

Haketilo - Support #95

Add JShelter in wiki: comparison with other extensions

10/01/2021 03:28 AM - Ogitnick

| | | | |
|---|----------|------------------------|------------|
| Status: | Rejected | Start date: | 10/01/2021 |
| Priority: | Normal | Due date: | |
| Assignee: | Ogitnick | % Done: | 70% |
| Category: | | Estimated time: | 0.00 hour |
| Target version: | | | |
| Description | | | |
| <p>The FSF just announced JShelter to combat nonfree JS: https://www.fsf.org/news/fsf-announces-jshelter-browser-add-on-to-combat-threats-from-nonfree-javascript</p> <p>It should be added to the 'comparison with other extensions' wiki page. It's probably a good idea to create a table with the extensions' names on the left and their functions at the top.</p> <p>Categories might be:</p> <ul style="list-style-type: none">• is a content blocker• enhances user freedom• enhances security/privacy• supports user script execution• supports firefox browser / forks• supports chrome browser / forks <p>I'm thinking there should be color classifications red yellow and green as well. LibreJS enhances user freedom, but does not enforce it as strongly as Haketilo. So LibreJS would be yellow in that category and Haketilo green. Haketilo would be red in the category 'doesn't break websites'. LibreJS would be yellow. UBO might be green.</p> <p>This could help people quickly grasp what Haketilo is about, by comparison.</p> | | | |

History

#1 - 10/01/2021 04:14 AM - jahoti

The table is a great idea! Just to make sure you're aware, however, it will need to be an image or at least use images for the cell contents (the wiki doesn't allow HTML tags).

#2 - 10/01/2021 12:12 PM - kozzko

The table is a great idea!

Indeed, a brilliant one :D

We could even have separate rows for "Haketilo (with scripts globally enabled)" and "Haketilo (with scripts globally disabled)" (and analogously with other extensions that can block scripts)

#3 - 10/03/2021 07:40 AM - Ogitnick

- File *chart.png* added
- Status changed from *New* to *In Progress*
- % Done changed from 0 to 30

Thoughts/comments/critiques on the attached chart?

koszko wrote:

We could even have separate rows for "Haketilo (with scripts globally enabled)" and "Haketilo (with scripts globally disabled)" (and analogously with other extensions that can block scripts)

I'll add that for Haketilo since it has a strong use case even without script blocking (custom site features/content). I don't think the other content blockers have a use case:

- LibreJS - with global script blocking disabled is useless.
- JSshelter - is better thought of as an API firewall not a content blocker. I'll correct that on the chart as well.
- Noscript - without script blocking is just anti-XSS. Not much.
- uBlock Origin - I don't think there's an option to disable script blocking globally. What would be the point?

#4 - 10/04/2021 05:14 AM - jahoti

Thoughts/comments/critiques on the attached chart?
Nice! I like it.

On a tangential note, we probably also need a non-visual version of the format to link to. Is there a good format for that?

uBlock Origin - I don't think there's an option to disable script blocking globally. What would be the point?

There is an option to *enable* script blocking globally. Inasmuch as uBlock Origin will (by default) allow scripts not blocked by a specific rule, it's arguable that the situation is just the reverse of Haketilo's.

Side note: now I see why uBlock Origin was/is one of the role models for Haketilo.

#5 - 10/05/2021 09:47 AM - koszko

How exactly do we define the "enhances user freedom" category?

How about we make "security" and "privacy" separate categories?

BTW, it seems like extensions are difficult to rate objectively :/

Ogitnick wrote:

I'll add that for Haketilo since it has a strong use case even without script blocking (custom site features/content). I don't think the other content blockers have a use case:

[...]

- Noscript - without script blocking is just anti-XSS. Not much.

Fair point. In that case we could just state the assumptions we make (e.g. that the user has scripts globally disabled in NoScript and enables them manually on certain sites). I think footnotes below the table would be a suitable place.

Btw, I patched `/usr/share/redmine/lib/redmine/wiki_formatting/macros.rb` on hydrillabugs to allow colors. I added the following to the macros there (I don't know Ruby btw):

```
desc "Insert color" + "\n\n" +
  "  {{color(FG_COLOR, BG_COLOR, TEXT)}}\n\n" +
  "Where FG_COLOR and BG_COLOR are given in hex as #xxx or #xxxxxx."
macro :color do |obj, args|
  out = ''.html_safe
  color_reg = /^#[0-9a-zA-Z]{3}|[0-9a-zA-Z]{6}$/
  if color_reg === args[0] && color_reg === args[1]
    out << content_tag('div', textilizable(args[2], :object => obj, :headings => false), :style =>
"color: #{args[0]}; background-color: #{args[1]}")
  else
    raise "Color is not of format #xxx nor #xxxxxx"
  end
  out
end
```

Use like this:

```
| Item          | {{color(#beb, #405040, *Prices*)}} | # In stock |
|-----|-----|-----|
| Juicy Apples | 1.99          | *7*        |
| Bananas      | **1.89**     | 5234       |
```

Result:

| Item | Prices | # In stock |
|--------------|--------|------------|
| Juicy Apples | 1.99 | 7 |
| Bananas | 1.89 | 5234 |

Table cells do retain non-colored padding but I don't think this is an issue worth spending more time on right now.

What I did is also safe with regard to XSS via `<script>` tags (unlike other solutions proposed on Redmine wikia...).

This:

```
{{color(#beb, #405040, <script>console.log("buuuu");</script>)}}
```

yields:

```
console.log("bUUUU");
```

#6 - 10/07/2021 06:58 AM - jahoti

Btw, I patched /usr/share/redmine/lib/redmine/wiki_formatting/macros.rb on hydrillabugs to allow colors. I added the following to the macros there (I don't know Ruby btw):

That will be really useful- thanks! I'll try configuring a version of the table on a wiki page now; however, if you're OK with it Nick, I think it would be good to still maintain a (possibly condensed) version as a graphic too. The style is much more aesthetically pleasing and slightly easier to read, which would make it ideal for a quick, clear explanation.

(Also, how did you put that together without knowing Ruby? Was there an existing solution that was XSS-safe and you then modified, or did you literally hack and RTFM your way through the entire thing?)

BTW, it seems like extensions are difficult to rate objectively :/

Very, very true. This is going to need explanations of each rating beneath the table for it to be properly informative.

#7 - 10/08/2021 10:07 AM - kozzko

(Also, how did you put that together without knowing Ruby? Was there an existing solution that was XSS-safe and you then modified, or did you literally hack and RTFM your way through the entire thing?)

I found a solution with an XSS-unsafe macro. I then looked at other macros in that file and based on that figured out how to write the one I need. For things like regex usage in Ruby, I searched online.

At the university we were often, without any earlier announcement, required to do a lab class using a language we've never learned before. This

happened with at least JS, C#, Julia and Python. It turns out that once you know a few languages, making a hack in another somewhat similar one is not that hard.

It would be more difficult with an unusual language, of course (e.g. Haskell, APL)

#8 - 10/11/2021 10:51 PM - Ogitnick

- *File chart.png added*

- *File table.htm added*

- *% Done changed from 30 to 60*

I've attached an update version of the chart with assumptions listed and footnotes. I've also included the html file just because it's easier to edit that way. Let me know if anything needs to be changed.

How exactly do we define the "enhances user freedom" category?

The user freedom column literally means "how good it is at blocking non-free JS", even if the extension isn't explicitly targeting non-free JS.

How about we make "security" and "privacy" separate categories?

Done.

BTW, it seems like extensions are difficult to rate objectively :/

Absolutely. It's not currently based on any quantitative data. Do you think it's necessary to test each extension on the top websites to see how they fare in each category? I already did this, but not methodically.

By the way, I excluded Haketilo+LibreJS for now because we haven't decided how that's going to work yet. When that gets ironed out, the chart should be updated. I will convert the chart to the redmine macro as soon as we get a finalized version everybody agrees on.

#9 - 10/12/2021 09:34 AM - koszko

Ogitnick wrote:

Let me know if anything needs to be changed.

Here we go

How exactly do we define the "enhances user freedom" category?

The user freedom column literally means "how good it is at blocking non-free JS", even if the extension isn't explicitly targeting non-free JS.

This explanation will go into footnotes, am I correct?

This one also seems to strongly depend on the user. One will enable scripts on every site that doesn't display properly while other will only do so on sites that are known to be servig free scripts (because, for example, they are instances of libre software like Etherpad). In the past I happened to use NoScript this way. Is there any way we could indicate some ratings also depend on user's skill and determination?

```
Hydrilla, the server-side counterpart to Haketilo, can still log which websites you request replacement JS for.
```

Won't some people interpret that Hydrilla snoops on users? Well, on api-demo.hachette-hydrilla.org requests are indeed being logged... by Apache's default logging facility. So it's true that it logs something. How about:

```
Administrator of a Hydrilla instance would still be able to look into web server logs to see which websites an IP address requested replacements for.
```

Btw, we can and should modify Haketilo to query Hydrilla with somewhat anonymized headers (no user-agent string, etc.)

```
For uBlock Origin, the user does not modify most of the default settings.
```

I think it would be worth to also show that uBO is pretty good when the user changes the settings to block JS globally. Actually, out of all extensions I tried (and I am sure uMatrix is better but I haven't tried it yet), uBO gives the finest control over resources loaded by sites. It allows applying different policies to different domains and also blocking of media elements and **fonts**. It is even possible to treat inline and external JS differently. I could say it replaces Third Party Request Blocker but it's actually better than that.

BTW, it seems like extensions are difficult to rate objectively :/

Absolutely. It's not currently based on any quantitative data. Do you think it's necessary to test each extension on the top websites to see how they fare in each category? I already did this, but not methodically.

Actually, the more I think about it, the more subjective our rating seems to me. But the table idea too cool and already looks to good to give up on it. How about instead of rating whether an extension has certain quality under some assumptions, we rate whether it has some (more strictly defined) features. Instead of "user freedom" we could list potential features like "can block (nonfree) javascript" and "blocks (nonfree) javascript under default settings". That would give even more information to the user and it would be easier to rate

I've also included the html file just because it's easier to edit that way.

I honestly thought you created this with an image editor or something- it's turned out very nicely for HTML!

Do you think it's necessary to test each extension on the top websites to see how they fare in each category? I already did this, but not methodically.

That would probably be a good way to do it. However, we should add in ten (somehow) "randomly selected" websites too; the top websites will probably be unusually JS-heavy and better tested with most extensions.

By the way, I excluded Haketilo+LibreJS for now because we haven't decided how that's going to work yet. When that gets ironed out, the chart should be updated. I will convert the chart to the redmine macro as soon as we get a finalized version everybody agrees on.

Sounds good!

["User freedom"] also seems to strongly depend on the user. One will enable scripts on every site that doesn't display properly while other will only do so on sites that are known to be servig free scripts (because, for example, they are instances of libre software like Etherpad). In the past I happened to use NoScript this way. Is there any way we could indicate some ratings also depend on user's skill and determination?

That's a very good point, applicable to other columns as well. Maybe we need two tables, one for users prioritizing site compatability and one for users prioritizing freedom, security and privacy. (Arguably that would also remove the need for Haketilo to have two entries.)

Hydrilla, the server-side counterpart to Haketilo, can still log which websites you request replacement JS for.

Won't some people interpret that Hydrilla snoops on users?

Perhaps; it seems reasonably safe to me with "can" in there. We could also write a privacy policy and then link that sentence to the relevant section of it (once this is in Markdown), which would make it possible for anyone who's concerned to learn more without putting too much information in the table.

I think it would be worth to also show that uBO is pretty good when the user changes the settings to block JS globally. Actually, out of all extensions I tried (and I am sure uMatrix is better but I haven't tried it yet), uBO gives the finest control over resources loaded by sites. It allows applying different policies to different domains and also blocking of media elements and fonts. It is even possible to treat inline and external JS differently. I could say it replaces Third Party Request Blocker but it's actually better than that.

If it weren't so much work to modify the UI (and add support for URL patterns), I would honestly suggest we rebuild Haketilo on uBlock Origin- it's hard to imagine a better blocker.

But the table idea too cool and already looks to good to give up on it. How about instead of rating whether an extension has certain quality under some assumptions, we rate whether it has some (more strictly defined) features. Instead of "user freedom" we could list potential features like "can block (nonfree) javascript" and "blocks (nonfree) javascript under default settings". That would give even more information to the user and it would be easier to rate

While I completely agree, there is a trade-off with readability: having longer column headings, potentially more of them, and only two colors (for "yes" or "no") would make it difficult to get useful information at a glance.

Maybe we could have two sets of tables: one would simply display these objective factors, while the other would condense them into categories more

like we have now. This could be done with a point-scoring system: each category gets a score based on adding up the number of relevant "objective factors" it has and then dividing by the maximum possible score. Depending on that it either gets "weak", "fair", or "strong" (the text would obviously be modified if something else is more suitable for a given entry).

#11 - 10/13/2021 12:20 AM - koszko

If it weren't so much work to modify the UI (and add support for URL patterns), I would honestly suggest we rebuild Haketilo on uBlock Origin- it's hard to imagine a better blocker.

I have to admit I thought about the same and reached the same conclusion ;)

How about instead of rating whether an extension has certain quality under some assumptions, we rate whether it has some (more strictly defined) features. [...]

While I completely agree, there is a trade-off with readability: having longer column headings, potentially more of them, [...] would make it difficult to get useful information at a glance.

Indeed, I considered transposing the table

Maybe we could have two sets of tables: one would simply display these objective factors, while the other would condense them into categories more like we have now. This could be done with a point-scoring system: each category gets a score based on adding up the number of relevant "objective factors" it has and then dividing by the maximum possible score. Depending on that it either gets "weak", "fair", or "strong" (the text would obviously be modified if something else is more suitable for a given entry).

Looks like you've found a way to rate extensions more objectively 😊

#12 - 10/13/2021 06:39 AM - jahoti

Looks like you've found a way to rate extensions more objectively ⁽⁰⁾

If we can determine a suitable list of factors: at least there's two already ("can block (nonfree) javascript" and "blocks (nonfree) javascript under default settings")!

#13 - 10/20/2021 01:04 AM - Ogitnick

- File *haketilo_table.htm* added

jahoti wrote:

If we can determine a suitable list of factors: at least there's two already ("can block (nonfree) javascript" and "blocks (nonfree) javascript under default settings")!

I haven't been able to come up with objective criteria for freedom/security/privacy. I attached a chart for the objective criteria but I'm very unsatisfied with it. Freedom seems the easiest to isolate, but I think both the objective categories you already suggested could be misleading.

For example UBO can block nonfree JS and does under default settings, but it has nothing to do with freedom. While your categories would be objectively correct, we don't want to mislead users that UBO cares about nonfree JS specifically.

There's not much substantive we can put on the objective chart without having lots of caveats. For the data chart I can't think of a good quantitative way to compare privacy, security and freedom for each extension.

There are certain scripts we can positively identify as privacy-invading. That could be quantitative but then how do we classify JSshelter? And how can we quantify something like site compatibility? That would have to be subjective. Perhaps it would be better to scrap the charts and just expand on the wiki?

#14 - 10/20/2021 10:36 AM - koszko

Ogitnick wrote:

I haven't been able to come up with objective criteria for freedom/security/privacy. I attached a chart for the objective criteria but I'm very unsatisfied with it. Freedom seems the easiest to isolate, but I think both the objective categories you already suggested could be misleading.

For example UBO can block nonfree JS and does under default settings

Does it? I remember having to toggle the switch to block JS in it.

but it has nothing to do with freedom

I admit uBO was not created with the goal of increasing software freedom as such. Despite that, I find it very useful for this purpose. There are many sites where I know scripts are (almost certainly) free (e.g. openstreetmap or Jitsi) and I would (sometimes, not on everyday basis) allow these sites in uBO while still blocking the rest.

While your categories would be objectively correct, we don't want to mislead users that UBO cares about nonfree JS specifically.

There's not much substantive we can put on the objective chart without having lots of caveats.

To avoid misleading users we could add a category like "targets nonfree JS specifically". I thought this would make things the most clear.

Also, I imagined there would be many categories (tens?).

For the data chart I can't think of a good quantitative way to compare privacy, security and freedom for each extension.

There are certain scripts we can positively identify as privacy-invading. That could be quantitative but then how do we classify JSshelter?

It would have "no" in categories related to script blocking and "yes" in privacy-related categories like "can prevent geolocation" or "can prevent fingerprinting by device memory size". Blockers like NS and uBO could have something like "when scripts are disabled" there.

And how can we quantify something like site compatibility? That would have to be subjective. Perhaps it would be better to scrap the charts and just expand on the wiki?

We could select a subset of most popular websites and make compatibility with each of them a separate category.

Elaborating more in terms of simple text is not a bad idea, though

#15 - 10/23/2021 05:57 AM - jahoti

Ogitnick wrote:

I haven't been able to come up with objective criteria for freedom/security/privacy. I attached a chart for the objective criteria but I'm very unsatisfied with it. Freedom seems the easiest to isolate, but I think both the objective categories you already suggested could be misleading.

You're right, actually- they probably aren't very helpful to use, nor or some other possible categories. However, having even an imperfect table would still be very useful IMO (certainly *don't* feel as though you have to work on it, or to do so exclusively!).

Your categories seem like a good start, Wojtek- in fact, perhaps they could be split even more finely into "does \$TASK by default", "(usually) does \$TASK when configured to block malware", and "(usually) does \$TASK when configured to minimize website breakage", which would remove the need to give some extensions multiple columns.

While the scheme (with or without splitting) will likely require the table to be transposed and make it very lengthy, a summary table can always be produced using more subjective categories derived from the full list.

#16 - 03/01/2022 08:13 PM - Ogitnick

- % Done changed from 60 to 70

I finally added JShelter to the wiki.

#17 - 10/13/2022 12:01 PM - koszko

- Status changed from In Progress to Rejected

Files

| | | | |
|--------------------|---------|------------|----------|
| chart.png | 44.9 KB | 10/03/2021 | Ogitnick |
| chart.png | 138 KB | 10/11/2021 | Ogitnick |
| table.htm | 6.52 KB | 10/11/2021 | Ogitnick |
| haketilo_table.htm | 4.2 KB | 10/19/2021 | Ogitnick |