POLITECNICO DI TORINO
Repository ISTITUZIONALE

Benchmark and comparison of tracker-blockers: Should you trust them?

Original
Benchmark and comparison of tracker-blockers: Should you trust them? / Traverso, Stefano; Trevisan, Martino; Giannantoni, Laura; Mellia, Marco; Metwalley, Hassan. - ELETTRONICO. - (2017), pp. 1-9. (Intervento presentato al convegno IEEE/IFIP Network Traffic Measurement and Analysis Conference (TMA) tenutosi a Dublin, IE nel 21-23 June 2017) [10.23919/TMA.2017.8002898].

Availability:
This version is available at: 11583/2679579 since: 2017-09-18T11:30:52Z

Publisher:
IEEE

Published
DOI:10.23919/TMA.2017.8002898

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

26 September 2024

Benchmark and Comparison of Tracker-blockers: Should You Trust Them?

Stefano Traverso[†⋆] Martino Trevisan[†] Leonardo Giannantoni[†] Marco Mellia[†⋆] Hassan Metwalley[†⋆]
[†]Department of Electronics and Telecommunications, Politecnico di Torino, Italy: e-mail: {first.last}@polito.it
[⋆]Ermes Cyber Security SRL

***Abstract*—People are getting more and more conscious and worried about privacy issues that arise when browsing the Web. Ad-blockers, anti-tracking extensions, privacy and anonymity plug-ins, etc. promise to protect users and their privacy from third-party tracking systems. But how effective are they? In this paper, we present the first experimental campaign aimed at benchmarking popular plug-ins for web privacy preservation to date. We select 7 different plug-ins and setup a testbed to automatically browse regular web pages, while collecting navigation data. We analyze this data to compare each plug-in, considering both privacy-protection and performance angles. Our results show that the picture is very variable, with no plug-in being able to guarantee complete protection while improving performance as promised.**

By considering different experimental setups, we also observe that the European ePrivacy Directive is ignored by the majority of considered web sites. The directive prevents web services from installing tracking and profiling cookies before explicit consent is given by the user, but apparently this is not observed for most of services.

To favor reproducibility, and repeatability, we share both the software and the data used to conduct this study with the community. Our aim is to let researchers and developers better understand the privacy threats in the Internet, possibly toward better performing privacy-preserving tools.

I. Introduction

When connected to the Internet to browse the web, users contact servers to fetch web pages, some of which contain images, videos, advertisement, etc., while others collect data on web page performance and users' browsing habits. The latter, the so-called "third-party trackers", may represent a serious threat to users' privacy. Some of these systems build their business on the massive collection and brokerage of personal data. Born to offer personalized advertisement, they track users across different web pages, and build profiles to be sold to other parties. Because of the lack of a coordinated and comprehensive regulation, trackers shadow users across web sites with practically no limits. Thousands of services are known to behave as tracking systems, but unfortunately it is hard to obtain an exhaustive list because of their hidden nature.

Many surveys report an increasing worry from users about their online privacy [1], [2]. This worry has translated in an actual demand for tools capable of protecting privacy during browsing. In fact, the most recent years have witnessed a proliferation of tracker-blockers. Ghostery [3], uBlock [4] and Blur [5] are among the most prominent ones. A recent study has estimated that these three together account for 23M users [6]. In parallel, other systems emerged to block advertisement content, with Adblock Plus [7] being among the most popular ones, and installed by about 20% of Internet users in their browsers [8], [9].[1]

Despite their momentum, little is known about tracker-blockers and their effectiveness. The research community has spent a significant effort in studying countermeasures to detect and defeat trackers (see Sec. VI for a detailed discussion), but - to the best of our knowledge - no study has focused on systematically benchmarking tracker-blocker effectiveness. Filling this gap is very important, also to check for questionable policies adopted by some of these systems [10], [11].

In this paper we build a testbed to systematically benchmark and compare seven popular freeware tracker-blockers. Our goal is to simulate the usage of the average Internet user, who installs one of these plug-ins, and enjoys the protection it offers. To this end, we design a custom tool that uses active measurements for our benchmark. Given the dynamic nature of web pages, and the complex relationships of objects they include, designing such a tool requires some ingenuity. Each page must be visited several times to ensure statistical significance in data, and the tool must be able to handle unpredictable events such as page timeouts and crashes which may halt the browsing emulation. In turn, this inflates the testing time, therefore a good balance must be considered.

We use the data we collect to assess the effectiveness of each plug-in to preserve users' privacy. We count the set of services contacted by the browser that are included in a superset of trackers we build from several sources. From a different angle, we verify the claims about the ability to improve the Quality of Experience (QoE), e.g., to speed up the web page loading time, and to reduce bandwidth usage ([12]).

In addition, we run experiments to observe the impact of the "Cookie Policy" notification and acceptance banner web sites must present when accessed by a user for the first time. This is imposed by the ePrivacy Directive of European Commission [13], [14]. In a nutshell, the law imposes the web site to ask the explicit user consent before installing any tracking and profiling cookies (or similar mechanisms) and before contacting any third-party service which uses persistent cookies (or other tracking mechanisms). Thus, we expect the second-visit to the same web site to be very different if the

[1] Ad-blockers and tracker-blockers aims at blocking two different type of services, even if this distinction is blurred for not expert users.

user provides authorization to the usage of third-party (TP) cookies.

Our experiments show surprising and unforeseen results. First, there is a large variance in the effectiveness of each plug-in. The only one offering a complete protection is Request Policy [15]. Unfortunately, it blocks *all* third-party content, and thus it breaks the web-page rendering. In our tests, the least effective is Privacy Badger [16], supported by the Electronic Frontier Foundation (EFF). Unfortunately, its internal algorithm used to identify trackers results ineffective with fresh browser installation. Differently, Ghostery [3] offers the best protection. uBlock [4], Disconnect [17] and Blur [5] provide good protection too, but surprisingly, they fail to block some very popular trackers.

Considering page loading performance, we observe that data being downloaded with any plug-in typically decreases due to less content being fetched by the browser. Despite this, the page load time may increase (Privacy Badger) or decrease (uBlock, Disconnect). This is due to the different anti-tracking approaches, and to the additional complexity of executing the plug-in code.

Finally, the ePrivacy Directive is mostly ignored by web designers, so that tracking systems get contacted before the user has accepted the cookie policy.

We believe the results presented in this paper are useful for the average Internet user to make an informed choice on tracker-blocker. To help the community in offering and updating independent, and scientifically sound experiments, we make available all software and data we used to conduct this study.

The rest of the paper is organized as follows. Sec. II provides background about web tracking and tracker-blockers. Sec. III describes the benchmark and datasets we collect. Sec. IV presents the metrics and Sec. V shows the results. Sec. VII describes the limitations of our methodology before discussing related work in Sec. VI. Finally, Sec. VIII concludes the paper.

II. Background

Online tracking. During their browsing, users are observed by both 'first parties', the sites explicitly visited, and 'third parties', support services that offload the main site and serve additional objects, code, images, etc. Among these, trackers are usually invisible services embedded in web pages by webmasters that would like to monetize the content they offer via, e.g., personalized advertisement. Trackers re-build users' browsing histories by employing several tracking technologies (e.g., cookies, super-cookies, fingerprinting [18], [19]) that let them uniquely identify users across different web sites. Combined with the "referer" in the HTTP header, they get the browsing history of the users. From the browsing habits, they build a profile for each user, that they sell to advertisers via online auctions [20] (e.g., for personalized advertisement).

TABLE I
Plug-ins considered in this study. Usage statistics provided by Mozilla Add-ons website.

Plug-in	Approach	Blockage	Firefox Users
Ghostery	Domain-based blacklist	Trackers	1.3M
Disconnect [2]	Domain-based blacklist	Trackers	0.3M
Blur [2]	Unknown	Trackers	0.16M
uBlock	Regexp-based blacklist	Trackers	3M
Privacy Badger	Behavior-based	Trackers	0.1M
Adblock Plus	Regexp-based blacklist	Ads and Trackers	19M
Request Policy	Cross-site-based	All third parties	0.07M

Tracking blocking. Trackers represent a menace for user privacy, thus, tracker-blockers were born to help users preserve their privacy during navigation. In most of the cases, tracker-blockers are deployed as browser plug-ins (also called extensions).

For our experiments we consider the seven most popular freeware plug-ins which offer tracker-blocking features. We report them in Table I, where we classify them based on the approach they use to block trackers, and the third-party services they aim to block. Some of them target advertisements, or/and trackers. Request Policy blocks all third-parties indiscriminately. Blur [5] does not provide information about the mechanisms to detect and stop trackers. In general, they inspect HTML code, and prevent the browser from visiting URLs headed to tracking services. To achieve this goal, they have to detect which URLs in a page refer to trackers. Ghostery [3] and Disconnect [17] leverage blacklists containing *domains* of services they classify as trackers. Adblock Plus [7] and uBlock [4] use *regular expressions* to match against the URLs to contact. Whenever the match is positive the request is blocked. Other tools act differently. Privacy Badger [16] is an open-source tool provided by the Electronic Frontier Foundation (EFF) that blocks more generically objectionable behaviors: if it detects the same third-party domain tracking the user across different sites, it blocks it. Hence, the same tracking domain can be blocked or not, depending, e.g., on the number of times the user meets it during its browsing trajectory. Request Policy [15] is much more severe. In fact, it prevents the browser to open cross-site connections to any third-party, independently from its nature or activity. This preserves users' from contacting third-party services, but dramatically hampers QoE as it blocks the delivery of objects, e.g., images, fonts, css, etc, from any third-party domain.

The second action tracker-blockers have to perform is the actual blockage of URL access. To this end, browsers make some APIs available. For instance, Mozilla Web API's `http-on-modify-request` is an event handler which allows to modify or cancel HTTP requests before they are sent.

Cookie Regulation. The first EU's attempt to provide guidelines about data protection and privacy in the digital age is 2002 E-Privacy Directive [13], namely 2002/58/EC. It specifically deals with the regulation of treatment of traffic data and cookies. It was then amended by directive 2009/136 [14], which made cookies subject to prior consent. In other words,

[2] We consider the free version of this plug-in.

since 2009 webmasters are compelled to ask users explicit consent before installing cookies on users' devices. This holds in particular for those cookies which are used to profile users, i.e., persistent in time and managed by a third party. Conversely, the directive states webmasters are not obliged to warn users about the usage of technical cookies, which typically expire at session end. Given this, we expect a user to download no persistent third-party cookies when she visits web sites for the first time.

III. Benchmark Definition and Dataset

In this section we describe the methodology for the data collection, and the datasets we obtain and use to compare the performance of tracker-blocker plug-ins.

A. Testbed setup

We use active measurements to setup and run our benchmark. The platform builds on automatically visiting a predefined set of web pages. We use Selenium [21] and the Mozilla Firefox browser configured to visit URLs and dump statistics via HAR (HTTP Archive [22]) files. In a nutshell, given a *set of pages* to visit, and a *set of profiles*, Selenium loads the profile, runs Firefox, lets it visit each page, and waits for the browser to return with the `OnLoad` event. If the event is not triggered within a timeout of 100s, we assume there has been some technical issue, and discard the visit. Between consecutive visits we insert an inactivity period of 6s. At the end of each visit, we extract the HAR from the navigation data generated by the browser. The HAR is a JSON-formatted container for recording HTTP(S) tracing information. It contains an entry for each object requested by a web page. This entry includes information such as timings (e.g., time to fetch DNS information, get a URL) and statistics about content (e.g., size, download time). We take care of erasing the browser cache after each visit. Each page is visited 10 times to increase experiments' reliability.

B. Measurement data collection

In this paper, we consider a scenario in which a user is browsing the web from her PC. We define the set of pages to visit by including 100 popular web sites. In more details, we consider 10 categories of web pages, and, for each category, we arbitrarily pick 10 distinct Italian popular sites. In particular, the first returned by Google Search for each category. We report the entire list of web pages, grouped by category in Table II.[3]

As set of profiles, we first build a baseline with no plug-in installed, that we call Plain in the remainder of the paper. Then, for each tracker-blocker, we create a fresh Firefox profile in which we manually install the corresponding plug-in. Thus, in total we obtain 8 different browser profiles. We install each plug-in from official Mozilla Firefox add-on page. We use the default configuration for all of them, except for Ghostery. Surprisingly, we discovered that by default Ghostery does not enable any filtering capability. Instead, it requires the user to i) create a Ghostery profile, ii) login to the system, iii) select advanced preferences, and iv) turn on protection for all sites. Without this cumbersome process Ghostery provides no protection.

To observe the impact of the EU ePrivacy Directive on cookies, we create two browser settings (thus doubling the number of profiles): in the first one we do not provide consent to third-party (TP) cookies, so that each visit we perform corresponds to a "first visit"; for the second one, we manually visited all the pages, and explicitly clicked on the "Accept Cookie" banner, when available. That is, the browser eventually accepts any cookie for each first- and third-party domains being visited, and the visit to the page would hence correspond to a "second visit". At the end, we erase the browser cache, but retain the cookie database. In total, we obtain 8×2 profiles, and 100 pages, that we visit 10 times each. Web pages can be very dynamic and change their content frequently during the day (e.g., news portals). Hence, we carefully design our experiments so that the same web page is visited by different profiles in a short time (a few minutes), thus maximizing the probability to encounter the same contents. For the same reason, we run two parallel experiments using two identical machines. The first for setup without consent to TP cookies and the second for setup with consent to TP cookies.

We use Linux-based Intel Core 2 Quad machines equipped with 6GB RAM, connected to the Internet through a 1Gb/s network, and using public IP addresses. In total, data collection lasted 180 hours approximately. At the end, $16,000$ HAR files have been collected, which account for about 15GB of data to process.[4]

C. List of trackers

To understand how effective plug-ins are at blocking connections with trackers, we extract from the corresponding HAR file the URLs contacted for downloading all objects for the page rendering. For each URL, we extract the second-level domain names, and we mark as *third-party* all URLs whose second-level domain does not match with the one of the visited page. We compare this approach with the one proposed in [23], which builds on domains' ADNS (Authoritative DNS) server to identify third parties, and we observe negligible differences for the considered set of domains.

We next label those third-party domains which correspond to trackers. Given the hidden nature of tracking systems, no ground truth is available. We manually build a super-set of regular expressions we obtain by joining blacklists from multiple sources: Ghostery, EasyPrivacy [24], Disconnect [25] and Princeton Web Census [26].[5] Similarly to [27], [9], we match each third-party domain from HAR files against the super-set

[3] We explicitly avoid using the Alexa ranking since it includes services which are questionable for some categories.

[4] All software and processed measurement are available for download at https://bitbucket.org/LGiannantoni/web-privacy-protection-systems. Given its size, we share the HAR collection on demand.

[5] Ghostery's blacklist has been extracted from the Firefox plug-in code directly.

News	Sport	Weather Forecast	E-Commerce	Forums
www.corriere.it	www.calcioinrosa.it	www.meteoitalia.it	www.trovaprezzi.it	www.fotopratica.it
www.repubblica.it	www.calciomercato.com	www.ilmeteo.it	www.amazon.it	www.clickblog.it
www.rainews.it	www.corrieredellosport.it	www.tempoitalia.it	www.ebay.it	www.lightroomcafe.it
www.ansa.it	www.fantagazzetta.com	www.meteo.it	www.glistockisti.it	www.zmphoto.it
www.huffingtonpost.it	www.figc.it	www.centrometeoitaliano.it	www.monclick.it	www.photo4u.it
www.oggi.it	www.gazzetta.it	www.nimbus.it	www.redcoon.it	www.pentaxiani.it
www.news.google.it	www.milannews.it	www.meteogiornale.it	www.subito.it	www.dphoto.it
www.tgcom24.mediaset.it	www.pianetamilan.it	www.datameteo.com	www.kijiji.it	www.maxartis.it
www.tg24.sky.it	www.raisport.rai.it	www.meteoconsult.it	www.kelkoo.it	www.nikonclub.it
www.panorama.it	www.sportmediaset.mediaset.it	www.meteogiuliacci.it	www.twenga.it	www.photographers.it

Games	Technology	Search Engines	Hobbies	Motors
www.spaziogames.it	www.punto-informatico.it	www.google.it	www.creazioni-or.it	www.autoscout24.it
www.gamesvillage.it	www.wired.it	duckduckgo.com	www.ideeperhobby.it	www.automobile.it
www.gamespot.com	www.hdblog.it	search.yahoo.com	www.fabiolamarchet.it	www.quattroruote.it
giochi-mmo.it	www.zeusnews.it	www.bing.com	www.fantasyehobby.it	annunci.quattroruote.it
www.gioco.it	www.hwupgrade.it	it.ask.com	www.ilbauledellanonna.it	www.fiat.it
www.flashgames.it	www.dynamick.it	www.libero.it	www.manididonna.it	www.autozona.it
www.giochixl.it	www.html.it	www.starpage.com	www.bricoio.it	www.blablacar.it
www.1001giochi.it	www.ilsoftware.it	www.virgilio.it	www.lemercerie.it	www.motori.it
www.giochigratisonline.it	www.mambro.it	www.istella.it	www.hobbyhobby.it	www.motorionline.com
giochi.disney.it	www.mrwebmaster.it	arianna.libero.it	verapaolagino.oneminutesite.it	www.motogp.com

TABLE II
WEB SITES CONSIDERED IN THIS STUDY, GROUPED BY CATEGORY. THESE ARE POPULAR SERVICES IN ITALY.

of rules, and we flag matching domains as trackers. The resulting super-set of regular expressions includes $15,245$ entries, and when applied on our dataset, we observe $1,140$ different tracking domains. Note that third-party domains might behave as trackers (i.e., use some user identifier) depending on the first party they are connected to [27]. In this work, we label as *trackers* all domains which match independently from the web page they are associated to.

IV. METRICS

We are interested in understanding how effective tracker-blockers are, and in quantifying their impact on the browsing QoE perceived by the users. Hence, we extract from HAR files the following set of metrics.

- *Contacted Trackers* (CTR): This set includes, for each page, the third-party domains actually contacted by the browser and present in our list of trackers.
- *Contacted Third Parties* (CTP): This set includes, for each page, the third-party domains actually contacted by the browser regardless of whether they are in our tracker list.
- *Loading Time [s]* (LT): the time needed to download and display all the elements contained in the web page. More precisely, we measure this by waiting for the browser to fire the `OnLoad` event.
- *Volume [bytes]* (V): the overall volume of bytes downloaded by the browser to build the web page.

CRT and CTP allow us to understand how effective each tracker-blocker is at protecting user's privacy. Intuitively, the larger CTR and CTP, the higher the privacy protection level. LT and V help us describing the impact on the page loading speed introduced by each plug-in. In this case, smaller LT and V should translate in better QoE perceived by the user, supposing that all content needed to render the page is loaded correctly.

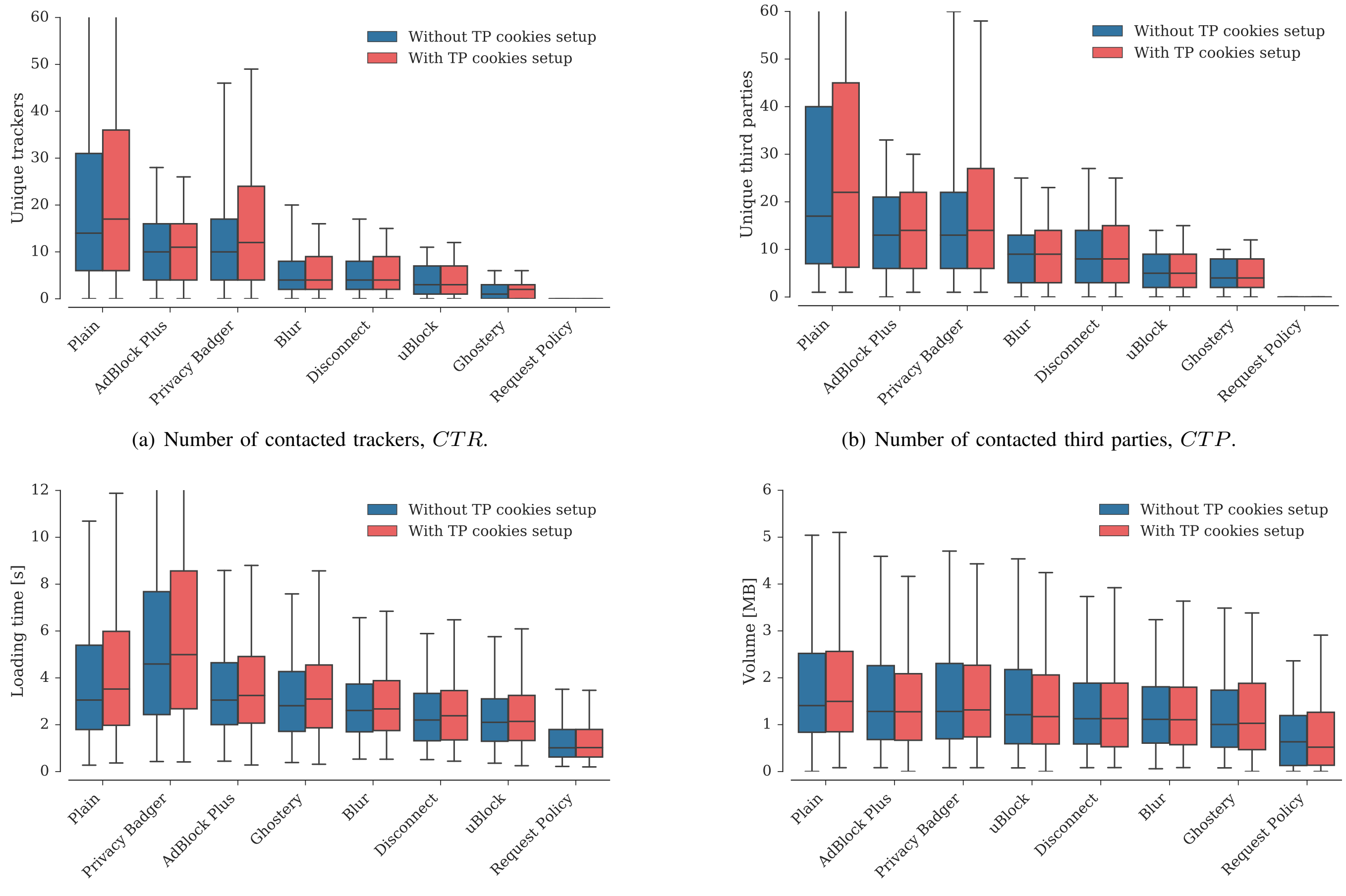
V. RESULTS

In this section we present the results we obtain by analyzing the data collected in our measurement campaign. For the presentation, we follow a top-down approach. We start presenting an overview of per-tracker-blocker results in Sec. V-A. Then, we provide more detailed per-category and per-web-site insights in the subsequent sections.

A. Performance overview

Protection from trackers: We start by analyzing the effectiveness of tracker-blockers to protect users' privacy. For each visit, we extract from the corresponding HAR file the list of distinct contacted trackers, CTR. Its size, $|CTR|$, represents the number of distinct trackers that have not been blocked. Hence, the larger $|CTR|$, the weaker the privacy protection for the considered browser profile. Considering all visits for a given profile, we obtain a set of $|CTR|$ samples that we use to build empirical cumulative distributions. We compute the 5th, 25th, 50th, 75th and 95th percentiles and use them to build the box plots in Figure 1(a). The blue box series refers to the setting without consent to TP cookies, and the red one to the case with consent to TP cookies. The leftmost pair of boxes represents the results for our baseline (Plain). The remaining browser profiles are sorted by the median value of $|CTR|$ without consent to TP cookies.

Focusing on the performance of tracker-blockers, we notice that they show quite different behaviors. Let us consider the setting without consent to TP cookies first. Request Policy blocks all cross-site connections, and thus to trackers too. It is effective at preserving users' privacy, but at the cost of breaking the rendering for the vast majority of web pages. The



(a) Number of contacted trackers, CTR.

(b) Number of contacted third parties, CTP.

(c) Loading time, LT.

(d) Volume, V.

Fig. 1. Box plots describing the distribution (5th, 25th, 50th, 75th and 95th) of different metrics for the Plain browser profile and profiles installing different tracker-blockers. Blue and red series refer to settings with and without consent to TP cookies, respectively.

second best is Ghostery, that apparently misses a few trackers. More precisely, 2 trackers for 50% of visits. By manually inspecting these, we observe that most of them belong to Google and Facebook ecosystems, e.g., *fonts.googleapis.com* and *facebook.com*. We label these as trackers because of our classification that does not take into account side information such as context and first parties. For instance, one of the pages we visit, *www.motorionline.com*, requires the browser to fetch support objects from the third-party platform *gstatic.com* which is present in our list because it can be found acting as a tracker in different contexts, although not in this case. We conclude that Ghostery filters out trackers with a considerable precision. Next in the rank we find uBlock, Disconnect and Blur. These miss domains such as *intellitxt.com* and *outbrain.com*. This is rather surprising as these are known trackers and included in Disconnect's blacklist [25]. Moving on, Privacy Badger exhibits poor performance and does not block a sizable number of trackers. However, this is a consequence of its behavioral-based anti-tracking mechanism, which needs time to understand which third parties to stop. We plan to run a more realistic test where Privacy Badger has been previously trained. At last, despite not being specialized in defeating trackers, Adblock Plus blocks some of them, if compared to the baseline. This is due to the fact that some of the tracker domains correspond to advertisement platforms too.

Finally, we observe that the number of contacted trackers for the setup without consent to TP cookies is always larger than 0. For instance, the baseline Plain shows that half of the visits include connections to more than 16 different trackers. On average, 29.5 distinct trackers are contacted in this configuration. This is very surprising as one would expect the browser to establish connections with trackers only when the user provided explicit consent to TP cookies, as required by the EU directives. Among the trackers involved, we count very popular ones such as *doubleclick.com*, *scorecardresearch.com* and *criteo.com*. On the other hand, as expected, we observe that $|CTR|$ for the setting with consent to TP cookies is always larger than the case without consent to TP cookies, but the increment is fairly limited. For instance, for the Plain configuration, the 75th percentile is equal to 38 for setup with consent to TP cookies and to 33 without consent to TP cookies. Similarly, the median increases from 14 to 18.

In summary, for the setting with consent to TP cookies Request Policy reduces the number of contacted distinct trackers by 99.9% with respect to the baseline, on average. Ghostery comes second, with a 91.3% reduction, followed by uBlock, Disconnect and Blur with 81.8%, 74.9%, and 74.3%, respectively. Finally, Adblock Plus and Privacy Badger, with 51.6% and 32.7%. The setting without consent to TP cookies shows similar results.

Contacted third parties: We now analyze the number of distinct third parties $|CTP|$ that are contacted at each visit. We report the results in Figure 1(b). As for the number of trackers, we observe that the amount of contacted third parties is quite large. The Plain browser contacts more than 19 (23) third parties in half of the visits in setting without consent to TP cookies (with consent to TP cookies). By comparing these numbers with Figure 1(a), we observe that a wide percentage –$78,9\%$ for setting without consent to TP cookies– of third parties are trackers. As a consequence, the results in this case are aligned with those in Figure 1(a). Request Policy blocks all third parties by design and this clearly breaks page rendering.

Page loading time: We now focus on understanding how much faster the browser is to render pages when installing tracker-blockers. Figure 1(c) reports the loading time, LT. First, we immediately notice that by consenting TP cookies, the user shall experience a worst QoE. Indeed, the loading time median increases by 1.3s for Plain profile when cookies are accepted. By comparing the results of Plain configuration with profiles installing tracker-blockers, we observe that the filtering of many third parties to contact considerably decreases the time needed to render the page. In fact, on average, Request Policy is 67.7% faster than the baseline for setting with consent to TP cookies. uBlock, Disconnect, and Blur improve the average loading time by 43.2%, 38.8% and 28.5%, respectively. Interestingly, Ghostery, which blocks most of trackers, comes fourth with 23.9% improvement. Adblock Plus increases the loading speed too, by a mere 16.4% improvement. Very unexpectedly, Privacy Badger even harms user QoE and it slows down the loading time by 44.1%. We speculate that the behavioral-based anti-tracking mechanism it implements, requires a larger processing time than other blacklist-based solutions.

Bandwidth saving: We now investigate the bandwidth saving provided by tracker-blockers. Figure 1(d) reports the results for the volume, V, i.e., the total amount of data downloaded to render the page. Focusing on the Plain profile in setting without consent to TP cookies, we see that V is lower than 5MB in 75% of the cases. The volume marginally increases when accepting cookies from third parties. Cookies are indeed small in size, and as we saw, the page does not change in content when third-party cookies are accepted or not. Let us now concentrate on tracker-blockers. They clearly decrease the amount of bytes to download. Intuitively, one would expect that the better the tracker-blocker is at blocking trackers, the larger the bandwidth it saves. This holds for Request Policy, which decreases the data to download by 65.2% (setup with consent to TP cookies), and Ghostery, with a 31.2% saving. Then, we notice Blur, with a 26.1% reduction, followed by Disconnect, and uBlock, which save 24.4% and 21.7% bandwidth, respectively. That is, the cost of downloading ads and tracking data sums as more than 30% of data volume. Interestingly, Adblock Plus, which specializes in blocking ads, does not offer the best results (14.7%). This is a little surprise as it should prevent the browser from downloading heavy advertisement content such as images and videos. Finally, as expected, Privacy Badger provides little bandwidth saving (12.2% only).

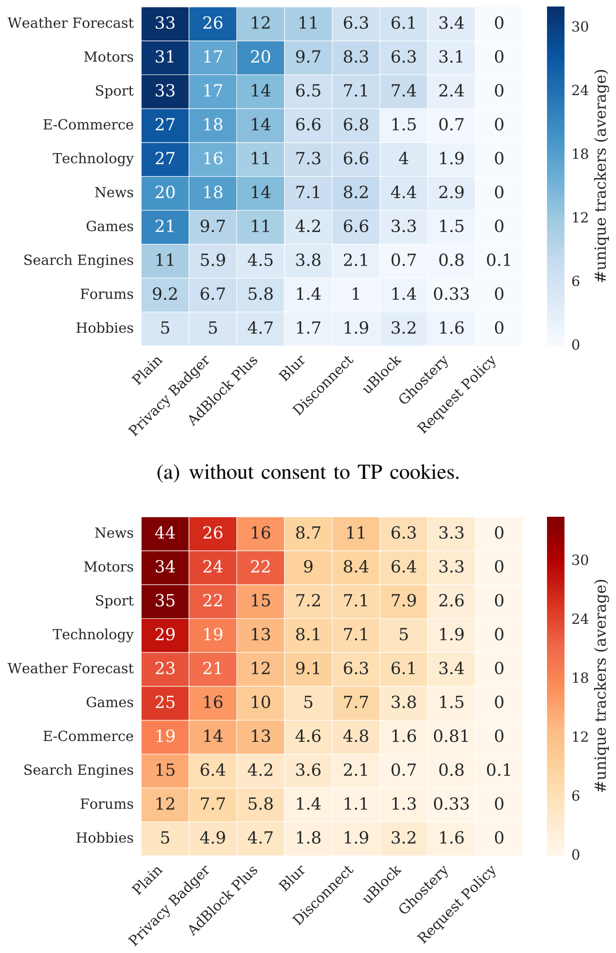
***Take-away*:** *First, for what concerns tracker-blockers' performance, we can conclude that the anti-tracking mechanisms they build upon do matter. Blacklist-based tracker-blockers, Ghostery, uBlock and Disconnect are fairly good at blocking trackers. They help save bandwidth and accelerate the page loading as stated by their producers. Instead, the behavior-based approach employed by Privacy Badger does not provide good protection since the first visits, and it introduces unnecessary extra delays which can severely impair the navigation experience.*[6]

*Second, we observe that EU ePrivacy Directive on cookies is not respected for a wide number of web sites. In fact, many trackers are contacted by the browser **before** the user provides consent to the usage of third-party cookies. They thus violate the directive. This said, by consenting to third-party cookies, the browsing experience degrades by a marginal factor since content is the same in most of cases.*

B. *Per-category results*

We are now interested in understanding how tracker-blockers behave for each web-page category. To this end, we use the results shown in the heatmaps in Figure 2, for both settings without consent to TP cookies (Figure 2(a)) and with consent to TP cookies (Figure 2(b)). Each cell reports the average number of contacted distinct trackers computed considering all visits generated by the corresponding browser profile (column) to web pages belonging to the corresponding category (row). We sort columns and rows based on the average number of contacted trackers, $|CTR|$, so that the cells in the top left corner refer to the cases with highest exposure to trackers, and the ones in the bottom right corner represent the most protected configurations. First, let us focus on Figure 2(a). As already observed in Sec. V-A, Plain profile (the leftmost column), our baseline, includes a large number of trackers, despite the fact that we did not provide explicit consent to the installation of TP cookies. Interestingly, not all categories exhibit the same behavior. In fact, web pages belonging to Weather Forecast, Motors, Sport, E-commerce and Technology host a larger number of trackers in general. This is explained by the fact that these categories are very popular among (Italian) users, and popular web sites tend to monetize their visits via ads [9]. Focusing on the performance of tracker-blockers, Figure 2(a) confirms the observation presented in Sec. V-A, with Ghostery blocking most of the trackers and Privacy Badger showing the weakest performance. Again, Request Policy blocks all

[6] We have contacted EFF and are discussing our results with them.



(a) without consent to TP cookies.

(b) with consent to TP cookies.

Fig. 2. Heatmaps describing the average number of distinct trackers encountered for each category and for each browser profile.

cross-site connections, thus breaking pages, but guaranteeing the best protection from trackers. Observe that values are larger than 0 in Search Engines categories because of Google Search site (*www.google.it*), which embeds support services (e.g., *ssl.gstatic.com*), that we label as trackers, but belonging to Google's ecosystem, and thus permitted by Request Policy.

Let us now focus on Figure 2(b). By comparing the Plain column for both settings with and without consent to TP cookies, we observe that more or less all categories are prone to violate EU ePrivacy Directive. However, the rank of categories changes: the pages in News and Games double the number of embedded trackers when TP cookies are accepted, on average. Instead, pages in other categories have the same number of trackers when user provides consent to the installation of TP cookies, confirming that the EU ePrivacy Directive is ignored. Notice also that for Weather Forecast and E-commerce, the average number of trackers decreases. This is due to a few pages, e.g., *www.centrometeoitaliano.it* that, once provided consent to third-party cookies, become so slow to trigger the 100s timeout, thus impeding our tool to generate the HAR file.

Take-away: *Tracker-blockers show consistent privacy-preserving performance across different web categories, with Ghostery being the most effective. We confirm that the EU ePrivacy Directive is not respected across all web categories considered in this study. Even the most popular categories, only partially respect the policy and already embed third-party trackers before the users provided consent to cookie usage.*

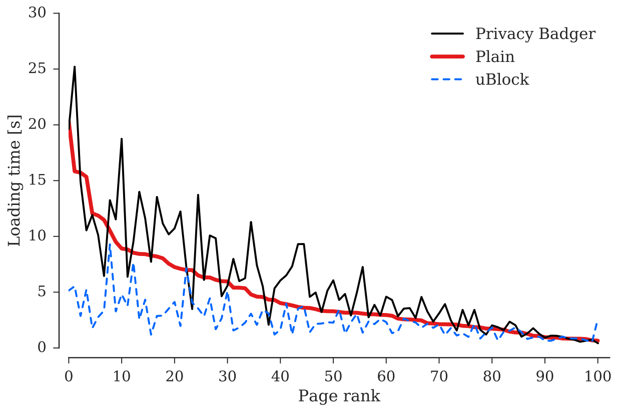


Fig. 3. Per-page average loading time, obtained with Plain, uBlock and Privacy Badger profiles. Pages are sorted based on values obtained with Plain profile. Setting with consent to TP cookies.

C. Per-page results

Now, we dig further and study the behavior of some tracker-blockers on a per-page basis. In particular, we are interested in better understanding the page load performance they provide, and if this is uniform across services. To this end, for each web page and each browser profile, we compute the average loading time we obtain over the 10 visits. We then sort web pages based on loading time with Plain profile, and use this as a reference for comparison. We report the results for uBlock and Privacy Badger in Figure 3. The plot confirms findings in Sec. V-A and shows that on average uBlock notably shortens loading time, whereas Privacy Badger slows down the page loading. However, this does not hold for all web pages. In fact, there exist pages whereby uBlock's improvement is negligible, and, vice versa, there are pages where Privacy Badger profile is faster than the baseline. By manually inspecting these cases, we observe that in general uBlock introduces some extra delays on web pages which are very lightweight and take less than 2.5s to load (mostly in the tail of curves in Sec. V-C). For instance, *www.google.it*, *duckduckgo.com* and *www.autoscout24.it*. These are pages on which the user pays the processing overhead introduced by the browser plug-in. However, this degradation is rather negligible. We observe similar results for the other tracker-blockers providing good privacy-protection. Conversely, Privacy Badger improves the baseline's loading time for web pages which embed many third parties, e.g., *www.lastampa.it*, *www.corrieredellosport.it* and *www.gamespot.com*, where it is able to effectively filter some trackers.

Take-away: *Tracker-blockers improve the loading time, but not uniformly for all web pages. In fact, pages which are fast to load suffer the overhead introduced by tracker-blocker plug-*

ins. The overall impact is however minimal. Similarly, the larger processing overhead introduced by Privacy Badger is amortized on pages rich of third-party content.

VI. Related work

In this section we briefly discuss the body of work related to our study.

Previous studies on web tracking. A number of studies have quantified the diffusion and pervasiveness of trackers in the last years. [23] provides an early snapshot of web tracking, showing that the largest third-party organizations had at least doubled their presence in sites between 2005 and 2008. Since then, many papers have shown a worryingly consistent growth of third-party trackers in the web [28], [29]. Using a passive measurement perspective, [9] shows that some trackers are so pervasive to be able to monitor the activity of 96% of the observed user population. Even more worryingly, trackers are contacted as soon as users switch on their smartphones or tablets. Authors of [30] show how Web tracking has also dramatically grown in complexity. In fact, web trackers leverage a wide catalog of fingerprinting techniques which are largely used to uniquely identify users in the web [27]. A second branch of research has focused on understating how to identify trackers in the wild. In particular, many studies focus on defining automatic methodologies to detect tracking domains [31], [32], [33], [34]. For instance, methodologies proposed in [33], [34] build on navigation data inspection to detect persistent user identifiers in order to label third parties as trackers. The results obtained with such methodologies can be employed to build accurate blacklists.

Performance of tracker-blockers. Surprisingly, only little is known about the tracker-blocking tools that are available in the market. To the best of our knowledge we are the first to present results of a comprehensive benchmarking of different tracker-blockers. Moreover, we compare both their effectiveness at blocking trackers, and their impact on users' QoE. The only very related work we could find are [35] and [27]. The former compares different tracker-blockers, but results are already outdated, and it does not analyze their impact on web QoE. The latter evaluates the privacy-preserving performance of Ghostery only. Similarly to us, it shows that overall Ghostery is effective at detecting and blocking connections to trackers. However, differently, our experiments are conducted to compare several tracker-blockers from different perspectives and with different settings for third-party cookies.

VII. Limitations and Future Work

Our analysis presents some limitations hereby briefly discussed.

i) Our measurement campaign is based on a limited number of web pages and categories. However, from our results clear trends emerge. We are confident that increasing the number of URLs would lead us to obtain similar results.

ii) The tool we designed to automatically browse URLs does not interact with pages as real users do. Hence, our results pertain only to homepages. This is common with studies based on similar measurement campaigns, e.g., [27].

iii) Privacy Badger shows weak performance because it is penalized by our testbed settings. In fact, our tool starts with a fresh profile at each visit, thus preventing Privacy Badger, whose tracker-detection algorithm builds on browser's history, to identify and block trackers effectively [36].

iv) Our measurements were collected from Linux-based servers in our lab, connected to the Internet with a 1Gb/s network, public IP addresses, and without proxy nor NATs. Hence, real users who access the Internet through residential or commercial connection might perceive different QoE. Furthermore, we used a browser from a single firm (Firefox). Other browsers might load web pages differently, and they offer their own sets of APIs, which might change tracker-blockers' performance.

v) Despite our method of classifying trackers being fairly standard in the literature [27], [9], the list of trackers we employ for this purpose can not be considered as an actual ground truth. For instance, our list might miss very novel trackers. However, we do not expect this to bias our results.

vi) Our analysis is limited to understanding the impact of tracker-blockers based on a few set of metrics. However, the amount of information contained in HAR files is rich and it includes data on cookies and objects installed by third parties that we plan to investigate in the future.

For our future work, we plan to run larger and more comprehensive measurement campaigns in order to overcome the limitations described above. In particular, we are running experiments on larger catalogs of URLs and involving more complex profiles, different browsers, operating systems, and network scenarios. We are also including other tracker-blockers not considered in this paper. Moreover, we will dig further in our data by analyzing other metrics.

VIII. Conclusion

We presented a systematic benchmark and comparison of tracker-blockers available in the market. We leveraged a sizable dataset of automatically generated traffic summaries to evaluate the effectiveness of several tracker-blockers to protect users' browsing from trackers and qualitatively estimate how they affect web QoE. We also considered different settings, based on users' consent to install third-party cookies as required by the EU ePrivacy Directive. To the best of our knowledge, we are the first to conduct this kind of study.

Our results are in part surprising.

First, we conclude that the most prominent tracker-blockers are rather effective at identifying and blocking traffic toward trackers. In particular, Ghostery, if properly enabled, offers the best protection from trackers (91.3% of trackers blocked), followed by uBlock, Disconnect and Blur which are surprisingly unable to capture connections to some very popular tracker domains.

We also tested tracker-blockers from another angle, and evaluated their impact on bandwidth usage and user web QoE. Enabling tracker-blockers reduces bandwidth usage up to 30%. For instance, Ghostery can reduce the amount of downloaded data by 31.2%. Instead, web QoE is impacted differently.

uBlock is the best at accelerating page loading being 43.2% faster than the baseline. Ghostery is only fourth in the rank, with 23.9% acceleration. Privacy Badger exhibits a negative impact on loading time, which increases by 44.1%.

Finally, we observe that the ePrivacy Directive is ignored by the vast majority of considered web pages, and a wide number of third-party trackers (29.5 on average) is contacted before users provide consent to third-party cookies.

We believe our results can guide the average Internet user to make an informed choice on tracker-blockers, and help developers and practitioners design better tracker-blocking technologies.

REFERENCES

[1] C. Hoofnagle, J. Urban, and S. Li, "Privacy and modern advertising," in *Proceedings of Amsterdam Privacy Conference*, 2012.

[2] J. Turow, J. King, C. J. Hoofnagle, A. Bleakley, and M. Hennessy, "Americans reject tailored advertising and three activities that enable it," 2009, available at SSRN 1478214.

[3] Ghostery, https://www.ghostery.com.

[4] uBlock, https://www.ublock.org.

[5] Blur, https://dnt.abine.com/.

[6] Quantable, "How Many Users Block Google Analytics?" https://www.quantable.com/analytics/how-many-users-block-google-analytics/.

[7] AdBlock Plus, https://adblockplus.org.

[8] E. Pujol, O. Hohlfeld, and A. Feldmann, "Annoyed users: Ads and ad-block usage in the wild," in *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*. ACM, 2015, pp. 93–106.

[9] H. Metwalley, S. Traverso, M. Mellia, S. Miskovic, and M. Baldi, "The online tracking horde: a view from passive measurements," in *International Workshop on Traffic Monitoring and Analysis*. Springer, 2015, pp. 111–125.

[10] Tom Simonite, "Popular Ad Blocker Also Helps the Ad Industry," http://mashable.com/2013/06/17/ad-blocker-helps-ad-industry/#97VsnvLoaGqM.

[11] Lara O'Reilly, "Google, Microsoft, and Amazon are paying Adblock Plus huge fees to get their ads unblocked," http://uk.businessinsider.com/google-microsoft-amazon-taboola-pay-adblock-plus-to-stop-blocking-their-ads-2015-2.

[12] Disconnect, "Disconnect loads pages 27% faster," https://disconnect.me/faster.

[13] Council of European Union, "Council regulation (EU) no 58/2002," 2002, http://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1491823875591&uri=CELEX:32002L0058.

[14] ——, "Council regulation (EU) no 136/2009," 2009, http://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32009L0136.

[15] Request Policy, https://requestpolicy.com.

[16] Privacy Badger, https://www.eff.org/privacybadger.

[17] Disconnect, https://disconnect.me.

[18] G. Acar, C. Eubank, S. Englehardt, M. Juarez, A. Narayanan, and C. Diaz, "The web never forgets: Persistent tracking mechanisms in the wild," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 674–689.

[19] T.-F. Yen, Y. Xie, F. Yu, R. P. Yu, and M. Abadi, "Host Fingerprinting and Tracking on the Web: Privacy and Security Implications," in *Proceedings of the 2012 Network and Distributed System Security Symposium*, 2012.

[20] Wikipedia, "Real-time Bidding," https://en.wikipedia.org/wiki/Real-time_bidding.

[21] Selenium Web Browser Automation, http://www.seleniumhq.org/.

[22] HAR 1.2 Spec, http://www.softwareishard.com/blog/har-12-spec/.

[23] B. Krishnamurthy and C. Wills, "Privacy diffusion on the web: a longitudinal perspective," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 541–550.

[24] EasyList, http://easylist.to.

[25] Disconnect, "Disconnect's tracker list," https://disconnect.me/trackerprotection/blocked.

[26] Princeton Web Census, https://webtransparency.cs.princeton.edu.

[27] S. Englehardt and A. Narayanan, "Online tracking: A 1-million-site measurement and analysis," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 1388–1401.

[28] J. R. Mayer and J. C. Mitchell, "Third-party web tracking: Policy and technology," in *Security and Privacy (SP), 2012 IEEE Symposium on*. IEEE, 2012, pp. 413–427.

[29] I. Altaweel, N. Good, and C. J. Hoofnagle, "Web privacy census," 2015.

[30] A. Lerner, A. K. Simpson, T. Kohno, and F. Roesner, "Internet jones and the raiders of the lost trackers: An archaeological study of web tracking from 1996 to 2016," in *25th USENIX Security Symposium (USENIX Security 16)*. USENIX Association, 2016.

[31] J. Bau, J. Mayer, H. Paskov, and J. C. Mitchell, "A promising direction for web tracking countermeasures," in *Proceedings of the 2013 Web 2.0 Security and Privacy conference*. IEEE, 2013.

[32] F. Roesner, T. Kohno, and D. Wetherall, "Detecting and defending against third-party tracking on the web," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association, 2012, pp. 12–12.

[33] S. Englehardt, D. Reisman, C. Eubank, P. Zimmerman, J. Mayer, A. Narayanan, and E. W. Felten, "Cookies that give you away: The surveillance implications of web tracking," in *Proceedings of the 24th International Conference on World Wide Web*. ACM, 2015, pp. 289–299.

[34] H. Metwalley, S. Traverso, and M. Mellia, "Unsupervised detection of web trackers," in *Global Communications Conference (GLOBECOM), 2015 IEEE*. IEEE, 2015, pp. 1–6.

[35] J. Mayer, "Tracking the Trackers: Self-Help Tools," http://cyberlaw.stanford.edu/blog/2011/09/tracking-trackers-self-help-tools.

[36] Kif Leswing, "Not all ad blockers are the same. Heres why the EFFs Privacy Badger is different," https://gigaom.com/2014/05/11/not-all-ad-blockers-are-the-same-heres-why-the-effs-privacy-badger-is-different/.