Haketilo - Bug #65

When a site fails to load, for example due to its IP address not being found, the injected value with settings remains in the URL

07/26/2021 12:15 PM - koszko

Status:	Closed	**Start date:**	07/26/2021
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

History

#1 - 08/17/2021 07:41 PM - koszko

We should investigate if we can use Set-Cookie header instead of URL for policy smuggling

EDIT: Looks very promising. Under Chromium at least. Cookie value can be checked synchronously at document_start. What's more, deleting the cookie then prevents it from being sent to server in Coookie: request header in connections originating from this site.

#2 - 08/18/2021 01:13 AM - jahoti

Sounds like a winner (and much safer than dealing with the URL fragment)! That said, is there any way to deal with a page trying to set a cookie with the same key as we use?

#3 - 08/18/2021 06:10 PM - koszko

- % Done changed from 0 to 90

> Sounds like a winner (and much safer than dealing with the URL fragment)!

It is indeed way more convenient. Safer? Not sure. With the old method it was very unlikely the smuggled value would somehow leak to the server. The only bugs were related to convenience. Now there is real danger cookie will not get deleted for some reason and will get sent to server. Anyway, I think this approach is sufficient for now :)

> That said, is there any way to deal with a page trying to set a cookie with the same key as we use?

I made key contain the signature. Signature is derived from nonce, so it should be OK.

I also dropped all the conditionals that made code execute under Firefox only. The code works just as well when there's no header caching in place. Plus who knows when Chrome devs decide to do the same thing with caching? As long as header operations don't slow things down too much, it can be like this :)

It's on my branch for now. I am yet to test under Mozilla

EDIT: Newest commit on my branch restores compatibility with IceCat 60. Testing on other browsers still welcome :)

#4 - 08/20/2021 01:49 AM - jahoti

> Now there is real danger cookie will not get deleted for some reason and will get sent to server. Anyway, I think this approach is sufficient for now :)

Ah, good point- that will be something to look at eventually.

> I also dropped all the conditionals that made code execute under Firefox only. The code works just as well when there's no header caching in place. Plus who knows when Chrome devs decide to do the same thing with caching? As long as header operations don't slow things down too much, it can be like this :)

Indeed, and anything that reduces the amount of browser specific code is a positive :)!

> It's on my branch for now. I am yet to test under Mozilla
>
> EDIT: Newest commit on my branch restores compatibility with IceCat 60. Testing on other browsers still welcome :)

I'll have a look soon!

#5 - 08/20/2021 07:20 AM - jahoti

> EDIT: Newest commit on my branch restores compatibility with IceCat 60. Testing on other browsers still welcome :)

It most works on LibreWolf and Tor Browser too! However, there is a small bug in content/main.js; the nonce variable is no longer set, yet the inject_csp function still uses it (and it's still declared).

#6 - 08/20/2021 11:06 AM - koszko

Thanks for pointing out. I'll fix it together with some bigger changes for issue 15 https://hachettebugs.koszko.org/issues/15

#7 - 08/23/2021 11:17 AM - koszko

> Now there is real danger cookie will not get deleted for some reason and will get sent to server. Anyway, I think this approach is sufficient for now :)

Ah, good point- that will be something to look at eventually.

We could use webRequest to remove our cookies from request headers in case they happen to get there

#8 - 08/26/2021 03:55 PM - koszko

> We could use webRequest to remove our cookies from request headers in case they happen to get there

Committed to `koszko-smuggle-policy' branch

#9 - 09/04/2021 07:33 PM - koszko

- Status changed from New to Closed

- % Done changed from 90 to 100

Merged to master