What is the difference between ad blocker dataset and tracker blocker dataset.#86

I found a very interesting article https://www.sba-research.org/wp-content/uploads/publications/block_me_if_you_can.pdf

Summary of Article

- This looks at the links between ad blocking and tracker blocking on the basis of overall privacy and security of Internet users and which tools are the most effective.
- It looks at the tools that internet users use to block adds and to block tracking.
- come from the browser extensions which differentiate between tracking and non-tracking HTTP requests. This will block the browser extensions and allow the non-tracking HTTP requests.

Blocking Trackers

We know that network-based blocking is done through DNS blocking where addresses are blacklisted in order to block access to certain domains. This will include ads and this method works independently of the used application. The DNS filtering can be used to block entire domains but not individual URIs.

When analyzing the data set, in relation ***to #24 – How does ad blocking relate to this dataset***, we are sure to find entire domains being blocked because of the use of DNS blocking, and an analysis of how to factor this into the code to build a heuristic of browser attributes should be considered.

With Blocking Trackers and Interception Proxies, third party content cannot always be reliably detected at a network level, which might skew the results when looking at ad blocking in a dataset.

Interception Proxies

Privacy is enhanced through the use of interception proxies. The article looks at *Privoxy* which has a URI-based filtering capability that can modify the content and headers of web requests allowing it to be used to remove individual cookies and block certain URIs and tracking code from the web pages.

With the use of proxies, we should look to see which method of proxies are being used as this will show how fingerprinting is detected and how far it goes into tracking a user. Because interception proxies cannot modify or intercept HTTP traffic, it is likely that we will still see the fingerprint of the user even if a proxy has been used, thus again, I think that this will be good to factor into code to build a heuristic of browser attributes.

Browser Extensions

Browser extensions are known to be reliably and easier to detect third party content including encrypted web traffic.

AdBlock Plus for example, focuses on blocking online advertisement, which ties into **topic #24** where we are seeking to establish how ad blocking relates to the dataset.

Tracker blockers will focus on blocking trackers through extensions. With *Ghostery* as one of the most popular extensions, it provides feedback on which third party trackers are included in each visited website. Specific rules for tracker blocking will include third-party domains, specific URIS and surrogates, such as the Like button on Facebook or Twitter. If you were to compare this to the ad blocker, blocking advertising does not stop a user being tracked as advertisers can still track interests through behavior profiles. This is another important factor to include in any code.

The use of general blocking filters and CSS filters differ significantly. When analyzing a dataset, the use of regular expressions in filters will affect the potential performance impact of a code designed to track users. This is because often the address changes every time you open a page. For example http://example.co.uk/ads/banners49669.gif, has **49669** as a random number and if we were to block the complete address, it would not help as a more general filter will be needed without the random number, for example http://example.co.uk/ads/banner*.gif or http://example.com/ads/*. It would be therefore better to use exceptions so that not all banners or ads re blocked.

Using regular expression to identify fingerprints can be done through specific URIs. This will show the effectiveness of existing browser extension to prohibit the execution of well-known and recently identified fingerprint scripts. If we are to use this method in the dataset, I believe we would be able to establish which fingerprint scripts to include into building a heuristic for browser attribute fingerprinting.

Blocking fingerprinting appears to be more difficult than blocking browser extensions because of the use of exchange of tracking information over unencrypted HTTP connectors.

If we were to consider our current tracker blocker dataset, the browser used will be the easiest to identify and track fingerprinting as they are uniquely identified. Factoring in other aspects such as ad blockers will look into specific filters that we must examine and analyze when creating a code.