	Subject
mediawiki/core	SECURITY: Better controls for logout interface buttons
mediawiki/core	SECURITY: Better controls for logout interface buttons

Customize query in [gerit](#)

Related Objects

Search...

Task Graph	Mentions	
Status	Assigned	Task
Resolved	Reedy	T240392 Release MediaWiki 1.31.7/1.33.3/1.34.1
Resolved	Reedy	T240393 Tracking bug for MediaWiki 1.31.7/1.33.3/1.34.1
Resolved	sbassett	T232932 User content can redirect the logout button to different URL (CVE-2020-10959)

-  **Fomafix** created this task. 2019-09-14 19:43:59 (UTC+0)
-   Restricted Application added a subscriber: **Aklapper**. · View Herald Transcript 2019-09-14 19:44:01 (UTC+0)
-  **Fomafix** updated the task description. (**Show Details**) 2019-09-14 19:47:51 (UTC+0)
-  **Fomafix** added a subscriber: **Krinkle**.
-  **Reedy** added a parent task: ~~T225152: Tracking bug for MediaWiki 1.31-4/1.32-6/1.33-1 security release~~. 2019-09-14 21:22:19 (UTC+0)
-  **Aklapper** added a project: **MediaWiki-General**. 2019-09-14 21:52:16 (UTC+0)
-  **Fomafix** updated the task description. (**Show Details**) 2019-09-17 09:46:20 (UTC+0)


 **Reedy** added a subscriber: **Reedy**. Edited · 2019-09-21 21:20:30 (UTC+0)

Does this only affect master? Or should we be back porting to REL1_33/REL1_32/REL1_31?



I'm aiming to get a security and maintenance release out next week (though there are no private security patches to go out)


It looks like there `1490e9dac828af097f0a15eca63eaa89e2d2bc50` and `97fffb3fd0b345db0a8999eda1ffa672311da9f1` might be needed as dependencies, just for REL1_33. But then there's some resources.php patch... Never mind any previous versions of MW...

Can someone take care of these backports where they're appropriate?

 **Krinkle** added a comment. Edited · 2019-09-25 19:58:29 (UTC+0)





Good catch. The link should be protected by `[data-mw="interface"]` which is illegal in user content (enforced, automatically filtered out).


-  **Krinkle** edited projects, added **MediaWiki-Interface**; removed **MediaWiki-General**. 2019-09-25 19:58:46 (UTC+0)
-  **Krinkle** added subscribers: **Ladsgroup**, **Jdlrobbson**.


 **Fomafix** added a comment. 2019-09-26 07:31:18 (UTC+0)


The click handler for the logout button was introduced in `rMW8f033911030d` included in REL1_34. This has two vulnerabilities by user content:

1. The target of the redirect of the logout button can be manipulated. This is fixed in `rMWd4a552e65bdf` included in REL1_34. A backport to older releases is not needed.
2. The user content can create an own logout button and do click catching. This is not fixed yet. This can be done by protecting the link with `[data-mw="interface"]`.


-  **Reedy** edited parent tasks, added: ~~T233495: Tracking bug for MediaWiki 1.31-6/1.32-6/1.33-2/1.34-0 security release~~, removed: ~~T225152: Tracking bug for MediaWiki 1.31-4/1.32-4/1.33-1 security release~~. 2019-10-07 18:49:32 (UTC+0)
- • **chasemp** triaged this task as *Medium* priority. 2019-12-09 15:57:24 (UTC+0)
-  **Reedy** edited parent tasks, added: ~~T240293: Tracking bug for MediaWiki 1.31-7/1.33-3/1.34-1~~, removed: ~~T233495: Tracking bug for MediaWiki 1.31-6/1.32-6/1.33-2/1.34-0 security release~~. 2019-12-10 23:03:10 (UTC+0)
-  • **chasemp** added a project: **Security**. 2020-02-10 22:56:44 (UTC+0) · 


 **Krinkle** added a project: **Security-Team**. Edited · 2020-02-10 23:40:00 (UTC+0)

 **Krinkle** removed a subscriber: **Krinkle**.

 **Krinkle** added a subscriber: **Krinkle**.

This seems important, tagging Security-Team directly to push, forward, or delegate accordingly.


-  **Krinkle** removed a subscriber: **Krinkle**. 2020-02-10 23:40:39 (UTC+0)


 **sbassett** added a subscriber: **sbassett**. 2020-02-11 23:11:12 (UTC+0)


In ~~T22932#5524054~~, @Fomafix wrote:

1. The user content can create an own logout button and do click catching. This is not fixed yet. This can be done by protecting the link with `[data-mw="interface"]`.


Is the fix as simple as this? Seems to work locally for me, I think.

 **T232932.patch** 970 B
Download


 **Fomafix** added a comment. 2020-02-12 07:33:37 (UTC+0)


 **T232932.patch** 970 B
Download


This is only the query part. The logout button doesn't have the attribute `data-mw="interface"` yet.

 **sbassett** added a comment. 2020-02-12 15:39:44 (UTC+0)

Second attempt. Not sure how hacky this is, though if `personal_urls` now require `data-mw` attribute support, this seems like the logical place to do it.

 **T232932-rev2.patch** 2 KB
Download

 **Fomafix** added a comment. 2020-02-12 17:23:59 (UTC+0)

 **T232932-rev2.patch** 2 KB
Download

mitigates the second vulnerability. Just the trailing whitespace in BaseTemplate.php is wrong.

sbassett added a comment.

2020-02-12 18:00:19 (UTC+0)

Ok, trailing whitespace removed.

T232932-rev3.patch

2 KB

Download

sbassett added a comment.

2020-02-13 16:16:57 (UTC+0)

If there are no objections to the rev3 patch above, I can plan to security-deploy it this afternoon (before SWAT and the train) or tomorrow (2020-02-14).

Fomafix added a comment.

2020-02-14 05:23:16 (UTC+0)

Can there a caching issue happen? The old selector also matches to the new HTML but the new selector doesn't match to the old HTML. If the deployment can't ensure that always a new HTML is loaded when the new JavaScript is delivered to the browsers then this patch have to split up.

sbassett added a comment.

2020-02-18 17:00:58 (UTC+0)

In **T232932#5884045**, @Fomafix wrote:
Can there a caching issue happen? The old selector also matches to the new HTML but the new selector doesn't match to the old HTML. If the deployment can't ensure that always a new HTML is loaded when the new JavaScript is delivered to the browsers then this patch have to split up.

I'm wondering if we could get away with performing a conditional check for the new selector and if it wasn't found, the old one, within `resources/src/mediawiki.page.ready/ready.js` for some period of time (24 hrs, a week, something else) to allow for various caches to clear.

sbassett moved this task from Incoming to In Progress on the Security-Team board.

2020-02-19 16:35:44 (UTC+0)

chasemp removed a project: **adl*security**.

2020-02-20 20:06:22 (UTC+0)

Fomafix added a comment.

2020-02-21 05:22:24 (UTC+0)

I guess there is no caching problem on deployment. And even when the new JavaScript is delivered to the browsers before the new HTML, then the fallback way of the logout button still works.

Jdlrobson added a comment.

2020-02-21 14:59:43 (UTC+0)

Logged in users bypass the varnish cache so I don't believe there are any caching issues here.

sbassett added a comment.

2020-02-21 15:10:35 (UTC+0)

Ok, I'll plan to deploy the patch as-is (**T232932#5877700**) today and keep an eye on logstash just in case.

sbassett added a project: **PermanentlyPrivate**.

2020-02-21 22:12:46 (UTC+0)

Patch deployed. Tested on testwiki (and oversighted) - everything worked fine for me. Just to note: this issue is being held for the [next security release](#), so please **keep this task private** for now (adding **PermanentlyPrivate** to be safe, until release) and refrain from backporting in gerrit for the time being. Thanks.

Restricted Application changed the visibility from "Custom Policy" to "Custom Policy". · View Herald Transcript

2020-02-21 22:12:47 (UTC+0)

Restricted Application changed the edit policy from "All Users" to "Custom Policy". · View Herald Transcript

sbassett mentioned this in **T240393 - Tracking bug for MediaWiki 1.34.7/1.35.3/1.34.1**.

2020-02-21 22:13:33 (UTC+0)

sbassett claimed this task.

2020-03-02 20:41:29 (UTC+0)

sbassett added a subscriber: Krinkle.

sbassett mentioned this in **T246602 - make Collapsible allows applying event handler to any CSS selector (CVE-2020-10960)**.

2020-03-02 22:51:00 (UTC+0)

Reedy added a comment.

2020-03-24 17:15:53 (UTC+0)

Patch applies cleanly to master and REL1_34.

Doesn't to REL1_33 and REL1_31. Time to poke further

Reedy added a comment.

Edited · 2020-03-24 17:18:36 (UTC+0)

Changes to be committed:

modified: includes/skins/BaseTemplate.php

modified: includes/skins/SkinTemplate.php

Unmerged paths:

(use "git add/rm <file>..." as appropriate to mark resolution)

deleted by us: resources/src/mediawiki.page.ready/ready.js

As **RMW8f033911030d26759c327145403f91ac6b1c5e66** in REL1_34 added "Turn logout to a POST action"... Can we just not apply the changes to `mediawiki.page.ready.js` but apply the two PHP ones (amend the commit summary too) and continue?

Ladsgroup added a comment.

2020-03-24 17:26:23 (UTC+0)

In **T232932#5995685**, @Reedy wrote:

Changes to be committed:

```
modified: includes/skins/BaseTemplate.php
modified: includes/skins/SkinTemplate.php

Unmerged paths:
(use "git add/rm <file>..." as appropriate to mark resolution)

deleted by us: resources/src/mediawiki.page.ready/ready.js
```

As `rMW8f033911030d26759c327145403f91ac6b1c5e66` in REL1_34 added "Turn logout to a POST action"... Can we just not apply the changes to `mediawiki.page.ready.js` but apply the two PHP ones (amend the commit summary too) and continue?

Yeah, this security issue has been introduced in by turning the log out button to a POST action which is rather recent (around a year now).

✔ Reedy closed this task as *Resolved*. 2020-03-24 17:32:12 (UTC+0)

Sorted then

✎ sbassett renamed this task from *User content can redirect the logout button to different URL* to *User content can redirect the logout button to different URL (CVE-2020-10959)*. 2020-03-26 03:40:07 (UTC+0)

🔔 sbassett mentioned this in ~~*T232932#5908474*~~ *Obtain CVEs for 1.34.7/1.35.0/1.34.1 security releases*. 2020-03-26 03:42:08 (UTC+0)

🔒 Reedy changed the visibility from **"Custom Policy"** to "Public (No Login Required)". 2020-03-26 17:41:58 (UTC+0)

🔒 Restricted Application changed the visibility from "Public (No Login Required)" to **"Custom Policy"**. · View Herald Transcript 2020-03-26 17:41:59 (UTC+0)

🔒 Restricted Application changed the edit policy from **"Custom Policy"** to **"Custom Policy"**. · View Herald Transcript

🔔 Reedy removed a project: **PermanentlyPrivate**. 2020-03-26 18:35:22 (UTC+0)

🔒 Reedy changed the visibility from **"Custom Policy"** to "Public (No Login Required)".

🔒 Reedy changed the edit policy from **"Custom Policy"** to "All Users".

👤 Krenair added a subscriber: **Krenair**. 2020-03-26 18:44:38 (UTC+0)

In ~~*T232932#5908474*~~, @sbassett wrote:

this issue is being held for the *next security release*, so please **keep this task private** for now (adding `🔒 PermanentlyPrivate` to be safe, until release)

Am I the only one confused about the use of PermanentlyPrivate here? It sounds like this was not permanently private, just private until the next security release. The visibility policy on the task already accounted for that.

💬 Ladsgroup added a comment. 2020-03-26 18:59:23 (UTC+0)

In ~~*T232932#6002996*~~, @Krenair wrote:

In ~~*T232932#5908474*~~, @sbassett wrote:

this issue is being held for the *next security release*, so please **keep this task private** for now (adding `🔒 PermanentlyPrivate` to be safe, until release)

Am I the only one confused about the use of PermanentlyPrivate here? It sounds like this was not permanently private, just private until the next security release. The visibility policy on the task already accounted for that.

There's a Herald rule that doesn't let users mistakenly "open" a ticket to public when it has `🔒 PermanentlyPrivate` tag on it. I think here it was used as more of a tool. Fixing it should not be that hard, create a tag like "StayPrivate" and use that instead (with the proper herald rules added)

💬 sbassett added a comment. 2020-03-26 19:10:44 (UTC+0)

In ~~*T232932#6003030*~~, @Ladsgroup wrote:

There's a Herald rule that doesn't let users mistakenly "open" a ticket to public when it has `🔒 PermanentlyPrivate` tag on it. I think here it was used as more of a tool. Fixing it should not be that hard, create a tag like "StayPrivate" and use that instead (with the proper herald rules added)

This is correct. Sorry for any confusion regarding my explanation above. If someone would like to file a bug to create a new "StayPrivate" tag, please feel free to do so.

🔔 ReleaseTaggerBot added projects: ~~*MMW-1.35-notes (1.35.0-mmfw.26, 2020-03-31)*~~, ~~*MMW-1.34-notes*~~. 2020-03-26 23:01:51 (UTC+0)

👤 RhinosF1 added a subscriber: **RhinosF1**. 2020-05-30 22:37:37 (UTC+0)

Is there any reason <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-10959> shows RESERVED?

This is public.

💬 Reedy added a comment. 2020-05-31 00:03:47 (UTC+0)

In ~~*T232932#6179379*~~, @RhinosF1 wrote:

Is there any reason <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-10959> shows RESERVED?

https://cve.mitre.org/about/faqs.html#reserved_signify_in_cve_entry

💬 RhinosF1 added a comment. 2020-05-31 06:48:47 (UTC+0)

In ~~*T232932#6179409*~~, @Reedy wrote:

In ~~*T232932#6179379*~~, @RhinosF1 wrote:

Is there any reason <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-10959> shows RESERVED?

https://cve.mitre.org/about/faqs.html#reserved_signify_in_cve_entry

Sorry, let me reword my question. Is there any reason the WMF have not yet asked for it to be populated or mitre are holding it?

I know why anything can be reserved, I meant this task specifically.


💬 sbassett added a comment. Edited · 2020-06-01 16:39:35 (UTC+0)

[@RhinosF1](#) - I can follow up with Mitre on this one, but note that the population of CVE data on their end is somewhat out of our hands. From the documentation [@Reedy](#) mentioned above:


"A CVE Entry can change from the RESERVED state to being populated at any time based on a number of factors both **internal** and **external** to the CVE List"

(emphasis mine)


Update: I just reached out to Mitre about updating the CVE description, etc. on their end. I'll post any relevant information here once I have it.

 [RhinosF1](#) added a comment. Edited · 2020-06-01 16:40:56 (UTC+0)


Thanks [@sbassett](#) , I just wondered as the last few times I've dealt with CVEs they've been deal with within around a day.

 [sbassett](#) added a comment. 2020-06-01 16:46:08 (UTC+0)

[@RhinosF1](#) - It depends. For issues that we fix and make public quickly, we can flip the acknowledge bit on Mitre's CVE request form and the CVE populates quickly on Mitre's end. For other issues (like this one) where we need to hold the patch for a quarterly security release, we can't really do that, and so sometimes the CVEs stay "RESERVED" for a long time.

 [RhinosF1](#) added a comment. 2020-06-01 16:46:44 (UTC+0)


Thanks for the clarification!

 [sbassett](#) added a comment. 2020-06-03 20:06:55 (UTC+0)

[@RhinosF1](#) -

The CVE appears to be populated on Mitre's end now: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-10959>

If, in the future, you notice any Wikimedia-related CVEs which still appear reserved and likely shouldn't be, feel free to comment on the relevant task or contact the [Security Team](#) via security-help@wikimedia.org. Thanks.

 [RhinosF1](#) added a comment. 2020-06-03 20:12:23 (UTC+0)

Thanks for the help, glad it's sorted!

[Log In to Comment](#)